Polska Akademia Nauk
Instytut Podstaw Informatyki

Agnieszka Prochenka

Delete or Merge Regressors algorithm

Supervisor:
dr hab.
Piotr Pokarowski

Warsaw, June 2016

Research of the author was partially supported by Project Information technologies: research and their interdisciplinary applications, POKL.04.01.01-00- 051/10-00.

Dziękuję promotorowi Piotrowi Pokarowskiemu za pomoc w pisaniu pracy,
Profesorowi Janowi Mielniczukowi za cenne rady,
narzeczonemu Michałowi i rodzinie: Hance, Donatowi i Markowi za wsparcie
oraz doktorantom i pracownikom IPI PAN
za przyjacielską atmosferę podczas pisania doktoratu.

Contents

Chapter 1

Introduction

When a categorical predictor is considered, in order to reduce model's complexity, we can either exclude the whole factor or merge its levels. In this dissertation we define the factorial selection problem as selection of a model consisting of a subset of continuous predictors and partitions of levels of factors. Furthermore, an algorithm which solves the factorial problem called Delete or Merge Regressors (DMR) described in Maj-Kańska et al. [2015] is examined and extended to high-dimensional data, where the number of predictors p is significantly larger than the number of observations n.

A very popular tool for model selection in high-dimensional regression when categorical as well as continuous predictors are present, is the group lasso proposed in Yuan and Lin [2006]. It uses the lasso (Tibshirani [1996]) penalty in order to decrease the model by selecting a subset of variables, both continuous and categorical, but without merging levels of factors. Since 2006 many improvements of the algorithm have been constructed, like Meier et al. [2008] or Breheny and Huang [2015].

The idea of partitioning a set of levels of a factor into non-overlapping groups has already been present in the literature. John W. Tukey (Tukey [1949]) described a stepwise backward procedure based on the studentised range which gives grouping of means for samples from normal distributions. However, the first algorithm that solves the factorial selection problem concerning linear regression model with both categorical and continuous variables, was introduced in Bondell and Reich [2009]. The algorithm called Collapsing And Shrinkage ANOVA (CAS-ANOVA) uses the lasso penalty imposed on differences between parameters corresponding to levels of each factor. This algorithm can be interpreted as a generalization of fused lasso (Tibshirani et al. [2004]) to data with categorical variables. In Gertheiss and Tutz [2010] one can find a modification of CAS-ANOVA, which is more computationally efficient because of using the least angle regression algorithm (LARS; Efron et al. [2004]). Another algorithm, based on regularized model selection with categorical predictors and effect modifiers (Oelker et al. [2014]) is implemented in `R` package `gvcm.cat`. It generalizes the lasso approach to simultaneous factor partitioning and selection of continuous variables to generalized linear models. The algorithm is based on local quadratic approximation of the penalty and iterated reweighted least squares.

Our DMR algorithm is based on a traditional stepwise method. In particular, it is a generalization of the Zheng-Loh greedy algorithm (Zheng and Loh [1995]), which assumes model selection of only continuous predictors. First, the squared t-statistics for the predictors are

sorted. Secondly, the final model is chosen according to the information criterion from the nested family of models created by accepting hypotheses in the ascending order of the squared t-statistic.

It has been described in the literature, for example in Zhang [2010a], that lasso regularization often chooses too large models. We claim that stepwise methods give sparser models with higher model selection accuracy and often also with lower prediction error. Our results are not exceptional in comparison to others in the literature. In Example 1 in Zou and Li [2008] a similar simulation setup to our Experiment 3, $n = 96$, has been considered. The adaptive Lasso method (denoted there as one-step LOG) was outperformed by exhaustive BIC (Bayes Information Criterion) with 66 to 73 percent of model selection accuracy. We repeated the simulations and got similar results with 76 percent for the Zheng-Loh algorithm.

Similar results for the case with only continuous predictors have been described recently in Pokarowski and Mielniczuk [2015], where a combination of screening of predictors by the lasso with the Zheng-Loh greedy selection for high-dimensional linear models has been proposed. The authors showed both theoretically and experimentally that such combination is competitive to the MCP (Minimax Concave Penalty) regularization described in Zhang [2010b]. MCP regularization was proposed to improve the performance of lasso by constructing convex penalty which interpolates L_1 and L_0 penalty, combining the lasso and the subset selection techniques. . It has been shown that it is better than lasso in terms of model selection accuracy and prediction. DMRnet algorithm is a generalization of the SOSnet.

In comparison to the Zheng-Loh algorithm, DMR allows categorical predictors. In the first step both constraints that the regression coefficient is equal to zero and that two regression coefficients are equal are considered. Ordering of the constraints is done using hierarchical clustering with dissimilarity defined as squared t-statistics. Finally, the final model is chosen according to the information criterion from the nested family of models.

In the thesis 6 variants of the DMR algorithm are described. The first, DMR4lm works for linear models and when $p < n$. The next two, DMR4glm and DMR4glm_wald work for generalized linear models and also when $p < n$. They use likelihood ratio test statistics and Wald statistics, respectively. The algorithms DMR4lm and DMR4glm_wald have been introduced in Maj-Kańska et al. [2015]. In order to make the DMR algorithm work for the high-dimensional setup, grid versions of the algorithms have been constructed: DMRnet4lm, DMRnet4glm and DMR4glm_wald. They consist of an additional screening step, using group lasso (group lasso for generalized linear models hs been described in Meier et al. [2008]), executed for a grid of parameters and after decreasing the problem to $p < n$ they call the DMR algorithm in the second step.

The main theoretical results are that DMR4lm and DMR4glm are consistent model selection methods. A version for DMR4lm has been published in Maj-Kańska et al. [2015], here it has been strengthened by finding upper bounds on the selection error. The main advantage of our theorems over the analogous ones for the lasso based methods is that we allow that the number of predictors grows to infinity.

The main experimental result is the comparison of the DMR algorithms with the lasso-based methods based on 6 real data sets and a simulation study with 12 simulation setups. We showed that the DMR algorithms usually chose sparser models, with higher model selection accuracy and often lower prediction error than CAS-ANOVA and gvcm. We decided to add to the compared methods also group lasso and group MCP algorithms. They both either delete or leave entire factors while selecting models. Despite the fact that their execution time was

lower than for DMR algorithms, their performance was in most cases worse in the sense that they chose larger models with not lower prediction error. Hence, we showed that considering factorial selection instead of deleting or leaving entire factors in the model is worthwhile.

The dissertation is organized as follows. The problem of factorial selection is defined in Section 2. Definitions of the DMR algorithms for linear regression for cases $p < n$ and $p \gg n$ and the asymptotic properties of the DMR4lm algorithm are described in Chapter 3. Formulations of the DMR algorithms for generalized linear regression for cases $p < n$ and $p \gg n$ and the asymptotic properties of the DMR4glm algorithm are described in Chapter 4. Analysis of real and simulated data sets for linear regression is given in Chapter 5. Experiments and real data examples for logistic regression are described in Chapter 6. Chapter 7 concerns the discussion and conclusions.

Chapter 2

Factorial selection

This chapter concerns definition of the factorial model selection. The material partially overlaps with Maj-Kańska et al. [2015].

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

$$\boldsymbol{\eta}^* = \mathbf{X}\boldsymbol{\beta}^* = \mathbf{1}\beta_{00}^* + \mathbf{X}_0\boldsymbol{\beta}_0^* + \mathbf{X}_1\boldsymbol{\beta}_1^* + \ldots + \mathbf{X}_l\boldsymbol{\beta}_l^*, \tag{2.1}$$

and

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 (2.1) as a linear constraint of one of two types:

$$\mathcal{H}_{jk}: \ \beta_{jk}^* = 0 \text{ where } j \in N_k \setminus \{0\}, \ k \in N, \tag{2.2}$$

$$\mathcal{H}_{ijk}: \ \beta_{ik}^* = \beta_{jk}^* \text{ where } i, j \in N_k, \ i \neq j, \ k \in N \setminus \{0\}. \tag{2.3}$$

2.1 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 (2.2) or (2.3) as the full model.

Example 1. For illustration, let us consider the linear predictor with one factor and one continuous variable:

$$\begin{aligned}\mathbf{y} = \mathbf{X}\boldsymbol{\beta}^* &= \mathbf{1}\cdot 1 + \mathbf{X}_0 \cdot 2 + \mathbf{X}_1 \cdot \begin{bmatrix} -2 \\ -2 \\ 0 \end{bmatrix} + \boldsymbol{\varepsilon} \\ &= \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \cdot 1 + \begin{bmatrix} -0.96 \\ -0.29 \\ 0.26 \\ -1.15 \\ 0.2 \\ 0.03 \\ 0.09 \\ 1.12 \end{bmatrix} \cdot 2 + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -2 \\ -2 \\ 0 \end{bmatrix}\end{aligned} \tag{2.4}$$

where $\mathbf{X}_0$ is a vector of length 8 generated independently from standard normal distribution, $\mathcal{N}(\mathbf{0}, \mathbb{I})$. 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 model corresponding to β^* is $(P_0 = \{1\}, P_1 = \{\{1, 4\}, \{2, 3\}\})$ and is the same as F 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), taking into account that the number of feasible models grows faster than exponentially with p. Since for the k-th factor the number of possible partitions is the Bell number $\mathcal{B}(p_k)$, the number of all feasible models is $2^{p_0} \prod_{k=1}^{l} \mathcal{B}(p_k)$. In order to significantly reduce the amount of computations, we propose a greedy backward search. First we precisely define the dimension of a feasible model used in GIC.

2.2 Unconstrained parametrization of feasible models

A feasible model can be defined by a linear space of parameters

$$\mathcal{L}_M = \{\boldsymbol{\beta} \in \mathbb{R}^p : \mathbf{A}_{0M}\boldsymbol{\beta} = 0\}, \tag{2.5}$$

where $\mathbf{A}_{0M}$ is a $(p - q) \times p$ matrix encoding q elementary constraints induced by the model. Such a constraint matrix can be expressed in many ways. In particular, every linear space can be spanned by different vectors. The number of such vectors can be greater than the dimension of the space when they are linearly dependent. In order to unify the form of a constraint matrix, we introduce the notion of its regular form, which is described in Section 2.2.1. We assume that $\mathbf{A}_{0M}$ is in regular form. We perform a standard change of constrained to an unconstrained problem. Let $\mathbf{A}_{1M}$ be a $q \times p$ complement of $\mathbf{A}_{0M}$ to invertible matrix A_M, that is:

$$\mathbf{A}_M = \left[\frac{\mathbf{A}_{1M}}{\mathbf{A}_{0M}}\right].$$

Denote:

$$\mathbf{A}_M^{-1} = \left[\begin{array}{c|c} \mathbf{A}^1_M & \mathbf{A}^0_M \end{array}\right], \tag{2.6}$$

where $\mathbf{A}^1_M$ is a $p \times q$ matrix. In order to replace a constrained by an unconstrained parametrization change of variables in model M is performed. Let $\boldsymbol{\beta}_M \in \mathcal{L}_M$ and $\boldsymbol{\xi}_M = \mathbf{A}_{1M}\boldsymbol{\beta}_M$. We have:

$$\boldsymbol{\beta}_M = \mathbf{A}^1_M \boldsymbol{\xi}_M. \tag{2.7}$$

Indeed,

$$\boldsymbol{\beta}_M = \mathbf{A}_M^{-1}\mathbf{A}_M\boldsymbol{\beta}_M = \mathbf{A}_M^{-1}\left[\frac{\mathbf{A}_{1M}\boldsymbol{\beta}_M}{\mathbf{A}_{0M}\boldsymbol{\beta}_M}\right] = \left[\begin{array}{c|c} \mathbf{A}^1_M & \mathbf{A}^0_M \end{array}\right]\left[\frac{\boldsymbol{\xi}_M}{\mathbf{0}}\right] = \mathbf{A}^1_M\boldsymbol{\xi}_M.$$

From equation 2.7 we obtain $\mathbf{X}\boldsymbol{\beta}_M = \mathbf{Z}_{1M}\boldsymbol{\xi}_M$, where $\mathbf{Z}_{1M} = \mathbf{X}\mathbf{A}^1_M$ and $\mathcal{L}_M = \{\mathbf{A}^1_M\boldsymbol{\xi} : \boldsymbol{\xi} \in \mathbb{R}^q\}$. Let us notice that $\mathcal{L}_M$ is a linear space spanned by columns of $\mathbf{A}^1_M$. The dimension of space $\mathcal{L}_M$ will be called the size of model M and denoted by $|M|$. Note that $|M| = q$.

Example 1 continued. Matrices $\mathbf{A}_M, \mathbf{A}^1_M, \mathbf{Z}_{1M}$ and $\boldsymbol{\xi}_M$ are:

$$\mathbf{A}_M = \left[\begin{array}{ccccc} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ \hline 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{array}\right], \ \mathbf{A}^1_M = \left[\begin{array}{ccc} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{array}\right], \ \mathbf{Z}_{1M} = \left[\begin{array}{ccc} 1 & -0.96 & 0 \\ 1 & -0.29 & 0 \\ 1 & 0.26 & 1 \\ 1 & -1.15 & 1 \\ 1 & 0.2 & 1 \\ 1 & 0.03 & 1 \\ 1 & 0.09 & 0 \\ 1 & 1.12 & 0 \end{array}\right],$$

$$\boldsymbol{\xi}_M = (\xi_1, \xi_2, \xi_3)^T \ , \ \xi_1 = \beta_{00}, \ \xi_2 = \beta_{10} \ , \ \xi_3 = \beta_{21} = \beta_{31}.$$

We define the inclusion relation between two models M_1 and M_2 by inclusion of linear spaces

$$M_1 \subseteq M_2 \text{ denotes } \mathcal{L}_{M_1} \subseteq \mathcal{L}_{M_2} \tag{2.8}$$

and intersection of two models M_1 and M_2 by intersection of linear spaces:

$$M_1 \cap M_2 \text{ as a model defined by } \mathcal{L}_{M_1} \cap \mathcal{L}_{M_2}. \tag{2.9}$$

A feasible model M will be called a true model if $\boldsymbol{\beta}^* \in \mathcal{L}_M$. A true model with minimal size will be denoted by T and its size by $t = |T|$. We assume that $p - t \geq 2$. Observe that T is unique because $\mathbf{X}$ is a full rank matrix.

Example 1 continued. For the example 1 the true model T is $T = (\{1\}, \{\{1, 4\}, \{2, 3\}\})$. The dimensions of the considered models are $|F| = p = 5$, $|T| = 3$.

2.2.1 Regular form of constraint matrix

We say that $\mathbf{A}_{0M}$ is in regular form if it can be complemented to $\mathbf{A}_M$ so that:

$$\mathbf{A}_M = \left[\frac{\mathbf{A}_{1M}}{\mathbf{A}_{0M}}\right] = \left[\frac{\mathbb{I} \quad 0}{\mathbf{B}_M \quad \mathbb{I}}\right], \tag{2.10}$$

where $\mathbf{B}_M$ is a matrix consisting of $0, -1, 1$. Then, using Schur complement we get:

$$\mathbf{A}_M^{-1} = \left[\begin{array}{c|c} \mathbb{I} & 0 \\ -\mathbf{B}_M & \mathbb{I} \end{array}\right] = \left[\begin{array}{c|c} \mathbf{A}^1_M & \mathbf{A}^0_M \end{array}\right]. \tag{2.11}$$

Constraint matrix in regular form can always be obtained by a proper permutation of model's parameters. Let us denote clusters in each partition: $P_k = (C_{ik})_{i=1}^{i_k}$, where i_k is the number of clusters, where $k \in N \setminus \{0\}$ and minimal elements in each cluster as $j_{ik} = \min\{j \in C_{ik}\}$. Let P_0 denote the subset of the set P_{F0} of all continuous variables in the full model. Sort model's parameters in the following order:

1. β_{00}, the intercept,
2. β_{j0}: $j \in P_0 \setminus \{0\}$, the parameters corresponding to continuous predictors present in the model,
3. $\beta_{j_{ik}k}$ for $i = 2, \ldots, i_k$, $k \in N \setminus \{0\}$, the parameters corresponding to minimal elements in each cluster in each factor (with which other parameters corresponding to factors are merged), apart from the first clusters, in which the parameters are merged with the intercept,
4. β_{j0}: $j \in P_{F0} \setminus P_0$, the parameters corresponding to continuous predictors not present in the model, which have to be deleted,
5. β_{jk}, $j \in C_{ik} \setminus \{j_{ik}\}$, $k \in N \setminus \{0\}$, the parameters corresponding to factors, which are not present in point 3.

Sort columns of model matrix $\mathbf{X}$ in the same way as vector $\boldsymbol{\beta}$.

Example 2. As an illustrative example consider a full model $F = (P_{F0}, P_{F1}, P_{F2})$, where

$$P_{F0} = \{1, 2\},\ P_{F1} = (\{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{6\}, \{7\}, \{8\}),$$
$$P_{F2} = (\{1\}, \{2\}, \{3\})$$

and $p_0 = 2, p_1 = 8, p_2 = 3, p = 12$. We denote a feasible model with 7 elementary constraints: $\beta_{10} = 0,\ \beta_{21} = 0,\ \beta_{71} = 0,\ \beta_{31} = \beta_{51},\ \beta_{41} = \beta_{61},\ \beta_{41} = \beta_{81},\ \beta_{22} = 0$ as $M = (P_0, P_1, P_2)$, where:

$$P_0 = \{2\},\ P_1 = (\{1, 2, 7\}, \{3, 5\}, \{4, 6, 8\}),\ P_2 = (\{1, 2\}, \{3\}).$$

Constraint matrix in regular form for model M, where each row corresponds to one of the 7 elementary constraints, is:

$$\mathbf{A}_{0M} = \begin{array}{c} \begin{array}{cccccccccccc} \beta_{00} & \beta_{20} & \beta_{31} & \beta_{41} & \beta_{32} & \beta_{10} & \beta_{21} & \beta_{71} & \beta_{51} & \beta_{61} & \beta_{81} & \beta_{22} \end{array} \\ \left[\begin{array}{cccccccccccc} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right] \end{array}.$$

and after inverting matrix $\mathbf{A}_M^{-1}$ is obtained

$$\mathbf{A}_M^{-1} = \left[\ \mathbf{A}_M^1 \mid \mathbf{A}_M^0\ \right]$$

$$= \left[\begin{array}{ccccc|cccccc} \beta_{00} & \beta_{20} & \beta_{31} & \beta_{41} & \beta_{32} & \beta_{10} & \beta_{21} & \beta_{71} & \beta_{51} & \beta_{61} & \beta_{81} & \beta_{22} \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right]$$

Notice that for regular constraint matrix $\mathbf{Z}_M$ is the full model matrix $\mathbf{X}$ with appropriate columns deleted or added to each other.

2.3 Generalized Information Criterion

Let us denote the likelihood function:

$$\mathscr{L}(\boldsymbol{\beta}) = f_{\boldsymbol{\beta},\sigma^2}(y) = \prod_{i=1}^{n} f_{\boldsymbol{\beta},\sigma^2}(y_i)$$

and the log-likelihood function as:

$$\ell(\boldsymbol{\beta}) = \log \mathscr{L}(\boldsymbol{\beta}).$$

Then the maximum likelihood estimator is:

$$\widehat{\boldsymbol{\beta}}_M = \underset{\boldsymbol{\beta} \in \mathcal{L}_M}{\arg\max}\, \mathscr{L}(\boldsymbol{\beta})$$

and Generalized Information Criterion is:

$$GIC_M = -2\ell(\boldsymbol{\beta}) + r|M|. \tag{2.12}$$

2.3.1 Generalized Information Criterion for linear regression

For the special case of linear regression, where

$$\mathbf{y} = \boldsymbol{\eta} + \boldsymbol{\varepsilon},\ \boldsymbol{\varepsilon} \sim \mathbb{N}(0, \mathbb{I}),\ \boldsymbol{\eta} = \mathbf{X}^T\boldsymbol{\beta}, \tag{2.13}$$

the ordinary least squares estimator is the maximum likelihood estimator. The OLS (Ordinary Least Squares) estimator of $\boldsymbol{\beta}^*$ constrained to $\mathcal{L}_M$ is given by the following expression:

$$\widehat{\boldsymbol{\beta}}_M = \mathbf{A}^1_M \widehat{\boldsymbol{\xi}}_M,\ \text{where } \widehat{\boldsymbol{\xi}}_M = \left(\mathbf{Z}^T_{1M}\mathbf{Z}_{1M}\right)^{-1}\mathbf{Z}^T_{1M}\mathbf{y}. \tag{2.14}$$

Note that $\mathbf{A}_{0M}\widehat{\boldsymbol{\beta}}_M = \mathbf{A}_{0M}\mathbf{A}^1_M\widehat{\boldsymbol{\xi}}_M = 0$ and thus indeed $\widehat{\boldsymbol{\beta}}_M \in \mathcal{L}_M$.

Let $\mathbf{H}_M = \mathbf{Z}_{1M}\left(\mathbf{Z}_{1M}^T\mathbf{Z}_{1M}\right)^{-1}\mathbf{Z}_{1M}^T$. Observe that $\mathbf{H}_M\mathbf{X}\boldsymbol{\beta}^* = \mathbf{X}\boldsymbol{\beta}^*$ for $M \supseteq T$. We define residual sum of squares for model M as $RSS_M = \|\mathbf{y} - \mathbf{X}\widehat{\boldsymbol{\beta}}_M\|^2$. From equation 2.14 we have:

$$RSS_M = \|\mathbf{y} - \mathbf{Z}_{1M}\widehat{\boldsymbol{\xi}}_M\|^2 = \|(\mathbb{I} - \mathbf{H}_M)\mathbf{y}\|^2.$$

Let us denote:

$$\Delta_M = \boldsymbol{\beta}^{*T}\mathbf{X}^T(\mathbb{I} - \mathbf{H}_M)\mathbf{X}\boldsymbol{\beta}^* = \|\mathbf{X}\boldsymbol{\beta}^* - \mathbf{X}\boldsymbol{\beta}_M^*\|^2, \tag{2.15}$$

where $\boldsymbol{\beta}_M^* = \arg\min_{\boldsymbol{\beta}\in\mathcal{L}_M}\|\mathbf{X}\boldsymbol{\beta}^* - \mathbf{X}\boldsymbol{\beta}\|^2$. Notice that if $\frac{1}{n}\mathbf{X}^T\mathbf{X} \xrightarrow{P} \boldsymbol{\Sigma} > 0$ with $n \to \infty$, then from the weak law of large numbers we have $\widehat{\boldsymbol{\beta}}_M \xrightarrow{P} \boldsymbol{\beta}_M^*$. The following decomposition of RSS in linear models is trivial, hence we omit the proof:

Proposition 1. *For linear regression we have:*

$$RSS_M = \Delta_M + 2\boldsymbol{\beta}^{*T}\mathbf{X}^T(\mathbb{I} - \mathbf{H}_M)\boldsymbol{\epsilon} + \boldsymbol{\epsilon}^T(\mathbb{I} - \mathbf{H}_M)\boldsymbol{\epsilon}.$$

In particular for $M \supseteq T$

$$RSS_M = \boldsymbol{\epsilon}^T(\mathbb{I} - \mathbf{H}_M)\boldsymbol{\epsilon} \sim \sigma^2\chi^2_{n-|M|}.$$

Generalized Information Criterion for model M can be defined in two ways, either we assume that σ^2 is unknown, then:

$$GIC_M = n\log RSS_M + r\cdot|M|, \tag{2.16}$$

or we assume that σ^2 is known, then:

$$GIC_M = RSS_M + r\cdot|M|\sigma^2. \tag{2.17}$$

2.4 Competitive algorithms

In the dissertation the DMR algorithm will be compared with methods based on regularization. The most popular algorithm used when both continuous and categorical variables are present in the model is group lasso introduced in Yuan and Lin [2006]. The group lasso estimator is:

$$\widetilde{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}}\left\{-\ell(\boldsymbol{\beta}) + \left(\sum_{j=1}^{p_0}\nu_\lambda(\beta_{j0}) + \sum_{k=1}^{l}\nu_\lambda(\boldsymbol{\beta}_k)\right)\right\}, \tag{2.18}$$

where:

$$\nu_\lambda(\beta_{j0}) = \lambda|\beta_{j0}| \text{ for continuous variables,}$$
$$\nu_\lambda(\boldsymbol{\beta}_k) = \lambda\sqrt{p_k - 1}\|\boldsymbol{\beta}_k\| \text{ for factors.}$$

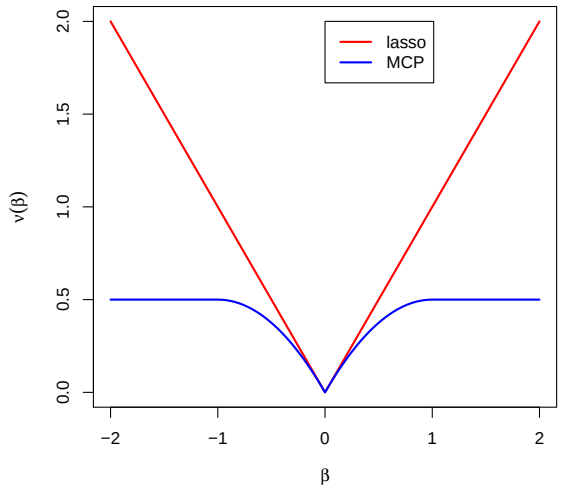
However, it has been described in the literature, for example in Zhang [2010a], that lasso regularization often chooses too large models. Many improvements of the method have been proposed. Among others, MCP regularization described in Zhang [2010b], which assumes a concave penalty and therefore uses more difficult optimization algorithms. It solves similar problem as group lasso (equation 2.18) but instead of the convex ν_λ function it uses concave $\nu_{\gamma,\lambda}$:

$$\nu_{\gamma,\lambda}(\beta_{j0}) = \begin{cases} \lambda|\beta_{j0}| - \frac{\beta_{j0}^2}{2\gamma} & \text{if } |\beta_{j0}| \leq \lambda\gamma \\ \frac{1}{2}\gamma\lambda^2 & \text{if } |\beta_{j0}| > \lambda\gamma \end{cases}$$

$$\nu_{\gamma,\lambda}(\boldsymbol{\beta}_k) = \begin{cases} \sqrt{p_k - 1}\left(\lambda\|\boldsymbol{\beta}_k\| - \frac{\|\boldsymbol{\beta}_k\|^2}{2\gamma}\right) & \text{if } \|\boldsymbol{\beta}_k\| \leq \lambda\gamma \\ \frac{1}{2}\sqrt{p_k - 1}\gamma\lambda^2 & \text{if } \|\boldsymbol{\beta}_k\| > \lambda\gamma \end{cases}$$

The penalty is between L_1 and L_0 penalty, combining the lasso and the subset selection techniques. The comparison of the penalties for continuous predictors when $\lambda = \gamma = 1$ is shown in Figure 2.1. It has been shown that it is better than lasso in terms of model selection accuracy and prediction.

Figure 2.1: Penalties for continuous predictors for lasso and MCP if $\lambda = \gamma = 1$.



Let us notice that the group lasso and group MCP choose entire factors: either leave it in the model or not. Both the algorithms are implemented in `R` package `grpreg`. The competitive methods to DMR which solve the factorial selection problem are CAS-ANOVA and gvcm. They both use lasso regularization to impose penalties on the differences between parameters in factors.

$$\widetilde{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}}\{-\ell(\boldsymbol{\beta}) + \nu_\lambda(\boldsymbol{\beta})\},$$

where

$$\nu_\lambda(\boldsymbol{\beta}) = \sum_{j=1}^{p_0} w_j^{(0)}|\beta_{j0}| + \sum_{k=1}^{l}\sum_{j=1}^{p_k} w_j^{(k)}|\beta_{jk}| + \sum_{k=1}^{l}\sum_{2\leq i\leq j\leq p_k} w_{ij}^{(k)}|\beta_{ik} - \beta_{jk}|.$$

In CAS-ANOVA weights w are the multiplication of adaptive lasso weights, namely the inverses of the ordinary least squares estimators and the normalizing constants depending on the number of levels in factors. In gvcm weights w can obtain different forms, including the

adaptive lasso case. In contrast to CAS-ANOVA, gvcm works for generalized linear models. Implementation of CAS-ANOVA can be found on the website of Howard Bondell, the author of Bondell and Reich [2009] and implementation of gvcm is in `R` package `gvcm`.

In chapters 5 and 6 a comparison of the described methods and DMR algorithm for linear and logistic regression is described.

Chapter 3

Delete or Merge Regressors algorithm for linear model

In this chapter an algorithm DMR4lm is described. It has been introduced in Maj-Kańska et al. [2015], here it is described with some modifications. The proof of consistency in Section 3.3 has been strengthened compared to Maj-Kańska et al. [2015]. Firstly, upper bounds on the selection error are given. Secondly, we let the dimension of the true model t grow to infinity.

Moreover, in this chapter an algorithm DMRnet4lm, which is a generalization of DMR4lm to high-dimensional data sets where $p \gg n$, is introduced. It uses group lasso in the screening step and DMR4lm after decreasing the column dimension to $p < n$.

In this chapter, if not otherwise stated, we assume $p < n$.

3.1 DMR4lm algorithm

In this section we introduce the DMR4lm algorithm. Because of cumbersome notations, in order to make the description of the algorithm more intuitive, we present here a general idea of the algorithm. In particular, we give the details of step 3 and 4 of the algorithm in Sections 3.1.1 and 3.1.2.

Assuming that $\mathbf{X}$ is full rank the QR decomposition of the model matrix is $\mathbf{X} = \mathbf{QR}$, where $\mathbf{Q}$ is $n \times p$ orthogonal matrix and $\mathbf{R}$ is $p \times p$ upper triangular matrix. Denote the minimum variance unbiased estimators of $\boldsymbol{\beta}$ and σ^2 for the full model F as:

$$\widehat{\boldsymbol{\beta}} = \mathbf{R}^{-1}\mathbf{z} \text{ and } \widehat{\sigma}^2 = \frac{\|\mathbf{y}\|^2 - \|\mathbf{z}\|^2}{n-p}, \text{ where } \mathbf{z} = \mathbf{Q}^T\mathbf{y}. \tag{3.1}$$

Let us denote

$$\widehat{\boldsymbol{\beta}} = [\widehat{\beta}_{jk}]_{\substack{j \in N_k \\ k \in N}}, \ \mathbf{R}^{-1} = [r_{jk,st}]_{\substack{j \in N_k \\ s \in N_t \\ k,t \in N}},$$

then

$$\widehat{\beta}_{jk} = \mathbf{r}_{jk}^T\mathbf{z}, \text{ where } j \in N_k, k \in N$$

and $\mathbf{r}_{jk}$ is a row of $\mathbf{R}^{-1}$.

Algorithm 1 DMR4lm (Delete or Merge Regressors for linear models)

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

1. Computation of t-statistics

Compute the QR decomposition of the full model matrix, obtaining matrix $\mathbf{R}^{-1}$, vector $\mathbf{z}$ and variance estimator $\widehat{\sigma}^2$ as in equation 3.1.

Calculate squared t-statistics for all elementary constraints defined in (2.2):

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

$$t^2_{1jk} = \frac{\widehat{\beta}^2_{jk}}{\widehat{Var}(\widehat{\beta}_{jk})} = \frac{(\mathbf{r}^T_{jk}\mathbf{z})^2}{\widehat{\sigma}^2\|\mathbf{r}_{jk}\|^2}$$

end for

Calculate squared t-statistics for all elementary constraints defined in (2.3):

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

$$t^2_{ijk} = \frac{(\widehat{\beta}_{ik} - \widehat{\beta}_{jk})^2}{\widehat{Var}(\widehat{\beta}_{ik} - \widehat{\beta}_{jk})} = \frac{((\mathbf{r}_{ik} - \mathbf{r}_{jk})^T z)^2}{\widehat{\sigma}^2\|\mathbf{r}_{ik} - \mathbf{r}_{jk}\|^2}$$

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**

$d_{1jk} = d_{j1k} = t^2_{1jk}$ for $j \in N_k$,

$d_{ijk} = t^2_{ijk}$ 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 $\mathbf{h}^T_1, \mathbf{h}^T_2, \ldots, \mathbf{h}^T_l$.

3. Sorting constraints (hypotheses) according to the squared t-statistics

Combine vectors of cutting heights: $\mathbf{h} = [0, \mathbf{h}^T_0, \mathbf{h}^T_1, \ldots, \mathbf{h}^T_l]^T$, where $\mathbf{h}_0$ is a vector of squared t-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. Details of clustering and sorting are given in Section 3.1.1.

4. Computation of RSS using a recursive formula in a nested family of models

Compute QR decomposition of the matrix $\mathbf{R}^{-T}\mathbf{A}^T_0$ obtaining the orthogonal matrix $\mathbf{W} = [\mathbf{w}_1, \ldots, \mathbf{w}_{p-1}]$. Set $\mathrm{RSS}_{M_0} = \|\mathbf{y}\|^2 - \|\mathbf{z}\|^2$ for a model without constraints.

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

$$\mathrm{RSS}_{M_m} = \mathrm{RSS}_{M_{m-1}} + (\mathbf{w}^T_m\mathbf{z})^2,$$

where M_m denotes a model with constraints defined by m first rows of $\mathbf{A}_0$. The constraints cause deleting or merging regressors. Derivation of the last formula are given in Section 3.1.2, see equation 3.8.

end for

Output: $\mathscr{M}^{DMR4lm} = \{M_0, \ldots, M_{p-1}\}$, $\mathbf{RSS}^{DMR4lm} = (RSS_{M_0}, \ldots, RSS_{M_{p-1}})^T$.

DMR4lm returns p nested models with dimensions from 1 to p. The choice of the final model can be based for example on cross-validation or information criterion.

The time complexities of successive steps of the DMR algorithm are $O(np^2)$ for QR decomposition in step 1, $O(p^2)$ for hierarchical clustering in step 2, $O(p^3)$ for QR decomposition used in step 4. The dominating operation in the described procedure is the QR decomposition of the full model matrix. Hence, the overall time complexity of the DMR algorithm is $O(np^2)$.

3.1.1 Sorting constraints according to the squared t-statistics

Since step 3 of the DMR algorithm needs complicated notations concerning hierarchical clustering, we decided to present them in the Appendix for the interested reader. In particular, we show here how the cutting heights vector $\mathbf{h}$ and matrix of constraints $\mathbf{A_0}$ are built.

Let us define vectors $\mathbf{a}(1,j,k)$ and $\mathbf{a}(i,j,k)$ (corresponding to the elementary constraints, being building blocks for $\mathbf{A_0}$) such that:

$$\mathbf{a}(1,j,k) = [a_{st}(j,k)]_{\substack{s \in N_t \\ t \in N}},\ a_{st}(j,k) = \mathbb{1}(s=j, t=k), \tag{3.2}$$

$$\mathbf{a}(i,j,k) = [a_{st}(i,j,k)]_{\substack{s \in N_t \\ t \in N}},\ a_{st}(i,j,k) = \mathbb{1}(s=i, t=k) - \mathbb{1}(s=j, t=k). \tag{3.3}$$

For each step s of the hierarchical clustering algorithm we use the following notation for the partitions of set $\{1\} \cup N_k = \{1, 2, \ldots, p_k\}$:

$$P_{sk} = \{C_{isk}\}_{i=1}^{p_k - s + 1},\ s = 1, \ldots, p_k.$$

We assume complete linkage clustering:

$$\begin{aligned} &d\left(C_{i_{s+1},s+1,k} = C_{i_s sk} \cup C_{j_s sk}, C_{j_{s+1},s+1,k} = C_{o_s sk}\right) \\ &\quad = \max\left\{d\left(C_{i_s sk}, C_{o_s sk}\right), d\left(C_{j_s sk}, C_{o_s sk}\right)\right\}. \end{aligned}$$

Cutting heights in steps $s = 1, \ldots, p_k - 1$ are defined as:

$$h_{sk} = \min_{i \neq j} d\left(C_{isk}, C_{jsk}\right).$$

Let us denote vector $\tilde{\mathbf{a}}_{sk}$ as an elementary constraint corresponding to cutting height h_{sk}, where:

$$\begin{aligned} \tilde{\mathbf{a}}_{sk} &= \mathbf{a}(i_*, j_*, k),\ i_* = \min_{i \in C_{i_1 sk}} i,\ j_* = \min_{j \in C_{j_1 sk}} j \text{ and } (i_1, j_1) \\ &= \arg\min_{i \neq j} d\left(C_{isk}, C_{jsk}\right). \end{aligned}$$

Step 3 of the algorithm can be now rewritten:

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 vector of cutting heights for constraints concerning continuous variables and 0 corresponds to model without constraints:

$$\mathbf{h}_k = [h_{sk}]_{s=1}^{p_k - 1},\ k \in N \setminus \{0\} \text{ and } \mathbf{h}_0 = [0, t^2_{110}, t^2_{120}, \ldots, t^2_{1p_0 0}]^T.$$

Sort elements of $\mathbf{h}$ in increasing order getting $\mathbf{h}_{:} = [h_{m:p}]_{m=1}^{p}$ and construct $(p-1) \times p$ matrix of constraints

$$\mathbf{A}_0 = [\tilde{\mathbf{a}}_{2:p}, \tilde{\mathbf{a}}_{3:p}, \ldots, \tilde{\mathbf{a}}_{p:p}]^T,$$

where $\tilde{\mathbf{a}}_{m:p}$ is the elementary constraint corresponding to cutting height $h_{m:p}$. Then proceed as described in Algorithm 1.

3.1.2 Recursive formula for RSS in a nested family of linear models

In this section we show some implementation facts concerning the DMR algorithm. In particular an effective way of calculation of residual sums of squares for nested models using QR decompositions is discussed.

Let us consider a linear model with linear constraints:

$$\mathcal{L} = \{\boldsymbol{\beta} \in \mathbb{R}^p, \mathbf{A}_0\boldsymbol{\beta} = \mathbf{0}\}, \tag{3.4}$$

where A_0 is $(p-q)\times p$ constraint matrix. The objective is to calculate residual sum of squares $RSS = \|\mathbf{y} - \mathbf{X}\widehat{\boldsymbol{\beta}}\|^2$. QR decomposition of the model matrix is performed

$$\mathbf{X} = \mathbf{QR},$$

where $\mathbf{Q}$ is $n \times p$ orthogonal matrix and $\mathbf{R}$ is $p \times p$ upper triangular matrix. Let us denote $\mathbf{S} = \mathbf{R}^{-T}\mathbf{A}_0^T$, then

$$\mathbf{Q}^T\mathbf{y} = \mathbf{R}\boldsymbol{\beta}^* + \mathbf{Q}^T\boldsymbol{\varepsilon} \text{ and } \mathbf{S}^T\mathbf{R}\boldsymbol{\beta}^* = \mathbf{0}.$$

After substitution $\mathbf{z} = \mathbf{Q}^T\mathbf{y}$, $\boldsymbol{\gamma}^* = \mathbf{R}\boldsymbol{\beta}^*$, $\boldsymbol{\eta} = \mathbf{Q}^T\boldsymbol{\varepsilon}$ we get

$$\mathbf{z} = \boldsymbol{\gamma}^* + \boldsymbol{\eta} \text{ and } \mathbf{U}^T\mathbf{W}^T\boldsymbol{\gamma}^* = \mathbf{0}, \tag{3.5}$$

where $\mathbf{W}$ and $\mathbf{U}$ are respectively $p \times (p-q)$ orthogonal matrix and $(p-q)\times(p-q)$ upper triangular matrix from the QR decomposition of matrix $\mathbf{S}$. We have

$$\mathbf{W}^T\boldsymbol{\gamma}^* = \mathbf{U}\mathbf{U}^T\mathbf{W}^T\boldsymbol{\gamma}^* = \mathbf{0}.$$

Let us denote $\overline{\mathbf{W}}$ as orthogonal complement of $\mathbf{W}$ to matrix with dimensions $p\times p$. We multiply equation 3.5 by $[\overline{\mathbf{W}}, \mathbf{W}]$:

$$[\overline{\mathbf{W}}, \mathbf{W}]^T\mathbf{z} = [\overline{\mathbf{W}}, \mathbf{W}]^T\boldsymbol{\gamma}^* + [\overline{\mathbf{W}}, \mathbf{W}]^T\boldsymbol{\eta} \text{ and } \mathbf{W}^T\boldsymbol{\gamma}^* = 0.$$

Therefore the OLS estimator $\widehat{\boldsymbol{\gamma}}$ of $\boldsymbol{\gamma}^*$ with constraints satisfies the following equation

$$\begin{bmatrix} \overline{\mathbf{W}}^T\mathbf{z} \\ 0 \end{bmatrix} = [\overline{\mathbf{W}}, \mathbf{W}]^T\widehat{\boldsymbol{\gamma}}. \tag{3.6}$$

Multiplying (3.6) by $[\overline{\mathbf{W}}, \mathbf{W}]$, we obtain $\overline{\mathbf{W}}\overline{\mathbf{W}}^T\mathbf{z} = \widehat{\boldsymbol{\gamma}}$, then

$$(\mathbb{I} - \mathbf{W}\mathbf{W}^T)\mathbf{z} = \widehat{\boldsymbol{\gamma}} = \mathbf{R}\widehat{\boldsymbol{\beta}}.$$

Let $\overline{\mathbf{Q}}$ be an orthogonal complement of $\mathbf{Q}$ to matrix with dimensions $n \times n$. The residual sum of squares for the model with linear constraints (3.4) can now be written as

$$\begin{aligned} RSS_M &= \|\overline{\mathbf{Q}}^T y\|^2 + \|\mathbf{Q}^T(\mathbf{y} - \mathbf{X}\widehat{\boldsymbol{\beta}}_M)\|^2 = \|\mathbf{y}\|^2 - \|\mathbf{z}\|^2 + \|\mathbf{Q}^T\mathbf{y} - \mathbf{R}\widehat{\boldsymbol{\beta}}_M\|^2 \\ &= \|\mathbf{y}\|^2 - \|\mathbf{z}\|^2 + \|\mathbf{W}\mathbf{W}^T\mathbf{z}\|^2 = \|\mathbf{y}\|^2 - \|\mathbf{z}\|^2 + \|\mathbf{W}^T\mathbf{z}\|^2 \\ &= \|\mathbf{y}\|^2 - \|\mathbf{z}\|^2 + \sum_{m=1}^{p-q}(\mathbf{w}_m^T\mathbf{z})^2, \end{aligned} \tag{3.7}$$

where $\mathbf{w}_m$ is the m-th column of $\mathbf{W}$.

Denote by $(\mathbf{A}_0)_{m,p}, \mathbf{S}_{m,p}, \mathbf{W}_{m,p}$ and $\mathbf{U}_{m,p}$ sub-matrices of $\mathbf{A}_0, \mathbf{S}, \mathbf{W}$ and $\mathbf{U}$ respectively, obtained by retaining first m rows and p columns. Let us consider a nested family of feasible models M_m, $m = 0, \ldots, p-q$ defined as

$$\mathcal{L}_{M_m} = \{\boldsymbol{\beta} \in \mathbb{R}^p, (\mathbf{A}_0)_{m,p}\boldsymbol{\beta} = \mathbf{0}\}.$$

For $m = 0, \ldots, p-q$ we have

$$\mathbf{S}_{p,m} = \mathbf{W}_{p,m}\mathbf{U}_{m,m},$$

because matrix $\mathbf{U}_{m,m}$ is upper triangular. Since $\mathbf{W}_{p,m}^T\mathbf{W}_{p,m} = \mathbb{I}$, then $\mathbf{W}_{p,m}\mathbf{U}_{m,m}$ is QR decomposition of $\mathbf{S}_{p,m}$. Then from equation 3.7 we get a recursive formula for residual sum of squares for nested models:

$$\begin{aligned} RSS_{M_0} &= \|\mathbf{y}\|^2 - \|\mathbf{z}\|^2, \\ RSS_{M_m} &= RSS_{M_{m-1}} + (\mathbf{w}_m^T\mathbf{z})^2 \text{ for } m = 1, \ldots, p-1. \end{aligned} \tag{3.8}$$

3.2 DMRnet4lm - Delete or Merge Regressors algorithm for linear model and high-dimensional data

DMR4lm algorithm does not work for $p > n$. We propose the DMRnet4lm algorithm by adding to DMR4lm a screening step so that it could be applied to high-dimensional data. The screening step is done using group lasso. After reduction of the dimension of the model to $p < n$, DMR4lm algorithm is used. In order to make the screening step more accurate and to better balance the impact of screening and the DMR4lm selection steps, the screening is done multiple times.

We rewrite the group lasso estimator, defined in equation 2.18:

$$\widetilde{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \left\{ \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2 + 2\lambda \left(\sum_{j=1}^{p_0} \nu(\beta_{j0}) + \sum_{k=1}^{l} \nu(\boldsymbol{\beta}_k) \right) \right\},$$

where:

$$\nu(\beta_{j0}) = |\beta_{j0}| \text{ for continuous variables,}$$
$$\nu(\boldsymbol{\beta}_k) = \sqrt{p_k - 1}\|\boldsymbol{\beta}_k\| \text{ for factors.}$$

Algorithm 2 `DMRnet4lm`: Delete or Merge Regressors algorithm for linear regression and high-dimensional data

Input: $\mathbf{y}$, $\mathbf{X}$, $(o, \lambda_1 < \ldots < \lambda_m)$ and $p^- = \min(p, n/2)$.
1. Solve the group lasso problem for grid: $\lambda_1, \ldots, \lambda_m$.
for $q = 1$ **to** m **do**
$\widehat{\beta}^{(q)} = \operatorname{argmin}_\beta \left\{ \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2 + 2\lambda_q \left(\sum_{j=1}^{p_0} \nu(\beta_{j0}) + \sum_{k=1}^{l} \nu(\boldsymbol{\beta}_k) \right) \right\}$.
Sort nonzero elements of $\{\nu(\beta_{10}), \ldots, \nu(\beta_{p_0 0}), \nu(\boldsymbol{\beta}_1), \ldots, \nu(\boldsymbol{\beta}_l)\}$ in non-increasing order, corresponding to s_{q0} continuous variables and $s_q - s_{q0}$ factors.
Select model with DMR4lm using the reduced model matrix.
for $t = 1$ **to** o **do**
$J = \{j_1, \ldots, j_{s_{q0t}}, j_{s_{q0t+1}}, \ldots, j_{s_{qt}}\}$, where $s_{qt} = \lfloor s_q \cdot t/o \rfloor$.
$\mathscr{M}_{qt} = \mathscr{M}^{DMR4lm}(\mathbf{X}^J, \mathbf{y})$ and $\mathbf{RSS}_{qt} = \mathbf{RSS}^{DMR4lm}(\mathbf{X}^J, \mathbf{y})$, where $\mathbf{X}^J$ is model matrix constructed from $j_1, \ldots, j_{s_{q0t}}$-th columns from $\mathbf{X}_0$ and $\mathbf{X}_{j_{s_{q0t+1}}}, \ldots, \mathbf{X}_{j_{s_{qt}}}$.
end for
end for
2. Choose the best models with 1 ***to*** p^- ***parameters that has minimal RSS.***
for $m = 1$ **to** p^- **do**
$M_m = \arg\min\{RSS_M : M \in \bigcup_{qt} \mathscr{M}_{qt},\ |M| = m\}$.
end for
Output: $\mathscr{M}^{DMRnet4lm} = \{\mathscr{M}_1, \ldots, \mathscr{M}_{p^-}\}$, $\mathbf{RSS}^{DMRnet4lm} = (RSS_{M_1}, \ldots, RSS_{M_{p^-}})^T$.

Remark 1. Let us notice that:

- DMRnet4lm improves the group lasso estimator by merging levels of factors.
- After solving the group lasso problem, model matrices $\mathbf{X}^J$ are built composed of the continuous variables and factors corresponding to the highest values of $\nu(\beta_{j0})$ and $\nu(\boldsymbol{\beta}_k)$. For example if $o = 5$ we will choose variables with $\frac{1}{5}$, $\frac{2}{5}$, $\frac{3}{5}$, $\frac{4}{5}$, and all variables with highest ν.
- DMRnet4lm returns p^- models with dimensions from 1 to p^-. The choice of the final model can be based for example on cross-validation or information criterion.

3.3 Bound for selection error of the DMR4lm algorithm

In this section the theorems about error bounds for selection error, consistency of model selection and asymptotic normality of the estimators are given for DMR4lm if we assume that the final model is chosen according to the GIC defined in equation 2.16, so we assume that σ^2 is unknown. The proof for GIC defined in equation 2.17 is a special case of the proof given in Chapter 4.

In Algorithm 1 and all the simulations and examples we assumed complete linkage in hierarchical clustering. The proof of consistency is more general: the linkage criterion has to be a convex combination of the minimum and maximum of the pairwise distances between clusters (see equation 3.14)

We distinguish the following subsets of the set of all feasible models $\mathcal{M}$:

1. Uniquely defined model T, which is fixed and does not depend on the sample size. We assume that the model consists of a finite number of continuous variables and a finite number of factors with finite numbers of levels.

2. A set $\mathcal{M}_{\mathcal{V}}^{F}$ of models obtained by imposing one false constraint on the full model:
$$\mathcal{M}_{\mathcal{V}}^{F} = \{M \subseteq F : |M| = |F| - 1 \text{ and } T \nsubseteq M\}.$$

3. A set $\mathcal{M}_{\mathcal{T}}^{F}$ of models obtained by imposing one true constraint on the full model:
$$\mathcal{M}_{\mathcal{T}}^{F} = \{M \subseteq F : |M| = |F| - 1 \text{ and } T \subseteq M\}.$$

4. A set $\mathcal{M}_{\mathcal{V}}^{T}$ of models obtained by imposing one false constraint on the true model:
$$\mathcal{M}_{\mathcal{V}}^{T} = \{M \subseteq T : |M| = |T| - 1\}.$$

We denote:
$$\delta_F = \min_{M \in \mathcal{M}_{\mathcal{V}}^{F}} \Delta_M, \tag{3.9}$$
$$\delta_T = \min_{M \in \mathcal{M}_{\mathcal{V}}^{T}} \Delta_M, \tag{3.10}$$
where δ_M was defined in equation 2.15.

Let $\widehat{T}$ be the model selected by DMR4lm, where linkage criterion for hierarchical clustering is a convex combination of minimum and maximum of the pairwise distances between clusters and the final model is chosen by GIC defined in equation 2.16.

Selection consistency. We denote the path of nested models from step 4 of the DMR4lm algorithm as $\mathscr{M} = \{M_0, \ldots, M_{p-1}\}$ and decompose selection error into three parts:
$$\{\widehat{T} \neq T\} = \{T \notin \mathscr{M}\} \cup \{T \in \mathscr{M}, \widehat{T} \subset T\} \cup \{T \in \mathscr{M}, \widehat{T} \supset T\}.$$
For sequences f_n and g_n we denote $f_n \prec g_n$ if $f_n = o(g_n)$, so $f_n/g_n \to 0$.

Theorem 1. *Assume that* $\mathbf{X}$ *is full rank. Then*

(i) $\mathbb{P}(T \notin \mathscr{M}) \leq \exp\left(-\frac{\delta_F}{12\sigma^2}(1 + o_1)\right)$,

(ii) $\mathbb{P}(T \in \mathscr{M}, \widehat{T} \subset T) \leq \exp\left(-\frac{(\delta_T/\sigma^2) \wedge n}{12}(1 + o_2)\right)$ *if* $r < \frac{n}{t}\log\left(\frac{\delta_T}{10n\sigma^2} + 1\right)$,

(iii) $\mathbb{P}(T \in \mathscr{M}, \widehat{T} \supset T) \leq \exp\left(-\frac{r}{2}(1 + o_3)\right)$ *if* $\frac{n(p-t+2)}{n-p-2} < r$,

where o_1, o_2, o_3 *are expressions defined in the proof.*

An important consequence of Theorem 1 is the following corollary.

Corollary 1 (Selection error bound). *Assuming that* $\mathbf{X}$ *is full rank and* $p \prec r \prec \delta_F \wedge \left(\frac{n \wedge \delta_T}{t}\right)$ *DMR4lm is a consistent model selection method:*
$$\mathbb{P}(\widehat{T} \neq T) \leq \exp\left(\frac{r}{2}(1 + o(1))\right).$$

Let us notice that for Bayesian Information Criterion is consistent if $p \prec \log(n)$.

Asymptotic normality. Let us notice that from equation 2.14 we get

$$\mathrm{Var}\left(\widehat{\boldsymbol{\beta}}_M\right) = \mathbf{A}_M^1 \mathrm{Var}\left(\widehat{\boldsymbol{\xi}}_M\right) \mathbf{A}_M^{1T} = \mathbf{A}_M^1 \left(\mathbf{A}_M^{1T}\mathbf{X}^T\mathbf{X}\mathbf{A}_M^1\right)^{-1} \mathbf{A}_M^{1T}\sigma^2.$$

Then

$$\mathrm{Var}\left(\sqrt{n}\left(\widehat{\boldsymbol{\beta}}_M - \boldsymbol{\beta}^*\right)\right) = n\mathbf{A}_M^1 \left(\mathbf{A}_M^{1T}\mathbf{X}^T\mathbf{X}\mathbf{A}_M^1\right)^{-1} \mathbf{A}_M^{1T}\sigma^2.$$

Additionally, for finite p, independent of n, if $\frac{1}{n}\mathbf{X}^T\mathbf{X} \to \boldsymbol{\Sigma} > 0$ then

$$\mathrm{Var}\left(\sqrt{n}\left(\widehat{\boldsymbol{\beta}}_M - \boldsymbol{\beta}^*\right)\right) \to \boldsymbol{\Sigma}_M = \mathbf{A}_M^1 \left(\mathbf{A}_M^{1T}\boldsymbol{\Sigma}\mathbf{A}_M^1\right)^{-1} \mathbf{A}_M^{1T}\sigma^2.$$

Theorem 2. *Assuming $t \prec p \prec r \prec \delta_F \wedge n \wedge (\frac{\delta_T}{t})$ and that p is finite and independent of n and $\frac{1}{n}\mathbf{X}^T\mathbf{X} \to \boldsymbol{\Sigma} > 0$,*

$$\sqrt{n}\left(\widehat{\boldsymbol{\beta}}_{\widehat{T}} - \boldsymbol{\beta}^*\right) \xrightarrow{d} \mathcal{N}(\mathbf{0}, \sigma^2\boldsymbol{\Sigma}_T).$$

3.4 Proofs

3.4.1 Residual sums of squares and t-statistics.

For a feasible model M let us define a following orthogonal projection matrix:

$$\overline{\mathbf{H}}_M = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{A}_{0M}^T \left(\mathbf{A}_{0M}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{A}_{0M}^T\right)^{-1} \mathbf{A}_{0M}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T.$$

Lemma 1. *We have*

$$\overline{\mathbf{H}}_M = \mathbf{H}_F - \mathbf{H}_M.$$

Proof. For simplicity of notations in the remainder of this subsection we omit subscript M. Let $\mathbf{Z}_1 = \mathbf{X}\mathbf{A}^1$, $\mathbf{Z} = \mathbf{X}\mathbf{A}^{-1}$ and $\mathbf{Z}_0 = \mathbf{X}\mathbf{A}^0$. We denote

$$\mathbf{G} = \begin{bmatrix} \mathbf{G}_{11} & \mathbf{G}_{10} \\ \mathbf{G}_{01} & \mathbf{G}_{00} \end{bmatrix} = \begin{bmatrix} \mathbf{Z}_1^T\mathbf{Z}_1 & \mathbf{Z}_1^T\mathbf{Z}_0 \\ \mathbf{Z}_0^T\mathbf{Z}_1 & \mathbf{Z}_0^T\mathbf{Z}_0 \end{bmatrix} = \mathbf{Z}^T\mathbf{Z} \text{ and } \mathbf{G}^{-1} = \begin{bmatrix} \mathbf{G}^{11} & \mathbf{G}^{10} \\ \mathbf{G}^{01} & \mathbf{G}^{00} \end{bmatrix}.$$

Note that

$$\mathbf{H}_F = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T = \mathbf{X}\mathbf{A}^{-1}(\mathbf{A}^{-T}\mathbf{X}^T\mathbf{X}\mathbf{A}^{-1})^{-1}\mathbf{A}^{-T}\mathbf{X}^T = \mathbf{Z}(\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T.$$

Moreover

$$\begin{aligned}(\mathbf{A}_0(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{A}_0^T)^{-1} &= \left(\mathbf{A}_0\mathbf{A}^{-1}\left(\mathbf{A}^{-T}\mathbf{X}^T\mathbf{X}\mathbf{A}^{-1}\right)^{-1}\mathbf{A}^{-T}\mathbf{A}_0^T\right)^{-1} \\ &= \left[\begin{bmatrix} \mathbf{0} & \mathbb{I} \end{bmatrix}(\mathbf{Z}^T\mathbf{Z})^{-1}\begin{bmatrix} \mathbf{0} \\ \mathbb{I} \end{bmatrix}\right]^{-1} = (\mathbf{G}^{00})^{-1}\end{aligned}$$

and

$$\mathbf{A}_0(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T = \mathbf{A}_0\mathbf{A}^{-1}(\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{A}^{-T}\mathbf{X}^T = \mathbf{A}_0\mathbf{A}^{-1}(\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T.$$

Then we get from the Schur complement:

$$\begin{aligned}
\mathbf{H}_F - \mathbf{H}_M &= \mathbf{Z}(\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T - \mathbf{Z}_1(\mathbf{Z}_1^T\mathbf{Z}_1)^{-1}\mathbf{Z}_1^T = \mathbf{Z}\mathbf{G}^{-1}\mathbf{Z}^T - \mathbf{Z}_1\mathbf{G}_{11}^{-1}\mathbf{Z}_1^T \\
&= \mathbf{Z}\mathbf{G}^{-1}\mathbf{Z}^T - \mathbf{Z}_1(\mathbf{G}^{11} - \mathbf{G}^{10}(\mathbf{G}^{00})^{-1}\mathbf{G}^{10})\mathbf{Z}_1^T \\
&= \begin{bmatrix} \mathbf{Z}_1 & \mathbf{Z}_0 \end{bmatrix} \begin{bmatrix} \mathbf{G}^{11} & \mathbf{G}^{10} \\ \mathbf{G}^{01} & \mathbf{G}^{00} \end{bmatrix} \begin{bmatrix} \mathbf{Z}_1^T \\ \mathbf{Z}_0^T \end{bmatrix} \\
&\quad - \begin{bmatrix} \mathbf{Z}_1 & \mathbf{Z}_0 \end{bmatrix} \begin{bmatrix} \mathbf{G}^{11} - \mathbf{G}^{10}(\mathbf{G}^{00})^{-1}\mathbf{G}^{10} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{Z}_1^T \\ \mathbf{0}^T \end{bmatrix} \\
&= \mathbf{Z} \begin{bmatrix} \mathbf{G}^{10} \\ \mathbf{G}^{00} \end{bmatrix} (\mathbf{G}^{00})^{-1} \begin{bmatrix} \mathbf{G}^{01} & \mathbf{G}^{00} \end{bmatrix} \mathbf{Z}^T \\
&= \mathbf{Z}(\mathbf{Z}^T\mathbf{Z})^{-1} \begin{bmatrix} \mathbf{0} \\ \mathbb{I} \end{bmatrix} (\mathbf{G}^{00})^{-1} \begin{bmatrix} \mathbf{0} & \mathbb{I} \end{bmatrix} (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T \\
&= \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{A}_M^T \left(\mathbf{A}_M(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{A}_M^T\right)^{-1} \mathbf{A}_M(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T = \overline{\mathbf{H}}_M.
\end{aligned}$$

□

From Lemma 1 we get that

$$RSS_M - RSS_F = \mathbf{y}^T(\mathbf{H}_F - \mathbf{H}_M)\mathbf{y} = \widehat{\boldsymbol{\beta}}^T\mathbf{A}_{0M}^T(\mathbf{A}_{0M}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{A}_{0M}^T)^{-1}\mathbf{A}_{0M}\widehat{\boldsymbol{\beta}},$$

where $\widehat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$. Hence for for each $M \in \mathcal{M}_{\mathcal{T}}^F \cup \mathcal{M}_{\mathcal{V}}^F$

$$\begin{aligned}
t_M^2 &= \frac{(\mathbf{A}_{0M}\widehat{\boldsymbol{\beta}})^2}{\widehat{\mathrm{Var}}(\mathbf{A}_{0M}\widehat{\boldsymbol{\beta}})} = \frac{(\mathbf{A}_{0M}\widehat{\boldsymbol{\beta}})^2}{\mathbf{A}_{0M}\widehat{\mathrm{Var}}(\widehat{\boldsymbol{\beta}})\mathbf{A}_{0M}^T} = \frac{(\mathbf{A}_{0M}\widehat{\boldsymbol{\beta}})^2}{\widehat{\sigma}^2\mathbf{A}_{0M}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{A}_{0M}^T} \\
&= \frac{RSS_M - RSS_F}{\widehat{\sigma}^2},
\end{aligned} \tag{3.11}$$

where $\widehat{\sigma}^2 = \frac{RSS_F}{n-|F|}$. Observe that $\mathbf{A}_{0M}$ is $1 \times |F|$ matrix, thus

$$t_M^2 = (n - |F|)\frac{RSS_M - RSS_F}{RSS_F}. \tag{3.12}$$

3.4.2 Correct ordering of constraints using hierarchical clustering

Temporarily let us limit the analysis to a model consisting of one factor and no continuous variables. The true partition of set $\{1, \ldots, p_1\}$ will be denoted by $P_1^* = (C_{i1}^*)_{i=1}^{|T|}$. We say that distance matrix $\mathbf{D} = [d_{ij}]_{ij}$ is consistent with the true partition if dissimilarities for elements within the same clusters are smaller than for elements from different clusters:

$$\max_{l\in\{1,\ldots,|T|\}} \max_{i,j\in C_{l1}^*} d_{ij} = d^{true} < d^{false} = \min_{\substack{l_1,l_2\in\{1,\ldots,|T|\} \\ l_1\neq l_2}} \min_{i\in C_{l_1 1}^*, j\in C_{l_2 1}^*} d_{ij}. \tag{3.13}$$

Let $P_{s1} = (C_{is1})_{i=1}^{p_1-s+1}$ denote a partition of set $\{1, \ldots, p_1\}$ in step s of hierarchical clustering algorithm, $s = 1, \ldots, p_1$. We will name aggregation of $C_{i_s s1}$ and $C_{j_s s1}$ in step s compatible with the true partition P_1^* if there exist $l \in \{1, \ldots, |T|\}$, $i_{s+1} \in \{1, \ldots, p_1 - s\}$ and $i_s \neq j_s$, $i_s, j_s \in \{1, \ldots p_1 - s + 1\}$ such that

$$C_{i_{s+1}s+11} = C_{i_s s1} \cup C_{j_s s1} \ , \ C_{i_{s+1}s+11} \subseteq C_{l1}^*.$$

Cutting height in step s is defined as $h_{s1} = d(C_{i_s s1}, C_{j_s s1})$ if $C_{i_s s1}$ and $C_{j_s s1}$ are aggregated in this step, $\mathbf{h}_1 = (h_{11}, \ldots, h_{p_1-1,1})$.

Lemma 2. *Assuming that the linkage criterion of hierarchical clustering algorithm satisfies:*

$$\begin{aligned} & d\left(C_{i_{s+1}s+1k} = C_{i_s sk} \cup C_{j_s sk}, C_{j_{s+1}s+1k} = C_{o_s sk}\right) \\ & = b \min\left\{d\left(C_{i_s sk}, C_{o_s sk}\right), d\left(C_{j_s sk}, C_{o_s sk}\right)\right\} \\ & + (1-b) \max\left\{d\left(C_{i_s sk}, C_{o_s sk}\right), d\left(C_{j_s sk}, C_{o_s sk}\right)\right\}, \end{aligned} \tag{3.14}$$

where $b \in [0,1]$ and the dissimilarity matrix has property (3.13), then the cutting heights for aggregations compatible with P_1^* are lower than d^{true} and cutting heights for aggregations not compatible with P_1^* are larger than d^{false}.

Proof. From (3.13) if $|T| = p_1$ the statement holds trivially and if $|T| < p_1$ aggregation in the first step is compatible with P_1^*. We assume that in step s aggregation is compatible with the true partition with cutting height not greater than d^{true}. If aggregation of $C_{i_{s+1}s+1,1} = C_{i_s s1} \cup C_{j_s s1}$ and $C_{j_{s+1}s+1,1} = C_{o_s s1}$ is compatible with P_1^* then

$$h_{s1} = d\left(C_{i_{s+1}s+11}, C_{j_{s+1}s+11}\right) \leq \max\left(d\left(C_{i_s s1}, C_{o_s s1}\right), d\left(C_{j_s s1}, C_{o_s s1}\right)\right) \leq d^{true}$$

If aggregation of $C_{i_{s+1}s+11} = C_{i_s s1} \cup C_{j_s s1}$ and $C_{j_{s+1}s+11} = C_{o_s s1}$ is not compatible with P_1^* then

$$h_{s1} = d\left(C_{i_{s+1}s+11}, C_{j_{s+1}s+11}\right) \geq \min\left(d\left(C_{i_s s1}, C_{o_s s1}\right), d\left(C_{j_s s1}, C_{o_s s1}\right)\right) \geq d^{false}$$

Hence, cutting heights $h_{11}, \ldots, h_{p_1-|T|,1}$ not greater than d^{true} are used until all aggregations compatible with P_1^* are performed. We have $C_{p_1-|T|+11} = P_1^*$ and in steps $s = p_1 - |T| + 2, \ldots, p_1$ the true partition P_1^* is a sub-partition of C_{s1} and cutting heights $h_{p_1-|T|+11}, \ldots, h_{p_1-11}$ are not less than d^{false}. □

Note that linkage criteria: single, complete and average satisfy assumption (3.14).

3.4.3 Tails of the Chi-squared and the Beta distributions.

Lemma 3 (Tails of the Chi squared distribution, Inglot and Ledwina [2006]). *Let $X \sim \chi_k^2$. Then for $k = 1$ and $x > 0$:*

$$w_{xk} l_{xk} \leq \mathbb{P}(X \geq x) \leq w_{xk},$$

where $w_{xk} = \exp\left(-\frac{x}{2}\right)\left(\frac{x}{2}\right)^{\frac{k}{2}-1}\Gamma^{-1}\left(\frac{k}{2}\right)$ and $l_{xk} = \frac{x}{x-k+2}$.

Let $k > 1$,$x > k-2$. Then:

$$w_{xk} \leq \mathbb{P}(X \geq x) \leq w_{xk} l_{xk}.$$

Lemma 4 (Stirling's formula for the gamma function, Jameson [2015].). *Let $w_x = x^{x-\frac{1}{2}}\exp(-x)(2\pi)^{\frac{1}{2}}$. Then for all $x > 0$,*

$$w_x \leq \Gamma(x) \leq w_x \exp\left(\frac{1}{12x}\right).$$

Lemma 5 (Tails of the Chi squared distribution, special case.). *Let $X \sim \chi_k^2$ and $x > k-2$. Then we have:*

(i)

$$\mathbb{P}(X \geq x) \leq \exp\left[-\frac{x}{2}(1 - o_\chi(x,k))\right],$$

where $o_\chi(x,k) = \frac{k}{x} - \frac{k-2}{x}\log\left(\frac{x}{k}\right) - \frac{1}{x}\log(k\pi) + \frac{2k}{x(x-k+2)}$ *for* $k \geq 2$ *and* $o_\chi(x,1) = 0$.

If we additionally assume that $k \prec x$, *then* $o_\chi(x,k) \prec 1$.

(ii) For $x = (1+a)k$, $k \geq 2$ *and* $a > 0$ *we have*

$$\mathbb{P}(X \geq x) \leq \exp\left[-\frac{k}{2}(a - \log(1+a))(1 - \tilde{o}_\chi(k,a))\right],$$

where $\tilde{o}_\chi(k,a) = \frac{2}{(a-\log(1+a))(ak+2)}$.

If we additionally assume that $1 \prec k$, *then* $\tilde{o}_\chi(k,a) \prec 1$.

Proof. The proof for $k = 1$ is easy. Here we assume $k \geq 2$. From Lemma 3 we have

$$\mathbb{P}(X \geq x) \leq \exp\left(-\frac{x}{2}\right)\left(\frac{x}{2}\right)^{\frac{k}{2}-1}\Gamma^{-1}\left(\frac{k}{2}\right)\frac{x}{x-k+2}$$

then from Lemma 4

$$\leq \exp\left(-\frac{x}{2}\right)\left(\frac{x}{2}\right)^{\frac{k}{2}-1}\left(\frac{k}{2}\right)^{-\frac{k}{2}+\frac{1}{2}}\exp\left(\frac{k}{2}\right)(2\pi)^{-\frac{1}{2}}\frac{x}{x-k+2}$$

$$= \exp\left(-\frac{x}{2}+\frac{k}{2}\right)\left(\frac{x}{k}\right)^{\frac{k}{2}-1}\left(\frac{1}{k\pi}\right)^{\frac{1}{2}}\left(1+\frac{k-2}{x-k+2}\right)$$

$$\leq \exp\left[-\frac{x}{2}\left(1-\frac{k}{x}+\frac{k-2}{x}\log\left(\frac{x}{k}\right)+\frac{1}{x}\log(k\pi)-\frac{2k}{x(x-k+2)}\right)\right]$$

$$= \exp\left[-\frac{x}{2}\left(1 - o_\chi(x,k)\right)\right].$$

If $x = (1+a)k$, we obtain:

$$\mathbb{P}(X \geq x)$$

$$\leq \exp\left[-\frac{(1+a)k}{2}\left(1-\frac{1}{1+a}+\frac{k-2}{(1+a)k}\log\left(\frac{1}{1+a}\right)+\frac{1}{(1+a)k}\log(k\pi)-\frac{2}{(1+a)(ak+2)}\right)\right]$$

$$= \exp\left[-\frac{k}{2}\left(1+a-1-\frac{k-2}{k}\log(1+a)+\frac{1}{k}\log(k\pi)-\frac{2}{ak+2}\right)\right]$$

$$\leq \exp\left[-\frac{(a-\log(1+a))k}{2}\left(1-\frac{2}{(a-\log(1+a))(ak+2)}\right)\right]$$

$$= \exp\left[-\frac{k}{2}\left(a-\log(1+a)\right)\left(1-\tilde{o}_\chi(k,a)\right)\right]$$

□

Lemma 6 (Tails of the Beta distribution, Pokarowski and Mielniczuk [2010]). *Let $a < 1$ and $x > \frac{a-1}{a+b}$ and $X \sim \mathcal{B}(a,b)$. Then*

$$w_{xab} l_{xab} \leq \mathbb{P}(X \geq x) \leq w_{xab}, \tag{3.15}$$

where $w_{xab} = \frac{(1-x)^b x^{a-1}}{B(a,b)b}$ and $l_{xab} = \frac{(b+1)x}{1-a+(a+b)x}$, where B denotes the Beta function.

Let $a \geq 1$ and $x > \frac{a-1}{a+b}$. Then:

$$w_{xab} \leq \mathbb{P}(X \geq x) \leq w_{xab} l_{xab}. \tag{3.16}$$

Proof. We will prove only equalities 3.16, inequalities in 3.15 can be proved analogously. Let $B_x(a,b) = \int_0^x t^{a-1}(1-t)^{b-1}dt$ be the incomplete Beta function. Let us notice that $B(a,b) = B_1(a,b)$ and $P(X < x) = B_x(a,b)/B(a,b)$. In the proof the following equality will be used:

$$aB_x(a,b) = x^a(1-x)^b + (a+b)B_x(a+1,b). \tag{3.17}$$

In order to prove it, we rewrite it using integrals:

$$a\int_0^x t^{a-1}(1-t)^{b-1}dt = \int_0^x \frac{d}{dt}\left(t^a(1-t)^b\right)dt + (a+b)\int_0^x t^a(1-t)^{b-1}dt.$$

Now, the equality can be proved by differentiating both sides of the equality with respect to x. The second useful equation is:

$$B_{1-x}(b,a) = B(a,b) - B_x(a,b), \tag{3.18}$$

which can be easily proved using integration by substitution.

Using equation 3.17 and assumption $x > \frac{a-1}{a+b}$ we have the following inequalities for $a \geq 1$:

$$\begin{aligned} bB_{1-x}(b,a) &= (1-x)^b x^a \left[1 + \frac{a+b}{b+1}(1-x) + \frac{(a+b)(a+b+1)}{(b+1)(b+2)}(1-x)^2 + \ldots\right] \\ &\leq (1-x)^b x^a \left[1 + \frac{a+b}{b+1}(1-x) + \left(\frac{a+b}{b+1}\right)^2(1-x)^2 + \ldots\right] \\ &= (1-x)^b x^a \sum_{n=0}^{\infty}\left[\frac{(a+b)(1-x)}{(b+1)}\right]^n = \frac{(b+1)(1-x)^b x^a}{1-a+(a+b)x}. \end{aligned}$$

Using the above inequality and equation 3.18 we get:

$$\mathbb{P}(X > x) = 1 - \frac{B_x(a,b)}{B(a,b)} = \frac{B_{1-x}(a,b)}{B(a,b)} \leq \frac{(1-x)^b x^{a-1}}{B(a,b)b}\frac{(b+1)x}{1-a+(a+b)x} = w_{xab} l_{xab}.$$

Using again equation 3.17 and assumption $x > \frac{a-1}{a+b}$ we have the following inequalities for $a \geq 1$:

$$\begin{aligned} bB_{1-x}(a,b) &= (1-x)^b x^a \left[1 + \frac{a+b}{b+1}(1-x) + \frac{(a+b)(a+b+1)}{(b+1)(b+2)}(1-x)^2 + \ldots\right] \\ &\geq (1-x)^b x^a \left[1 + (1-x) + (1-x)^2 + \ldots\right] = (1-x)^b x^{a-1}. \end{aligned}$$

Using the above inequality and equation 3.18 we get:

$$\mathbb{P}(X > x) = 1 - \frac{B_x(a,b)}{B(a,b)} \geq \frac{(1-x)^b x^{a-1}}{B(a,b)b} = w_{xab}.$$

□

Lemma 7 (Tails of the Beta distribution, special case.). *Let $X \sim \mathcal{B}(a,b)$, $a \geq 1$ and $x > \frac{a-1}{a+b}$. Then*

$$\mathbb{P}(X > x) \leq \exp\left[-bx(1 - o_{\mathcal{B}}(a,b,x))\right],$$

where $o_{\mathcal{B}}(a,b,x) = -\frac{a-1}{bx}\log\left(\frac{a}{(a+b)x}\right) + \frac{(b+\frac{1}{2})a}{b^2x} + \frac{1}{12bx(a+b)} + \frac{a}{bx(1-a+(a+b)x)}$.

If we additionally assume that $a \prec bx$, we have $o_{\mathcal{B}}(a,b,x) \prec 1$.

Proof. Using Lemma 4 we get:

$$B^{-1}(a,b) \leq \frac{(a+b)^{a+b-\frac{1}{2}}\exp(-a-b)(2\pi)^{\frac{1}{2}}\exp\left(\frac{1}{12(a+b)}\right)}{a^{a-\frac{1}{2}}b^{b-\frac{1}{2}}\exp(-a-b)2\pi}.$$

From Lemma 6 we have:

$$\mathbb{P}(X > x) \leq \frac{(1-x)^b x^{a-1}}{B(a,b)b}\left(1 + \frac{(a-1)(1-x)}{1-a+(a+b)x}\right)$$

$$\leq (1-x)^b\left(\frac{(a+b)x}{a}\right)^{a-1}\left(1+\frac{a}{b}\right)^{b+\frac{1}{2}}\exp\left(\frac{1}{12(a+b)}\right)\left(1+\frac{(a-1)(1-x)}{1-a+(a+b)x}\right)$$

$$= \exp\left[b\log(1-x) + (a-1)\log\left(\left(\frac{a+b}{a}\right)x\right) + \left(b+\frac{1}{2}\right)\log\left(1+\frac{a}{b}\right)\right]$$

$$\cdot\exp\left[\frac{1}{12(a+b)} + \log\left(1+\frac{(a-1)(1-x)}{1-a+(a+b)x}\right)\right]$$

$$\leq \exp\left[-bx\left(1 + \frac{a-1}{bx}\log\left(\frac{a}{(a+b)x}\right) - \frac{(b+\frac{1}{2})a}{b^2x} - \frac{1}{12bx(a+b)} - \frac{a}{bx(1-a+(a+b)x)}\right)\right]$$

□

3.4.4 Proof of Theorem 1(i).

Lemma 8.

$$\mathbb{P}(T \notin \mathscr{M}) = \mathbb{P}\{\exists_{M_1 \in \mathcal{M}_{\mathcal{T}}^F}\exists_{M_2 \in \mathcal{M}_{\mathcal{V}}^F}\ t_{M_1}^2 \geq t_{M_2}^2\},$$

where t_{M_1} and t_{M_2} are t-statistics for hypothesis accepted in M_1 and M_2 (see equation 3.11), respectively.

Proof. Let us observe that:

$$\left\{\forall_{M_1 \in \mathcal{M}_{\mathcal{T}}^F}\forall_{M_2 \in \mathcal{M}_{\mathcal{V}}^F}\ t_{M_1}^2 < t_{M_2}^2\right\} = \left\{\exists_{h^*}\max_{M_1 \in \mathcal{M}_{\mathcal{T}}^F} t_{M_1}^2 < h^* < \min_{M_2 \in \mathcal{M}_{\mathcal{V}}^F} t_{M_2}^2\right\} = \{T \in \mathscr{M}\}.$$

The last equality follows because dissimilarity matrices used in the algorithm are consistent with the partitions for model T. Then, applying Lemma 2 for each factor, we get that the cutting heights for aggregations compatible with the true partitions are not greater than h_* and for incompatible ones not smaller than h_*. Hence, in the DMR4lm algorithm accepting true constraints precede accepting false ones. □

We can decompose RSS_M into:

$$RSS_M = \boldsymbol{\beta}^{*T}\mathbf{X}^T(\mathbb{I}-\mathbf{H}_M)\mathbf{X}\boldsymbol{\beta}^* + 2\boldsymbol{\beta}^{*T}\mathbf{X}^T(\mathbb{I}-\mathbf{H}_M)\boldsymbol{\varepsilon} + \boldsymbol{\varepsilon}^T(\mathbb{I}-\mathbf{H}_M)\boldsymbol{\varepsilon}. \tag{3.19}$$

Let $M_1 \in \mathcal{M}_{\mathcal{T}}^F$ and $M_2 \in \mathcal{M}_{\mathcal{V}}^F$. When $T \subseteq M$ we have $\mathbf{H}_M\mathbf{X}\boldsymbol{\beta}^* = \mathbf{X}\boldsymbol{\beta}^*$ and $RSS_{M_1} = \boldsymbol{\varepsilon}^T(\mathbb{I}-\mathbf{H}_M)\boldsymbol{\varepsilon}$. Let $\sigma^2 W_{M_1M_2} = \boldsymbol{\varepsilon}^T(\mathbf{H}_{M_1}-\mathbf{H}_{M_1\cap M_2})\boldsymbol{\varepsilon}$, $\sigma^2 W_{M_2M_1} = \boldsymbol{\varepsilon}^T(\mathbf{H}_{M_2}-\mathbf{H}_{M_1\cap M_2})\boldsymbol{\varepsilon}$ and $Z_{M_2} = \frac{\boldsymbol{\beta}^{*T}\mathbf{X}^T(\mathbb{I}-\mathbf{H}_{M_2})\boldsymbol{\varepsilon}}{\sigma\sqrt{\delta_{M_2}}}$. Then we know that $W_{M_1M_2} \geq 0$, $W_{M_2M_1} \sim \chi^2_1$ and $Z_{M_2} \sim \mathbb{N}(0,1)$.

$$\begin{aligned}
&RSS_{M_2} - RSS_{M_1} \\
&= \boldsymbol{\beta}^{*T}\mathbf{X}^T(\mathbb{I}-\mathbf{H}_{M_2})\mathbf{X}\boldsymbol{\beta}^* + 2\boldsymbol{\beta}^{*T}\mathbf{X}^T(\mathbb{I}-\mathbf{H}_{M_2})\boldsymbol{\varepsilon} + \boldsymbol{\varepsilon}^T(\mathbb{I}-\mathbf{H}_{M_2})\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^T(\mathbb{I}-\mathbf{H}_{M_1})\boldsymbol{\varepsilon} \\
&= \Delta_{M_2} + 2\sqrt{\Delta_{M_2}}\sigma Z_{M_2} - \sigma^2 W_{M_2M_1} + \sigma^2 W_{M_1M_2} \\
&\geq \Delta_{M_2}\left(1 + \frac{2\sigma Z_{M_2}}{\sqrt{\Delta_{M_2}}} - \frac{\sigma^2 W_{M_2M_1}}{\Delta_{M_2}}\right).
\end{aligned}$$

From Lemma 8 and equation 3.12 we have:

$$\begin{aligned}
&\mathbb{P}(T \notin \mathscr{M}) = \mathbb{P}(\exists_{M_1\in\mathcal{M}_{\mathcal{T}}^F}\exists_{M_2\in\mathcal{M}_{\mathcal{V}}^F}\ t^2_{M_1} \geq t^2_{M_2}) \\
&\leq \sum_{M_1\in\mathcal{M}_{\mathcal{T}}^F}\sum_{M_2\in\mathcal{M}_{\mathcal{V}}^F}\mathbb{P}(t^2_{M_1} \geq t^2_{M_2}) = \sum_{M_1\in\mathcal{M}_{\mathcal{T}}^F}\sum_{M_2\in\mathcal{M}_{\mathcal{V}}^F}\mathbb{P}(RSS_{M_1} \geq RSS_{M_2}) \\
&\leq \sum_{M_1\in\mathcal{M}_{\mathcal{T}}^F}\sum_{M_2\in\mathcal{M}_{\mathcal{V}}^F}\mathbb{P}\left(-\frac{2\sigma Z_{M_2}}{\sqrt{\Delta_{M_2}}} + \frac{\sigma^2 W_{M_2M_1}}{\Delta_{M_2}} \geq 1\right) \\
&\leq \sum_{M_1\in\mathcal{M}_{\mathcal{T}}^F}\sum_{M_2\in\mathcal{M}_{\mathcal{V}}^F}\left(\mathbb{P}\left(-\frac{2\sigma Z_{M_2}}{\sqrt{\Delta_{M_2}}} \geq c\right) + \mathbb{P}\left(\frac{\sigma^2 W_{M_2M_1}}{\Delta_{M_2}} \geq 1-c\right)\right) \\
&\leq |\mathcal{M}_{\mathcal{T}}^F||\mathcal{M}_{\mathcal{V}}^F|\left(\mathbb{P}\left(Z^2 \geq \frac{c^2\delta_F}{4\sigma^2}\right) + \mathbb{P}\left(W \geq \frac{(1-c)\delta_F}{\sigma^2}\right)\right)
\end{aligned}$$

choosing c such that $\frac{c^2}{4} = 1-c$ ($c \approx 0.83$), because $1-c > \frac{1}{6}$ and $|\mathcal{M}_{\mathcal{T}}^F||\mathcal{M}_{\mathcal{V}}^F| \leq |M_{\mathcal{T}}^F \cup \mathcal{M}_{\mathcal{V}}^F|^2 \leq \frac{p^2(p+1)^2}{4}$, we get from Lemma 5:

$$\leq \frac{p^2(p+1)^2}{4}\frac{3}{2}\exp\left[-\frac{\delta_F}{12\sigma^2}\left(1 - o_\chi\left(\frac{\delta_F}{12\sigma^2}, 1\right)\right)\right].$$

Finally, we put $o_1 = -o_\chi\left(\frac{\delta_F}{12\sigma^2}, 1\right) - \frac{12\sigma^2}{\delta_F}\log\left(\frac{3p^2(p+1)^2}{8}\right)$.

3.4.5 Proof of Theorem 1(ii).

Let $M \in \mathcal{M}_{\mathcal{V}}^T$, $\sigma^2 W_{MT} = \boldsymbol{\varepsilon}^T(\mathbf{H}_M - \mathbf{H}_{M\cap T})\boldsymbol{\varepsilon}$, $\sigma^2 W_{TM} = \boldsymbol{\varepsilon}^T(\mathbf{H}_T - \mathbf{H}_{M\cap T})\boldsymbol{\varepsilon}$ and $\sigma Z_M = \frac{\boldsymbol{\beta}^{*T}\mathbf{X}^T(\mathbb{I}-\mathbf{H}_M)\boldsymbol{\varepsilon}}{\sqrt{\delta_M}}$. Then we know that $W_{MT} = 0$, $W_{TM} \sim \chi_1^2$ and $Z_M \sim \mathbb{N}(0,1)$.

$$\begin{aligned} &RSS_M - RSS_T \\ &= \boldsymbol{\beta}^{*T}\mathbf{X}^T(\mathbb{I}-\mathbf{H}_M)\mathbf{X}\boldsymbol{\beta}^* + 2\boldsymbol{\beta}^{*T}\mathbf{X}^T(\mathbb{I}-\mathbf{H}_M)\boldsymbol{\varepsilon} + \boldsymbol{\varepsilon}^T(\mathbb{I}-\mathbf{H}_M)\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^T(\mathbb{I}-\mathbf{H}_T)\boldsymbol{\varepsilon} \\ &= \Delta_M + 2\sqrt{\Delta_M}\sigma Z_M - \sigma^2 W_{MT} + \sigma^2 W_{TM} \\ &= \Delta_M\left(1 + \frac{2\sigma Z_M}{\sqrt{\Delta_M}} - \frac{\sigma^2 W_{MT}}{\Delta_M}\right). \end{aligned}$$

Let us denote $\tilde{r} = (\exp\left(\frac{tr}{n}\right) - 1)n$. From assumption that $\mathbf{X}$ is full rank we have $\Delta_M > \delta_T > 0$ and we get:

$$\begin{aligned} \left\{\log(RSS_M) - \log(RSS_T) < \frac{tr}{n}\right\} &= \left\{\log\left(1 + \frac{RSS_M - RSS_T}{RSS_T}\right) < \frac{tr}{n}\right\} \\ &= \left\{\frac{RSS_M - RSS_T}{RSS_T} < \exp\left(\frac{tr}{n}\right) - 1\right\} \\ = \left\{RSS_M - RSS_T < \frac{\tilde{r}}{n}RSS_T\right\} &= \left\{1 + \frac{2\sigma Z_M}{\sqrt{\Delta_M}} - \frac{\sigma^2 W_{MT}}{\Delta_M} - \frac{\tilde{r}}{n\Delta_M}RSS_T < 0\right\} \\ &\subset \left\{-\frac{2\sigma Z_M}{\sqrt{\Delta_M}} + \frac{\tilde{r}}{n\Delta_M}RSS_T > 1\right\}. \end{aligned}$$

Let us notice that:

$$\left\{T \in \mathscr{M}, \widehat{T} \subset T\right\} \subseteq \bigcup_{M\subset T}\{GIC_M < GIC_T\} \subseteq \bigcup_{M\in\mathcal{M}_{\mathcal{V}}^T}\left\{\log(RSS_M) - \log(RSS_T) < \frac{tr}{n}\right\}.$$

Hence, reasoning as previously we obtain:

$$\begin{aligned} \mathbb{P}\left(T \in \mathscr{M}, \widehat{T} \subset T\right) &\leq \sum_{M\in\mathcal{M}_{\mathcal{V}}^T}\mathbb{P}\left(-\frac{2\sigma Z_M}{\sqrt{\Delta_M}} + \frac{\tilde{r}}{n\Delta_M}RSS_T > 1\right) \\ &\leq \sum_{M\in\mathcal{M}_{\mathcal{V}}^T}\left(\mathbb{P}\left(-\frac{2\sigma Z_M}{\sqrt{\Delta_M}} \geq c\right) + \mathbb{P}\left(\frac{\tilde{r}}{n\Delta_M}RSS_T \geq 1 - c\right)\right) \\ &\leq |\mathcal{M}_{\mathcal{V}}^T|\left(\frac{1}{2}\mathbb{P}\left(Z^2 \geq \frac{c^2\delta_T}{4\sigma^2}\right) + \mathbb{P}\left(\frac{RSS_T}{\sigma^2} \geq \frac{(1-c)n\delta_T}{\tilde{r}\sigma^2}\right)\right) \\ &\leq \frac{t(t+1)}{2}\left(\frac{1}{2}\mathbb{P}\left(Z^2 \geq \frac{c^2\delta_T}{4\sigma^2}\right) + \mathbb{P}\left(\frac{\|\boldsymbol{\varepsilon}\|^2}{\sigma^2} \geq \frac{(1-c)n\delta_T}{\tilde{r}\sigma^2}\right)\right). \end{aligned}$$

Let a be the solution to equation $1/6 = a - \log(1+a)$, then $a \approx 0.69$. We have from the assumptions of the theorem:

$$\frac{n\delta_T}{\tilde{r}\sigma^2} > 10n \geq \frac{1+a}{1-c}n \approx 9.6n.$$

Hence

$$\frac{(1-c)\delta_T n}{\tilde{r}\sigma^2} \geq (1+a)n$$

and from Lemma 5 we get:

$$\mathbb{P}\left(T \in \mathscr{M}, \widehat{T} \subset T\right) \leq \frac{t(t+1)}{2}\left(\frac{1}{2}\exp\left[-\frac{\delta_T}{12\sigma^2}\left(1-o_\chi\left(\frac{\delta_T}{6},1\right)\right)\right]+\exp\left[-\frac{n}{12}\left(1-\tilde{o}_\chi\left(n,a\right)\right)\right]\right).$$

Finally, we put $o_2 = o_\chi \vee \tilde{o}_\chi + \frac{12}{(\delta_T/\sigma^2)\wedge n}\log\left(t(t+1)3/4\right)$.

3.4.6 Proof of Theorem 1(iii)

First, observe that:

$$\mathbb{P}(T \in \mathscr{M}, \widehat{T} \supset T) \leq \mathbb{P}\left(\log(RSS_F) + \frac{r}{n} \leq \log(RSS_T)\right)$$

$$= \mathbb{P}\left(\log\left(\frac{RSS_F}{RSS_T}\right) \leq -\frac{r}{n}\right) = \mathbb{P}\left(\log\left(1+\frac{RSS_F - RSS_T}{RSS_T}\right) \leq -\frac{r}{n}\right)$$

$$= \mathbb{P}\left(1+\frac{RSS_F - RSS_T}{RSS_T} \leq \exp\left(-\frac{r}{n}\right)\right) = \mathbb{P}\left(\frac{RSS_T - RSS_F}{RSS_T} \geq 1-\exp\left(-\frac{r}{n}\right)\right) = \mathbb{P}\left(D \geq \frac{\tilde{r}}{n}\right),$$

where $D = \frac{RSS_T - RSS_F}{RSS_T}$ has Beta distribution $\mathcal{B}(\frac{p-t}{2}, \frac{n-p}{2})$ and $\frac{\tilde{r}}{n} = 1-\exp\left(-\frac{r}{n}\right)$.

Next, from assumption $\frac{p-2-t}{n-p+2} < \frac{r}{n}$ we have

$$\log\left(1+\frac{p-2-t}{n-p+2}\right) < \frac{r}{n}$$

and thus

$$-\frac{r}{n} < \log\left(\frac{n-p+2}{n-t}\right),$$

which is equivalent to

$$\frac{p-t-2}{n-t} < \frac{\tilde{r}}{n}. \tag{3.20}$$

Hence, if we put $a = \frac{p-t}{2}$, $b = \frac{n-p}{2}$ and $x = \frac{\tilde{r}}{n}$, then from equation 3.20 the assumption of Lemma 7 $x > \frac{a-1}{a+b}$ is fulfilled and we obtain finally the third part of the theorem.

3.4.7 Proof of Theorem 2

Let us denote

$$\mathbf{g}_n = \sqrt{n}\left(\widehat{\boldsymbol{\beta}}_T - \boldsymbol{\beta}^*\right) \text{ and } \mathbf{b}_n = \sqrt{n}\left(\widehat{\boldsymbol{\beta}}_{\widehat{T}} - \boldsymbol{\beta}^*\right),$$

Notice that $\mathbf{g}_n = \mathbf{b}_n$ if $\widehat{T} = T$. From Theorem 1

$$\mathbb{P}\left(\mathbb{1}(\widehat{T} \neq T) = 0\right) \xrightarrow{P} 1.$$

Since

$$\left\{\mathbb{1}(\widehat{T} \neq T) = 0\right\} \subseteq \left\{\mathbf{b}_n \mathbb{1}(\widehat{T} \neq T) = 0\right\},$$

hence $\mathbf{b}_n \mathbb{1}(\widehat{T} \neq T) \xrightarrow{P} 0$. From properties of the OLS estimator we have

$$\mathbf{g}_n \mathbb{1}(\widehat{T} = T) \xrightarrow{d} \mathbb{N}(0, \sigma^2 \mathbf{\Sigma}_T).$$

Henceforth, from multi-dimensional Slutsky's theorem we get

$$\mathbf{b}_n = \mathbf{b}_n \mathbb{1}(\widehat{T} \neq T) + \mathbf{b}_n \mathbb{1}(\widehat{T} = T) = \mathbf{b}_n \mathbb{1}(\widehat{T} \neq T) + \mathbf{g}_n \mathbb{1}(\widehat{T} = T) \xrightarrow{d} \mathbb{N}(0, \sigma^2 \mathbf{\Sigma}_T).$$

Chapter 4

Delete or Merge Regressors algorithm for generalized linear model

In this chapter we introduce DMR4glm algorithm, which works for generalized linear models. The difference between DMR4glm and DMR4lm is that it uses likelihood ratio statistics instead of t-statistics. Furthermore, we present a bound on the selection error of the DMR4glm algorithm which is true even when p tends to infinity with n. Let us observe that if σ^2 is known, DMR4lm with with $GIC = RSS_M + r \cdot |M|$ is a special case of the DMR4glm algorithm.

We also describe DMR4glm_wald algorithm, which uses Wald statistics instead of likelihood ratio test statistics. This algorithm has been introduced in Maj-Kańska et al. [2015]. It has an advantage that it works considerably faster than DMR4glm. However, no proof of consistency is given for this case.

Moreover, in this chapter an algorithm DMRnet4glm, which is a generalization of DMR4glm to high-dimensional data sets is introduced. It is similar to DMRnet, so it uses group lasso in the screening step and DMR4glm after decreasing the column dimension of $\mathbf{X}$ to $p < n$.

Let $\dot{f}, \ddot{f}$ denote the first and the second derivative of function f and $\mathbf{x}_{i\cdot}$ denote i-th row of matrix $\mathbf{X}$. In this chapter, if not otherwise stated, we assume $p \leq n$.

4.1 Generalized linear models

Likelihood function for generalized linear models with canonical link can be written as:

$$\mathscr{L}(\boldsymbol{\beta}) = \exp\{\boldsymbol{\eta}^T \mathbf{y} - \gamma(\boldsymbol{\eta})\}, \ \boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta}.$$

Then the log likelihood function is:

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n} \eta_i y_i - \gamma_0(\eta_i),$$

where γ_0 is the cumulant generating function.

For example, let us define the likelihood function for logistic regression:

$$\mathscr{L}(\boldsymbol{\beta}) = \prod_{i=1}^{n} \left(\frac{\exp(\eta_i)}{1+\exp(\eta_i)} \right)^{y_i} \left(1 - \frac{\exp(\eta_i)}{1+\exp(\eta_i)} \right)^{1-y_i}. \tag{4.1}$$

Then the log-likelihood function is:

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n} y_i \eta_i - \gamma_0(\eta_i),\ \gamma_0(\eta_i) = \log(1 + \exp(\eta_i)).$$

We define:

$$\mathbf{W}_{\boldsymbol{\beta}} = \operatorname{diag}(\ddot{\gamma_0}(\mathbf{x}_{1.}^T\boldsymbol{\beta}), \ldots, \ddot{\gamma_0}(\mathbf{x}_{n.}^T\boldsymbol{\beta})),$$

$$\gamma(\boldsymbol{\beta}) = \sum_{i=1}^{n} \gamma_0(\mathbf{x}_{i.}^T\boldsymbol{\beta}).$$

Let us notice that $\dot{\gamma}(\boldsymbol{\beta}) = \mathbf{X}^T\dot{\gamma_0}(\mathbf{X}\boldsymbol{\beta})$.

For the true parameter vector β^* we have:

$$\mathbb{E}_{\boldsymbol{\beta}^*}(y_i) = \dot{\gamma_0}(\mathbf{x}_{i.}^T\boldsymbol{\beta}^*),\ i = 1 \ldots, n,$$

$$\operatorname{Var}_{\boldsymbol{\beta}^*}(y_i) = \sigma^2\ddot{\gamma_0}(\mathbf{x}_{i.}^T\boldsymbol{\beta}^*),\ i = 1 \ldots, n.$$

We obtain:

$$\mathbf{X}_* = \mathbf{W}_{\boldsymbol{\beta}^*}^{\frac{1}{2}}\mathbf{X},$$

$$\mathbf{H}_* = \mathbf{X}_*(\mathbf{X}_*^T\mathbf{X}_*)^{-1}\mathbf{X}_*^T,$$

$$\boldsymbol{\varepsilon} = \mathbf{W}_{\boldsymbol{\beta}^*}^{\frac{1}{2}}(\mathbf{y} - \mathbb{E}_{\boldsymbol{\beta}^*}(y)).$$

Then $\mathbb{E}_{\boldsymbol{\beta}^*}(\boldsymbol{\varepsilon}) = 0$, $\operatorname{Var}_{\boldsymbol{\beta}^*}(\boldsymbol{\varepsilon}) = \sigma^2\mathbb{I}$.

We can rewrite the log-likelihood function as:

$$\ell(\boldsymbol{\beta}) = \boldsymbol{\beta}^T\mathbf{X}^T\mathbf{y} - \gamma(\boldsymbol{\beta}) = \boldsymbol{\beta}^T\mathbf{X}^T(\dot{\gamma_0}(\mathbf{X}\boldsymbol{\beta}) + \boldsymbol{\varepsilon}) - \gamma(\boldsymbol{\beta})$$

$$= \boldsymbol{\beta}^T\dot{\gamma}(\boldsymbol{\beta}^*) + \boldsymbol{\beta}^T\mathbf{X}_*^T\boldsymbol{\varepsilon} - \gamma(\boldsymbol{\beta}).$$

Then we see that $\dot{\ell}(\boldsymbol{\beta}^*) = \mathbf{X}_*^T\boldsymbol{\varepsilon}$.

Let us define:

$$\dot{\ell}(\boldsymbol{\beta}) = \frac{\partial \ell}{\partial \boldsymbol{\beta}} = \mathbf{X}^T\mathbf{y} - \dot{\gamma}(\boldsymbol{\beta}) = \mathbf{X}^T\left(\mathbf{y} - \dot{\gamma_0}(\mathbf{X}\boldsymbol{\beta})\right),$$

$$-\ddot{\ell}(\boldsymbol{\beta}) = -\frac{\partial \ell^2}{\partial \boldsymbol{\beta}\partial \boldsymbol{\beta}^T} = \ddot{\gamma}(\boldsymbol{\beta}) = \mathbf{X}^T\ddot{\gamma_0}(\mathbf{X}\boldsymbol{\beta})\mathbf{X} = \mathbf{X}^T\mathbf{W}_{\boldsymbol{\beta}}\mathbf{X}.$$

Let us notice that since $\mathbf{X}^T\mathbf{X} > 0$, thus $\mathbf{X}^T\mathbf{W}_{\boldsymbol{\beta}}\mathbf{X} > 0$. We denote the maximum likelihood estimator of $\boldsymbol{\beta}$ as $\widehat{\boldsymbol{\beta}} = \arg\max_{\boldsymbol{\beta}} \ell(\boldsymbol{\beta})$. In order to find it, usually an equation $s(\boldsymbol{\beta}) = 0$ is solved by the Newton-Raphson algorithm:

$$\widehat{\boldsymbol{\beta}}^{(k+1)} = \widehat{\boldsymbol{\beta}}^{(k)} + \left[\ddot{\ell}(\widehat{\boldsymbol{\beta}}^{(k)})\right]^{-1}\dot{\ell}(\widehat{\boldsymbol{\beta}}^{(k)})$$

$$= \widehat{\boldsymbol{\beta}}^{(k)} + \left[\mathbf{X}^T\mathbf{W}_{\widehat{\boldsymbol{\beta}}^{(k)}}\mathbf{X}\right]^{-1}\mathbf{X}^T\left[\mathbf{y} - \dot{\gamma_0}(\mathbf{X}\boldsymbol{\beta}^{(k)})\right]$$

$$= \left[\mathbf{X}^T\mathbf{W}_{\widehat{\boldsymbol{\beta}}^{(k)}}\mathbf{X}\right]^{-1}\mathbf{X}^T\mathbf{W}_{\widehat{\boldsymbol{\beta}}^{(k)}}\left[\mathbf{X}\boldsymbol{\beta}^{(k)} + \mathbf{y} - \dot{\gamma_0}(\mathbf{X}\boldsymbol{\beta}^{(k)})\right].$$

In order to make the steps computationally effective, the Iteratively Related Least Squares algorithm is used.

Algorithm 3 (IRLS) Iteratively Reweighted Least Squares

Input: $\mathbf{y}$, $\mathbf{X}$, ξ
$k = 0$, $\widehat{\boldsymbol{\beta}}^{(k)} = 0$;
while $\|\dot{\ell}(\widehat{\boldsymbol{\beta}}^{(k)})\|^2 > \xi$ **do**
$\mathbf{z}^{(k)} = \mathbf{X}\widehat{\boldsymbol{\beta}}^{(k)} + \mathbf{y} - \dot{\gamma}_0(\mathbf{X}\widehat{\boldsymbol{\beta}}^{(k)})$,
$\mathbf{W}_{\widehat{\boldsymbol{\beta}}^{(k)}} = \mathrm{diag}(\ddot{\gamma}_0(\mathbf{x}_{1.}^T\widehat{\boldsymbol{\beta}}^{(k)}), \ldots, \ddot{\gamma}_0(\mathbf{x}_{n.}^T\widehat{\boldsymbol{\beta}}^{(k)}))$
$\mathbf{y}^{(k)} = \mathbf{W}^{\frac{1}{2}}_{\widehat{\boldsymbol{\beta}}^{(k)}}\mathbf{z}^{(k)}$,
$\mathbf{X}^{(k)} = \mathbf{W}^{\frac{1}{2}}_{\widehat{\boldsymbol{\beta}}^{(k)}}\mathbf{X}$,
$\widehat{\boldsymbol{\beta}}^{(k+1)} = \arg\min_{\boldsymbol{\beta}} \|\mathbf{y}^{(k)} - \mathbf{X}^{(k)}\boldsymbol{\beta}\|^2$ solved using least squares.
$k = k + 1$;
end while;
Output: $\widehat{\boldsymbol{\beta}}^{(k)}$

4.2 Unconstrained parametrization of feasible models

Let us consider a feasible model defined by a linear space of parameters (equation 2.5):

$$\mathcal{L}_M = \{\boldsymbol{\beta} \in \mathbb{R}^p : \mathbf{A}_{0M}\boldsymbol{\beta} = 0\},$$

Similarly to the linear case, in order to replace a constrained by an unconstrained parametrization a change of variables in model M is performed getting $\boldsymbol{\xi}_M = \mathbf{A}_{1M}\boldsymbol{\beta}_M$. Let us recall that $\mathbf{X}\boldsymbol{\beta}_M = \mathbf{Z}_{1M}\boldsymbol{\xi}_M$. For all feasible models we have:

$$\mathscr{L}(\boldsymbol{\beta}_M) = \exp\{\boldsymbol{\eta}^T\mathbf{y} - \gamma(\boldsymbol{\eta})\},\ \boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta} = \mathbf{Z}_{1M}\boldsymbol{\xi}_M.$$

Let us denote:

$$\mathbf{Z}_{*1M} = \mathbf{X}_*\mathbf{A}^1_M = \mathbf{W}^{\frac{1}{2}}_{\boldsymbol{\beta}^*}\mathbf{X}\mathbf{A}^1_M,$$

$$\mathbf{H}_{M*} = \mathbf{Z}_*(\mathbf{Z}_*^T\mathbf{Z}_*)\mathbf{Z}_*^T.$$

Let us define GIC for feasible models as:

$$GIC_M = -2\ell(\widehat{\boldsymbol{\beta}}_M) + r \cdot |M|,$$

where $\widehat{\boldsymbol{\beta}}_M = \arg\min_{\boldsymbol{\beta}_M} \ell(\boldsymbol{\beta}_M)$.

4.3 DMR4glm algorithm

Algorithm 4 DMR4glm (Delete or Merge Regressors for generalized linear models)

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

1. Computation of likelihood ratio test statistics

Calculate likelihood ratio test statistics for all elementary constraints defined in (2.2):

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

$$LRT_{1jk} = 2\ell(\widehat{\boldsymbol{\beta}}_{M_{jk}}) - 2\ell(\widehat{\boldsymbol{\beta}}_F),$$

where M_{jk} is the full model with one elementary constraint.

end for

Calculate likelihood ratio test statistics for all elementary constraints defined in (2.3):

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

$$LRT_{ijk} = 2\ell(\widehat{\boldsymbol{\beta}}_{M_{ijk}}) - 2\ell(\widehat{\boldsymbol{\beta}}_F),$$

where M_{ijk} is the full model with one elementary constraint.

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**

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

$d_{ijk} = LRT_{ijk}$ 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: $\mathscr{M}^{DMR4glm} = \{M_0, \ldots, M_{p-1}\}$, $\mathbf{L}^{DMR4glm} = (L_{M_0}, \ldots, L_{M_{p-1}})^T$.

DMR4glm returns p nested models with dimensions from 1 to p. The choice of the final model can be based for example on crossvalidation or information criterion. In the next section we assume that Generalized Information Criterion is used.

4.4 DMR4glm_wald - Delete or Merge Regressors algorithm for generalized linear model with Wald statistics

Algorithm DMR4glm_wald works similarly as DMR4glm, but instead of likelihood ratio test statistics it uses Wald statistics. It will be shown later in Chapter 5 that this algorithm works faster than DMR4glm and has similar prediction and selection accuracy. However, we don't give the proof of consistency for this algorithm.

4.5 DMRnet4glm - Delete or Merge Regressors algorithm for generalized linear model and high-dimensional data

DMR4glm algorithm does not work for $p > n$. We propose the DMRnet4glm algorithm, analogous to the DMRnet4lm, by adding to DMR4glm a screening step so that it could be applied to high-dimensional data. The screening step is done using group lasso. After reduction of the dimension of the model to $p < n$, DMR4glm algorithm is used. In order to make the screening step more accurate and to better balance the impact of screening and the DMR4glm selection steps, the screening is done multiple times.

We rewrite the group lasso estimator, defined in equation 2.18:

$$\widetilde{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \left\{ -\ell(\boldsymbol{\beta}) + 2\lambda \left(\sum_{j=1}^{p_0} \nu(\beta_{j0}) + \sum_{k=1}^{l} \nu(\boldsymbol{\beta}_k) \right) \right\},$$

where:

$$\nu(\beta_{j0}) = |\beta_{j0}| \text{ for continuous variables,}$$

$$\nu(\boldsymbol{\beta}_k) = \sqrt{p_k - 1}\|\boldsymbol{\beta}_k\| \text{ for factors.}$$

Ordering is done by ordering the nonzero values of $\nu(\beta_{j0})$ and $\nu(\boldsymbol{\beta}_k)$ and building the final model using the model matrix composed of the continuous variables and factors corresponding to the highest values of $\nu(\beta_{j0})$ and $\nu(\boldsymbol{\beta}_k)$. In the DMRnet4glm algorithm, for example if $o = 5$ we will choose variables with $\frac{1}{5}$, $\frac{2}{5}$, $\frac{3}{5}$, $\frac{4}{5}$, and all variables with highest ν.

Algorithm 5 `DMRnet4glm`: Delete or Merge Regressors algorithm for generalized linear model and high-dimensional data

Input: $\mathbf{y}$, $\mathbf{X}$, $(o, \lambda_1 < \ldots < \lambda_m)$ and $p^- = \min(p, n/4)$.
1. Solve the group lasso problem for grid: $\lambda_1, \ldots, \lambda_m$***.***
for $q = 1$ **to** m **do**
$\widehat{\beta}^{(q)} = \text{argmin}_\beta \left\{ -\ell(\boldsymbol{\beta}) + \lambda_q \left(\sum_{j=1}^{p_0} \nu(\beta_{j0}) + \sum_{k=1}^{l} \nu(\boldsymbol{\beta}_k) \right) \right\}$.
Sort nonzero elements of $\{\nu(\beta_{10}), \ldots, \nu(\beta_{p_0 0}), \nu(\boldsymbol{\beta}_1), \ldots, \nu(\boldsymbol{\beta}_l)\}$ in non-increasing order, corresponding to s_{q0} continuous variables and $s_q - s_{q0}$ factors.
Select model with DMR4glm using the reduced model matrix.
for $t = 1$ **to** o **do**
$J = \left\{ j_1, \ldots, j_{s_{q0t}}, j_{s_{q0t+1}}, \ldots, j_{s_{qt}} \right\}$, where $s_{qt} = \lfloor s_q \cdot t/o \rfloor$.
$\mathscr{M}_{qt} = \mathscr{M}^{DMR4glm}(\mathbf{X}^J, \mathbf{y})$ and $\mathbf{L}_{qt} = \mathbf{L}^{DMR4glm}(\mathbf{X}^J, \mathbf{y})$, where $\mathbf{X}^J$ is model matrix constructed from $j_1, \ldots, j_{s_{q0t}}$-th columns from $\mathbf{X}_0$ and $\mathbf{X}_{j_{s_{q0t+1}}}, \ldots, \mathbf{X}_{j_{s_{qt}}}$.
end for;
end for;
2. Choose the best models with 1 ***to*** p^- ***parameters that has maximal log-likelihood.***
for $m = 1$ **to** p^- **do**
$M_m = \arg\max\{L_M : M \in \bigcup_{qt} \mathscr{M}_{qt},\ |M| = m\}$.
end for
Output: $\mathscr{M}^{DMRnet4glm} = \{\mathscr{M}_1, \ldots, \mathscr{M}_{p^-}\}$, $\mathbf{L}^{DMRnet4glm} = (L_{M_1}, \ldots, L_{M_{p^-}})^T$.

Remark 2. Let us notice that:

- DMRnet4glm improves the group lasso estimator by merging levels of factors.
- After solving the group lasso problem, model matrices $\mathbf{X}^J$ are built composed of the continuous variables and factors corresponding to the highest values of $\nu(\beta_{j0})$ and $\nu(\boldsymbol{\beta}_k)$. For example if $o = 5$ we will choose variables with $\frac{1}{5}$, $\frac{2}{5}$, $\frac{3}{5}$, $\frac{4}{5}$, and all variables with highest ν.
- DMRnet4glm returns p^- models with dimensions from 1 to p^-. The choice of the final model can be based for example on crossvalidation or information criterion.
- In comparison to DMRnet4lm, the number of models returned by DMRnet4glm equal to p^- is twice lower, because the algorithm is much more computationally demanding.
- Algorithm DMRnet4glm_wald works similarly, using DMR4glm_wald instead of DMR4glm.

4.6 Bound for selection error of the DMR4glm algorithm

Let us define for $M \not\subseteq T$:

$$\Delta_M = \min_{\boldsymbol{\beta}_M} \|\mathbf{X}_* \boldsymbol{\beta}^* - \mathbf{X}_* \boldsymbol{\beta}_M\|^2$$

$$\boldsymbol{\beta}_M^* = \arg \min_{\boldsymbol{\beta} \in \mathcal{L}_M} \|\mathbf{X}_* \boldsymbol{\beta}^* - \mathbf{X}_* \boldsymbol{\beta}\|^2,$$

$$\Delta_M = \|\mathbf{X}_*\boldsymbol{\beta}^* - \mathbf{X}_*\boldsymbol{\beta}^*_M\|^2 = \|(\mathbb{I} - \mathbf{H}_{*M})\mathbf{X}_*\boldsymbol{\beta}^*\|^2,$$

$$\delta = \min_M \Delta_M,$$

$$B_\delta = \{\boldsymbol{\beta} : \|\mathbf{X}_*\boldsymbol{\beta} - \mathbf{X}_*\boldsymbol{\beta}^*\|^2 \leq \delta\},$$

where $\boldsymbol{\beta}^{*T}\mathbf{X}_*^T(\mathbb{I} - \mathbf{H}_{*M})\mathbf{X}_*\boldsymbol{\beta}^* = \|\mathbf{X}_*\boldsymbol{\beta}^* - \mathbf{X}_*\boldsymbol{\beta}^*_M\|^2$.

Let us recall that:

$$\mathcal{M}_{\mathcal{V}}^F = \{M \subseteq F : |M| = |F| - 1 \text{ and } T \nsubseteq M\},$$

$$\mathcal{M}_{\mathcal{T}}^F = \{M \subseteq F : |M| = |F| - 1 \text{ and } T \subseteq M\}.$$

Theorem 3. *Assume that $p \leq n$ and exists $c > 0$ such that:*

$$\forall \boldsymbol{\beta} \in B_\delta \; c\ddot{\gamma}(\boldsymbol{\beta}^*) \leq \ddot{\gamma}(\boldsymbol{\beta}), \tag{4.2}$$

$$\mathbf{X} \textit{ is full rank},$$

$$\frac{\sigma^2 p}{c} \leq r \leq \frac{c\delta}{8t} \tag{4.3}$$

Let $\widehat{T}$ be the model selected by DMR4glm, where linkage criterion for hierarchical clustering is a convex combination of minimum and maximum of the pairwise distances between clusters. Then

$$\mathbb{P}\left(\widehat{T}^{DMR} \neq T\right) \leq \frac{\sigma^2 p}{cr}.$$

An important consequence of Theorem 3 is the following corollary.

Corollary 2 (Selection error bound)**.** *Assuming that $\mathbf{X}$ is full rank and $p \prec r \prec \frac{c\delta}{8t}$ DMR4glm is a consistent model selection method:*

$$\mathbb{P}(\widehat{T} \neq T) = o(1).$$

4.6.1 Proof of Theorem 3

The proof of Theorem 3 has similarities with proofs in Fahrmeir and Kaufmann [1985]. Let us denote the path of nested models from step 5 of the DMR4glm algorithm by $J = \{M_0, \ldots, M_{p-1}\}$. The event of erroneous selection of the model by the DMR4glm algorithm equals:

$$\begin{aligned}\{\widehat{T} \neq T\} &\subseteq \{T \notin J\} \cup \{T \in J, \mathrm{GIC}_T \geq \min_{M \subsetneq T} \mathrm{GIC}_M\} \\ &\cup \{T \in J, \mathrm{GIC}_T \geq \min_{T \subsetneq M} \mathrm{GIC}_M\} \\ &\subseteq \{T \notin J\} \cup \{\mathrm{GIC}_T \geq \min_{M \subsetneq T} \mathrm{GIC}_M\} \cup \{\mathrm{GIC}_T \geq \min_{T \subsetneq M} \mathrm{GIC}_M\}.\end{aligned}$$

Let us denote:

$$\{T \notin J\} = \mathscr{E}_1,$$

$$\{\exists_{M\supset T}\ \mathrm{GIC}_T \geq \mathrm{GIC}_M\} \subseteq \{-\ell(\widehat{\boldsymbol{\beta}}_T) \geq -\ell(\widehat{\boldsymbol{\beta}}_F) + \frac{r}{2}\} = \mathscr{E}_2,$$

$$\{\exists_{M\not\supseteq T}\ \mathrm{GIC}_T \geq \mathrm{GIC}_M\} \subseteq \{\exists_{M\not\supseteq T}\ -\ell(\widehat{\boldsymbol{\beta}}_T) + \frac{tr}{2} \geq -\ell(\widehat{\boldsymbol{\beta}}_M)\}$$

$$\subseteq \{\exists_{M\not\supseteq T}\ -\ell(\widehat{\boldsymbol{\beta}}_T) + \frac{ac\delta}{2} \geq -\ell(\widehat{\boldsymbol{\beta}}_M)\} = \mathscr{E}_{3a}$$

for some a so that $tr \leq ac\delta$. Then we get the following decomposition:

$$\{\widehat{T} \neq T\} \subseteq \mathscr{E}_1 \cup \mathscr{E}_2 \cup \mathscr{E}_{3a}.$$

We will show that sets $\mathscr{E}_1$, $\mathscr{E}_2$ and $\mathscr{E}_{3a}$ belong to sets of the form:

$$\mathscr{C}(b) = \{\boldsymbol{\varepsilon}^T\mathbf{H}_*\boldsymbol{\varepsilon} \geq b\}.$$

We use Taylor expansion. For $\beta \in \partial B_\delta$ and some $\tilde{\beta} \in [\beta^*, \beta]$ from Schwartz inequality and assumption (4.2) we get:

$$\ell_*(\boldsymbol{\beta}) = \ell(\boldsymbol{\beta}^*) - \ell(\boldsymbol{\beta}) = (\boldsymbol{\beta}^* - \boldsymbol{\beta})^T\mathbf{X}_*^T\mathbf{H}_*\boldsymbol{\varepsilon} + \frac{1}{2}(\boldsymbol{\beta}^* - \boldsymbol{\beta})^T\ddot{\gamma}(\tilde{\boldsymbol{\beta}})(\boldsymbol{\beta}^* - \boldsymbol{\beta})$$

$$\geq -\left(\delta\boldsymbol{\varepsilon}^T\mathbf{H}_*\boldsymbol{\varepsilon}\right)^{\frac{1}{2}} + \frac{c}{2}\delta.$$

Since the last expression does not depend on $\boldsymbol{\beta}$, we have for $a \geq 0$:

$$\begin{aligned}\mathscr{L}_a &= \left\{\min_{\boldsymbol{\beta}\in\partial B_\delta} \ell_*(\boldsymbol{\beta}) \leq \frac{ac\delta}{2}\right\} \\ &\subseteq \left\{-\left(\delta\boldsymbol{\varepsilon}^T\mathbf{H}_*\boldsymbol{\varepsilon}\right)^{\frac{1}{2}} + \frac{c}{2}\delta \leq \frac{ac\delta}{2}\right\} = \mathscr{C}\left(\frac{c^2(1-a)^2\delta^2}{4}\right). \end{aligned} \tag{4.4}$$

Lemma 9.

$$\forall_{0\leq a<1}\ \mathscr{E}_{3a} \subseteq \mathscr{L}_a \subseteq \mathscr{C}\left(\frac{c^2(1-a)^2\delta^2}{4}\right).$$

Proof. Let us notice that for $M \not\supseteq T$ $\|\mathbf{X}_*(\widehat{\boldsymbol{\beta}}_M - \boldsymbol{\beta}^*)\|^2 \geq \delta$, so $\widehat{\boldsymbol{\beta}}_M \notin \mathrm{int}(B_\delta)$. Moreover, $\ell_*(\boldsymbol{\beta}^*) = 0$. Since ℓ_* is convex we have:

$$\mathscr{L}_a \supseteq \left\{\exists_{M\not\supseteq T}\ \ell_*(\widehat{\boldsymbol{\beta}}_M) \leq \frac{ac\delta}{2}\right\} \supseteq \mathscr{E}_{3a}.$$

□

Let:

$$\mathscr{L}_0^C = \left\{\min_{\boldsymbol{\beta}\in\partial B_\delta} \ell_*(\boldsymbol{\beta}) > 0\right\}.$$

Lemma 10.

$$\mathscr{E}_2 \cap \mathscr{L}_0^C \subseteq \mathscr{C}(cr).$$

Proof. Let us notice that from convexity of ℓ_* and the fact that $\ell_*(\boldsymbol{\beta}^*) = 0$ we have:

$$\mathscr{L}_0^C \subseteq \left\{\forall_{M \supseteq T}\ \widehat{\boldsymbol{\beta}}_M \in \operatorname{int}(B_\delta)\right\} = \mathscr{B}.$$

Moreover, from the mean value theorem we get:

$$\mathbf{X}_*^T\boldsymbol{\varepsilon} = \dot{\ell}(\boldsymbol{\beta}^*) = -\left[\int_0^1 \ddot{\gamma}(\boldsymbol{\beta}^* + u(\boldsymbol{\beta}^* - \widehat{\boldsymbol{\beta}}_F))du\right](\boldsymbol{\beta}^* - \widehat{\boldsymbol{\beta}}_F).$$

Using Taylor expansion, from assumption (4.2) on $\mathscr{B}$ we get:

$$\ell(\widehat{\boldsymbol{\beta}}_T) - \ell(\widehat{\boldsymbol{\beta}}_F) \geq \ell_*(\widehat{\boldsymbol{\beta}}_F) = (\boldsymbol{\beta}^* - \widehat{\boldsymbol{\beta}}_F)^T\mathbf{X}_*^T\boldsymbol{\varepsilon} + \frac{1}{2}(\boldsymbol{\beta}^* - \widehat{\boldsymbol{\beta}}_F)^T\ddot{\gamma}(\tilde{\boldsymbol{\beta}})(\boldsymbol{\beta}^* - \widehat{\boldsymbol{\beta}}_F)$$

$$\geq (\boldsymbol{\beta}^* - \widehat{\boldsymbol{\beta}}_F)^T\mathbf{X}_*^T\boldsymbol{\varepsilon} = -\boldsymbol{\varepsilon}^T\mathbf{X}_*\left[\int_0^1 \ddot{\gamma}(\boldsymbol{\beta}^* + u(\boldsymbol{\beta}^* - \widehat{\boldsymbol{\beta}}_F))du\right]^{-1}\mathbf{X}_*^T\boldsymbol{\varepsilon}$$

$$\geq -c^{-1}\boldsymbol{\varepsilon}^T\mathbf{X}_*\ddot{\gamma}(\boldsymbol{\beta}^*)^{-1}\mathbf{X}_*^T\boldsymbol{\varepsilon} = -c^{-1}\boldsymbol{\varepsilon}^T\mathbf{H}_*\boldsymbol{\varepsilon}.$$

As a result we get $\mathscr{E}_2 \cap \mathscr{L}_0^C \subseteq \mathscr{E}_2 \cap \mathscr{B} \subseteq \left\{\boldsymbol{\varepsilon}^T\mathbf{H}_*\boldsymbol{\varepsilon} \geq cr\right\}$. □

If $0 \leq a \leq 2 - \sqrt{3} \approx 0.257$ we have $a \leq \frac{(1-a)^2}{2}$, so for $a = \frac{1}{4}$ from assumptions (4.3) we get:

$$cr \leq \frac{c^2 a\delta}{2t} \leq \frac{c^2(1-a)^2\delta}{4} \leq \frac{c^2\delta}{4}.$$

Then from Lemma 9 $\mathscr{E}_{3\frac{1}{4}} \subseteq \mathscr{C}\left(\frac{c^2(1-\frac{1}{4})^2\delta}{4}\right) \subseteq \mathscr{C}(cr)$, from equation 4.4 $\mathscr{L}_0 \subseteq \mathscr{C}\left(\frac{c^2\delta}{4}\right) \subseteq \mathscr{C}(cr)$ and from Lemma 10 $\mathscr{E}_2 \cap \mathscr{L}_0^C \subseteq \mathscr{C}(cr)$.

As a result we get $\mathscr{E}_{3\frac{1}{4}} \cup \mathscr{E}_2 \subseteq \mathscr{E}_{3\frac{1}{4}} \cup \mathscr{L}_0 \cup (\mathscr{E}_2 \cap \mathscr{L}_0^C) \subseteq \mathscr{C}(cr)$. Finally, from the Markov inequality and from the fact that $\mathbb{E}(\boldsymbol{\varepsilon}^T\mathbf{H}_*\boldsymbol{\varepsilon}) = \operatorname{tr}(\mathbf{H}_*\mathbb{I}) = \sigma^2 p$, we get

$$\mathbb{P}\left(\mathscr{C}(cr)\right) \leq \frac{\sigma^2 p}{cr}.$$

Let

$$\mathscr{E}_1 = \left\{\max_{M \in \mathcal{M}_{\mathcal{V}}^F}\left(LRT_M\right) \geq \min_{M \in \mathcal{M}_{\mathcal{T}}^F}\left(LRT_M\right)\right\}.$$

Lemma 11.

$$\mathscr{E}_1 \subseteq \mathscr{E}_{3\frac{1}{4}}.$$

Proof.

$$\mathscr{E}_1 = \left\{\max_{M \in \mathcal{M}_{\mathcal{V}}^F} \ell(\widehat{\boldsymbol{\beta}}_M) \geq \min_{M \in \mathcal{M}_{\mathcal{T}}^F} \ell(\widehat{\boldsymbol{\beta}}_M\right\}$$

$$\mathscr{E}_{3\frac{1}{4}} = \{\exists_{M \not\supseteq T}\ \ell(\widehat{\boldsymbol{\beta}}_M) + \frac{c\delta}{8} \geq \ell(\widehat{\boldsymbol{\beta}}_T)\}$$

Since $\min_{M \in \mathcal{M}_{\mathcal{T}}^F} \ell(\widehat{\boldsymbol{\beta}}_M) \geq \ell(\widehat{\boldsymbol{\beta}}_T)$, $\mathscr{E}_1 \subseteq \mathscr{E}_{3\frac{1}{4}}$.

□

Proof of Theorem 3 Let us consider constant h_* such that

$$\max_{M\in\mathcal{M}_{\mathcal{V}}^F}\left(LRT_M\right) < h_* < \min_{M\in\mathcal{M}_{\mathcal{T}}^F}\left(LRT_M\right)$$

It is obvious that cutting heights for true constraints for continuous variables are smaller than h_* and for false ones greater than h_*. It also follows from Lemma 8 that dissimilarity matrices used in the algorithm are consistent with the partitions for model T. Then, applying Lemma 2 for each factor, we get that the cutting heights for aggregations compatible with the true partitions are not greater than h_* and for incompatible ones not smaller than h_*. Hence, the probability that in the DMR4glm algorithm accepting true constraints precede accepting false ones is contained in $\mathscr{E}_1$ and, from Lemma 11, in $\mathscr{E}_{3\frac{1}{4}}$.

Hence, we can bound the error of erroneous selection by the DMR4glm algorithm:

$$\mathbb{P}\left(\widehat{T}\neq T\right) \leq \mathbb{P}\left(\mathscr{E}_1\cup\mathscr{E}_2\cup\mathscr{E}_{3\frac{1}{4}}\right) \leq \mathbb{P}\left(\mathscr{C}(cr)\right) \leq \frac{\sigma^2 p}{c\_r}.$$

Corollary 3 (Consistency)**.** *Under the assumptions from Theorem 3, DMR4glm is a consistent model selection method:*

$$\mathbb{P}(\widehat{T}=T)\xrightarrow{P}1.$$

Chapter 5

Numerical experiments for linear regression

In order to compare the performance of our algorithms with those described in the literature, analysis of 3 real data sets and a simulation study based on 6 scenarios have been performed: in the first 3 $p < n$ and in the remaining 3 $p \gg n$. We compared performances of 6 algorithms: our DMR4lm (denoted as DMR) and DMRnet4lm (DMRnet) and 4 competitors: CAS-ANOVA, gvcm, grpreg MCP (group MCP) and grpreg group lasso (group lasso). Let us notice that while CAS-ANOVA and gvcm solve the same factorial selection problem as the DMR algorithms, group MCP and group lasso choose whole factors. We want to answer 2 questions: if the DMR algorithms are better than the lasso based methods in terms of minimizing prediction error and choosing sparse models and whether the effort of merging levels of factors really pays off? We have to keep in mind that in terms of computation time group MCP and group lasso are more efficient, obviously.

In all the cases for the DMRnet algorithm we set parameters o=5, m=50 and for the DMR and DMRnet algorithms the complete linkage in hierarchical clustering was used. For group MCP and group lasso there are 2 ways of calculating the degrees of freedom when calculating GIC: df_1 implemented in the package `grpreg` and described in Breheny and Huang [2015] and df_2 equal to |M|. We present only the results for df_1 since the results for df_2 were similar. For CAS-ANOVA and gvcm λ grids (with 100 λ values) were chosen on the log scale so that $|M|$ took many values from 1 to p. Moreover, for group MCP and group lasso the default 100-elements λ grids were used.

5.1 Real data examples

Model selection was done using 6 algorithms on 3 data sets: Miete, Barley and Antigua. We used 10-fold cross-validation (C-V) to calculate mean prediction error PE and mean model dimension MD for 100 λ values for CAS-ANOVA, gvcm, group MCP and group lasso and for model dimension from 1 to p for DMR and from 1 to $\min\{p, \frac{n}{2}\}$ for DMRnet. Moreover, PE and MD were calculated for models selected by $GIC_M = RSS_M + c \cdot \log(p) \cdot \widehat{\sigma}^2 \cdot df$, where df stands for degrees of freedom and equals $|M|$ in all of the cases except for df_1 for group MCP and group lasso, for different values of $c \in \mathcal{C}_1 = \{.05, .1, .15, .25, .5, .75, \ldots, 7.5\}$, where $\widehat{\sigma}$ was computed by least squares using the full model.

The results are organized in tables and 2 types of plots. In the first type points corresponding to $(MD(\lambda), PE(\lambda))$, $\lambda = \lambda_1, \ldots, \lambda_{100}$ for each algorithm: CAS-ANOVA, gvcm, group MCP and group lasso and analogously points $(MD(m), PE(m))$ $m = 1, \ldots, p$ for DMR algorithm and $m = 1, \ldots, \min\{p, \frac{n}{2}\}$ for DMRnet algorithm are plotted together. In the second type points corresponding to $(MD(c), PE(c))$, $c \in \mathcal{C}_1$, where $MD(c)$ and $PE(c)$ are calculated for models chosen by GIC with parameter c, for all the algorithms are plotted together.

5.1.1 Estimation of prediction error using 10-fold cross-validation

The estimator of the prediction error was calculated as follows. We assume for ease of notation that the folds have equal number of observations n_{test}. Let w_{tl} denote the error calculated for learning sample $\Lambda_l = \{y_i^{(l)},\ (\mathbf{x}_{i\cdot}^{(l)})^T,\ i = 1, \ldots, 9 \cdot n_{test}\}$ consisting of 9 folds of data and test sample $\{y_t^{(l)},\ (\mathbf{x}_{t\cdot}^{(l)})^T\}$ consisting of the remaining fold, $t = 1, \ldots, n_{test}$, $l = 1, \ldots, N = 10$. We computed the squared loss function:

$$w_{tl} = L(y_t^{(l)}, \widehat{y}_t^{(l)}) = \left(y_t^{(l)} - (\mathbf{x}_{t\cdot}^{(l)})^T \widehat{\boldsymbol{\beta}}^{(l)}\right)^2,$$

where $\widehat{\boldsymbol{\beta}}^{(l)}$ is the parameter vector estimated using Λ_l. Let

$$\overline{w}_{\cdot l} = \frac{1}{n_{test}} \sum_{t=1}^{n_{test}} w_{tl},\ \overline{w} = \frac{1}{N} \sum_{l=1}^{N} \overline{w}_{\cdot l} = \frac{1}{N} \frac{1}{n_{test}} \sum_{l=1}^{N} \sum_{t=1}^{n_{test}} w_{tl}.$$

We will use $PE = \overline{w}/\widehat{\sigma}^2$ as the estimator of the prediction error. In order to assess its accuracy, we use standard deviation:

$$sd(PE) = \frac{\frac{1}{N}\sqrt{\sum_{l=1}^{N} (\overline{w}_{\cdot l} - \overline{w})^2}}{\widehat{\sigma}^2}$$

The estimator of the model dimension (MD) and its standard deviation were calculated similarly. Let d_l denotes the model dimension for $l = 1, \ldots, N$ and $MD = \frac{1}{N} \sum_{l=1}^{N} d_l$ and its standard deviation

$$sd(MD) = \frac{1}{N} \sqrt{\sum_{l=1}^{N} (d_l - MD)^2}.$$

5.1.2 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 Tutz [2011]. 8 categorical and 2 continuous variables give 36 and 3 (including the intercept) parameters. This gives $p = 39$.

In Figure 5.1 a plot of PE vs 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. The picture shows results for small values of PE in order to make the differences between the methods more visible. 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.

In Figure 5.2 PE vs MD for models chosen by GIC, $c \in \mathcal{C}_1$ are presented. Again, the DMRnet algorithm gives the smallest prediction error among the competition and sparse models.

In Table 5.1 there are characteristics of models chosen by GIC (MD and PE with standard deviations) for optimal c: $c = 2.5$ gives minimal PE for DMRnet and DMR, $c = .75$ for CAS-ANOVA, $c = .5$ for group lasso and $c = .25$ for gvcm and group MCP.

Table 5.1: MD and PE with standard deviations for Miete data set for the models selected using GIC with optimal c.

Algorithm	c	MD (sd)	PE (sd)
DMRnet	2.5	12 (0)	1.021 (.049)
DMR	2.5	12 (0)	1.022 (.049)
CAS-ANOVA	.75	20.7 (.8)	1.027 (.047)
gvcm	.25	25.7 (.9)	1.026 (.049)
group MCP	.25	39 (0)	1.027 (.049)
group lasso	.5	39 (0)	1.027 (.048)

Figure 5.1: PE vs MD calculated by 10-fold C-V for Miete data set.

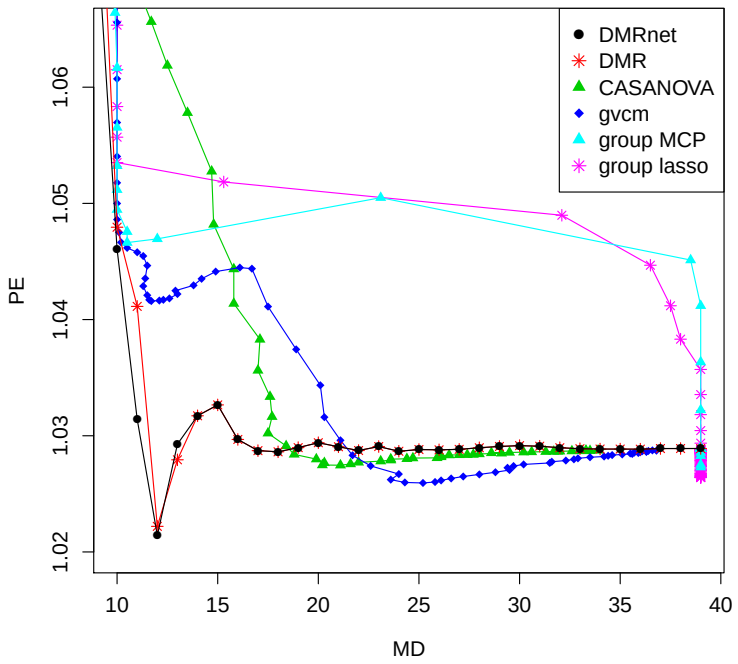
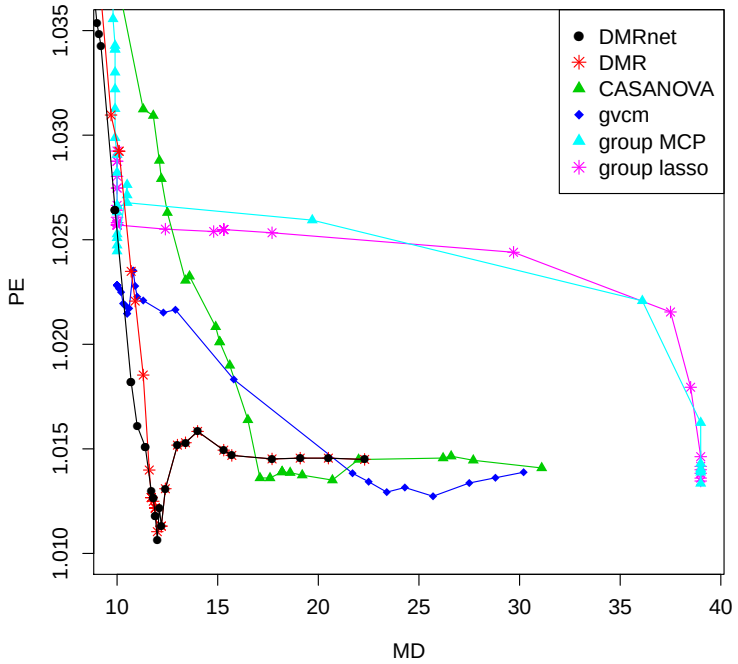


Figure 5.2: PE vs MD calculated by 10-fold C-V for Miete data set for models chosen by GIC.



5.1.3 Barley

The data set Barley from R library `lattice` has already been discussed in the literature, for example in Bondell and Reich [2009]. The response is the barley yield for each of 5 varieties (Svansota, Manchuria, Velvet, Peatland and Trebi) at 6 experimental farms in Minnesota for each year of the years 1931 and 1932 giving a total of $n = 60$ and $p = 11$.

In Figure 5.3 a plot of PE vs MD calculated by 10-fold C-V for 100 λ values for CAS-ANOVA, gvcm, group MCP and group lasso and for model dimension 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 = 7, for CAS-ANOVA and gvcm when MD = 10.3 and for group MCP and group lasso for the full model, MD=11. If we chose models with the lowest prediction error, DMRnet would have both the smallest error and the smallest number of parameters.

In Figure 5.4 PE vs MD for models chosen by GIC, $c \in \mathcal{C}_1$ are presented. The values of interest for c in terms of PE for DMRnet are: 0.5 giving MD = 7 and PE = 1.193 and 1.75, 2, 2.25, 2.5 all giving MD = 5.1 and PE = 1.293. However, the values from 1.75 to 2.5 give a much smaller model with only a slight increase of prediction error. Again, the DMRnet algorithm gives the smallest prediction error among the competition and sparse models.

In Table 5.2 there are characteristics of models chosen by GIC (MD and PE with standard deviations) for optimal c: $c = .5$ gives minimal PE for DMRnet, DMR, group MCP and group lasso, $c = .05$ for CAS-ANOVA and gvcm. Additionally, results for $c = 2.5$ for DMRnet and DMR are given.

Table 5.2: MD and PE with standard deviations for Barley data set for the models selected using GIC with optimal c.

Algorithm	c	MD (sd)	PE (sd)
DMRnet	.5	7 (.1)	1.193 (.179)
DMR	.5	7 (.1)	1.193 (.179)
CAS-ANOVA	.05	9.5 (.3)	1.229 (.178)
gvcm	.05	9.6 (.4)	1.224 (.178)
MCP	.5	11 (0)	1.21 (.186)
group lasso	.5	11 (0)	1.209 (.178)
DMRnet	2.5	5.1 (.1)	1.293 (.233)
DMR	2.5	5.1 (.1)	1.293 (.233)

Figure 5.3: PE vs MD calculated by 10-fold C-V for Barley data set.

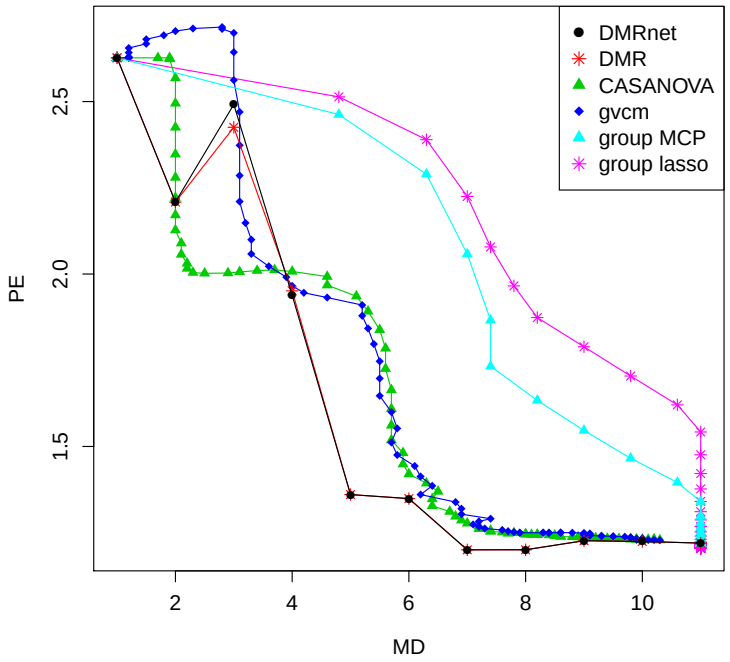
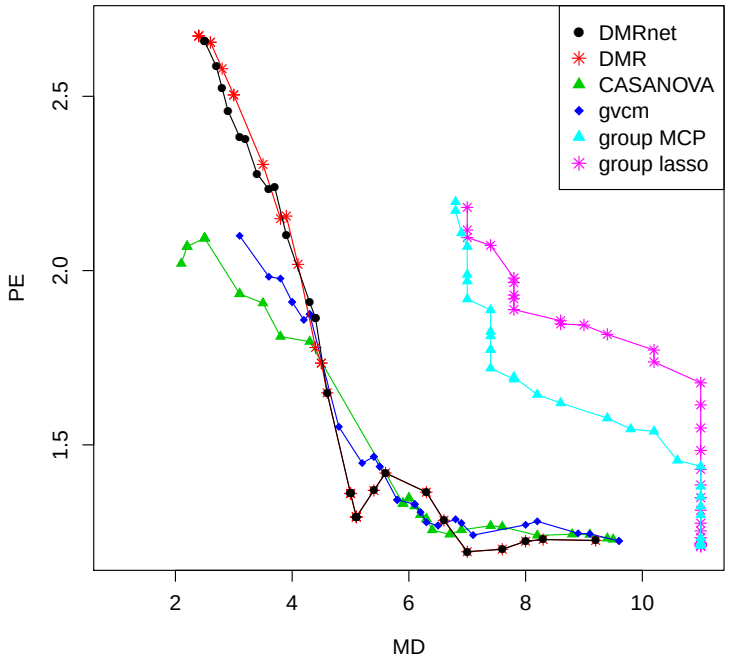


Figure 5.4: PE vs MD calculated by 10-fold C-V for Barley data set for models chosen by GIC.



5.1.4 Antigua

Data set Antigua used in this example comes from `R` library `DAAG`. The data concerns maize fertilizer experiments on the Island of Antigua. The average weight of good maize ears is under investigation and $n = 287$ observations of 3 categorical and 1 continuous variable are given. Type of fertilizer has 12 levels: 000 111 113 131 002 020 200 202 022 220 222 311, where the three digits represent the amount of nitrogen, phosphorus and potassium in the fertilizer, respectively. Name of the site, where the maize was planted was included in the model since differences between them in terms of yield was significant, for example site TEAN was demaged by goats. There are 8 levels of factor site: DBAN, LFAN, NSAN, ORAN, OVAN, TEAN, WEAN, WLAN. The last categorical predictor is parcel with 4 levels. This gives $p = 24$.

In Figure 5.5 a plot of PE vs MD calculated by 10-fold C-V for 100 λ values for CAS-ANOVA, gvcm, group MCP and group lasso and for model dimension 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 when MD = 8, for DMR when MD = 21, for CAS-ANOVA when MD = 21.5, for gvcm when MD = 14, for group MCP when MD = 20 and for group lasso when MD = 22.7. The smallest prediction error has group MCP (1.063), but DMRnet has not much greater PE (1.093) and much smaller MD.

In Figure 5.6 PE vs MD for models chosen by GIC, $c \in \mathcal{C}_1$ are presented. The values of interest for c in terms of PE for DMRnet are: 0.05 giving MD = 18.7 and PE = 1.102 and 2.25, 2.5, 2.75, 3 all giving MD = 7.9 and PE = 1.108. However, the values from 2.25 to 3 give a much smaller model with only a slight increase of prediction error.

In Table 5.3 there are characteristics of models chosen by GIC (MD and PE with standard deviations) for optimal c: $c = .05$ gives minimal PE for DMRnet, DMR and CAS-ANOVA, $c = .1$ for gvcm and $c = .5$ for group MCP and group lasso. Additionally, results for $c = 2.5$ for DMRnet and DMR are given.

Table 5.3: MD and PE with standard deviations for Antigua data set for the models selected using GIC with optimal c.

Algorithm	c	MD (sd)	PE (sd)
DMRnet	.05	18.7 (.03)	1.102 (.104)
DMR	.05	18.4 (.05)	1.103 (.104)
CAS-ANOVA	.05	21.5 (.03)	1.118 (.1)
gvcm	.1	18.4 (.06)	1.112 (.108)
group MCP	.5	20 (0)	1.061 (.106)
group lasso	.5	23 (0)	1.084 (.104)
DMRnet	2.5	7.9 (.1)	1.108 (.102)
DMR	2.5	7.9 (.1)	1.141 (.103)

Figure 5.5: PE vs MD calculated by 10-fold C-V for Antigua data set.

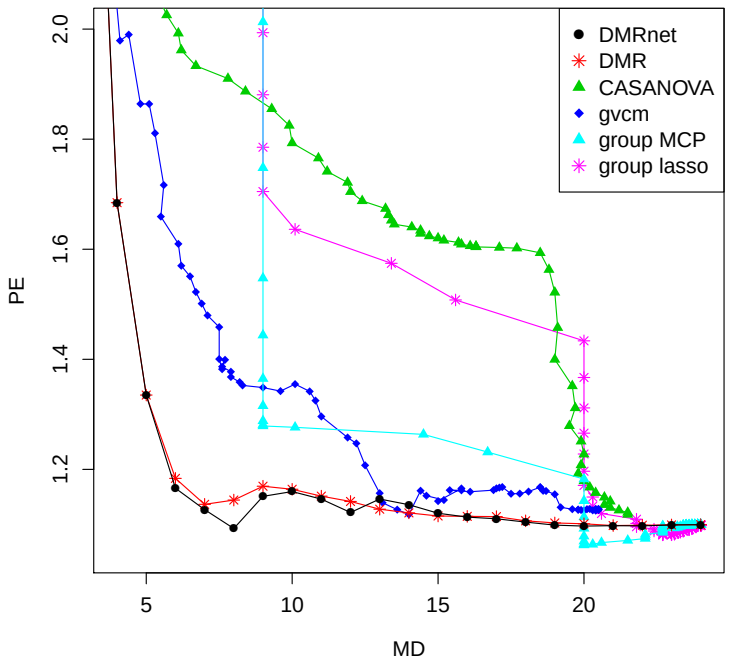
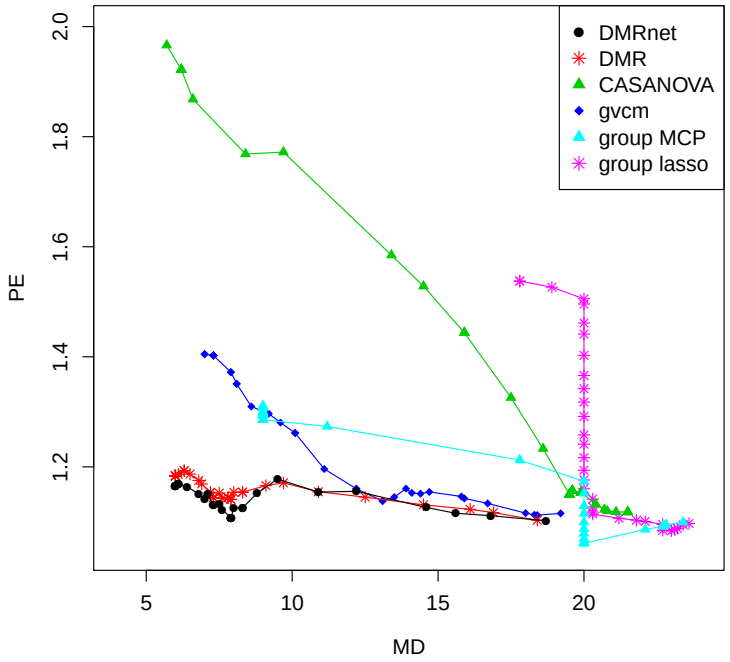


Figure 5.6: PE vs MD calculated by 10-fold C-V for Antigua data set for models chosen by GIC.



5.2 Simulation study

A simulation study was conducted in order to compare model selection methods. We assumed a true relationship between $\mathbf{y}$ and $\mathbf{X}$, data sets were generated, a linear model was fitted using different algorithms and finally, the outcome was compared to the truth.

Independent random samples $\{\mathbf{x}_{i\cdot}^T, y_i\}$, $i = 1, \ldots, n$ from the linear model (2.13) were generated. Plan of experiments is presented in Tables 5.4 for $p < n$ scenarios and 5.5 for $p \gg n$ scenarios, where l stands for the number of factors with equal numbers o levels, t for the dimension of the true model. Parameter vector $\boldsymbol{\beta}^*$ can have one of three forms:

- the same as in Bondell and Reich [2009]:

$$\boldsymbol{\beta}_1^* = (2, 0, -3, -3, -3, -3, -2, -2, \mathbf{0}_{p-8}^T)^T, \tag{5.1}$$

- with opposite effects:

$$\boldsymbol{\beta}_2^* = (1, -2, 2, -2, 2, -2, 3, -3, 3, -3, 3, \mathbf{0}_{p-11}^T)^T. \tag{5.2}$$

- the same as in Yuan and Lin [2006], which corresponds to Tibshirani [1996]:

$$\boldsymbol{\beta}_3^* = (0.3, 3, 1.2, 0, 0, 0.5, -0.5, 0, 0, 0, -1, \mathbf{0}_{p-11}^T)^T, \tag{5.3}$$

In order to get the model matrix $\mathbf{X}$, first the rows of a matrix $\mathbf{Z}$ were generated iid from l-dimensional normal distribution $x_i \sim N(0_l, \boldsymbol{\Xi}(\rho))$, $i = 1, \ldots, n$. First-order auto-regressive structure of covariance matrix was considered, $\boldsymbol{\Xi}(\rho) = AR(\rho) = (\xi_{ij})_{i,j=1,..l}$, $\xi_{ij} = \rho^{|i-j|}$. Next, Z_{ij}, $i = 1, \ldots, n$, $j = 1, \ldots, l$ were replaced by $0, 1, \ldots, p_k - 1$ (p_k denotes the number of levels in factors) if it belonged to $(-\infty, \Phi^{-1}(\frac{1}{p_k}))$, $[\Phi^{-1}(\frac{1}{p_k}), \Phi^{-1}(\frac{2}{p_k}))$, $\ldots$, $[\Phi^{-1}(\frac{p_{k-1}}{p_k}), \infty)$, respectively, where Φ^{-1} is the empirical quantile function.

The noise was generated from normal distribution $\varepsilon \sim N(\mathbf{0}_n, \sigma^2 \mathbf{I}_n)$. Signal to noise ratio averaged over replications (SNR) $\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(N)}$ equals:

$$SNR = \frac{1}{N} \sum_{l=1}^{N} \frac{\|\mathbf{X}^{(l)} \boldsymbol{\beta}^*\|}{\sqrt{(n)}\sigma}.$$

For every experiment the results are based on $N = 1000$ simulation runs.

Table 5.4: Plan of experiments for linear regression, $p < n$.

exp	n	p	l	t	$\boldsymbol{\beta}^*$	$\Xi(\rho)$	σ	SNR
1	100	31	6	3	$\boldsymbol{\beta}_1^*$	AR(.5)	2	.87
2	100	31	6	5	$\boldsymbol{\beta}_2^*$	AR(.5)	1.5	2.26
3	100	31	15	6	$\boldsymbol{\beta}_3^*$	AR(.5)	.5	3.86

Table 5.5: Plan of experiments for linear regression, $p \gg n$.

exp	n	p	l	t	$\boldsymbol{\beta}^*$	$\Xi(\rho)$	σ	SNR
1	200	3001	600	3	$\boldsymbol{\beta}_1^*$	AR(.5)	2.5	.7
2	200	2001	400	5	$\boldsymbol{\beta}_2^*$	AR(.5)	2	1.7
3	200	2001	1000	6	$\boldsymbol{\beta}_3^*$	AR(.5)	.5	3.86

5.2.1 Estimation of prediction error

In the simulations we reported the number of times the true model was correctly identified (denoted by TM), the number of times all of the true factors were kept and false factors set to zero (denoted by TF), mean model dimension (denoted by MD), mean prediction error on a test data set with $n_{test} = 1000$ observations (denoted by PE) and mean execution time in seconds (denoted by time). Moreover, the statistics were calculated for models selected by $GIC_M = RSS_M + c \cdot \log(p) \cdot \sigma^2 \cdot df$ (df stands for degrees of freedom and equals $|M|$ in all of the cases except for df_1 for group MCP and group lasso), for different values of $c \in \mathcal{C}_2 = \{.25, .5, .75, \ldots, 7.5\}$.

The estimated prediction error (PE) was calculated as follows. Let w_{tl} denote the error calculated for learning sample $\Lambda_l = \{y_i^{(l)},\ (\mathbf{x}_{i\cdot}^{(l)})^T,\ i = 1, \ldots, n\}$ and independent test sample $\{y_t^{(l)},\ (\mathbf{x}_{t\cdot}^{(l)})^T\}$, $t = 1, \ldots, n_{test}$, $l = 1, \ldots, N$. We have $N = 1000$ and $n_{test} = 1000$. We assume the squared loss function:

$$w_{tl} = L\left((\mathbf{x}_{t\cdot}^{(l)})^T\boldsymbol{\beta}^*, (\mathbf{x}_{t\cdot}^{(l)})^T\widehat{\boldsymbol{\beta}}^{(l)}\right) = \left((\mathbf{x}_{t\cdot}^{(l)})^T\boldsymbol{\beta}^* - (\mathbf{x}_{t\cdot}^{(l)})^T\widehat{\boldsymbol{\beta}}^{(l)}\right)^2,$$

where $\widehat{\boldsymbol{\beta}}^{(l)}$ is the parameter vector estimated using Λ_l. Let

$$\overline{w}_{\cdot l} = \frac{1}{n_{test}}\sum_{t=1}^{n_{test}} w_{tl},\ \overline{w} = \frac{1}{N}\sum_{l=1}^{N}\overline{w}_{\cdot l} = \frac{1}{N}\frac{1}{n_{test}}\sum_{l=1}^{N}\sum_{t=1}^{n_{test}} w_{tl}.$$

We will use $PE = \overline{w}/\sigma^2$ as the estimator of the expected prediction error $\mathbb{E}\overline{w}$. In order to assess the accuracy of the estimator, we calculate its standard deviation. Since the learning data sets are independent and have identical distributions,

$$\mathrm{Var}(\overline{w}) = \mathbb{E}\left(\overline{w} - \mathbb{E}\overline{w}\right)^2 = \frac{1}{N^2}\sum_{l=1}^{N}\mathbb{E}\left(\overline{w}_{\cdot l} - \mathbb{E}\overline{w}_{\cdot l}\right)^2 = \frac{1}{N}\mathbb{E}\left(\overline{w}_{\cdot l} - \mathbb{E}\overline{w}_{\cdot l}\right)^2$$

for some l. Using the decomposition of variance we get

$$\mathrm{Var}(\overline{w}) = \frac{1}{N}\left\{\mathbb{E}\mathbb{E}\left[(\overline{w}_{\cdot l} - \mathbb{E}[\overline{w}_{\cdot l}|\Lambda_l])^2|\Lambda_l\right] + \mathbb{E}\left(\mathbb{E}[\overline{w}_{\cdot l}|\Lambda_l] - \mathbb{E}\mathbb{E}[\overline{w}_{\cdot l}|\Lambda_l]\right)^2\right\}.$$

Because the observations in the test set are independent given the training set, we have

$$\mathrm{Var}(\overline{w}) = \frac{1}{N}\left\{\frac{1}{n_{test}}\mathbb{E}\mathrm{Var}[w_{tl}|\Lambda_l] + \mathrm{Var}\left(\mathbb{E}[w_{tl}|\Lambda_l]\right)\right\}.$$

We estimate it by

$$\overline{V} = \frac{1}{N}\left\{\frac{1}{n_{test}}\frac{1}{N}\sum_{l=1}^{N}\frac{1}{n_{test}}\sum_{t=1}^{n_{test}}\left(w_{tl} - \overline{w}_{\cdot l}\right)^2 + \frac{1}{N}\sum_{l=1}^{N}\left(\overline{w}_{\cdot l} - \overline{w}\right)^2\right\}.$$

The standard deviation of the prediction error is $sd(PE) = \frac{\sqrt{\overline{V}}}{\sigma^2}$.

The estimator of the model dimension (MD) and its standard deviation were calculated as follows. If we denote d_l as model dimension in each simulation run, $l = 1, \ldots, N$, then $MD = \frac{1}{N}\sum_{l=1}^{N} d_l$ and its standard deviation equals

$$sd(MD) = \sqrt{\frac{1}{N^2}\sum_{l=1}^{N}(d_l - MD)^2}.$$

5.2.2 Results

The results for all the simulation setups are organized in 4 types of figures and 3 types of tables:

1. Figures:

   (i) Points corresponding to $(MD(\lambda), PE(\lambda))$, $\lambda = \lambda_1, \ldots, \lambda_{100}$ for each algorithm: CAS-ANOVA, gvcm, group MCP and group lasso and $(MD(m), PE(m))$ $m = 1, \ldots, p$ for DMR algorithm and $m = 1, \ldots, \min\{p, \frac{n}{2}\}$ for DMRnet algorithm are plotted together. The dark red vertical lines denote the dimension of the true model. They are given on the left sides of Figures: 5.7, 5.9, 5.11, 5.13, 5.15, 5.17.

   (ii) 50 runs of the simulations for $p < n$ and 100 runs for $p \gg n$ are given showing the variability of the results for DMRnet, gvcm and group MCP when $p < n$ and for DMRnet, group MCP and group lasso when $p \gg n$. The dark red vertical lines denote the dimension of the true model. They are given on the right sides of Figures: 5.7, 5.9, 5.11, 5.13, 5.15, 5.17.

   (iii) Points corresponding to $(MD(c), PE(c))$, $c \in \mathcal{C}_2$, where $MD(c)$ and $PE(c)$ are calculated for models chosen by GIC with parameter c, for all the algorithms are plotted together. The dark red vertical lines denote the dimension of the true model. They are given on the left sides of Figures: 5.8, 5.10, 5.12, 5.14, 5.16, 5.18.

   (iv) Points corresponding to $(MD(c), SE(c))$, $c \in \mathcal{C}_2$, where the selection error is calculated as $SE = \frac{N-TM}{N}$, for all the algorithms are given. The dark red vertical lines denote the dimension of the true model. They are given on the right sides of Figures: 5.8, 5.10, 5.12, 5.14, 5.16, 5.18.

2. Tables:

   (i) Values of minimal PE with the corresponding MD corresponding to figures (i). They are given in Tables: 5.6, 5.9, 5.12, 5.15, 5.18, 5.21.

   (ii) Optimal values of c in terms of PE and TM (giving minimal PE and maximal TM) for models were chosen by GIC. They are given in Tables: 5.7, 5.10, 5.13, 5.16, 5.19, 5.22.

   (iii) Characteristics of models: TM, TF, MD with standard deviation, PE with standard deviation and time for models chosen by GIC for the values of c given in the second type of tables and additionally for $c = 2.5$, which is the value which usually gives good results for DMR and DMRnet. They are given in Tables: 5.8, 5.11, 5.14, 5.17, 5.20, 5.23.

For experiments when $p \gg n$ there are no results for DMR and CAS-ANOVA since they work only if $p < n$ and no results for gvcm since the problem was too computationally intensive for this implementation and it ended with an error.

Looking at figures (i) and tables (i) for experiments 1 and 2 when $p < n$ and experiment 1 when $p \gg n$, we can see that if we chose models with the lowest prediction error, DMRnet would have the smallest error and the smallest number of parameters. In experiments 3 when $p < n$ and 2 when $p \gg n$, group MCP gives the smallest PE among competition, but it chooses

larger models than DMRnet. In experiment 3 when $p \gg n$ the results for group MCP and DMRnet are very similar.

Looking at figures (ii) for experiments $p < n$, the variability of the results for gvcm is higher than for DMRnet and group MCP. For all the experiments the variability of DMRnet, compared to the variability of the competitive methods, is lower around the dimension of the true model and higher for bigger models.

Looking at figures (iii) and tables (iii) for experiments 1, 2 when $p < n$ and 1 when $p \gg n$, the DMRnet algorithm gives the smallest prediction error among the competition and the sparsest models. In experiments 3 when $p < n$ and 2 when $p \gg n$, group MCP gives the smallest PE among competition, but it chooses larger models than DMRnet. In experiment 3 when $p \gg n$ the results for group MCP and DMRnet are very similar in terms of MD, but group MCP has lower PE.

Looking at figures (iv) DMRnet chooses the true model most often in all the experiments except for experiment 3 when $p < n$, where CAS-ANOVA gives lowest SE. Let us notice that group MCP and group lasso always give SE equal to 1 (TM=0) since they don't split factors.

Looking at figures (iii) and (iv) let us notice that for DMR and DMRnet approximately the same number of parameters (around t, which is the true model dimension) gives minimal PE and minimal SE, which is not true for other algorithms. Looking at tables (ii) we can see that similar values of c are optimal in terms of PE and SE (TM) for DMR and DMRnet.

Looking at tables (iii) we can see that in terms of time, group MCP and group lasso have much smaller values than DMRnet and DMR, but they don't split factors. CAS-ANOVA and gvcm are much more time demanding.

Value of $c = 2.5$ gives good results for DMRnet in all the setups.

5.2.3 Experiment 1, linear regression, $p < n$

Table 5.6: Values of minimal PE with the corresponding MD for Experiment 1, $p < n$, linear regression.

	MD	PE
DMRnet	3	.08
DMR	3	.128
CAS-ANOVA	7.86	.125
gvcm	8.1	.121
group MCP	14.16	.138
group lasso	24.88	.157

Table 5.7: Values of interest for c for Experiment 1, $p < n$, linear regression.

Algorithm	optimal c in terms of PE	optimal c in terms of TM
DMRnet	3.25	3.75
DMR	3.25	3.5
CAS-ANOVA	1	4
gvcm	1	3.5
group MCP	.75	-
group lasso	.75	-

Table 5.8: Characteristics of models chosen by GIC for some c for Experiment 1, $p < n$, linear regression.

Algorithm	c	TM	TF	MD (sd)	PE (sd)	time
DMRnet	3.25	676	860	3 (.01)	.091 (.003)	.27
DMR	3.25	505	783	3 (.02)	.13 (.004)	.15
CAS-ANOVA	1	54	321	6.76 (.09)	.144 (.003)	11.29
gvcm	1	64	329	6.7 (.09)	.141 (.003)	9.33
group MCP	.75	0	453	14.6 (.13)	.139 (.002)	.01
group lasso	.75	0	18	21.34 (.14)	.165 (.002)	.01
DMRnet	3.75	687	863	2.93 (.01)	.091 (.003)	.27
DMR	3.5	511	785	2.93 (.02)	.131 (.004)	.15
CAS-ANOVA	4	198	496	2.77 (.04)	.281 (.007)	11.29
gvcm	3.5	212	532	2.97 (.04)	.245 (.006)	9.33
DMRnet	2.5	597	780	3.22 (.02)	.099 (.003)	.27
DMR	2.5	467	735	3.25 (.02)	.136 (.004)	.15

Figure 5.7: Left side: PE vs MD for Experiment 1, $p < n$, linear regression. Right side: 50 runs of Experiment 1, linear regression, $p < n$, PE vs MD.

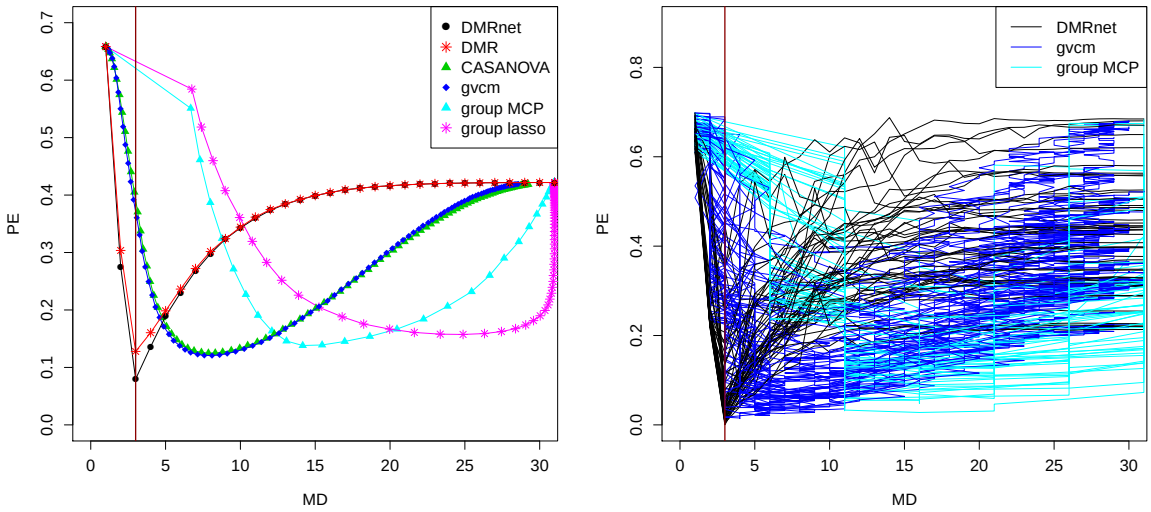
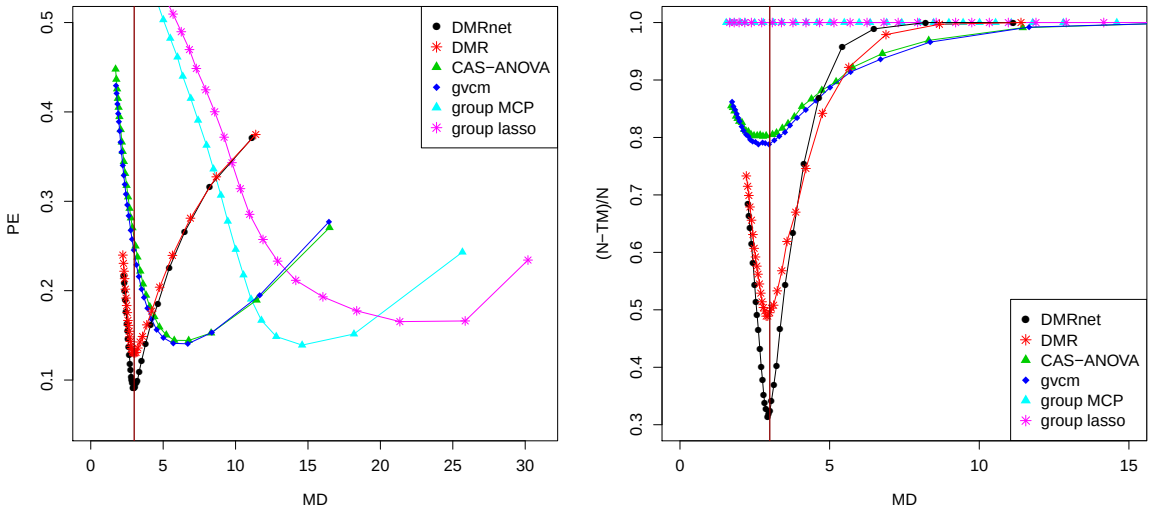


Figure 5.8: Left side: PE vs MD for Experiment 1, $p < n$, linear regression for models chosen by GIC. Right side: (N-TM)/N vs MD for Experiment 1, $p < n$, linear regression for models chosen by GIC.



5.2.4 Experiment 2, linear regression, $p < n$

Table 5.9: Values of minimal PE with the corresponding MD for Experiment 2, $p < n$, linear regression.

	MD	PE
DMRnet	5	.096
DMR	5	.147
CAS-ANOVA	9.02	.125
gvcm	8.58	.108
group MCP	11.71	.119
group lasoo	25.96	.192

Table 5.10: Values of interest for c for Experiment 2, $p < n$, linear regression.

Algorithm	optimal c in terms of PE	optimal c in terms of TM
DMRnet	1.5	1.5
DMR	1.25	1.5
CAS-ANOVA	.75	5.25
gvcm	.75	5.5
group MCP	.75	-
group lasso	.5	-

Table 5.11: Characteristics of models chosen by GIC for some c for Experiment 2, $p < n$, linear regression.

Algorithm	c	TM	TF	MD (sd)	PE (sd)	time
DMRnet	1.5	616	842	4.97 (.02)	.122 (.004)	.29
DMR	1.25	483	727	5.21 (.03)	.145 (.004)	.17
CAS-ANOVA	.75	132	704	7 (.06)	.144 (.003)	13.51
gvcm	.75	232	731	6.47 (.05)	.127 (.003)	10.61
group MCP	.75	0	973	11.15 (.03)	.122 (.002)	.02
group lasso	.5	0	18	21.7 (.13)	.203 (.003)	.02
DMR	1.5	506	836	4.95 (.02)	.145 (.004)	.17
CAS-ANOVA	5.25	371	994	5.02 (.03)	.335 (.006)	13.51
gvcm	5.5	499	990	4.9 (.02)	.269 (.008)	10.61
DMRnet	2.5	360	988	4.34 (.02)	.188 (.004)	.29
DMR	2.5	348	977	4.37 (.02)	.195 (.004)	.17

Figure 5.9: Left side: PE vs MD for Experiment 2, $p < n$, linear regression. Right side: 50 runs of Experiment 2, linear regression, $p < n$, PE vs MD.

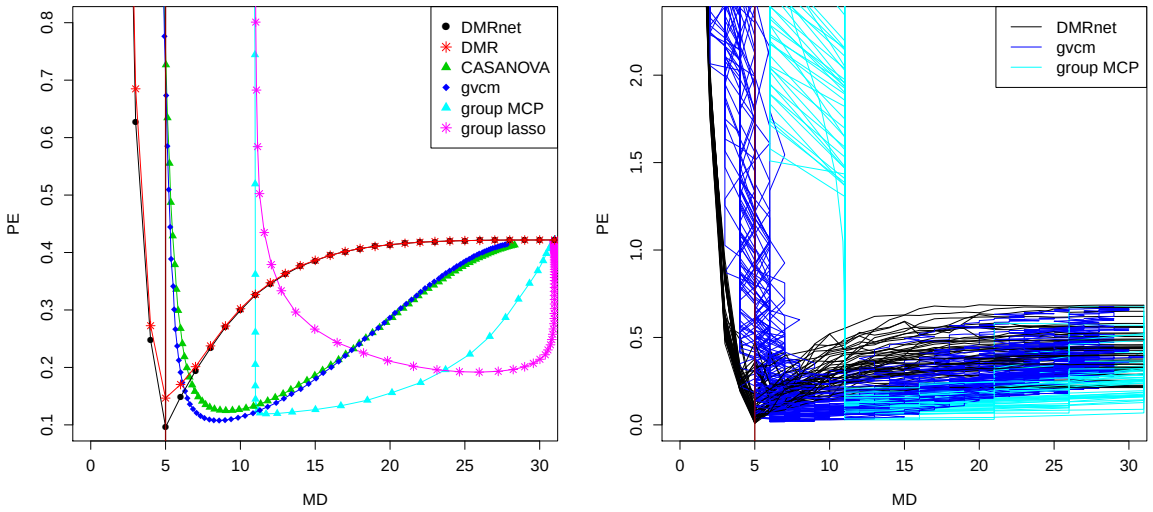
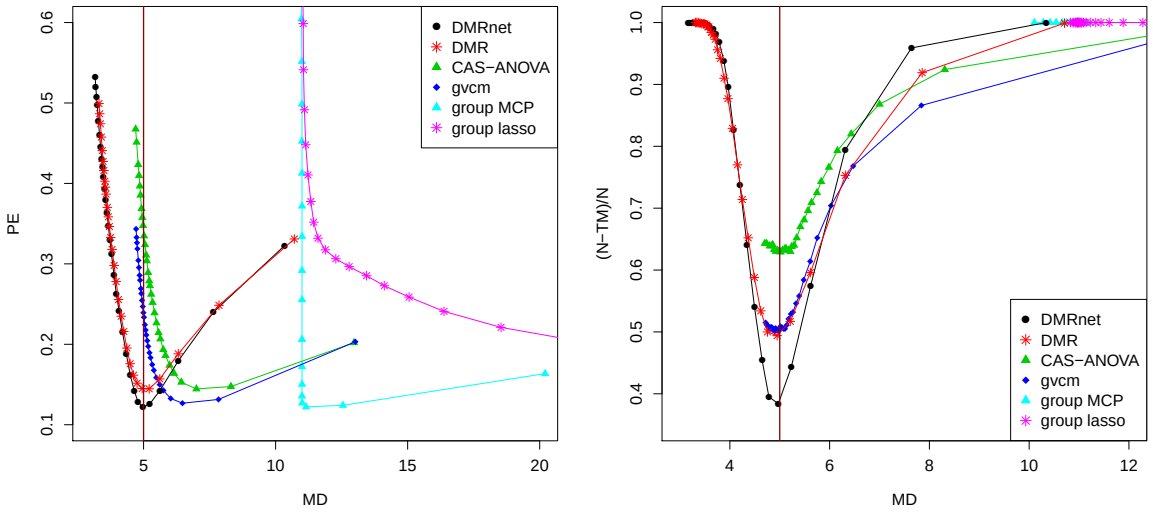


Figure 5.10: Left side: PE vs MD for Experiment 2, $p < n$, linear regression for models chosen by GIC. Right side: (N-TM)/N vs MD for Experiment 2, $p < n$, linear regression for models chosen by GIC.



5.2.5 Experiment 3, linear regression, $p < n$

Table 5.12: Values of minimal PE with the corresponding MD for Experiment 3, $p < n$, linear regression.

	MD	PE
DMRnet	6	.101
DMR	6	.154
CAS-ANOVA	8.12	.103
gvcm	8.12	.103
group MCP	8.13	.079
group lasso	18.13	.154

Table 5.13: Values of interest for c for Experiment 3, $p < n$, linear regression.

Algorithm	optimal c in terms of PE	optimal c in terms of TM
DMRnet	2.25	2.25
DMR	2.25	2.25
CAS-ANOVA	1.25	4
gvcm	1.25	4
group MCP	1.5	-
group lasso	.75	-

Table 5.14: Characteristics of models chosen by GIC for some c for Experiment 3, $p < n$, linear regression.

Algorithm	c	TM	TF	MD (sd)	PE (sd)	time
DMRnet	2.25	612	806	5.94 (.02)	.122 (.003)	1.72
DMR	2.25	473	771	5.91 (.02)	.143 (.003)	.63
CAS-ANOVA	1.25	525	609	6.76 (.04)	.116 (.002)	22.33
gvcm	1.25	491	559	6.78 (.05)	.117 (.002)	25.08
group MCP	1.5	0	744	7.78 (.05)	.089 (.002)	.03
group lasso	.75	0	0	17.93 (.1)	.157 (.002)	.03
CAS-ANOVA	4	740	952	5.86 (.02)	.142 (.003)	22.33
gvcm	4	699	893	5.73 (.02)	.147 (.003)	25.08
DMRnet	2.5	591	863	5.82 (.02)	.126 (.003)	1.72
DMR	2.5	457	823	5.78 (.02)	.144 (.003)	.63

Figure 5.11: Left side: PE vs MD for Experiment 3, $p < n$, linear regression. Right side: 50 runs of Experiment 2, linear regression, $p < n$, PE vs MD.

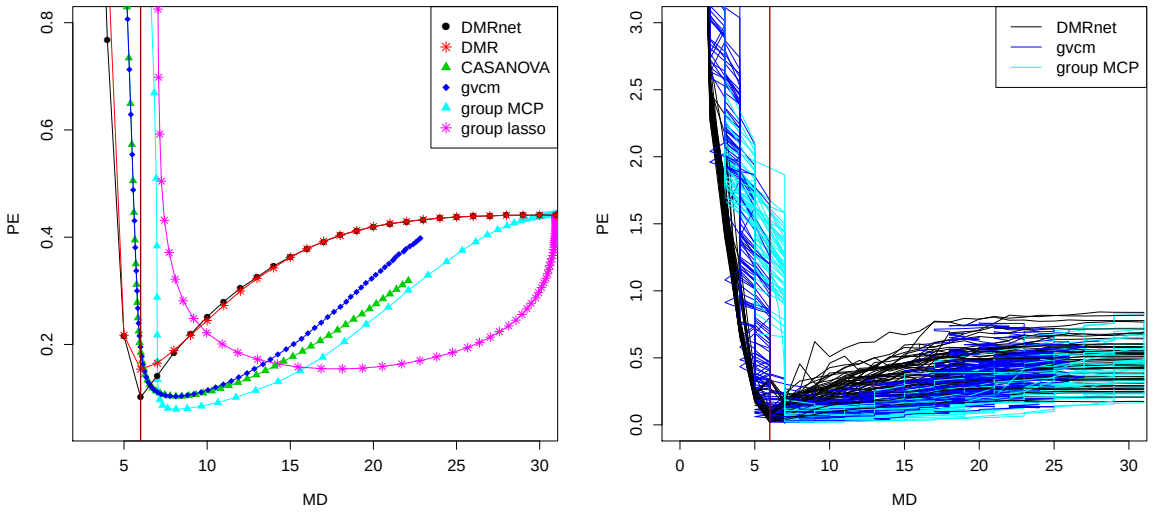
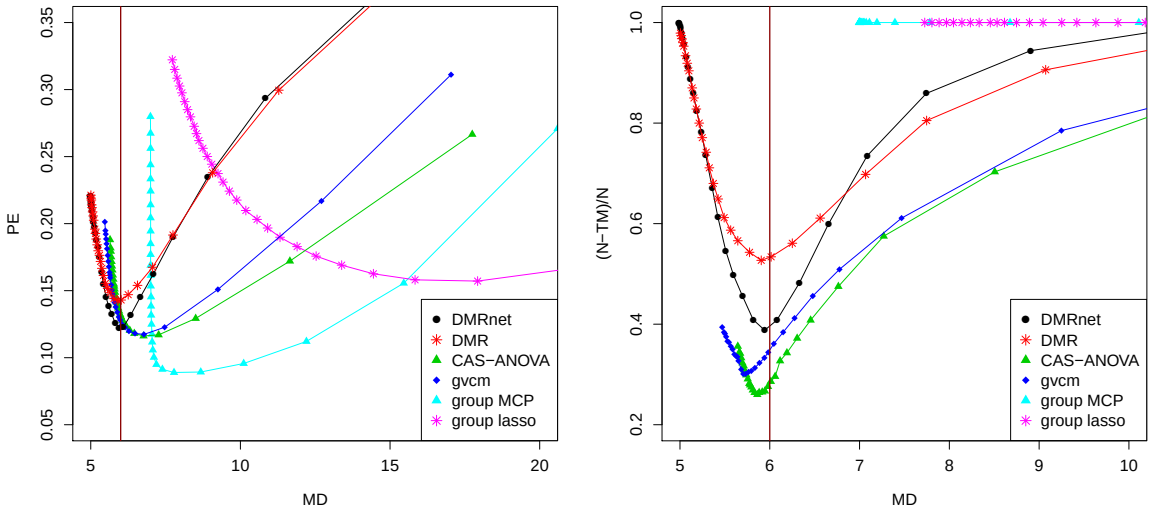


Figure 5.12: Left side: PE vs MD for Experiment 3, $p < n$, linear regression for models chosen by GIC. Right side: (N-TM)/N vs MD for Experiment 3, $p < n$, linear regression for models chosen by GIC.



5.2.6 Experiment 1, linear regression, $p \gg n$

Table 5.15: Values of minimal PE with the corresponding MD for Experiment 1, $p \gg n$, linear regression.

	MD	PE
DMRnet	3	.066
group MCP	51.84	.122
group lasso	97.97	.180

Table 5.16: Values of interest for c for Experiment 1, $p \gg n$, linear regression.

Algorithm	optimal c in terms of PE	optimal c in terms of TM
DMRnet	2.5	2.5
group MCP	.75	-
group lasso	.5	-

Table 5.17: Characteristics of models chosen by GIC for some c for Experiment 1, $p \gg n$, linear regression.

Algorithm	c	TM	TF	MD (sd)	PE (sd)	time
DMRnet	2.5	631	699	2.91 (.02)	.069 (.003)	7.99
group MCP	.75	0	14	32.33 (.35)	.126 (.002)	.75
group lasso	.5	0	0	113.43 (.67)	.178 (.002)	.66

Figure 5.13: Left side: PE vs MD for Experiment 1, $p \gg n$, linear regression. Right side: 100 runs of Experiment 1, linear regression, $p \gg n$, PE vs MD.

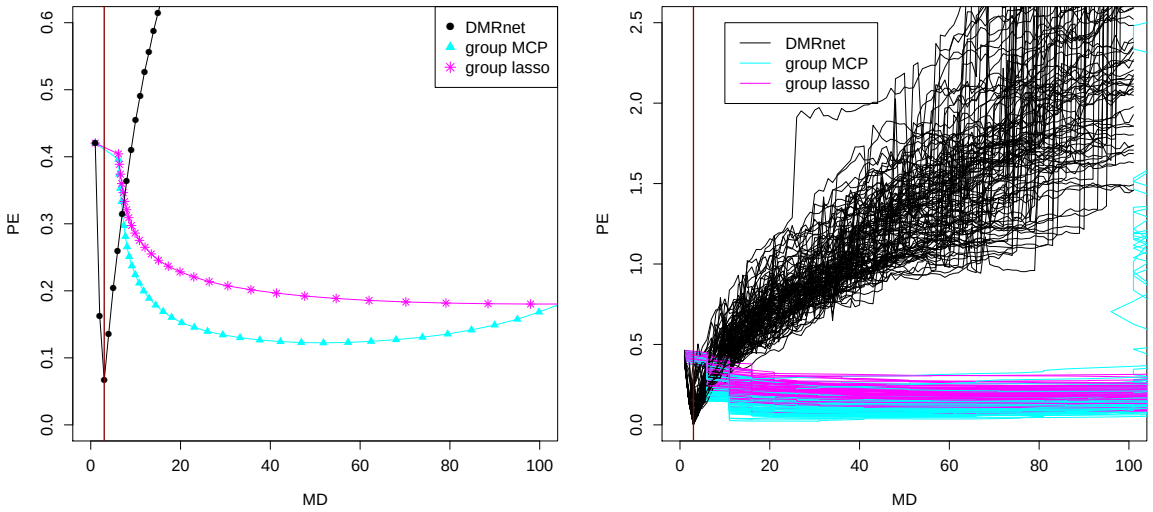
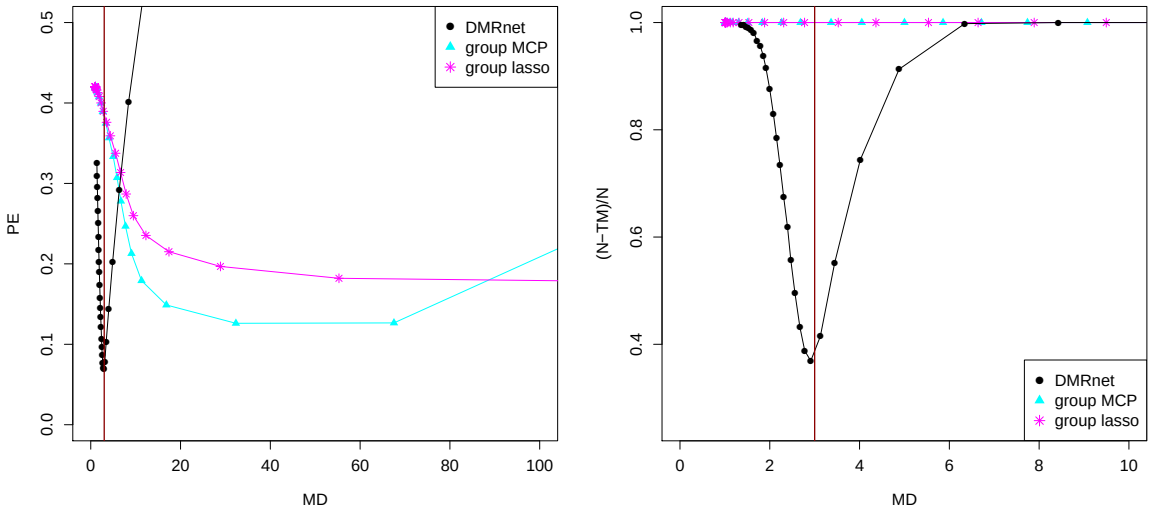


Figure 5.14: Left side: PE vs MD for Experiment 1, $p \gg n$, linear regression for models chosen by GIC. Right side: (N-TM)/N vs MD for Experiment 1, $p \gg n$, linear regression for models chosen by GIC.



5.2.7 Experiment 2, linear regression, $p \gg n$

Table 5.18: Values of minimal PE with the corresponding MD for Experiment 2, $p \gg n$, linear regression.

	MD	PE
DMRnet	5	.115
group MCP	15.3	.061
group lasso	101.48	.194

Table 5.19: Values of interest for c for Experiment 2, $p \gg n$, linear regression.

Algorithm	optimal c in terms of PE	optimal c in terms of TM
DMRnet	2.25	2.25
group MCP	1	-
group lasso	.5	-

Table 5.20: Characteristics of models chosen by GIC for some c for Experiment 2, $p \gg n$, linear regression.

Algorithm	c	TM	TF	MD (sd)	PE (sd)	time
DMRnet	2.25	272	701	4.69 (.02)	.133 (.003)	9.21
group MCP	1	0	938	11.39 (.05)	.064 (.001)	.48
group lasso	.5	0	0	112.31 (.59)	.194 (.002)	.44
DMRnet	2.5	215	843	4.37 (.02)	.133 (.003)	9.21

Figure 5.15: Left side: PE vs MD for Experiment 2, $p \gg n$, linear regression. Right side: 100 runs of Experiment 2, linear regression, $p \gg n$, PE vs MD.

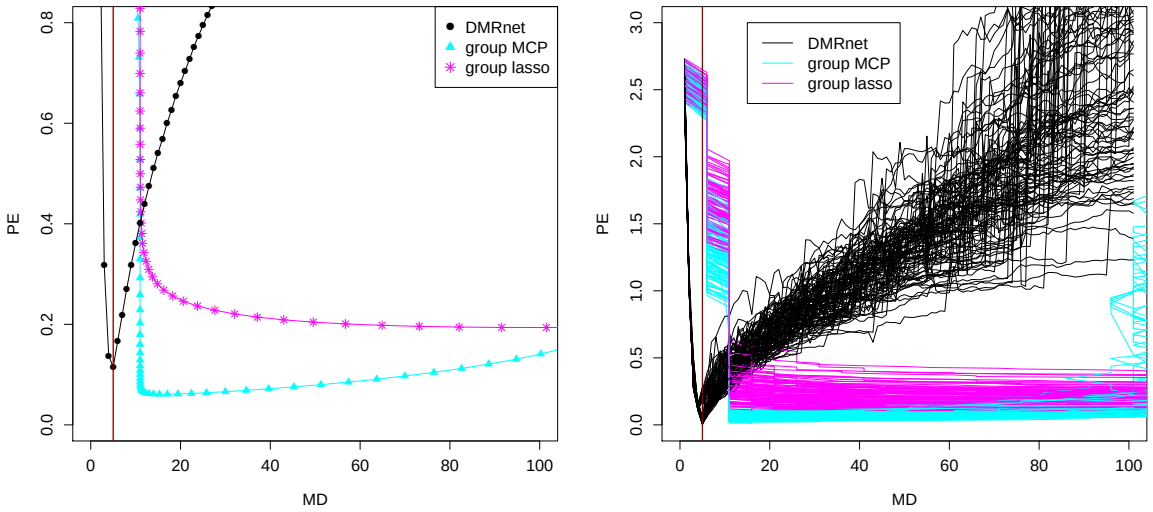
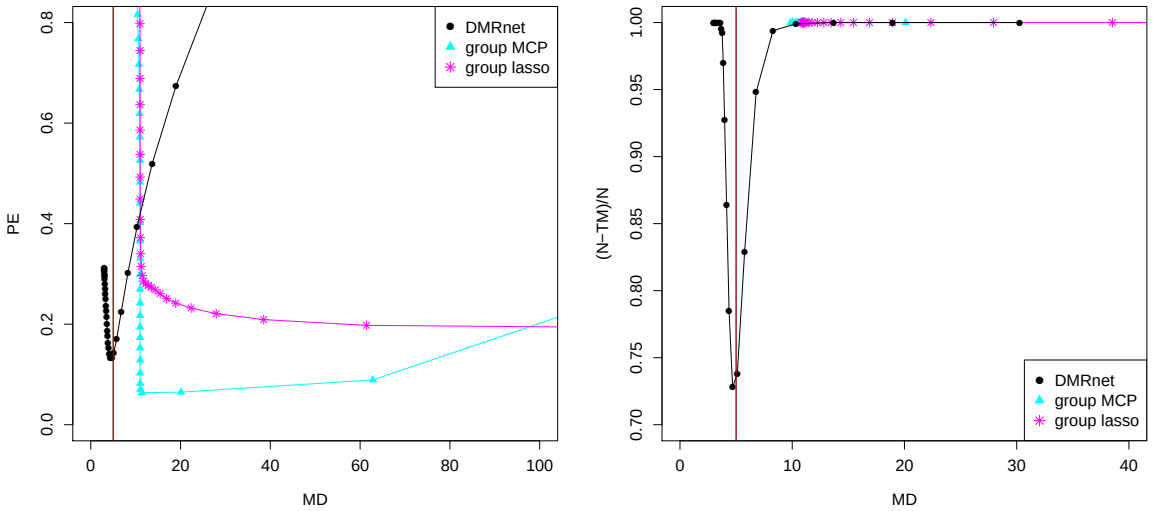


Figure 5.16: Left side: PE vs MD for Experiment 2, $p \gg n$, linear regression for models chosen by GIC. Right side: (N-TM)/N vs MD for Experiment 2, $p \gg n$, linear regression for models chosen by GIC.



5.2.8 Experiment 3, linear regression, $p \gg n$

Table 5.21: Values of minimal PE with the corresponding MD for Experiment 3, $p \gg n$, linear regression.

	MD	PE
DMRnet	6	.05
group MCP	9.16	.04
group lasso	52.15	.19

Table 5.22: Values of interest for c for Experiment 3, $p \gg n$, linear regression.

Algorithm	optimal c in terms of PE	optimal c in terms of TM
DMRnet	2.25	2.25
group MCP	1.5	-
group lasso	1	-

Table 5.23: Characteristics of models chosen by GIC for some c for Experiment 3, $p \gg n$, linear regression.

Algorithm	c	TM	TF	MD (sd)	PE (sd)	time
DMRnet	2.25	769	898	5.95 (.01)	.063 (.002)	19.7
group MCP	1.5	0	697	9.11 (.12)	.049 (.001)	1.16
group lasso	1	0	0	49.1 (.27)	.192 (.002)	1.16
DMRnet	2.5	736	953	5.82 (.01)	.069 (.002)	19.7

Figure 5.17: Left side: PE vs MD for Experiment 3, $p \gg n$, linear regression. Right side: 100 runs of Experiment 3, linear regression, $p \gg n$, PE vs MD.

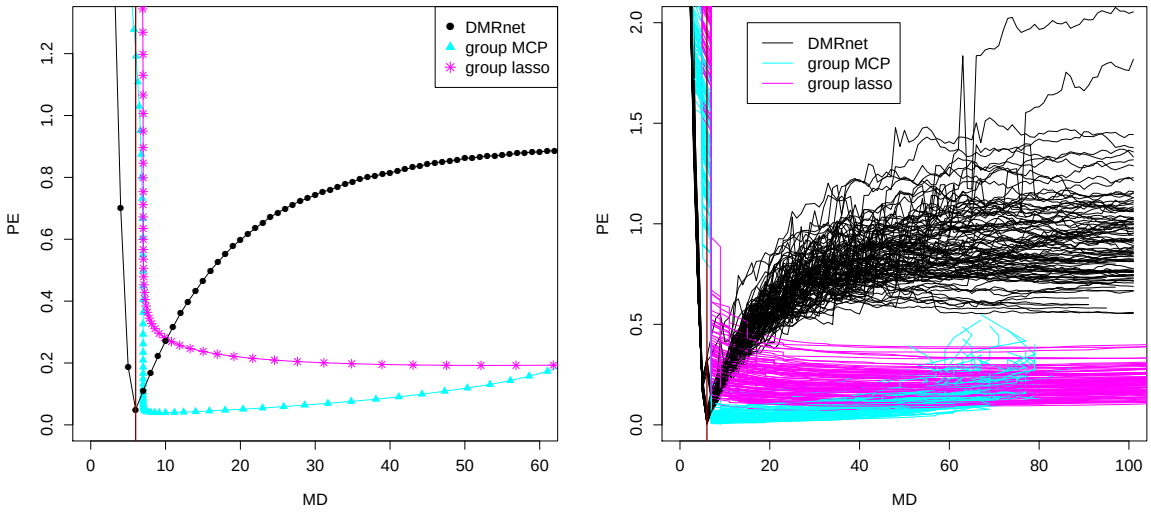
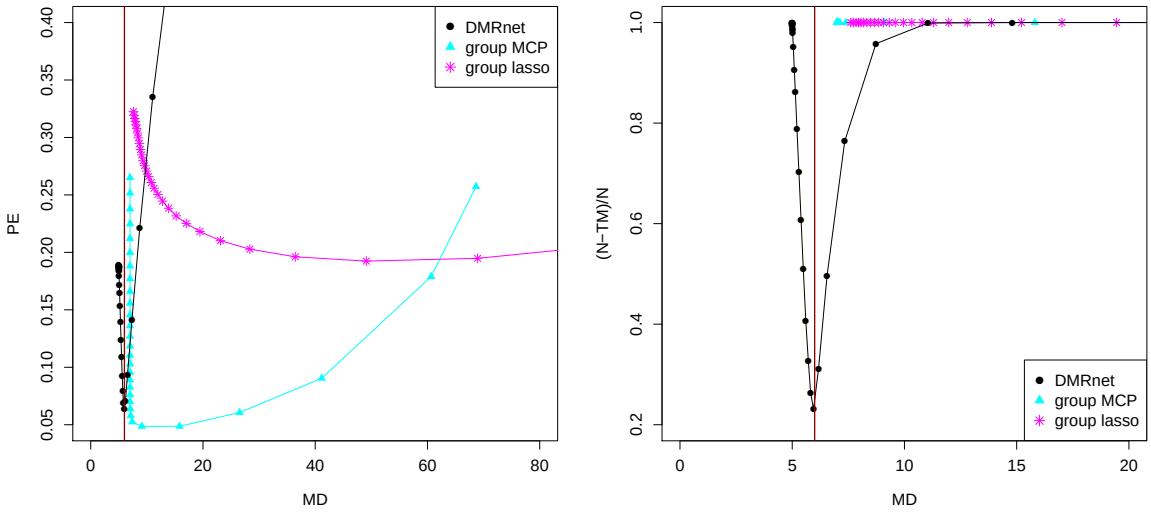


Figure 5.18: Left side: PE vs MD for Experiment 3, $p \gg n$, linear regression for models chosen by GIC. Right side: (N-TM)/N vs MD for Experiment 3, $p \gg n$, linear regression for models chosen by GIC.



Chapter 6

Numerical experiments for logistic regression

In order to compare the performance of our algorithms with those described in the literature, analysis of 3 real data sets and a simulation study based on 6 scenarios have been performed. We confronted performances of 7 algorithms: our DMR4glm (denoted as DMR), DMR4glm_wald (DMR_wald), DMRnet4glm (DMRnet) and DMRnet4glm_wald (DMRnet_wald) and 3 competitive: gvcm, grpreg MCP (group MCP) and grpreg group lasso (group lasso). In opposition to the linear regression case, CAS-ANOVA does not work for logistic regression. While gvcm solves the same factorial selection problem as the DMR algorithms, group MCP and group lasso choose whole factors. We want two answer two questions: if the DMR algorithms are better than the lasso based methods in terms of minimizing prediction error and choosing sparse models and whether the effort of merging levels of factors really pays of? We have to keep in mind that in terms of computation time group MCP and group lasso are more efficient, obviously. We also want to check which DMR algorithms are better: the ones using likelihood ratio test or Wald statistics?

In all the cases for the DMRnet algorithms we set parameters o=5, nlambda=20 and for DMR and DMRnet algorithms complete linkage in hierarchical clustering were used. Since the `glm` function often gives NA results when the data is linearly separable, we use ridge regression instead (`glmnet` package) with a very low value of penalty for the second norm of the parameters vector, usually $r_l = 10^{-7}$. For group MCP and group lasso there are two ways of calculating the degrees of freedom when calculating GIC: df_1 implemented in the package `grpreg` and described in Breheny and Huang [2015] and df_2 equal to |M|. We present only the results for df_1 since the results for df_2 were similar. For gvcm λ grid (with 100 λ values) was chosen on the log scale so that $|M|$ took many values from 1 to p. Moreover, for group MCP and group lasso the default 100-elements λ grids were used.

6.1 Real data examples

Model selection was done using 7 algorithms on 3 data sets: Promoter, Mem and Knee. We used 10-fold cross-validation (C-V) to calculate mean prediction error (PE) and mean model dimension (MD) for 100 λ values for CAS-ANOVA, gvcm, group MCP and group lasso and for model dimension from 1 to p for DMR and DMR_wald and from 1 to $\min\{p, \frac{n}{4}\}$ for

DMRnet and DMRnet_wald. Moreover, PE and MD were calculated for models selected by $GIC_M = -2\ell(\widehat{\boldsymbol{\beta}}_M) + c \cdot \log(p) \cdot df$ (df stands for degrees of freedom and equals $|M|$ in all of the cases except for df_1 for group MCP and group lasso) for different values of $c = .25, .5, .75, \ldots, 7, 8.5, 10$.

The results are organized in tables and 2 types of plots. In the first type points corresponding to $(MD(\lambda), PE(\lambda))$, $\lambda = \lambda_1, \ldots, \lambda_{100}$ for each algorithm: gvcm, group MCP and group lasso and analogously $(MD(m), PE(m))$ $m = 1, \ldots, p$ for DMR algorithms and $m = 1, \ldots, \min\{p, \frac{n}{4}\}$ for DMRnet algorithms are plotted together. In the second type points corresponding to $(MD(c), PE(c))$, $c \in \mathcal{C}_3 = \{.25, .5, .75, \ldots, 7, 8.5, 10\}$, where $MD(c)$ and $PE(c)$ are calculated for models chosen by GIC with parameter c, for all the algorithms are plotted together.

6.1.1 Estimation of prediction error using 10-fold cross-validation

The estimated prediction error (PE) and the standard deviation were calculated similarly as in Section 5.1.1, differing only in definition of the loss function:

$$w_{tl} = L(y_t^{(l)}, \widehat{y}_t^{(l)}) = \mathbb{1}(y_t^{(l)} \neq \widehat{y}_t^{(l)}), \text{ where } \widehat{y}_t^{(l)} = \begin{cases} 1 & (\mathbf{x}_{t\cdot}^{(l)})^T \widehat{\boldsymbol{\beta}}^{(l)} > 0 \\ 0 & \text{otherwise} \end{cases},$$

where $\widehat{\boldsymbol{\beta}}^{(l)}$ is the parameter vector estimated using Λ_l.

The estimation of MD and standard deviation is exactly the same as in Section 5.1.1.

6.1.2 Promoter data set

The data is available from the UCI machine learning repository: https://archive.ics.uci.edu/ml/datasets/Molecular+Biology+%28Promoter+Gene+Sequences%29 and has been discussed in Towell et al. [1990]. It consists of E. Coli promoter gene sequences starting at position -50 (p-50) and ending at position +7 (p7). Each of these 57 fields is filled by one of {a, g, t, c}. The task is to recognize promoters, which are genetic regions which initiate the first step in the expression of adjacent genes (transcription). There are 53 promoters and 53 nonpromoter sequences ($n = 106$). Changing each of 57 fields to factors with 4 levels gives $p = 172$ parameters. Algorithms DMR and DMR_wald could not be used since $p > n$.

In Figure 6.1 a plot of PE vs MD calculated by 10-fold C-V for 100 λ values for gvcm, group MCP and group lasso and from 1 to $\min\{p, \frac{n}{4}\}$ for DMRnet and DMRnet_wald is shown. For every algorithm we can find a global minimum: for DMRnet and DMRnet_wald these are when MD = 5, for gvcm when MD = 36.8, for group MCP when MD = 13.6 and for group lasso when MD=21.4. If we chose models with the lowest prediction error, DMRnet_wald would have both the smallest error and the smallest number of parameters.

In Figure 6.2 PE vs MD for models chosen by GIC, $c \in \mathcal{C}_3$ are presented. Again, the DMRnet_wald algorithm gives the smallest prediction error among the competition and sparse models.

In Table 6.1 there are characteristics of models chosen by GIC (MD and PE with standard deviations) for optimal c: $c = 1.5$ gives minimal PE for DMRnet, $c = 3$ for DMRnet_wald, $c = .25$ for gvcm, group MCP and group lasso. Additionally, results for $c = 2$ for DMRnet and DMRnet_wald are given.

Table 6.1: MD and PE with standard deviations for Promoter data set for the models selected using GIC for optimal c.

Algorithm	c	MD (sd)	PE (sd)
DMRnet	3	4.9 (.1)	.065 (.024)
DMRnet_wald	1.5	5 (0)	.047 (.021)
gvcm	.25	30.1 (1.6)	.074 (.035)
group MCP	.25	13.6 (.4)	.074 (.023)
group lasso	.25	36.4 (1.1)	.074 (.027)
DMRnet	2	5 (0)	.084 (.032)
DMRnet_wald	2	5 (0)	.047 (.021)

Figure 6.1: PE vs MD calculated by 10-fold C-V for Promoter data set.

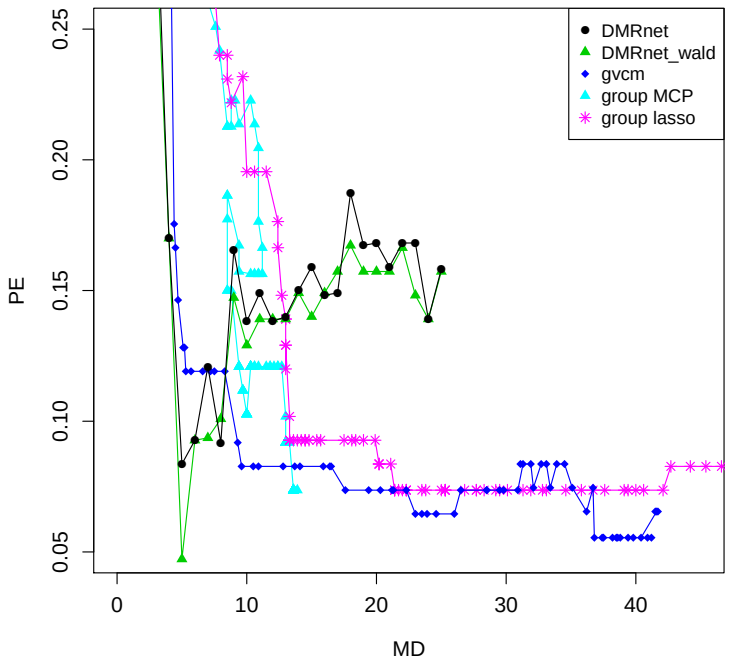
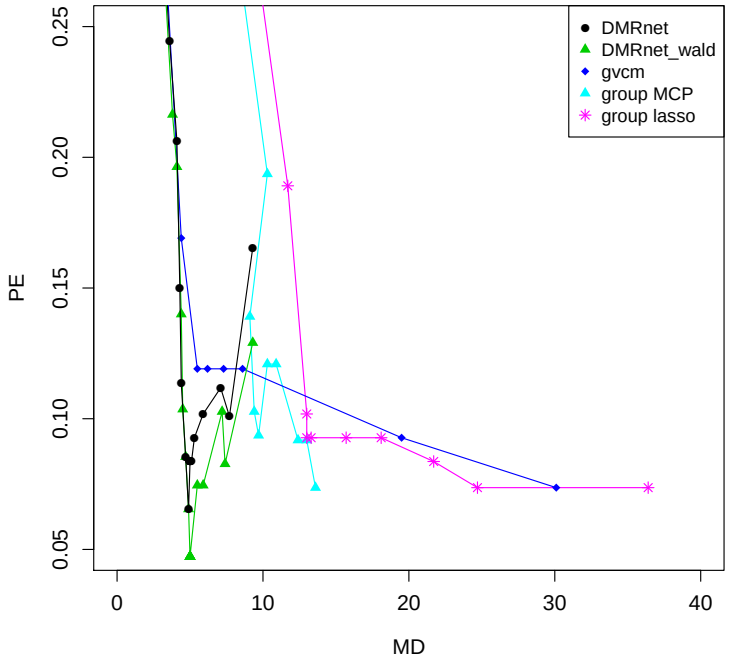


Figure 6.2: PE vs MD calculated by 10-fold C-V for Promoter data set for models chosen by GIC.



6.1.3 Mem data set

Mem data set concerns prediction of splice sites. Splice sites are regions of DNA between coding (exons) and non-coding (introns) segments. The 5' end of an intron is called a donor splice site. A donor site whose two first intron positions are letters "GT" is called canonical. A sequence of a real splice site consists of the last three bases of the exon and the first six bases of the intron. False splice sites are sequenced on the DNA which match the consensus sequence at positions 4 and 5. Removing the consensus GT results in sequence length of 7 with values in {A,C,G,T}, thus the predictor variables are seven factors, each having four levels. The data are available at `http://genes.mit.edu/burgelab/maxent/ssdata/` and described in more detail in Yeo and Burge [2004]. Training set consists of 8 415 true and 179 438 false human donor sites. These data has been analysed in Bühlmann and Van De Geer [2011] and Meier et al. [2008].

The original training data set is used to build a smaller balanced training data set (5610 true and 5610 false donor sites, $n = 11220$) chosen randomly without replacement. The candidate model that was used for the logistic group lasso consists of all two-way and lower order interactions involving 29 terms with $p = 1 + 3 * 7 + 21 * 9 = 211$ parameters. This is the same setup as described in Meier et al. [2008]. The gvcm function could not be used since an error occurred since the problem was too computationally intensive.

In Figure 6.3 a plot of PE vs MD calculated by 10-fold C-V for 100 λ values for group MCP and group lasso and from 1 to p for DMR and DMR_wald and from 1 to $\min\{p, \frac{n}{4}\}$ for DMRnet and DMRnet_wald is shown. For every algorithm we can find a global minimum: for DMRnet and DMR this is when MD = 108, for DMR this is when MD = 120, for DMRnet_wald and DMR_wald when MD = 153, for group MCP when MD = 142.6 and for group lasso when MD = 194.8. If we chose models with the lowest prediction error, DMRnet_wald would have both the smallest error and the smallest number of parameters. Additionally, we have local minimums for DMRnet and DMRnet_wald when MD = 8 . In particular, for DMRnet the PE for MD = 8 is not much greater than PE for bigger models.

In Figure 6.4 PE vs MD for models chosen by GIC, $c \in \mathcal{C}_3$ are presented. Again, the DMRnet algorithm gives the smallest prediction error among the competition and sparse models.

In Table 6.2 there are characteristics of models chosen by GIC (MD and PE with standard deviations) for optimal c: $c = 1.5$ gives minimal PE for DMRnet, $c = 3$ for DMRnet_wald, $c = .25$ for group MCP and group lasso. Additionally, results for $c = 2$ for DMRnet and DMR are given.

Figure 6.3: PE vs MD calculated by 10-fold C-V for Mem data set.

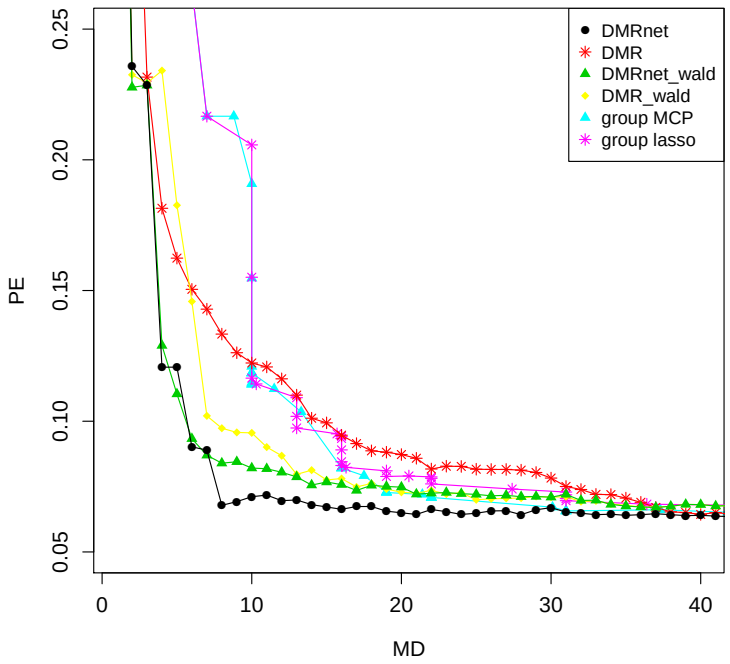


Figure 6.4: PE vs MD calculated by 10-fold C-V for Mem data set for models chosen by GIC.

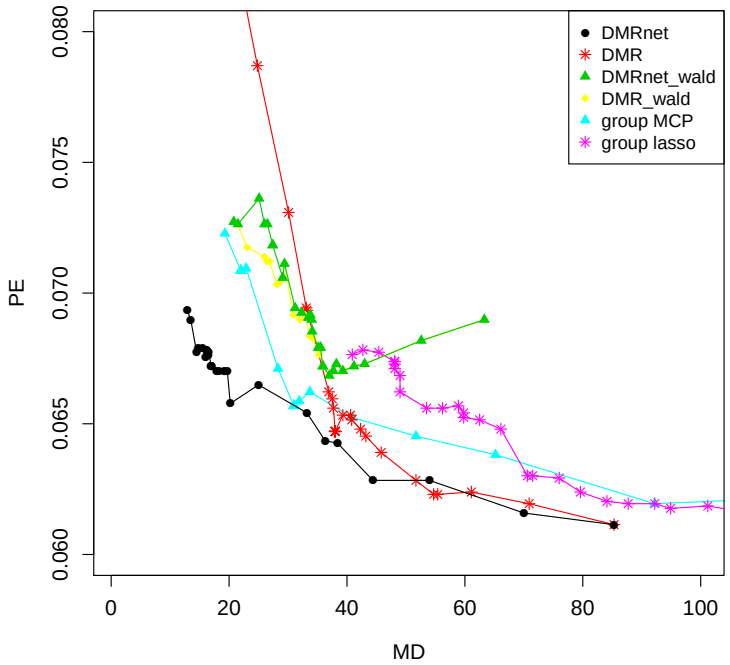


Table 6.2: MD and PE with standard deviations for Mem data set for the models selected using GIC for optimal c.

Algorithm	c	MD (sd)	PE (sd)
DMRnet	.25	85.3 (3)	.061 (.002)
DMR	.25	85.3 (3)	.061 (.002)
DMRnet_wald	2	37 (.5)	.067 (.002)
DMR_wald	2	37 (.5)	.067 (.002)
group MCP	.5	92.2 (3.5)	.062 (.002)
group lasso	1	130.9 (.9)	.06 (.002)
DMRnet	2	25 (2)	.066 (.002)
DMR	2	43.2 (1.9)	.065 (.002)

6.1.4 Knee data set

Data set `knee` comes from `R` package `catdata` accompanying Tutz [2011]. It consists of $n = 127$ observations of patients with sport related injuries have been treated with two different therapies (chosen by random design, 63 patients with placebo and 64 with treatment). There are 6 explanatory variables: age, gender and intensification of pain (no pain = 1, severe pain = 5) in 4 time moments: before the therapy and after 3, 7 and 10 days, $p = 19$.

In Figure 6.5 a plot of PE vs MD calculated by 10-fold C-V for 100 λ values for gvcm, group MCP and group lasso and from 1 to p for DMR and DMR_wald and from 1 to $\min\{p, \frac{n}{4}\}$ for DMRnet and DMRnet_wald is shown. For every algorithm we can find a global minimum: for DMRnet this is when MD = 10, for DMRnet_wald, DMR, gvcm, group MCP and group lasso these are when MD = 19 (the full model) and for DMR_wald when MD = 15. Additionally, we have local minimums for some algorithms when MD $\approx$ 5. In particular, for DMRnet and DMRnet_wald the PE for MD = 5 is not much greater than PE for bigger models.

In Figure 6.6 PE vs MD for models chosen by GIC, $c \in \mathcal{C}_3$ are presented.

In Table 6.3 there are characteristics of models chosen by GIC (MD and PE with standard deviations) for optimal c: $c = .5$ gives minimal PE for DMRnet, $c = .25$ for DMR, DMR_wald, gvcm and group lasso and $c = .75$ for DMRnet_wald and group MCP. Additionally, results for $c = 2$ for DMRnet, DMRnet_wald, DMR and DMR_wald are given.

Figure 6.5: PE vs MD calculated by 10-fold C-V for Knee data set.

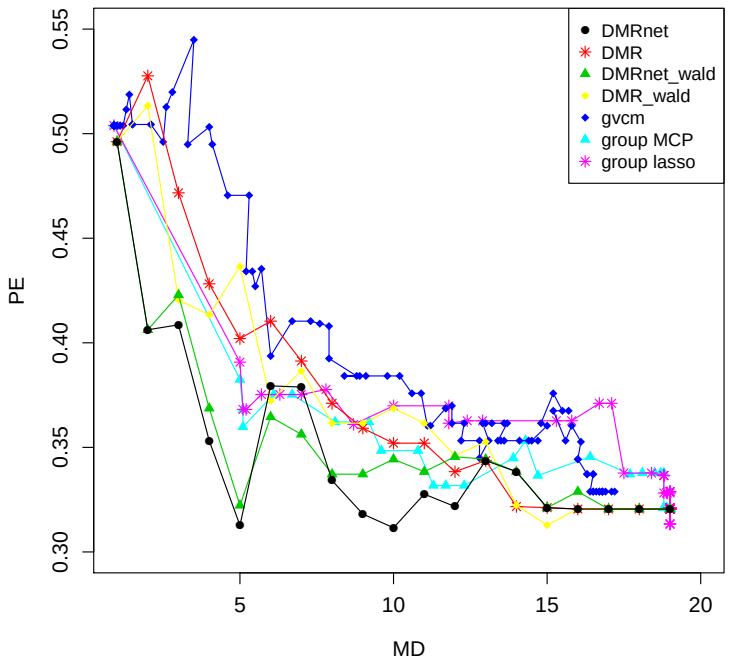


Figure 6.6: PE vs MD calculated by 10-fold C-V for Knee data set for models chosen by GIC.

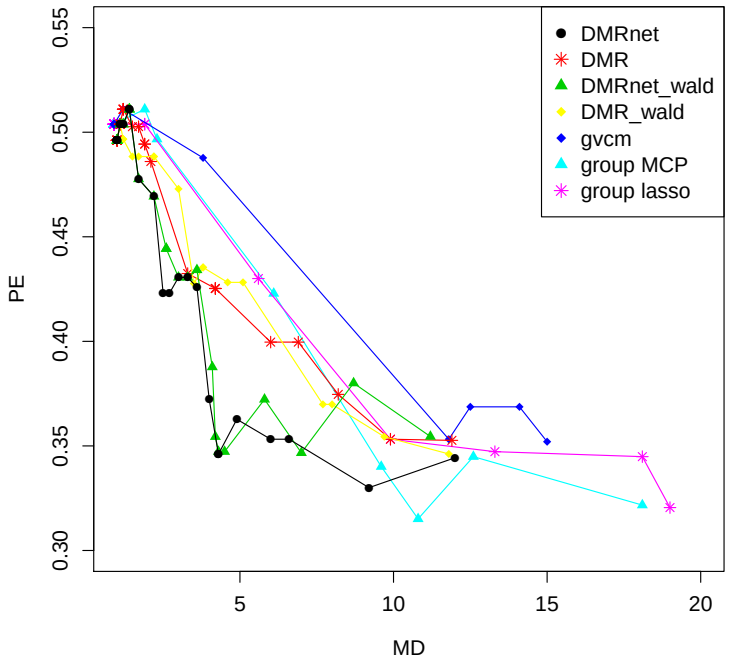


Table 6.3: MD and PE with standard deviations for Knee data set for the models selected using GIC for optimal c.

DMRnet	.5	9.2 (.7)	.330 (.045)
DMR	.25	11.9 (.4)	.353 (.05)
DMRnet_wald	.75	7 (.6)	.347 (.042)
DMR_wald	.25	11.8 (.8)	.346 (.048)
gvcm	.25	15 (.7)	.352 (.055)
group MCP	.75	10.8 (.6)	.315 (.046)
group lasso	.25	19 (0)	.321 (.055)
DMRnet	2	4 (.1)	.372 (.051)
DMR	2	4.2 (.4)	.425 (.032)
DMRnet_wald	2	4.1 (.2)	.388 (.049)
DMR_wald	2	3.8 (.5)	.435 (.034)

6.2 Simulation study

The simulation study for logistic regression was designed in a similar way as for linear regression. The following 7 algorithms were compared: DMR4glm (DMR), DMR4glm_wald (DMR_wald), DMRnet4glm (DMRnet), DMRnet4glm_wald (DMRnet_wald), gvcm, grpreg MCP (group MCP) and grpreg group lasso (group lasso).

Independent random samples $\{\mathbf{x}_{i\cdot}^T, y_i\}$, $i = 1, \ldots, n$ from the logistic regression model (4.1) were generated: y_i was sampled from Bernoulli distribution with probability $\exp(\mathbf{x}_{i\cdot}\boldsymbol{\beta}^*)/(1 + \exp(\mathbf{x}_{i\cdot}\boldsymbol{\beta}^*))$. Plan of experiments is presented in Tables 6.4 (for 3 $p < n$ scenarios) and 6.5 (for 3 $p \gg n$ scenarios), where l stands for the number of factors with equal numbers o levels, t for the dimension of the true model. Parameter vector $\boldsymbol{\beta}^*$ can have one of three forms the same as in experiments for linear regression (equations 5.1, 5.2 and 5.3). The model matrix $\mathbf{X}$ was sampled in the same way as for the linear regression case.

For every experiment the results are based on $N = 1000$ simulation runs.

Table 6.4: Plan of experiments for logistic regression, $p < n$.

exp	n	p	l	t	$\boldsymbol{\beta}^*$	$\Xi(\rho)$
1	300	31	6	3	$\boldsymbol{\beta}_1^*$	AR(.5)
2	300	31	6	5	$\boldsymbol{\beta}_2^*$	AR(.5)
3	300	31	15	6	$\boldsymbol{\beta}_3^* \cdot 4$	AR(.5)

Table 6.5: Plan of experiments for logistic regression, $p \gg n$.

exp	n	p	l	t	$\boldsymbol{\beta}^*$	$\Xi(\rho)$
1	400	3001	600	3	$\boldsymbol{\beta}_1^*$	AR(.5)
2	400	2001	400	5	$\boldsymbol{\beta}_2^* \cdot 2$	AR(.5)
3	400	2001	1000	6	$\boldsymbol{\beta}_3^* \cdot 6$	AR(.5)

6.2.1 Estimation of prediction error

We reported the number of times the true model was correctly identified (denoted by TM), the number of times all of the true factors were kept and false factors set to zero (denoted by TF), mean model dimension (denoted by MD), mean prediction error on a test data set with 1000 observations (denoted by PE) and mean execution time (denoted by time). Moreover, statistics were calculated for models selected by $GIC_M = -2\ell(\widehat{\boldsymbol{\beta}}_M) + c \cdot \log(p) \cdot df$ (df stands for degrees of freedom and equals $|M|$ in all of the cases except for df_1 for MCP and group lasso), for different values of $c \in \mathcal{C}_3$.

The estimated prediction error (PE) and the standard deviation were calculated similarly as in Section 5.2.1, differing only in definition of the loss function:

$$w_{tl} = L(y_t^{(l)}, \widehat{y}_t^{(l)}) = \mathbb{1}(y_t^{(l)} \neq \widehat{y}_t^{(l)}), \text{ where } \widehat{y}_t^{(l)} = \begin{cases} 1 & \frac{\exp(\widehat{\eta}^{(l)})}{1+\exp(\widehat{\eta}^{(l)})} > \frac{1}{2} \\ 0 & \text{otherwise} \end{cases},$$

$\widehat{\eta}^{(l)} = (\mathbf{x}_{t\cdot}^{(l)})^T \widehat{\boldsymbol{\beta}}^{(l)}$ and $\widehat{\boldsymbol{\beta}}^{(l)}$ is the parameter vector estimated using Λ_l.

The estimation of MD and standard deviation is exactly the same as in Section 5.2.1.

6.2.2 Results

The results for all the simulation setups are organized in 4 types of figures and 3 types of tables:

1. Figures:

   (i) Points corresponding to $(MD(\lambda), PE(\lambda))$, $\lambda = \lambda_1, \ldots, \lambda_{100}$ for each algorithm: gvcm, group MCP and group lasso and $(MD(m), PE(m))$ $m = 1, \ldots, p$ for DMR and DMR_wald algorithms and $m = 1, \ldots, \min\{p, \frac{n}{4}\}$ for DMRnet and DMRnet_wald algorithms are plotted together. The dark red vertical lines denote the dimension of the true model. They are given on the left sides of Figures: 6.7, 6.9, 6.11, 6.13, 6.15, 6.17.

   (ii) 50 runs of the simulations for $p < n$ and 100 runs for $p \gg n$ are given showing the variability of the results for DMRnet, gvcm and group MCP when $p < n$ and for DMRnet and group MCP when $p \gg n$. The dark red vertical lines denote the dimension of the true model. They are given on the right sides of Figures: 6.7, 6.9, 6.11, 6.13, 6.15, 6.17.

   (iii) Points corresponding to $(MD(c), PE(c))$, $c \in \mathcal{C}_3$, where $MD(c)$ and $PE(c)$ are calculated for models chosen by GIC with parameter c, for all the algorithms are plotted together. The dark red vertical lines denote the dimension of the true model. They are given on the left sides of Figures: 6.8, 6.10, 6.12, 6.14, 6.16, 6.18.

   (iv) Points corresponding to $(MD(c), SE(c))$, $c \in \mathcal{C}_3$, where the selection error is calculated as $SE = \frac{N-TM}{N}$, for all the algorithms are given. The dark red vertical lines denote the dimension of the true model. They are given on the right sides of Figures: 6.8, 6.10, 6.12, 6.14, 6.16, 6.18.

2. Tables:

(i) Values of minimal PE with the corresponding MD corresponding to figures (i). They are given in Tables: 6.6, 6.9, 6.12, 6.15, 6.18, 6.21.

(ii) Optimal values of c in terms of PE and TM (giving minimal PE and maximal TM) for models were chosen by GIC. They are given in Tables: 6.7, 6.10, 6.13, 6.16, 6.19, 6.22.

(iii) Characteristics of models: TM, TF, MD with standard deviation, PE with standard deviation and time for models chosen by GIC for the values of c given in the second type of tables and additionally for $c = 2$, which is the value which usually gives good results for DMR and DMRnet. They are given in Tables: 6.8, 6.11, 6.14, 6.17, 6.20, 6.23.

For experiments when $p \gg n$ there are no results for DMR and DMR_wald since they work only if $p < n$ and no results for gvcm since the problem was too computationally intensive for this implementation and it ended with an error. Additionally, there are no results for gvcm for experiment 3 when $p < n$ since it also ended with an error.

Looking at figures (i) and tables (i) for experiments 1 when $p < n$ and 1 when $p \gg n$, we can see that if we chose models with the lowest prediction error, DMRnet algorithms would have the smallest error and the smallest number of parameters. In experiments 2 when $p < n$ and 2 when $p \gg n$, group MCP gives the smallest PE among competition (slightly smaller than DMRnet algorithms), but it chooses much larger models than DMRnet algorithms. In experiment 3 when $p < n$ group MCP gives the smallest PE among competition, but it chooses larger models than DMRnet algorithms. In experiment 3 when $p \gg n$ the results for group MCP and DMRnet algorithms are very similar.

Looking at figures (ii) for experiments $p < n$, the variability of the results for gvcm is higher than for DMRnet and group MCP. For all the experiments the variability of DMRnet, compared to the variability of the competitive methods, is lower around the dimension of the true model and higher for bigger models.

Looking at figures (iii) and tables (iii) for experiments 1 when $p < n$ and 1 when $p \gg n$, the DMRnet algorithms give the smallest prediction error among the competition and the sparsest models. In experiments 2 when $p < n$ and 2 when $p \gg n$, group MCP gives the smallest PE among competition (slightly smaller than DMRnet algorithms), but it chooses much larger models than DMRnet algorithms. In experiment 3 when $p < n$ group MCP gives the smallest PE among competition, but it chooses larger models than DMRnet algorithms. In experiment 3 when $p \gg n$ the results for group MCP and DMRnet algorithms are very similar.

Looking at figures (iv) DMRnet chooses the true model most often in all the experiments. Let us notice that group MCP and group lasso always give SE equal to 1 (TM=0) since they don't split factors.

Looking at figures (iii) and (iv) let us notice that for DMR and DMRnet algorithms approximately the same number of parameters (around t, which is the true model dimension) gives minimal PE and minimal SE, which is not true for other algorithms. Looking at tables (ii) we can see that similar values of c are optimal in terms of PE and SE (TM) for DMR and DMRnet algorithms.

Looking at tables (iii) we can see that in terms of time, group MCP and group lasso have much smaller values than DMRnet and DMR algorithms, but they don't split factors. Algorithm gvcm is much more time demanding.

In all the setups DMRnet_wald is not worse than DMRnet in terms of PE and SE.

Value of $c = 2$ gives good results for DMRnet algorithms in all the setups.

6.2.3 Experiment 1, logistic regression, $p < n$

Table 6.6: Values of minimal PE with the corresponding MD for Experiment 1, $p < n$, logistic regression.

	MD	PE
DMRnet	3	.226
DMR	3	.226
DMRnet_wald	3	.226
DMR_wald	3	.226
gvcm	4.68	.226
group MCP	8.65	.226
group lasso	12.66	.227

Table 6.7: Values of interest for c for Experiment 1, $p < n$, logistic regression.

Algorithm	optimal c in terms of PE	optimal c in terms of TM
DMRnet	5.75	4.5
DMR	5.75	4.5
DMRnet_wald	5.75	4.5
DMR_wald	5.75	4.5
gvcm	2	4.75
group MCP	2	-
group lasso	1.5	-

Figure 6.7: Left side: PE vs MD for Experiment 1, $p < n$, logistic regression. Right side: 50 runs of Experiment 1, logistic regression, $p < n$, PE vs MD.

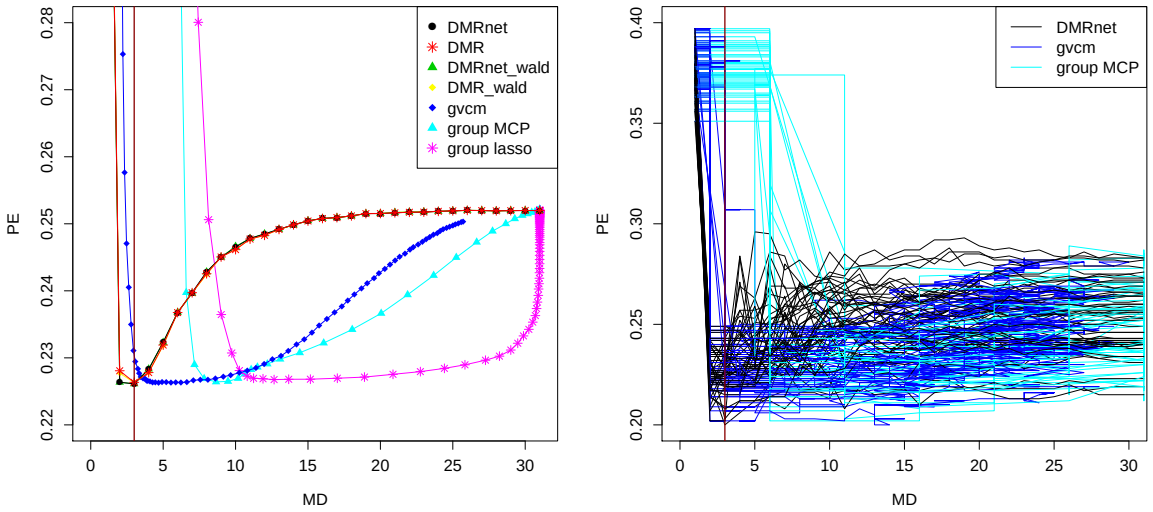


Figure 6.8: Left side: PE vs MD for Experiment 1, $p < n$, logistic regression for models chosen by GIC. Right side: (N-TM)/N vs MD for Experiment 1, $p < n$, logistic regression for models chosen by GIC.

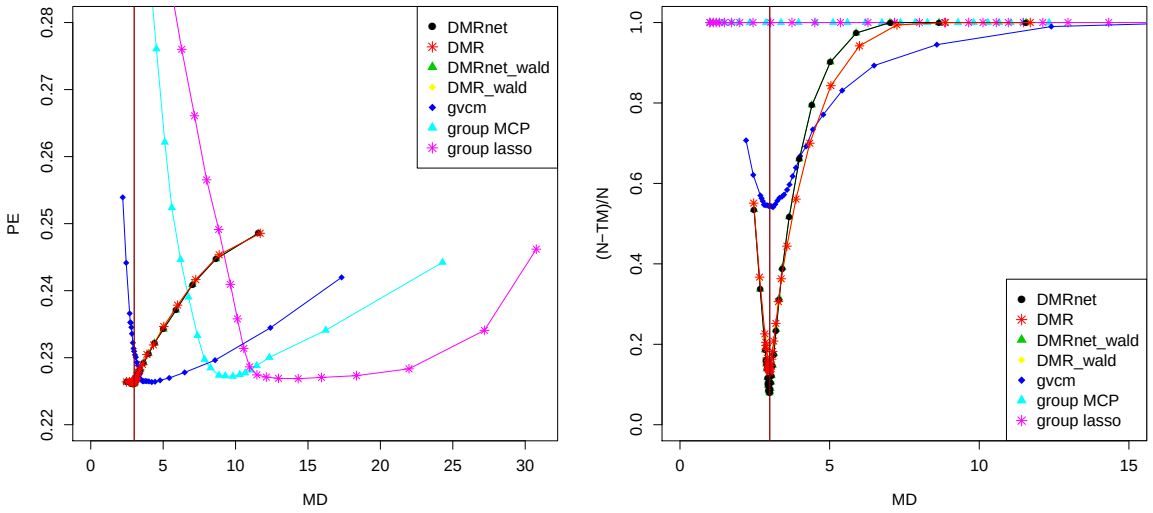


Table 6.8: Characteristics of models chosen by GIC for some c for Experiment 1, $p < n$, logistic regression.

Algorithm	c	TM	TF	MD (sd)	PE (sd)	time
DMRnet	5.75	897	938	2.94 (.01)	.226 (.001)	2.47
DMR	5.75	847	925	2.93 (.01)	.226 (.001)	.75
DMRnet_wald	5.75	896	938	2.94 (.01)	.226 (.001)	.99
DMR_wald	5.75	853	926	2.93 (.01)	.226 (.001)	.29
gvcm	2	308	734	4.21 (.04)	.226 (.001)	14.08
group MCP	2	0	764	9.81 (.07)	.227 (.001)	.03
group lasso	1.5	0	400	14.32 (.1)	.227 (.001)	.04
DMRnet	4.5	922	972	2.99 (.01)	.226 (.001)	2.47
DMR	4.5	869	966	2.99 (.01)	.226 (.001)	.75
DMRnet_wald	4.5	921	972	2.99 (.01)	.226 (.001)	.99
DMR_wald	4.5	875	966	2.99 (.01)	.226 (.001)	.29
gvcm	4.75	459	754	3.11 (.03)	.23 (.001)	14.08
DMRnet	2	484	573	3.65 (.02)	.229 (.001)	2.47
DMR	2	556	659	3.57 (.03)	.229 (.001)	.75
DMRnet_wald	2	483	571	3.65 (.02)	.229 (.001)	.99
DMR_wald	2	557	656	3.57 (.03)	.229 (.001)	.29

6.2.4 Experiment 2, logistic regression, $p < n$

Table 6.9: Values of minimal PE with the corresponding MD for Experiment 2, $p < n$, logistic regression.

	MD	PE
DMRnet	6	.146
DMR	6	.147
DMRnet_wald	5	.146
DMR_wald	6	.148
gvcm	10.38	.143
group MCP	11.02	.14
group lasso	26.45	.145

Figure 6.9: Left side: PE vs MD for Experiment 2, $p < n$, logistic regression. Right side: 50 runs of Experiment 2, logistic regression, $p < n$, PE vs MD.

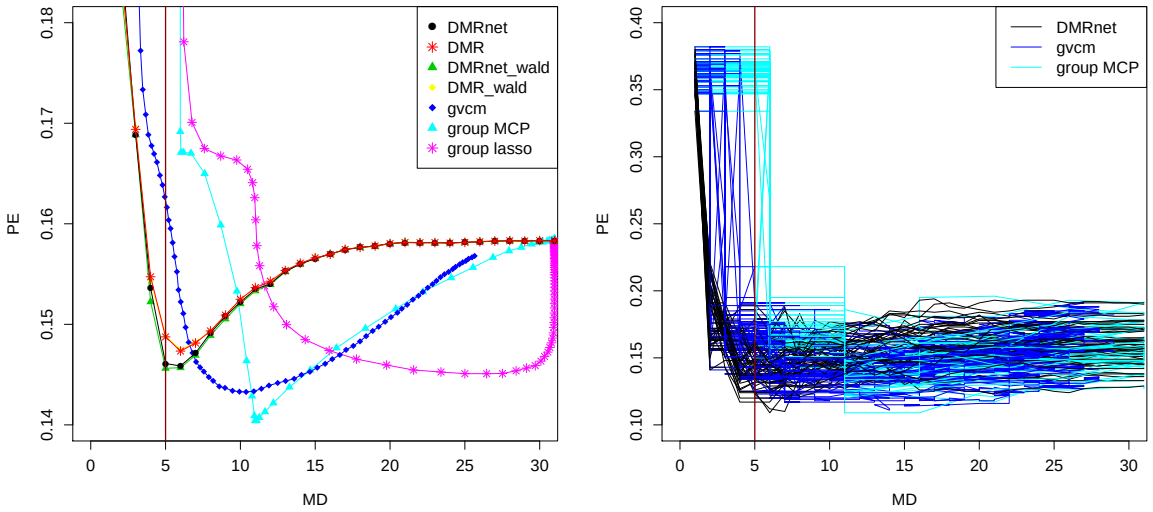


Figure 6.10: Left side: PE vs MD for Experiment 2, $p < n$, logistic regression for models chosen by GIC. Right side: (N-TM)/N vs MD for Experiment 2, $p < n$, logistic regression for models chosen by GIC.

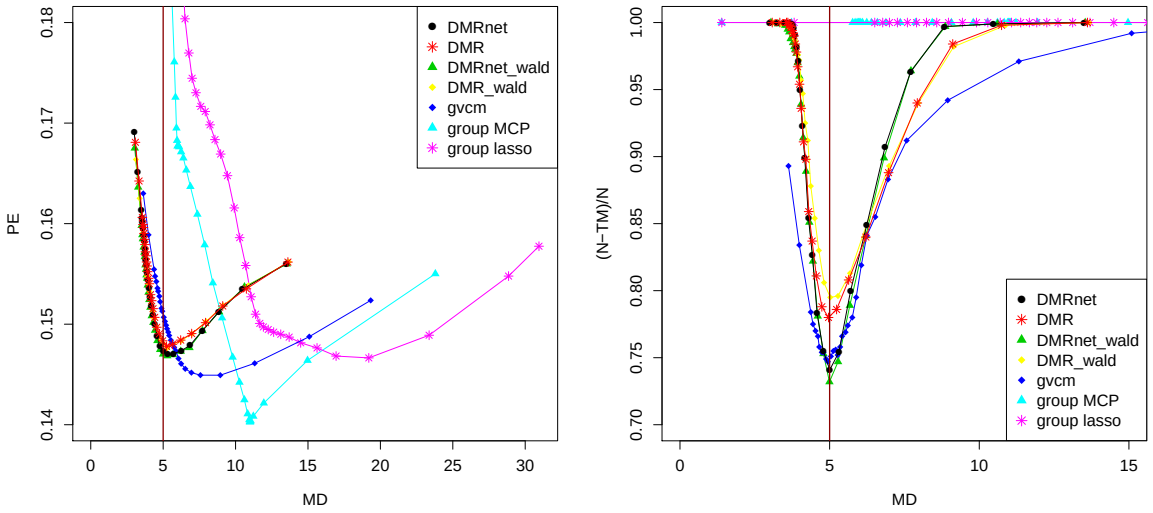


Table 6.10: Values of interest for c for Experiment 2, $p < n$, logistic regression.

Algorithm	optimal c in terms of PE	optimal c in terms of TM
DMRnet	2	2.25
DMR	2	2.25
DMRnet_wald	2	2.25
DMR_wald	2	2.25
gvcm	1	5
group MCP	1.5	-
group lasso	1	-

Table 6.11: Characteristics of models chosen by GIC for some c for Experiment 2, $p < n$, logistic regression.

Algorithm	c	TM	TF	MD (sd)	PE (sd)	time
DMRnet	2	246	542	5.31 (.03)	.147 (.001)	3.29
DMR	2	214	596	5.23 (.03)	.148 (.001)	1.04
DMRnet_wald	2	253	546	5.28 (.03)	.147 (.001)	1.37
DMR_wald	2	204	571	5.28 (.04)	.148 (.001)	.4
gvcm	1	58	303	8.95 (.1)	.145 (.001)	23.5
group MCP	1.5	0	999	11.01 (0)	.14 (.001)	.06
group lasso	1	0	58	19.2 (.13)	.147 (.001)	.1
DMRnet	2.25	259	668	5 (.03)	.147 (.001)	3.29
DMR	2.25	220	690	4.96 (.03)	.148 (.001)	1.04
DMRnet_wald	2.25	268	669	4.99 (.03)	.147 (.001)	1.37
DMR_wald	2.25	205	652	5.02 (.03)	.148 (.001)	.4
gvcm	5	253	912	4.94 (.03)	.151 (.001)	23.49

6.2.5 Experiment 3, logistic regression, $p < n$

For experiment 3 results for gvcm are not presented since the call of this function ended with an error.

Table 6.12: Values of minimal PE with the corresponding MD for Experiment 3, $p < n$, logistic regression.

	MD	PE
DMRnet	6	.099
DMR	6	.098
DMRnet_wald	4	.103
DMR_wald	4	.101
group MCP	7.04	.096
group lasso	18.97	.102

Table 6.13: Values of interest for c for Experiment 3, $p < n$, logistic regression.

Algorithm	optimal c in terms of PE	optimal c in terms of TM
DMRnet	2.5	2
DMR	3	1.75
DMRnet_wald	3	2.25
DMR_wald	4	2.5
group MCP	2.5	-
group lasso	1	-

Table 6.14: Characteristics of models chosen by GIC for some c for Experiment 3, $p < n$, logistic regression.

Algorithm	c	TM	TF	MD (sd)	PE (sd)	time
DMRnet	2.5	246	833	5.45 (.02)	.1 (.001)	5.62
DMR	3	121	893	5.13 (.02)	.102 (.001)	.9
DMRnet_wald	3	419	880	5.26 (.03)	.098 (.001)	3.18
DMR_wald	4	9	930	4.16 (.02)	.101 (.001)	.48
group MCP	2.5	0	978	6.98 (.01)	.096 (0)	.13
group lasso	1	0	0	15.46 (.12)	.104 (0)	.19
DMRnet	2	294	660	5.9 (.03)	.101 (.001)	5.62
DMR	1.75	185	559	6.13 (.04)	.103 (.001)	.9
DMRnet_wald	2.25	491	716	5.99 (.03)	.099 (.001)	3.18
DMR_wald	2.5	13	662	5.14 (.05)	.102 (.001)	.48
DMR	2	181	661	5.79 (.03)	.102 (.001)	.9
DMRnet_wald	2	465	655	6.23 (.03)	.099 (.001)	3.18
DMR_wald	2	12	444	6.12 (.06)	.104 (.001)	.48

Figure 6.11: Left side: PE vs MD for Experiment 3, $p < n$, logistic regression. Right side: 50 runs of Experiment 3, logistic regression, $p < n$, PE vs MD.

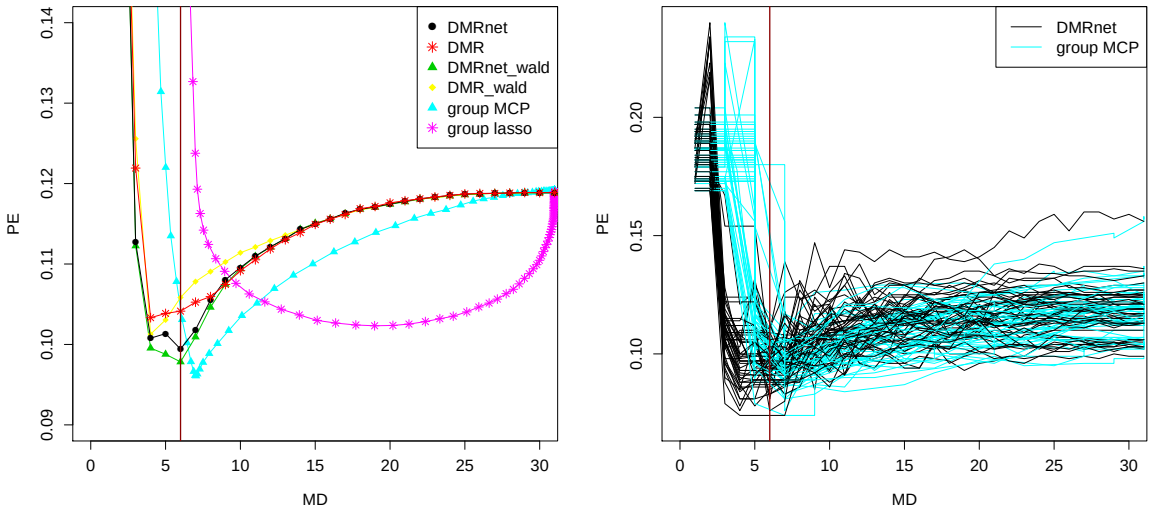
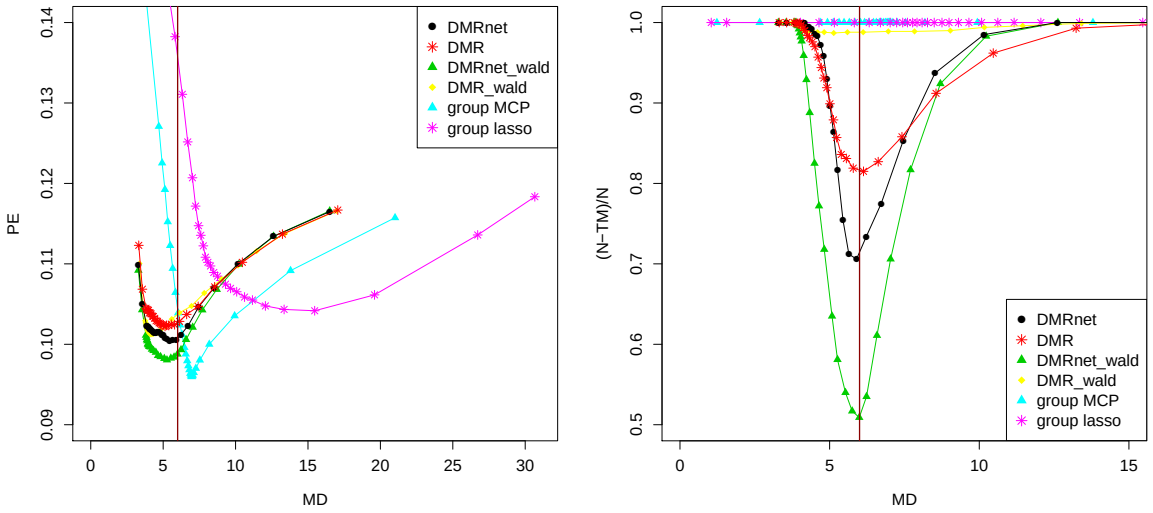


Figure 6.12: Left side: PE vs MD for Experiment 3, $p < n$, logistic regression for models chosen by GIC. Right side: (N-TM)/N vs MD for Experiment 3, $p < n$, logistic regression for models chosen by GIC.



6.2.6 Experiment 1, logistic regression, $p \gg n$

Table 6.15: Values of minimal PE with the corresponding MD for Experiment 1, $p \gg n$, logistic regression.

	MD	PE
DMRnet	3	.227
DMRnet_wald	3	.227
group MCP	9.63	.227
group lasso	17.23	.227

Table 6.16: Values of interest for c for Experiment 1, $p \gg n$, logistic regression.

Algorithm	optimal c in terms of PE	optimal c in terms of TM
DMRnet	5.75	3
DMRnet_wald	5.75	3
group MCP	1	-
group lasso	.75	-

Table 6.17: Characteristics of models chosen by GIC for some c for Experiment 1, $p \gg n$, logistic regression.

Algorithm	c	TM	TF	MD (sd)	PE (sd)	time
DMRnet	5.75	444	452	2.45 (.02)	.227 (.001)	62.66
DMRnet_wald	5.75	444	452	2.45 (.02)	.227 (.001)	17.76
group MCP	1	0	796	10.58 (.07)	.227 (.001)	2.19
group lasso	0.75	0	5	27.7 (.19)	.227 (.001)	.93
DMRnet	3	928	951	2.98 (.01)	.227 (.001)	62.66
DMRnet_wald	3	928	951	2.98 (.01)	.227 (.001)	17.76
DMRnet	2	444	460	3.9 (.04)	.234 (.001)	62.66
DMrnet_wald	2	447	460	3.9 (.04)	.234 (.001)	17.76

Figure 6.13: Left side: PE vs MD for Experiment 1, $p \gg n$, logistic regression. Right side: 50 runs of Experiment 1, logistic regression, $p \gg n$, PE vs MD.

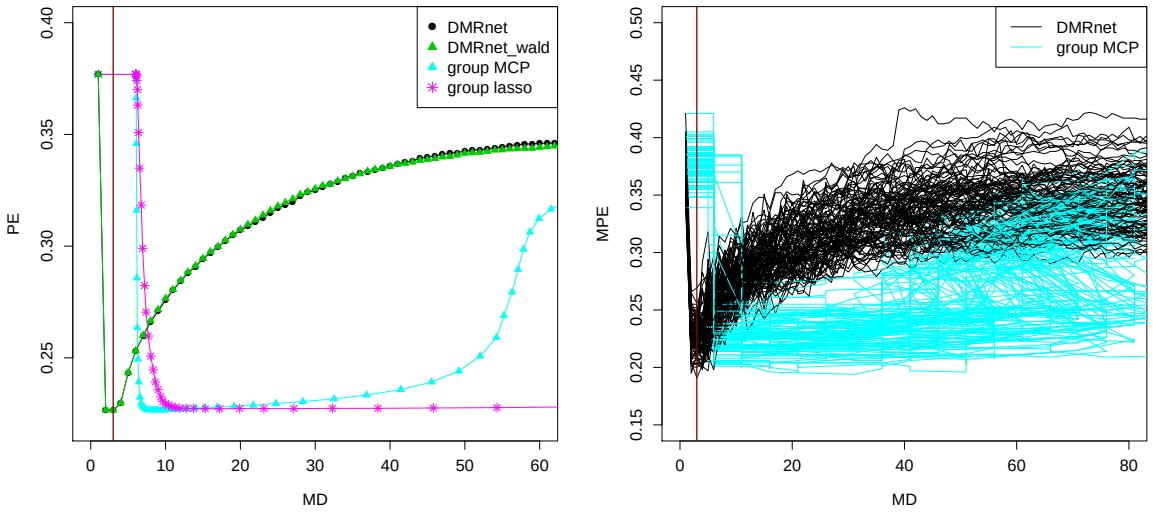
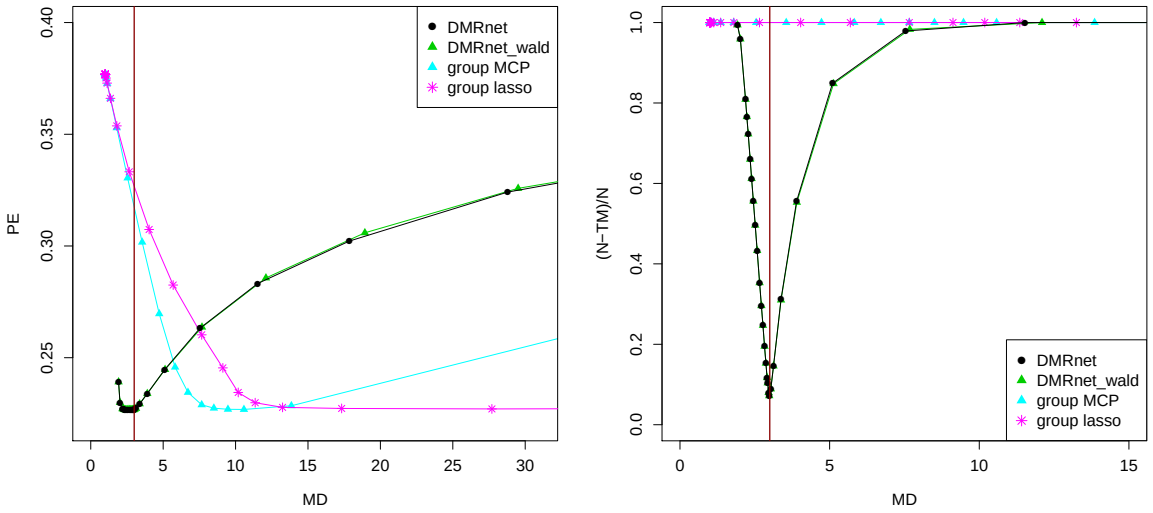


Figure 6.14: Left side: PE vs MD for Experiment 1, $p \gg n$, logistic regression for models chosen by GIC. Right side: (N-TM)/N vs MD for Experiment 1, $p \gg n$, logistic regression for models chosen by GIC.



6.2.7 Experiment 2, logistic regression, $p \gg n$

Table 6.18: Values of minimal PE with the corresponding MD for Experiment 2, $p \gg n$, logistic regression.

	MD	PE
DMRnet	5	.087
DMRnet_wald	5	.087
group MCP	11	.084
group lasso	87.13	.092

Table 6.19: Values of interest for c for Experiment 2, $p \gg n$, logistic regression.

Algorithm	optimal c in terms of PE	optimal c in terms of TM
DMRnet	2.5	2.25
DMRnet_wald	2.5	2.25
group MCP	1.25	-
group lasso	.25	-

Table 6.20: Characteristics of models chosen by GIC for some c for Experiment 2, $p \gg n$, logistic regression.

Algorithm	c	TM	TF	MD (sd)	PE (sd)	time
DMRnet	2.5	312	877	4.55 (.02)	.086 (0)	121.82
DMRnet_wald	2.5	310	874	4.55 (.02)	.086 (0)	43.06
group MCP	1.25	0	1000	11 (0)	.084 (0)	1.67
group lasso	.25	0	0	75.96 (.37)	.092 (0)	.92
DMRnet	2.25	365	746	4.86 (.02)	.086 (0)	121.82
DMRnet_wald	2.25	358	746	4.84 (.02)	.086 (0)	43.06
DMRnet	2	325	515	5.37 (.03)	.088 (0)	121.82
DMRnet_wald	2	323	517	5.35 (.03)	.088 (0)	43.06

Figure 6.15: Left side: PE vs MD for Experiment 2, $p \gg n$, logistic regression. Right side: 100 runs of Experiment 2, logistic regression, $p \gg n$, PE vs MD.

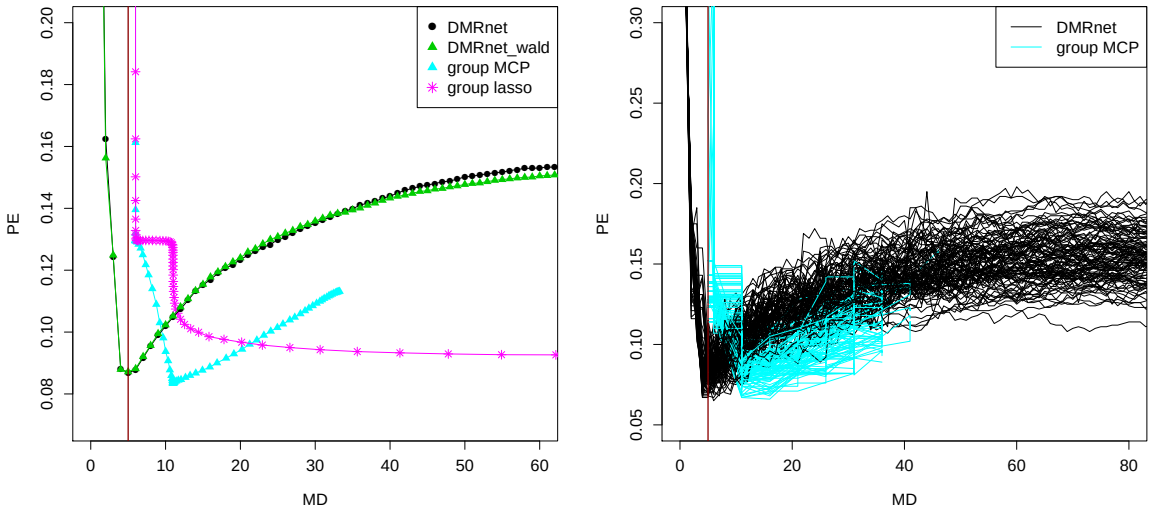
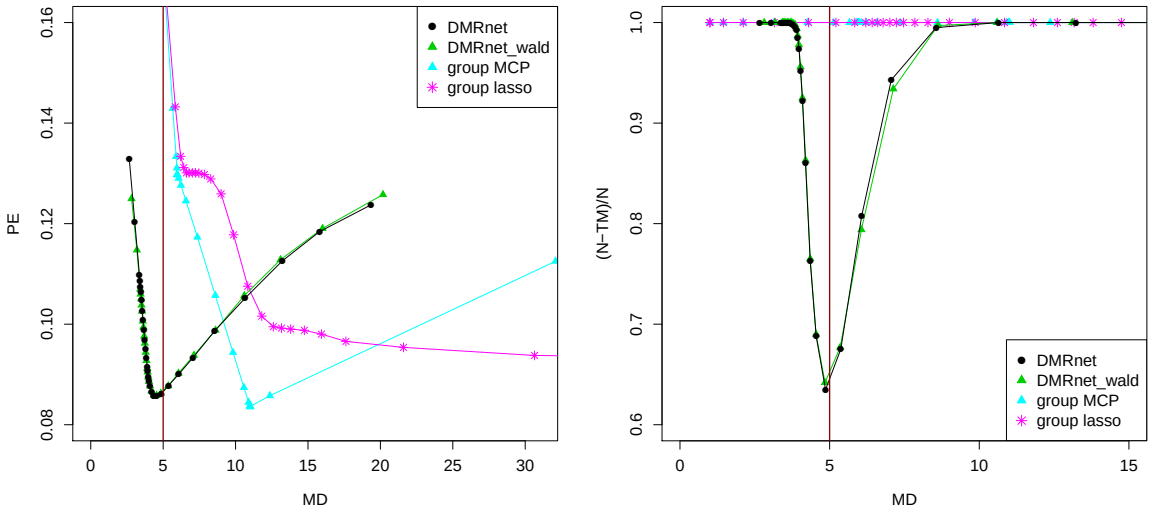


Figure 6.16: Left side: PE vs MD for Experiment 2, $p \gg n$, logistic regression for models chosen by GIC. Right side: (N-TM)/N vs MD for Experiment 2, $p \gg n$, logistic regression for models chosen by GIC.



6.2.8 Experiment 3, logistic regression, $p \gg n$

Table 6.21: Values of minimal PE with the corresponding MD for Experiment 3, $p \gg n$, logistic regression.

	MD	PE
DMRnet	4	.071
DMRnet_wald	4	.07
group MCP	7.02	.07
group lasso	77.79	.089

Table 6.22: Values of interest for c for Experiment 3, $p \gg n$, logistic regression.

Algorithm	optimal c in terms of PE	optimal c in terms of TM
DMRnet	3.75	1.75
DMRnet_wald	3.75	1.75
group MCP	1.75	-
group lasso	.25	-

Table 6.23: Characteristics of models chosen by GIC for some c for Experiment 3, $p \gg n$, logistic regression.

Algorithm	c	TM	TF	MD (sd)	PE (sd)	time
DMRnet	3.75	0	987	4.01 (.01)	.071 (0)	162.62
DMRnet_wald	3.75	0	987	4.01 (.01)	.071 (0)	104.92
group MCP	1.75	0	997	7.01 (0)	.07 (0)	1.29
group lasso	0.25	0	0	77.8 (.28)	.089 (0)	.92
DMRnet	1.75	72	556	5.51 (.03)	.075 (0)	162.62
DMRnet_wald	1.75	87	559	5.52 (.03)	.075 (0)	104.92
DMRnet	2	44	765	4.95 (.02)	.073 (0)	162.62
DMRnet_wald	2	49	774	4.93 (.02)	.073 (0)	104.92

Figure 6.17: Left side: PE vs MD for Experiment 3, $p \gg n$, logistic regression. Right side: 100 runs of Experiment 3, logistic regression, $p \gg n$, PE vs MD.

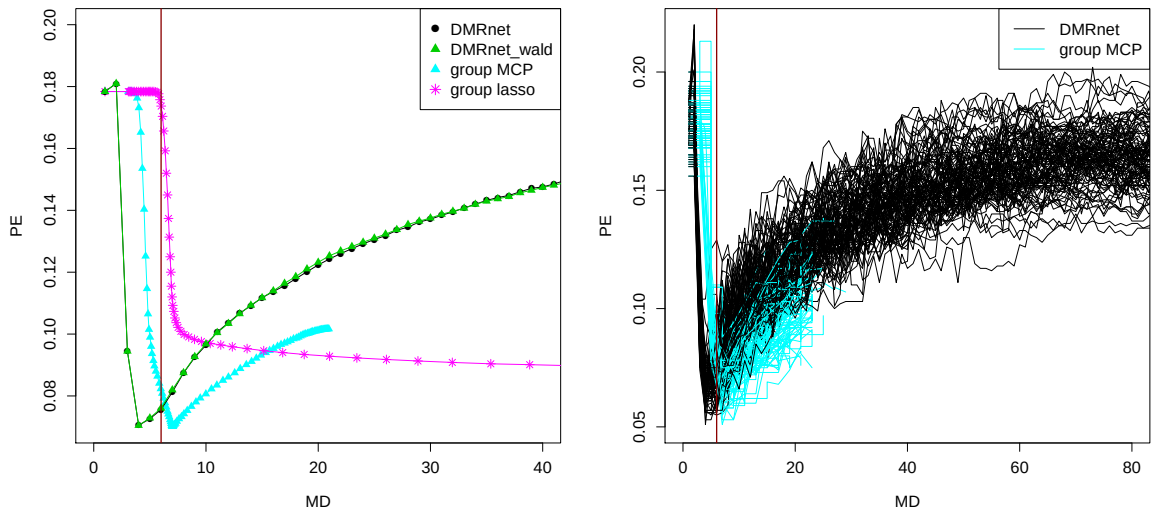
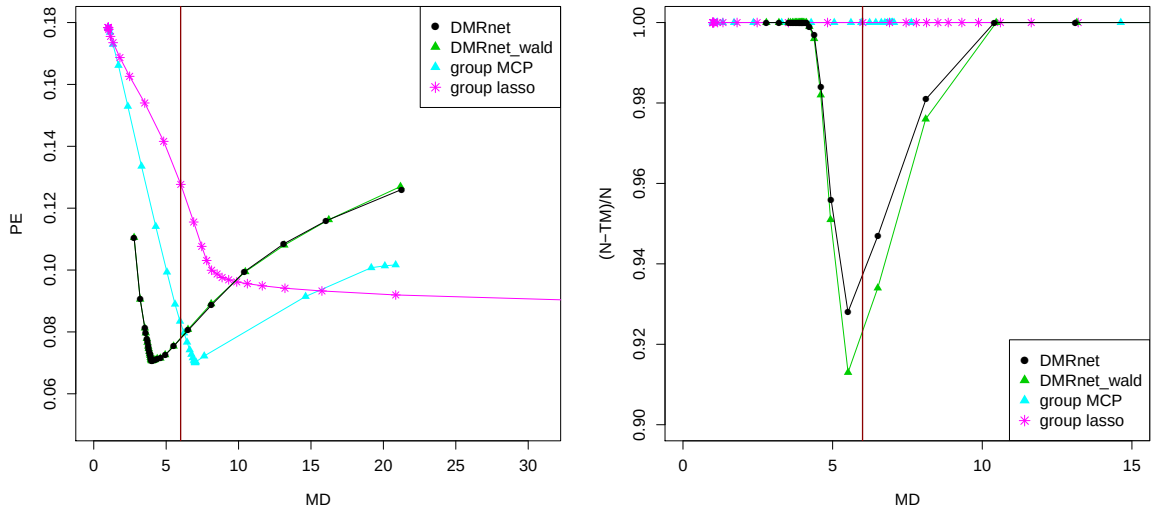


Figure 6.18: Left side: PE vs MD for Experiment 3, $p \gg n$, logistic regression for models chosen by GIC. Right side: (N-TM)/N vs MD for Experiment 3, $p \gg n$, logistic regression for models chosen by GIC.



Chapter 7

Discussion and conclusions

In this dissertation the problem of factorial selection was defined, namely model selection in the presence of both continuous and categorical predictors. It assumes considering models consisting of a subset of continuous variables and partitions of factors. Moreover, an algorithm DMR for factorial selection problem was presented.

It was shown that both DMR for linear model (DMR4lm) and for generalized linear model (DMR4glm) are consistent factorial selection methods and bounds on the selection error were given. In comparison to the lasso-based factorial selection methods (CAS-ANOVA and gvcm), the consistency holds even if p tends to infinity with n. The algorithm DMR4glm uses likelihood ratio test statistics. In order to make it more computationally effective, an algorithm DMR4glm_wald was developed, which uses Wald statistics. However, we haven't yet proven that it is a consistent model selection method.

Furthermore, a new algorithm DMRnet was introduced. It is a generalization of DMR to high-dimensional data. It uses group lasso in the screening step and DMR algorithm after reducing the data to $p < n$. The procedure is repeated for different parameters of the screening step. DMRnet works for both linear (DMRnet4lm) and generalized linear models (DMRnet4glm and DMRnet4glm_wald). Despite its very good practical outcomes, we still work on the proof of its consistency.

In the thesis a thorough practical comparison based on 6 real data examples and 12 experimental setups of the DMR algorithms with the lasso-based factorial selection methods and group lasso and MCP algorithms was presented. Compared to CAS-ANOVA and gvcm methods, DMRnet algorithm in almost all real data sets and simulation setups gave smaller prediction error, sparser models and higher selection accuracy. Moreover, CAS-ANOVA does not work either for generalized linear models nor for high-dimensional data and gvcm implementation often gives an error, among others in all the high-dimensional simulation setups. DMRnet is also better in terms of time efficiency than the lasso-based factorial selection methods.

Group lasso and group MCP do not solve the factorial selection problem, but choose entire continuous and categorical predictors. Group lasso, as has been observed in the literature, chose too big models and group MCP had better results. For almost all real data sets and in approximately half of the simulation setups DMRnet gave smaller prediction error and sparser models than group MCP and in the cases where group MCP was better in terms of PE, DMRnet gave smaller MD. Group MCP, obviously, gave good results when the number

of levels in factors was small and when the true model did not assume many merges within factors as in experiment 3. In terms of time efficiency DMRnet was slower, but the group lasso and MCP solve an easier problem.

In all the cases DMRnet was not worse than DMR in terms of minimizing prediction error and choosing sparse models. In simulations for DMRnet and DMR the dimension of the model which gave smallest prediction error also gave the maximal frequency of choosing the true model. In terms of choice of c in GIC, we would recommend $c = 2.5$ for linear regression and $c = 2$ for logistic regression since they often gave very good results. DMRnet is computationally more intensive than DMRnet_wald and since they give similar results, we recommend DMRnet_wald for practical use.

Bibliography

Howard D Bondell and Brian J Reich. Simultaneous factor selection and collapsing levels in ANOVA. *Biometrics*, 65:169–177, 2009.

Patrick Breheny and Jian Huang. Group descent algorithms for nonconvex penalized linear and logistic regression models with grouped predictors. *Statistics and Computing*, 25:173–187, 2015.

Peter Bühlmann and Sara Van De Geer. *Statistics for high-dimensional data: methods, theory and applications*. Springer Science & Business Media, 2011.

Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. Least angle regression. *Annals of Statistics*, 32:407–499, 2004.

Ludwig Fahrmeir and Heinz Kaufmann. Consistency and asymptotic normality of the maximum likelihood estimator in generalized linear models. *Annals of Statistics*, 13:342–368, 1985.

Jan Gertheiss and Gerhard Tutz. Sparse modeling of categorial explanatory variables. *Annals of Applied Statistics*, 4:2150–2180, 2010.

Tadeusz Inglot and Teresa Ledwina. Asymptotic optimality of new adaptive test in regression model. In *Annales de l'IHP Probabilités et Statistiques*, volume 42, pages 579–590, 2006.

GJO Jameson. A simple proof of Stirling's formula for the Gamma function. *The Mathematical Gazette*, 99:68–74, 2015.

Aleksandra Maj-Kańska, Piotr Pokarowski, Agnieszka Prochenka, et al. Delete or merge regressors for linear model selection. *Electronic Journal of Statistics*, 9:1749–1778, 2015.

Lukas Meier, Sara Van De Geer, and Peter Bühlmann. The group lasso for logistic regression. *Journal of the Royal Statistical Society: Series B*, 70:53–71, 2008.

Margret-Ruth Oelker, Jan Gertheiss, and Gerhard Tutz. Regularization and model selection with categorical predictors and effect modifiers in generalized linear models. *Statistical Modelling*, 14:157–177, 2014.

Piotr Pokarowski and Jan Mielniczuk. Linear model selection using p-values. *Unpublished manuscript*, 16:961–992, 2010.

Piotr Pokarowski and Jan Mielniczuk. Combined l_1 and greedy l_0 penalized least squares for linear model selection. *Journal of Machine Learning Research*, 19, 2015.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B*, pages 267–288, 1996.

Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B*, 67: 91–108, 2004.

Geofrey G Towell, Jude W Shavlik, and Michiel O Noordewier. Refinement of approximate domain theories by knowledge-based neural networks. In *Proceedings of the eighth National conference on Artificial intelligence*, pages 861–866. Boston, MA, 1990.

John W Tukey. Comparing individual means in the analysis of variance. *Biometrics*, 5:99–114, 1949.

Gerhard Tutz. *Regression for categorical data*. Cambridge University Press, 2011.

Gene Yeo and Christopher B Burge. Maximum entropy modeling of short sequence motifs with applications to RNA splicing signals. *Journal of Computational Biology*, 11:377–394, 2004.

Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B*, 68:49–67, 2006.

Cun-Hui Zhang. Nearly unbiased variable selection under minimax concave penalty. *Annals of statistics*, 38:894–942, 2010a.

Tong Zhang. Analysis of multi-stage convex relaxation for sparse regularization. *Journal of Machine Learning Research*, 11:1081–1107, 2010b.

Xiaodong Zheng and Wei-Yin Loh. Consistent variable selection in linear models. *Journal of the American Statistical Association*, 90:151–156, 1995.

Hui Zou and Runze Li. One-step sparse estimates in nonconcave penalized likelihood models. *Annals of Statistics*, 36:1509, 2008.