**Working paper**

**2019-07**

Statistics and Econometrics
ISSN 2387-0303

# Quantile regression: a penalization approach

Álvaro Méndez Civieta, M. Carmen Aguilera-Morillo, Rosa E. Lillo

uc3m | Universidad Carlos III de Madrid

# Quantile regression: a penalization approach

Álvaro Méndez Civieta*† M. Carmen Aguilera-Morillo*†
Rosa E. Lillo*†

May 28, 2019

**Abstract**

Sparse group LASSO (SGL) is a penalization technique used in regression problems where the covariates have a natural grouped structure and provides solutions that are both between and within group sparse. In this paper the SGL is introduced to the quantile regression (QR) framework, and a more flexible version, the adaptive sparse group LASSO (ASGL), is proposed. This proposal adds weights to the penalization improving prediction accuracy. Usually, adaptive weights are taken as a function of the original non-penalized solution model. This approach is only feasible in the $n > p$ framework. In this work, a solution that allows using adaptive weights in high-dimensional scenarios is proposed. The benefits of this proposal are studied both in synthetic and real datasets.

**keywords:** quantile regression; group variable selection; adaptive sparse group LASSO; high dimension; weight calculation

## 1 Introduction

Along years, regression has become a key method in statistics. Ordinary least squares (OLS) regression estimates the conditional mean response of a variable as a function of the covariates. However, OLS estimators relies on certain hypothesis over the first two moments that are not always verified in practical applications. Ever since the seminal work of [Koenker and Bassett,

*Department of Statistics, University Carlos III of Madrid.
†UC3M-Santander Big Data Institute.

1978], quantile regression (QR) models have gained importance when dealing with this kind of situations. QR models allow a relaxation of the classical first two moment conditions over the model error, and it offers robust estimators capable of dealing with heteroscedasticity and outliers. QR models can also estimate different quantile levels of a response variable, giving a precise insight of the relation between response and covariates at upper and lower tails. This can provide a much richer point of view than OLS regression. For a full review on quantile regression, we recommend [Koenker, 2005].

In recent years, high-dimensional data in which the number of covariates $p$ is larger than the number of observations $n$ ($p \gg n$), has become increasingly common. This problem can be found in many different areas like computer vision and pattern recognition [Wright et al., 2010], climate data over different land regions [Chatterjee et al., 2011], and prediction of cancer recurrence based on patients genetic information [Simon et al., 2013]. In these scenarios, variable selection gains special importance offering sparse modeling alternatives that help identifying significant covariates and enhancing prediction accuracy. One of the first and more popular sparse regularization alternatives is LASSO, which was proposed by [Tibshirani, 1996] and adapted to the QR framework by [Li and Zhu, 2008], that developed the piece-wise linear solution of this technique. LASSO is a technique that penalizes each variable individually, enhancing thus individual sparsity. However, in many real applications variables are structured into groups, and group sparsity rather than individual sparsity is desired. One can think for example in a genetic dataset grouped into gene pathways. This problem was faced by the group LASSO penalization of [Yuan and Lin, 2006], and opened the doors to more complex penalizations like the sparse group LASSO [Friedman et al., 2010], which is a linear combination of LASSO and group LASSO providing solutions that are both between and within group sparse. With the same objective in mind, [Zhou and Zhu, 2010] proposed a hierarchical LASSO. To the best of our knowledge, the SGL technique has not been studied in the framework of QR models, so this gap is addressed first, extending the SGL penalization to quantile regression.

[Zou, 2006] was the first to propose the usage of specific weights for each variable on LASSO penalization as a way to increase the model flexibility. This idea, generally known as the adaptive idea, was then extended to other penalizations. The weights of the adaptive idea are typically defined in the

literature based on the results of non-penalized models. This definition is a key step for the demonstration of the oracle properties of the estimators [Fan and Li, 2001], but it is restrictive in the sense that limits the usage of adaptive penalizations just to the case in which solving a non-penalized model is a feasible first step. This approach, focused on the oracle properties under low-dimensional scenarios is observed in [Nardi and Rinaldo, 2008] for the adaptive group lasso, [Ciuperca, 2019] for the adaptive group LASSO and the adaptive fused LASSO [Ciuperca, 2017] in QR, [Wu and Liu, 2009] for the adaptive LASSO and scad penalizations in QR, and [Zhao et al., 2014] for an adaptive hierarchical LASSO in QR regression among others. It is interesting to remark especially the work developed in [Poignard, 2018], in which an adaptive sparse group LASSO estimator suitable for low-dimensional scenarios (with n>p) is proposed, studying its theoretical properties for a set of general convex loss functions.

The main contribution of this work lies here. An adaptive sparse group lasso (ASGL) for quantile regression estimator is defined, working specially on enabling the usage of the ASGL estimator in high-dimensional scenarios (with $p \gg n$). In order to achieve this objective, four alternatives for the weight calculation are proposed. It is worth noting that these weight calculation alternatives can be used no only in the case of the ASGL estimator, but also in the rest of the adaptive-based estimators available in the literature. The performance of this alternatives is also studied in the case of low-dimensional scenarios, making the proposed work a good alternative for both high-dimensional and low-dimensional problems.

The rest of the paper is organized as follows. In Section 2 some basic theoretical concepts are introduced, along with the formal definition of the sparse group LASSO in quantile regression. This definition is extended to the adaptive idea in Section 3, proposing the ASGL estimator. Section 4 introduces the weights calculation alternatives for high-dimensional scenarios. Section 5 shows the advantages of this proposal in different synthetic datasets in high and low-dimensional scenarios. In Section 6 the proposed model is used in a real dataset, a genomic dataset including gene expression data of rat eye disease first shown in [Scheetz et al., 2006]. The computational aspects of the problem are briefly commented on Section 7, and the conclusions are provided in Section 8.

## 2 Penalized quantile regression

Consider a sample of $n$ observations structured as $\mathbb{D} = (y_i, \boldsymbol{x}_i),\ i = 1, \ldots, n$ from some unknown population and define the following linear model,

$$y_i = \boldsymbol{x}_i^t \boldsymbol{\beta} + \epsilon_i,\ i = 1, \ldots, n \tag{1}$$

where $y_i$ is the i-*th* observation of the response variable, $\boldsymbol{x}_i \equiv (x_{i1}, \ldots, x_{ip})$ is the vector of $p$ covariates for observation $i$ and $\epsilon_i$ is the error term.

Let us introduce now the quantile regression framework by defining the loss check function,

$$\rho_\tau(u) = u(\tau - I(u < 0)) \tag{2}$$

where $I(\cdot)$ is the indicator function. In their seminal work [Koenker and Bassett, 1978] proved that the $\tau$-*th* quantile of the response variable can be estimated by solving the following optimization problem,

$$\tilde{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\arg\min} \left\{ \frac{1}{n} \sum_{i=1}^{n} \rho_\tau(y_i - \boldsymbol{x}_i^t \boldsymbol{\beta}) \right\}. \tag{3}$$

Quantile regression models allow a relaxation of the classical first two moment conditions over the model error, and it offers robust estimators capable of dealing with heteroscedasticity and outliers.

We call high-dimensional scenarios to the datasets in which $p$ is much larger than $n$ ($p \gg n$). This problem is becoming more and more common nowadays, and can be observed in many different fields of research such as computer vision and pattern recognition [Wright et al., 2010], climate data over different land regions [Chatterjee et al., 2011] or prediction of cancer recurrence based on patients genetic information [Simon et al., 2013]. An alternative that has been intensively studied in recent years for dealing with these scenarios is the penalization approach. By penalizing a regression model it is possible to perform variable selection and improve the accuracy and interpretability of the models. Since it is a continuous process, it also offers more stable solutions than other alternatives.

One of the best known variable selection penalization methods is the least absolute selection and shrinkage operator, generally known as LASSO, proposed initially by [Tibshirani, 1996] which, in the case of the QR framework solves,

$$\hat{\boldsymbol{\beta}}(\lambda) = \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\arg\min} \left\{ \frac{1}{n} \sum_{i=1}^{n} \rho_\tau(y_i - \boldsymbol{x}_i^t \boldsymbol{\beta}) + \lambda \left\| \boldsymbol{\beta} \right\|_1 \right\}, \tag{4}$$

where $\rho_\tau(\cdot)$ is the QR check function defined in (2). The LASSO penalization sends many $\boldsymbol{\beta}$ components to zero, offering sparse solutions and performing automatic variable selection. In the last years, many LASSO-based algorithms have been proposed. [Yuan and Lin, 2006] introduced the group LASSO penalization as an answer for the need to select variables not individually but at the group level. This penalization solves the following problem,

$$\hat{\boldsymbol{\beta}}(\lambda) = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\arg\min} \left\{ \frac{1}{n}\sum_{i=1}^{n} \rho_\tau(y_i - \boldsymbol{x}_i^t\boldsymbol{\beta}) + \lambda \sum_{l=1}^{K} \sqrt{p_l} \left\|\boldsymbol{\beta}^l\right\|_2 \right\}, \tag{5}$$

where $K$ is the number of groups, $\boldsymbol{\beta}^l \in \mathbb{R}^{p_l}$ are vectors of components of $\boldsymbol{\beta}$ from the l-*th* group, and $p_l$ is the size of the l-*th* group. The group LASSO penalization performs in a similar way to LASSO, but while LASSO enhances sparsity at individual level, group LASSO enhances sparsity at group level, selecting, or sending to zero whole groups of variables.

Initially proposed by [Friedman et al., 2010], the sparse group LASSO (SGL) is a linear combination of LASSO and group LASSO penalizations. Well known in linear regression and other GLM models, to the best of our knowledge SGL has not been adapted to QR, and as a first step in the paper, this penalization is introduced in the framework.

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\arg\min} \left\{ \frac{1}{n}\sum_{i=1}^{n} \rho_\tau(y_i - \boldsymbol{x}_i^t\boldsymbol{\beta}) + \alpha\lambda \left\|\boldsymbol{\beta}\right\|_1 + (1-\alpha)\lambda \sum_{l=1}^{K} \sqrt{p_l} \left\|\boldsymbol{\beta}^l\right\|_2 \right\}. \tag{6}$$

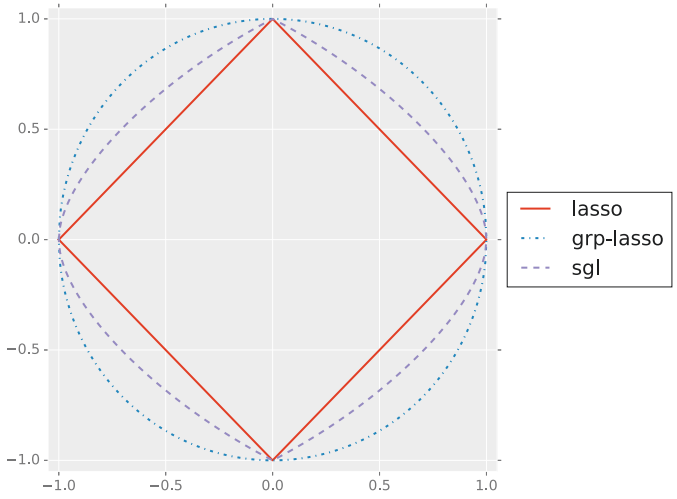
As in LASSO and group LASSO, SGL solutions are, in general, sparse, sending many of the predictor coefficients to zero. However, while LASSO solutions are sparse at individual level, and group LASSO solutions are sparse at group level, SGL offers both between and within group sparsity, outperforming both alternatives.

From an optimization perspective, equation (6) defines a sum of convex functions. This convexity ensures that the solution of the minimization problem is a global minimum. Figure 1 shows the constrains defined by LASSO, group LASSO and SGL in the case of a single 2-dimensional group of predictors.

## 3 Adaptive sparse group LASSO

Variable penalization in SGL is somehow restrictive in the sense that it penalizes equally all the variables in LASSO part (giving them the same

Figure 1: Contour lines for LASSO, group-LASSO and sparse-group-LASSO penalties in the case of a single 2-dimensional group




importance), and it penalizes only based on the group size in the group LASSO part. The usage of the adaptive idea, initially introduced by [Zou, 2006], is proposed here as a way to solve this limitations. In this work, a variant of the SGL penalization, the adaptive sparse group LASSO (ASGL) for quantile regression is defined. The ASGL estimator for QR is the result of the following minimization process,

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\arg\min} \left\{ \frac{1}{n}\sum_{i=1}^{n} \rho_\tau(y_i - \boldsymbol{x}_i^t\beta_i) + \alpha\lambda\sum_{j=1}^{p} \tilde{w}_j|\beta_j| + (1-\alpha)\lambda\sum_{l=1}^{K} \sqrt{p_l}\tilde{v}_l \left\|\boldsymbol{\beta}^l\right\|_2 \right\}, \tag{7}$$

where $\tilde{\boldsymbol{w}} \in \mathbb{R}^p$ and $\tilde{\boldsymbol{v}} \in \mathbb{R}^K$ are known weights vectors. The intuition behind these weights is that if a variable (or group of variables) is important, it should have a small weight, and this way would be lightly penalized. On the other hand, if it is not important, by setting a large weight it is heavily penalized. This enhances the model flexibility and improves prediction accuracy. It is worth saying that this formulation defines a convex function and thus, the global minimum can be found.

The adaptive idea has been used in many LASSO-based formulations in recent years. One can see for instance [Wu and Liu, 2009] that introduces the adaptive LASSO in quantile regression, [Ciuperca, 2017] where an adaptive fused group LASSO is defined, or [Zhao et al., 2014] that proposes an

adaptive hierarchical LASSO among others. When dealing with an adaptive estimator, the general tendency found in the literature is to study the so-called oracle properties of the estimator, defined initially in [Fan and Li, 2001]. One estimator is oracle if it is (asymptotically) normally distributed and recovers the true underlying sparse model.

A major drawback of this approach in our opinion is precisely that it is focused on the asymptotic behavior of the estimators, when the number of observations $n$ diverges, or in the double-asymptotic behavior, when the number of parameters $p$ diverges with the number of observations $n$. However, it does not consider the case of high-dimensional scenarios, where $p \gg n$. A typical step in the demonstration of the oracle behavior of the estimators is to define the adaptive weights based on the result of the unpenalized model,

$$\tilde{\boldsymbol{w}} = \frac{1}{|\tilde{\boldsymbol{\beta}}|^{\gamma}}, \tag{8}$$

where $|\cdot|$ denotes the absolute value function, $\gamma$ is a non negative constant and $\tilde{\boldsymbol{\beta}}$ is the solution vector obtained from the unpenalized model (described, in the case of the QR framework, in equation (3)). This approach makes unfeasible the usage of adaptive models in high-dimensional scenarios in which solving unpenalized models is not a realistic approach. The work developed by [Poignard, 2018] is of special interest. Here an adaptive sparse group lasso estimator is defined in a general set of convex functions. However, the author is centered on studying the asymptotic behavior of the estimator, and thus, its usage in real applications is restricted to low-dimensional scenarios. The proposal developed in this work, on the other hand, can be used both in high-dimensional and low-dimensional scenarios.

# 4 Adaptive weights calculation

The objective of this section is to introduce different alternatives for the calculation of weights in the adaptive framework. The intuitive idea is to find a way to substitute $\tilde{\beta}$, the solution from the unpenalized model, unfeasible in high-dimensional scenarios, in the calculation of the adaptive weights. This problem will be faced making use of two dimensionality reduction techniques, principal component analysis (PCA) and partial least squares (PLS). The proposed weight calculation alternatives can be used both in high-dimensional and low-dimensional scenarios. It is worth remark-

ing that these alternatives can be applied not only to the ASGL algorithm, but also to other adaptive-based algorithms.

## 4.1 Principal components analysis

Given the covariates matrix $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ defined in equation (1), with maximum rank $r = \min\{n, p\}$, consider the matrix of principal components $\boldsymbol{Q} \in \mathbb{R}^{p \times r}$ defined in a way such that the first principal component has the largest possible variance, and each succeeding component has the largest possible variance under the constraint that it is orthogonal to the preceding components. From an algebra perspective, the principal components in $\boldsymbol{Q}$ define an orthogonal change of basis matrix that maximize the variance explained from $\boldsymbol{X}$. Consider $\boldsymbol{Z} = \boldsymbol{X}\boldsymbol{Q} \in \mathbb{R}^{n \times r}$ the projection of $\boldsymbol{X}$ into the principal components subspace. Two weight calculation alternatives based on principal components are proposed.

### 4.1.1 Based on a subset of components

Consider the sub-matrix $\boldsymbol{Q}_d = [\boldsymbol{q}_1, \ldots, \boldsymbol{q}_d]^t$ where $\boldsymbol{q}_i \in \mathbb{R}^p$ is the i-*th* column of the matrix $\boldsymbol{Q}$, and $d \in \{1, \ldots, r\}$ is the number of components chosen. Let $\alpha_{pca,d} \in [0, 100]$ be the percentage of variability from $\boldsymbol{X}$ that the principal components in $\boldsymbol{Q}_d$ are able to explain. If $d = r$ then the principal components in $\boldsymbol{Q}_d$ are able to explain all the original variability from $\boldsymbol{X}$, and $\alpha_{pca,d} = 100$, if $d < r$ then $\alpha_{pca,d} < 100$. The number of components chosen in order to explain up to a certain percentage of variability is fixed by the researcher. Obtain $\boldsymbol{Z}_d = \boldsymbol{X}\boldsymbol{Q}_d \in \mathbb{R}^{n \times d}$ the projection of $\boldsymbol{X}$ into the subspace generated by $\boldsymbol{Q}_d$ and solve the unpenalized model,

$$\tilde{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\arg\min} \left\{ \frac{1}{n} \sum_{i=1}^{n} \rho_\tau(y_i - \boldsymbol{z}_i^t \boldsymbol{\beta}) \right\}. \tag{9}$$

This model defines a low-dimensional scenario where $\tilde{\boldsymbol{\beta}} \in \mathbb{R}^d$. Using this solution, it is possible to obtain an estimation of the high-dimensional scenario solution, $\hat{\boldsymbol{\beta}} = \boldsymbol{Q}_d \tilde{\boldsymbol{\beta}} \in \mathbb{R}^p$. Finally, the weights are estimated as,

$$\tilde{w_j} = \frac{1}{|\hat{\beta}_j|^{\gamma_1}} \quad \text{and} \quad \tilde{v_l} = \frac{1}{\left\| \hat{\boldsymbol{\beta}}^l \right\|_2^{\gamma_2}}, \tag{10}$$

where $\hat{\beta}_j$ is the j-*th* component from $\hat{\boldsymbol{\beta}}$, $\hat{\boldsymbol{\beta}}^l$ is the vector of components of $\boldsymbol{\beta}$ from the l-*th* group, and $\gamma_1$ and $\gamma_2$ are non negative constants usually taken in $[0, 2]$.

### 4.1.2 Based on the first component

A more straightforward approach based on the first principal component is also proposed. The principal components are no more than lineal combinations of the original variables. Therefore, the first principal component $\boldsymbol{q}_1 \in \mathbb{R}^p$, which is the first column of the matrix $\boldsymbol{Q}$, includes one weight for each of the $p$ original variables. This proposal consists of calculating the weights as,

$$\tilde{w_j} = \frac{1}{|q_{1j}|^{\gamma_1}} \quad \text{and} \quad \tilde{v_l} = \frac{1}{\left\|\boldsymbol{q}_1^l\right\|_2^{\gamma_2}}, \tag{11}$$

where $q_{1j}$ is the j-*th* component from $\boldsymbol{q}_1$ and defines the weight associated to the j-*th* original variable, $\boldsymbol{q}_1^l$ is the vector of components of $\boldsymbol{q}_1$ from the l-*th* group and $\gamma_1$ and $\gamma_2$ are non negative constants usually taken in $[0, 2]$.

## 4.2 Partial least squares

The principal components are defined in a way such that they capture the maximum possible variance from $\boldsymbol{X}$ under the constraint that they are orthogonal to the rest of the principal components. However, being relevant for describing the variance of $\boldsymbol{X}$ does not necessarily mean that a principal component is relevant for predicting the value of $\boldsymbol{y}$. Partial least squares (PLS) is a dimensionality reduction technique centered on maximizing the covariance between $\boldsymbol{X}$ and $\boldsymbol{y}$.

Given the covariates matrix $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ defined in equation (1), with maximum rank $r = \min\{n, p\}$, consider the matrix of PLS components $\boldsymbol{T} \in \mathbb{R}^{p \times s}$ and the projection of $\boldsymbol{X}$ into the subspace generated by $\boldsymbol{T}$: $\boldsymbol{U} = \boldsymbol{X}\boldsymbol{T} \in \mathbb{R}^{n \times s}$. The matrix of PLS components $\boldsymbol{T}$ defines a non-orthogonal change of basis matrix whose projection $\boldsymbol{U}$ is computed in a way such that the first projection vector, $\boldsymbol{u_1} \in \mathbb{R}^n$ has the largest possible covariance with $\boldsymbol{y}$, and each succeeding projection vector has the largest possible covariance with $\boldsymbol{y}$ under the constraint that it is uncorrelated to the rest of the projection vectors.

Given the sub-matrix $\boldsymbol{T}_d = [\boldsymbol{t}_1, \ldots, \boldsymbol{t}_d]^t$ where $\boldsymbol{t}_i \in \mathbb{R}^p$ is the i-*th* column of the matrix $\boldsymbol{T}$, and $d \in \{1, \ldots, s\}$ is the number of components chosen, let $\alpha_{pls,d} \in [0, 100]$ be the percentage of variability from $\boldsymbol{X}$ that the PLS components in $\boldsymbol{T}_d$ are able to explain. The non-orthogonality of $\boldsymbol{T}$ implies that the total number of PLS components available to be computed is smaller

than the rank of $\boldsymbol{X}$, $s \leq r$, and then the maximum possible percentage of variability explained by the PLS components $\alpha_{pls,s}$ is lower than 100%.

In the case of principal components analysis, the matrix of principal components $\boldsymbol{Q}$ defines an orthogonal change of basis matrix that results into an orthogonal projection matrix $\boldsymbol{Z}$ maximizing the variance of $\boldsymbol{X}$. On the other hand, PLS defines a non-necesarily orthogonal change of basis matrix $\boldsymbol{T}$ that results into an uncorrelated projection matrix $\boldsymbol{U}$ maximizing the covariance between $\boldsymbol{U}$ and $\boldsymbol{y}$. Similarly to the alternatives proposed for PCA, two alternatives of weight calculation using PLS are proposed: based on a subset of PLS components, and based just on the first PLS component.

# 5 Simulation studies

This section shows the performance of the proposed ASGL model under different synthetic dataset examples. Given that the ASGL formulation in equation (7) includes a weight penalization on the group LASSO part based on the group size (the term $\sqrt{p_l}$), two model formulations are considered:

- Adaptive LASSO in sparse group LASSO (AL-SGL), where $\tilde{\boldsymbol{w}} \neq \mathbf{1}$ but $\tilde{\boldsymbol{v}} = \mathbf{1}$, in which the adaptive idea is only applied to the LASSO part.
- Adaptive sparse group LASSO (ASGL), where $\tilde{\boldsymbol{w}} \neq \mathbf{1}$ and $\tilde{\boldsymbol{v}} \neq \mathbf{1}$.

Furthermore, the four weight calculation alternatives proposed are studied:

- PCA weights based on regression on a subset of principal components, we denote this as $pca_d$;
- PCA weights based on the first principal component, we denote this as $pca_1$;
- PLS weights based on regression on a subset of PLS components, we denote this as $pls_d$;
- PLS weights based on the first PLS component, we denote this as $pls_1$.

The total number of components $d$ used in the weight estimation in $pls_d$ and $pca_d$ is chosen such that in both cases the percentage of variability explained from the original matrix $\boldsymbol{X}$ is $\alpha_{pca,d} = 80\%$, $\alpha_{pls,d} = 80\%$. As commented along Section 4, due to the non orthogonality of the PLS components it is possible for the maximum possible variability explained by the

PLS components $\alpha_{pls,s}$ to be smaller than 80%. In these cases we consider $d$ such that $\alpha_{pls,d} = \alpha_{pls,s}$.

The results obtained by the models proposed in this work are compared with the results from LASSO and SGL formulations. For each dataset $\mathbb{D}$, a partition into three disjoint subsets, $\mathbb{D}_{train}$, $\mathbb{D}_{val}$ and $\mathbb{D}_{test}$ is considered. $\mathbb{D}_{train}$ is used for training the models, this is, solving the model equations. $\mathbb{D}_{val}$ is used for validation, this is, optimizing the model parameters. This optimization is performed based on grid-search. Finally, $\mathbb{D}_{test}$ is used for testing the models prediction accuracy. The model parameters are optimized based on the minimization of the quantile error, defined as,

$$E_v = \frac{1}{\#\mathbb{D}_{val}} \sum_{(y_i, \boldsymbol{x_i}) \in \mathbb{D}_{val}} \rho_\tau(y_i - \boldsymbol{x}_i^t \hat{\boldsymbol{\beta}}), \tag{12}$$

where $\rho_\tau(\cdot)$ denotes the quantile function defined at (2), and # denotes the cardinal of a set. The final model error is calculated over $\mathbb{D}_{test}$ as,

$$E_t = \frac{1}{\#\mathbb{D}_{test}} \sum_{(y_i, \boldsymbol{x_i}) \in \mathbb{D}_{test}} \rho_\tau(y_i - \boldsymbol{x}_i^t \hat{\boldsymbol{\beta}}). \tag{13}$$

Additionally, the following metrics for evaluating the performance of the methods are considered:

- $\left\| \hat{\boldsymbol{\beta}} - \boldsymbol{\beta} \right\|_2$ the euclidean distance between the estimated vector and the true vector;
- true positive rate (TPR)= $\mathrm{P}(\hat{\beta}_i \neq 0 | \beta_i \neq 0)$;
- true negative rate (TNR)= $\mathrm{P}(\hat{\beta}_i = 0 | \beta_i = 0)$;
- correct selection rate (CSR)= $\mathrm{P}(\hat{\beta} = \beta)$.

In these simulations we consider $\tau = 0.5$. Each simulation example has been executed 50 times considering 100/100/5000 observations in the train / validate / test samples. The results have been summarized in terms of the mean and standard deviation values, and the best result from each metric is highlighted.

The general simulation scheme comes from the model,

$$y = X\beta + \epsilon, \ \epsilon \sim t(3),$$

where $X$ is generated from a standard Gaussian distribution. Variables are organized in groups, and within group correlation of 0.5 and 0 otherwise

is considered. The scheme used here is an adaptation of other simulation schemes used in [Wu and Liu, 2009] and [Zhao et al., 2014].

We are interested on studying the performance of the proposed models under different situations, for this reason three main scenarios are defined, the first two are high-dimensional and the third one is low-dimensional. The scenario from Section 5.1 is a high-dimensional scenario that considers a sparse distribution of the significant variables along many groups, the scenario from Section 5.2 is a high-dimensional scenario that considers a dense distribution of the significant variables in a few number of groups. It is interesting to test the performance of the models as the number of variables increase. For this reason, two cases are considered in each of these two scenarios. A dataset having 225 variables and a dataset having 625 variables.

As it was commented in Section 3, the general tendency found in the literature regarding the weights in adaptive models is to define them based on the results of the unpenalized model,

$$\tilde{\boldsymbol{w}} = \frac{1}{|\tilde{\boldsymbol{\beta}}|^{\gamma}}, \tag{14}$$

where $|\cdot|$ denotes the absolute value function, $\gamma$ is a non negative constant and $\tilde{\boldsymbol{\beta}}$ is the solution vector obtained from the unpenalized model (described, in the case of the QR framework, in equation (3)). This approach is limited just to low-dimensional scenarios, where the unpenalized model can actually be solved. For this reason, a third scenario, a low-dimensional dataset is considered in Section 5.3, comparing the results from the weights based on the unpenalized model with the weights based on the proposal made in this work.

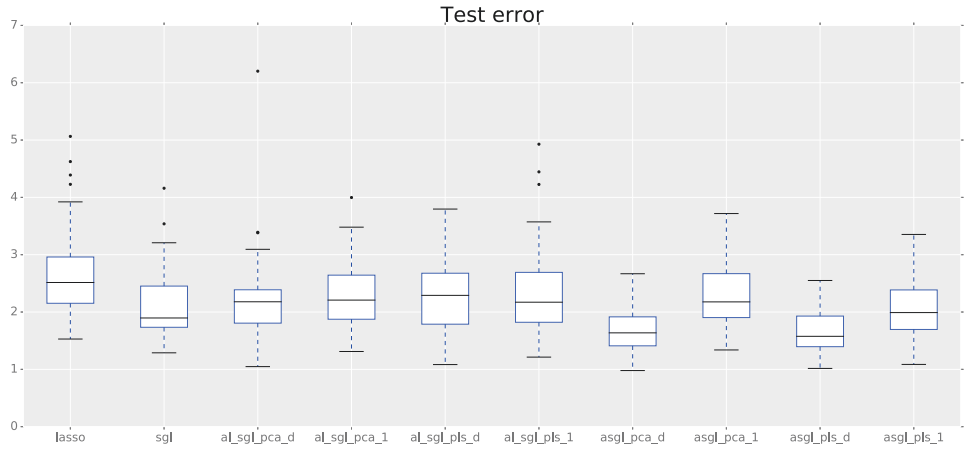
## 5.1 Simulation 1: sparse distribution of significant variables

*Case 1:* 225 *variables*

There are 15 groups of size 15 each, a total number of 225 variables. Among these groups, 7 groups with 8 significant variables each are defined, a total number of 56 significant variables. For $l \in \{1 \ldots, 15\}$, coefficients inside each group are defined as,

$$\begin{cases} \beta^l &= (1, 2, \ldots, 8, \underbrace{0, \ldots, 0}_{7}), \ l = 1, \ldots, 7 \\ \beta^l &= (\underbrace{0, \ldots, 0}_{15}), \ l = 8, \ldots, 15. \end{cases}$$

Figure 2: Simulation 1. Dataset with non-zero variables spread along many groups. 225 variables. Box-plots showing the test error of the different models.




*Case 2:* 625 *variables*

There are 25 groups of size 25 each, a total number of 625 variables. Among these groups, 7 groups with 8 significant variables each are defined, a total number of 56 significant variables. For $l \in \{1 \dots, 25\}$, coefficients inside each group are defined as,

$$\begin{cases} \beta^l &= (1, 2, \dots, 8, \underbrace{0, \dots, 0}_{17}), \ l = 1, \dots, 7 \\ \beta^l &= (\underbrace{0, \dots, 0}_{25}), \ l = 8, \dots, 25. \end{cases}$$

The results from this simulation scheme are displayed in Table 1. The best results are obtained by the ASGL model using $pls_d$ weights, closely followed by $pca_d$ weights. These models outperforms LASSO and SGL both in terms of the distance between predicted and true $\boldsymbol{\beta}$, and in terms of the test error $E_t$, improving prediction accuracy. Given that LASSO enhances individual sparsity, LASSO solutions are more sparse than the solutions obtained by the proposed models models, and this is shown in the TNR values. However, the proposed models produce a better selection of the truly significant variables, fact that is displayed by the TPR values. If the number of

Table 1: Simulation 1. Dataset with non-zero variables spread along many groups, and 100/100/5000 train/validate/test observations. Experiments run 50 times and displayed as mean value, with standard deviations given in parenthesis

| | $\left\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right\|$ | $E_t$ | CSR | TPR | TNR |
|---|---|---|---|---|---|
| | | $p = 225$ variables | | | |
| LASSO | 8.09 (2.48) | 2.66 (0.81) | **0.80** (0.02) | 0.96 (0.03) | **0.75** (0.02) |
| SGL | 6.43 (2.02) | 2.12 (0.60) | 0.76 (0.06) | 0.98 (0.02) | 0.69 (0.07) |
| AL-SGL-$pca_d$ | 6.66 (2.33) | 2.20 (0.76) | 0.78 (0.06) | 0.97 (0.03) | 0.71 (0.08) |
| AL-SGL-$pca_1$ | 7.06 (1.98) | 2.30 (0.61) | 0.73 (0.06) | 0.98 (0.02) | 0.65 (0.09) |
| AL-SGL-$pls_d$ | 6.95 (1.79) | 2.28 (0.56) | 0.77 (0.06) | 0.97 (0.02) | 0.70 (0.08) |
| AL-SGL-$pls_1$ | 7.27 (2.46) | 2.39 (0.78) | 0.74 (0.06) | 0.98 (0.02) | 0.66 (0.08) |
| ASGL-$pca_d$ | 5.09 (1.32) | 1.70 (0.38) | 0.73 (0.09) | **0.99** (0.01) | 0.65 (0.12) |
| ASGL-$pca_1$ | 7.07 (1.98) | 2.31 (0.62) | 0.75 (0.06) | 0.98 (0.02) | 0.67 (0.07) |
| ASGL-$pls_d$ | **5.05** (1.30) | **1.68** (0.37) | 0.74 (0.09) | **0.99** (0.02) | 0.66 (0.12) |
| ASGL-$pls_1$ | 6.21 (1.78) | 2.04 (0.52) | 0.74 (0.05) | 0.98 (0.02) | 0.66 (0.06) |
| | | $p = 625$ variables | | | |
| LASSO | 23.37 (4.61) | 7.85 (1.70) | **0.89** (0.01) | 0.76 (0.07) | **0.90** (0.01) |
| SGL | 19.62 (3.28) | 6.29 (1.08) | 0.76 (0.10) | 0.90 (0.04) | 0.75 (0.12) |
| AL-SGL-$pca_d$ | 17.97 (3.56) | 5.68 (1.13) | 0.83 (0.07) | 0.88 (0.05) | 0.83 (0.08) |
| AL-SGL-$pca_1$ | 21.41 (2.78) | 6.88 (0.93) | 0.70 (0.10) | 0.90 (0.04) | 0.68 (0.12) |
| AL-SGL-$pls_d$ | 17.60 (3.28) | 5.78 (1.14) | 0.83 (0.06) | 0.89 (0.04) | 0.83 (0.07) |
| AL-SGL-$pls_1$ | 19.40 (2.99) | 6.23 (0.99) | 0.78 (0.09) | 0.90 (0.04) | 0.77 (0.10) |
| ASGL-$pca_d$ | 15.19 (3.43) | 4.65 (1.04) | 0.84 (0.04) | **0.92** (0.03) | 0.83 (0.04) |
| ASGL-$pca_1$ | 21.38 (2.58) | 6.80 (0.87) | 0.73 (0.10) | 0.91 (0.04) | 0.71 (0.11) |
| ASGL-$pls_d$ | **13.23** (3.35) | **4.07** (0.99) | 0.85 (0.03) | 0.91 (0.04) | 0.84 (0.04) |
| ASGL-$pls_1$ | 17.56 (3.98) | 5.61 (1.33) | 0.81 (0.01) | 0.91 (0.04) | 0.80 (0.07) |

Figure 3: Simulation 1. Dataset with non-zero variables spread along many groups. 625 variables. Box-plots showing the test error of the different models.


variables is increased, the difference in performance between the proposed models, and LASSO and SGL gets larger, indicating that the performance of the proposed models is better than LASSO and SGL as the number of variables increase. Figures 2 and 3 display box-plots of the test error $E_t$ for the different models, showing that the spread of $E_t$ is much smaller in the ASGL $pls_d$ and $pca_d$ than in the LASSO and SGL, indicating that these models provide more stable solutions in terms of prediction accuracy. A second simulation example in which there are a few groups that account for all the significant variables is shown.

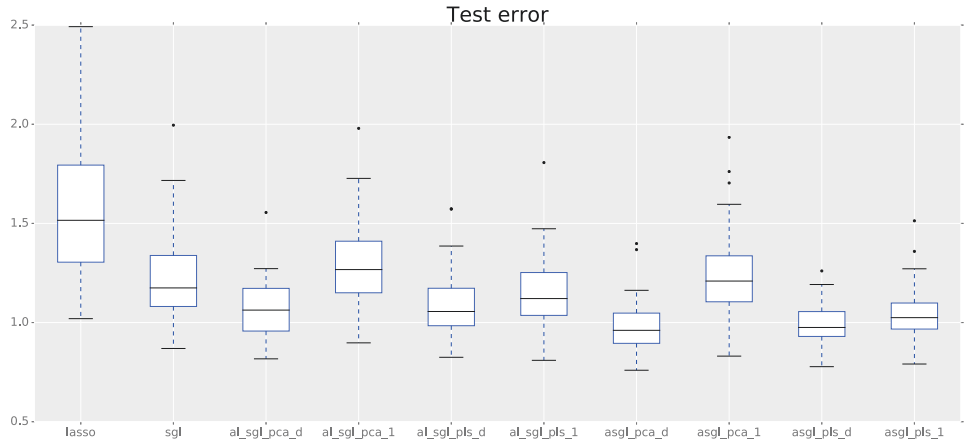
### 5.2 Simulation 2: dense distribution of significant variables

*Case 1:* 225 *variables*

There are 15 groups of size 15 each, a total number of 225 variables. Among these groups, we define 3 groups with 15 significant variables each, a total number of 45 significant variables. For $l \in \{1 \dots, 15\}$, coefficients inside each group are defined as,

$$\begin{cases} \beta^l &= (1, 2, \dots, 15),\ l = 1, \dots, 3 \\ \beta^l &= (\underbrace{0, \dots, 0}_{15}),\ l = 4, \dots, 15. \end{cases}$$

Figure 4: Simulation 2. Dataset with non-zero variables concentrated in a small number of fully significant groups. 225 variables. Box-plots showing the test error of the different models.




*Case 2:* 625 *variables*

There are 25 groups of size 25 each, a total number of 625 variables. Among these groups, 3 groups with 25 significant variables each are defined, a total number of 75 significant variables. For $l \in \{1 \dots, 25\}$, coefficients inside each group are defined as,

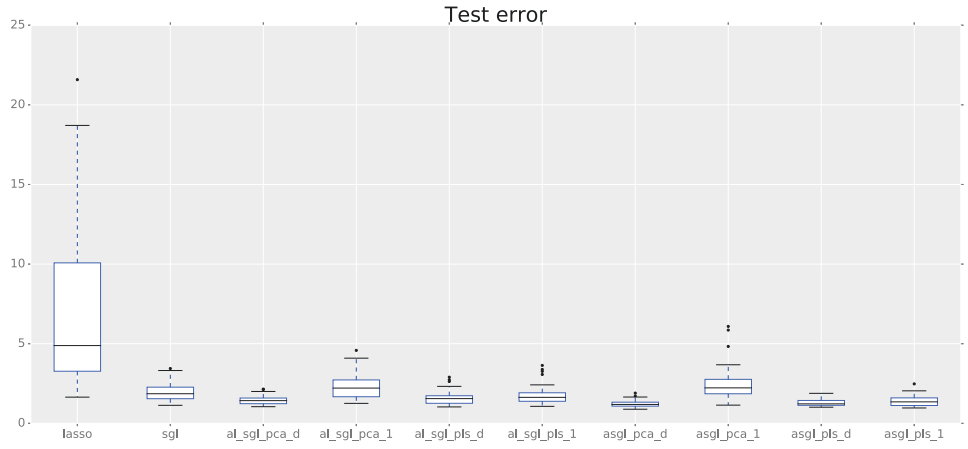
$$\begin{cases} \beta^l &= (1, 2, \dots, 25),\ l = 1, \dots, 3 \\ \beta^l &= (\underbrace{0, \dots, 0}_{25}),\ l = 4, \dots, 25. \end{cases}$$

The results from this simulation scheme are displayed in Table 2. Similarly to the situation displayed in Section 5.1, the ASGL model using $pls_d$ or $pca_d$ weights shows the best results in terms of the distance between predicted and true $\boldsymbol{\beta}$, and the value of $E_t$. It is worth saying that under a more "compact" distribution of the significant variables in a small number of groups, the proposed methods show a great improvement in terms of prediction accuracy compared to LASSO and SGL. Figures 4 and 5 display box-plots of test error value $E_t$ showing, as in the previous simulation scheme, that ASGL models with $pls_d$ or $pca_d$ weights also provide more stable results in terms of spread.

Table 2: Simulation 2. Dataset with non-zero variables concentrated in a small number of fully significant groups, and 100/100/5000 train/validate/test observations. Experiments run 50 times and displayed as mean value, with standard deviations given in parenthesis

| | $\left\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right\|$ | $E_t$ | CSR | TPR | TNR |
|---|---|---|---|---|---|
| | | $p = 225$ variables | | | |
| LASSO | 4.43 (1.10) | 1.57 (0.35) | 0.87 (0.03) | 0.99 (0.01) | 0.83 (0.05) |
| SGL | 3.29 (0.75) | 1.21 (0.21) | 0.73 (0.13) | 0.99 (0.01) | 0.64 (0.17) |
| AL-SGL-$pca_d$ | 2.88 (0.50) | 1.07 (0.14) | 0.78 (0.06) | **1.00** (0.01) | 0.84 (0.11) |
| AL-SGL-$pca_1$ | 3.63 (0.73) | 1.30 (0.22) | 0.61 (0.15) | 0.99 (0.01) | 0.47 (0.21) |
| AL-SGL-$pls_d$ | 2.92 (0.57) | 1.09 (0.16) | 0.84 (0.12) | **1.00** (0.01) | 0.78 (0.16) |
| AL-SGL-$pls_1$ | 3.14 (0.65) | 1.16 (0.18) | 0.76 (0.14) | **1.00** (0.01) | 0.67 (0.20) |
| ASGL-$pca_d$ | **2.56** (0.49) | **0.98** (0.13) | **0.89** (0.12) | **1.00** (0.01) | **0.85** (0.16) |
| ASGL-$pca_1$ | 3.49 (0.79) | 1.25 (0.22) | 0.62 (0.15) | **1.00** (0.01) | 0.49 (0.21) |
| ASGL-$pls_d$ | 2.59 (0.43) | 0.99 (0.10) | 0.88 (0.16) | **1.00** (0.01) | 0.83 (0.21) |
| ASGL-$pls_1$ | 2.80 (0.53) | 1.05 (0.14) | 0.81 (0.12) | **1.00** (0.01) | 0.74 (0.17) |
| | | $p = 625$ variables | | | |
| LASSO | 21.00 (13.00) | 7.13 (4.67) | **0.95** (0.01) | 0.96 (0.03) | **0.95** (0.01) |
| SGL | 6.02 (1.77) | 1.99 (0.56) | 0.82 (0.09) | **1.00** (0.01) | 0.80 (0.10) |
| AL-SGL-$pca_d$ | 4.32 (0.99) | 1.45 (0.28) | 0.94 (0.04) | **1.00** (0.01) | 0.93 (0.05) |
| AL-SGL-$pca_1$ | 7.17 (2.47) | 2.30 (0.75) | 0.72 (0.09) | **1.00** (0.01) | 0.68 (0.11) |
| AL-SGL-$pls_d$ | 4.81 (1.47) | 1.60 (0.44) | 0.92 (0.06) | **1.00** (0.01) | 0.90 (0.07) |
| AL-SGL-$pls_1$ | 5.38 (1.20) | 1.77 (0.57) | 0.87 (0.08) | **1.00** (0.01) | 0.85 (0.09) |
| ASGL-$pca_d$ | **3.61** (0.78) | **1.23** (0.20) | 0.92 (0.10) | **1.00** (0.01) | 0.90 (0.12) |
| ASGL-$pca_1$ | 7.60 (3.20) | 2.46 (1.01) | 0.74 (0.09) | **1.00** (0.01) | 0.71 (0.11) |
| ASGL-$pls_d$ | 3.85 (0.83) | 1.29 (0.21) | 0.85 (0.03) | **1.00** (0.01) | 0.89 (0.13) |
| ASGL-$pls_1$ | 4.17 (1.17) | 1.40 (0.32) | 0.90 (0.11) | **1.00** (0.01) | 0.87 (0.09) |

Figure 5: Simulation 2. Dataset with non-zero variables concentrated in a small number of fully significant groups. 625 variables. Box-plots showing the test error of the different models.




Based on the simulations described in Sections 5.1 and 5.2, we conclude that the best performance is achieved by ASGL models with $pls_d$ or $pca_d$ weights.

### 5.3 Simulation 3: n > p

This simulation defines a low-dimensional scenario that studies the performance of the ASGL model with $pls_d$ and $pca_d$ weights compared to an ASGL model with weights from an unpenalized model. The simulation scheme considers 200/200/5000 observations in the train / validate / test samples, and 100 variables divided into 10 groups of size 10 each. Two different cases depending on the distribution of the significant variables are considered:

*Case 1: Sparse distribution of variables*
Here, 5 groups with 6 significant variables each are defined, a total number of 30 significant variables. For $l \in \{1 \dots, 10\}$, coefficients inside each group are defined as,

$$\begin{cases} \beta^l &= (1, 2, \dots, 6, \underbrace{0, \dots, 0}_{4}), \ l = 1, \dots, 5 \\ \beta^l &= (\underbrace{0, \dots, 0}_{10}), \ l = 6, \dots, 10. \end{cases}$$

Table 3: Simulation 3. Dataset with 200/200/5000 train/validate/test observations. Experiments run 50 times and displayed as mean value, with standard deviations given in parenthesis

| | $\left\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right\|$ | $E_t$ | CSR | TPR | TNR |
|---|---|---|---|---|---|
| Sparse distribution of variables | | | | | |
| LASSO | 1.08 (0.14) | 0.67 (0.02) | 0.77 (0.09) | **1.00** (0.00) | 0.67 (0.12) |
| SGL | 1.08 (0.13) | 0.67 (0.02) | 0.72 (0.12) | **1.00** (0.00) | 0.60 (0.16) |
| ASGL-$pca_d$ | 0.96 (0.13) | 0.64 (0.02) | 0.79 (0.10) | **1.00** (0.00) | 0.70 (0.15) |
| ASGL-$pls_d$ | **0.78** (0.11) | **0.62** (0.02) | **0.9**4 (0.07) | **1.00** (0.00) | **0.92** (0.09) |
| ASGL-unpenalized | 0.79 (0.11) | **0.62** (0.02) | **0.94** (0.07) | **1.00** (0.00) | **0.92** (0.09) |
| Dense distribution of variables | | | | | |
| LASSO | 0.90 (0.15) | 0.65 (0.02) | 0.85 (0.09) | **1.00** (0.00) | 0.79 (0.13) |
| SGL | 0.87 (0.15) | 0.64 (0.02) | 0.74 (0.16) | **1.00** (0.00) | 0.64 (0.24) |
| ASGL-$pca_d$ | 0.76 (0.13) | **0.61** (0.02) | 0.93 (0.08) | **1.00** (0.00) | 0.90 (0.12) |
| ASGL-$pls_d$ | **0.74** (0.13) | **0.61** (0.02) | **0.95** (0.09) | **1.00** (0.00) | **0.93** (0.13) |
| ASGL-unpenalized | 0.75 (0.12) | **0.61** (0.02) | 0.94 (0.13) | **1.00** (0.00) | 0.91 (0.18) |

*Case 2: Dense distribution of variables*

Here, 3 groups with 10 significant variables each are defined, a total number of 30 significant variables. For $l \in \{1 \dots, 10\}$, coefficients inside each group are defined as,

$$\begin{cases} \beta^l &= (1, 2, \dots, 10),\ l = 1, \dots, 3 \\ \beta^l &= (\underbrace{0, \dots, 0}_{10}),\ l = 4, \dots, 10. \end{cases}$$

Table 3 shows the results from this simulation. In both cases, the sparse and the dense distribution of variables, the best results in terms of the distance between predicted and true $\boldsymbol{\beta}$, and in terms of the test error $E_t$ are obtained by the ASGL model using $pls_d$, closely followed by the ASGL model with unpenalized model weights and by ASGL with $pca_d$ weights. All the models achieve a TPR of 1.00, indicating that all the models selected correctly all the significant variables. However, the largest TNR is achieved by ASGL with $pls_d$ or unpenalized weights, indicating that these models tend to select less non-important variables than the other models. This simulation example shows that even though the proposals made in Section 4 are

specially suited for high-dimensional scenarios, the performance achieved in low-dimensional scenarios is very good, and is at least as good as the performance of the models with unpenalized weights.

# 6 Real application

The performance of the ASGL estimator is shown here using a genomic dataset first reported in [Scheetz et al., 2006]. The dataset consists of 120 twelve-week-old male offspring animals chosen for tissue harvesting from the eyes and for micro-array analysis. The dataset contains expression values from 31042 different probe-sets (Affymetric GeneChip Rat Genome 230 2.0 Array) on a logarithmic scale. As described in [Huang et al., 2008] and [Wang et al., 2012], a two-steps preprocessing is performed, selecting, among the 31042 probe-sets, the ones that are sufficiently expressed, and sufficiently variable. A probe is considered to be sufficiently expressed if the maximum expression value observed for that probe among the 120 animals is greater than the 25-*th* percentile of the entire set of RMA expression values. A probe is considered to be sufficiently variable if it shows at least 2-fold variation in the expression value among the 120 rats. There are 18986 probes that meet these criteria.

We study how expression level of gene TRIM32, corresponding to probe 1389163_at, is related to expression levels at other probes. [Chiang et al., 2006] pointed out that gene TRIM32 was found to cause Bardet-Biedl syndrome, a disease of multiple organ systems including the retina. [Scheetz et al., 2006] said: "Any genetic element that can be shown to alter the expression of a specific gene or gene family known to be involved in a specific disease is itself an excellent candidate for involvement in the disease, either primarily or as a genetic modifier." Here the sample size is 120 (the number of animals selected for micro-array analysis), and the number of covariates (probes that pass the preprocessing steps) is 18985. The correlation coefficients of the 18985 probes and the probe corresponding to gene TRIM32 is calculated, and the genes in which the absolute value of the correlation exceeds 0.5 are selected. There are 3734 probes meeting this criteria. Finally, this dataset is standardized. Only a few genes are expected to be related to gene TRIM32, making this a high-dimensional sparse problem.

From a biological perspective it is clear that genes do not work individu-

ally. The problem of grouping genes based on a medical criteria is nowadays under intense study, and it is possible to find some group structures for human genetic information based, for example, in cytogenetic positions [Subramanian et al., 2005]. It is interesting to remark that groups built based on biological criteria are usually formed just by a few dozens of genes. For example, in the case of groups based on cytogenetic positions, groups averaged 30 genes, as stated in [Simon et al., 2013]. However, these group structures are not available for all the genetic information, and to the best of our knowledge there is no genetic grouping alternative for the dataset under study here.

We address the grouping problem from an statistical perspective, using principal components analysis to create groups of genes that are similar. It is worth to remark that in Section 4.1 PCA was used for estimating the ASGL weights, while here it will be used for variable clustering.

*Variable clustering using PCA*

1. Given a matrix of covariates $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ as in Section 4.1, obtain the matrix of principal components $\boldsymbol{Q} \in \mathbb{R}^{p \times r}$ $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ defined in Section 4.1.

2. Consider $r$ possible groups, as many as principal components.

3. Each principal component $\boldsymbol{q}_i \in \boldsymbol{Q}$, $i \in 1, \ldots, r$, is a linear combination of the original variables from $\boldsymbol{X}$. Assign each original variable to the group associated to the principal component in which that variable had its maximum weight (in absolute value).

The intuition behind this process is that variables with a large weight in the same principal component are likely to be related and should be included in the same group.

In the case of the dataset used in this section, there are 120 observations from 3734 different genes. The maximum rank of $\boldsymbol{X}$ here is 120, for this reason 120 possible groups are initially considered. Each gene is assigned to the group associated to the principal component in which that gene had its maximum weight. No gene was assigned to one of the groups, and therefore 119 groups averaging 32 genes per group are created this way. It is worth remarking that the average group size obtained based on this proposal is

Figure 6: Gene expression data of rat eye disease. Box-plot showing the sizes of the groups built using PCA.

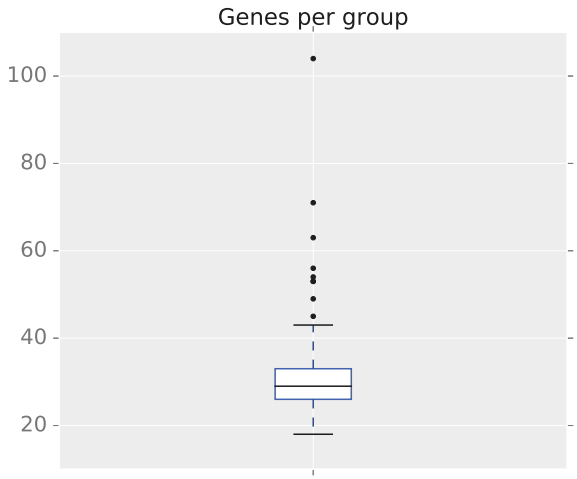



Table 4: Gene expression data of rat eye disease. 20 random dataset divisions were considered. Results displayed as mean value, with standard errors in parenthesis.

| | $E_t$ | # Variables selected |
|---|---|---|
| LASSO | 0.34 (0.08) | 18.9 (15.4) |
| SGL | 0.31 (0.07) | 189.5 (156.6) |
| ASGL-$pca_d$ | **0.28** (0.06) | 56.35 (70.86) |
| ASGL-$pls_d$ | 0.29 (0.06) | 101.7 (85.56) |

close to the expected group size in terms of the cytogenetic position. Figure 6 shows a box-plot of the group sizes.

The dataset is randomly divided into 80/20/20 train / validate / test observations and LASSO, SGL, ASGL $pls_d$ and ASGL $pca_d$ models are solved. For each model, the test error $E_t$ and the significant variables selected are obtained. This process is repeated 20 times as a way to gain stability.

The results obtained are shown in Table 4. The best results in terms of the test error are obtained by the proposed ASGL models. LASSO offers a test error approximately 20% greater while SGL test error is 11% greater. Figure 7 displays box-plots of the test error $E_t$, showing that the spread of $E_t$ is also smaller in the proposed ASGL models providing more stable results. Figure 8 displays box-plots of the number of genes each model selected as

Figure 7: Gene expression data of rat eye disease. 20 random dataset divisions were considered. Box-plot showing the test error.

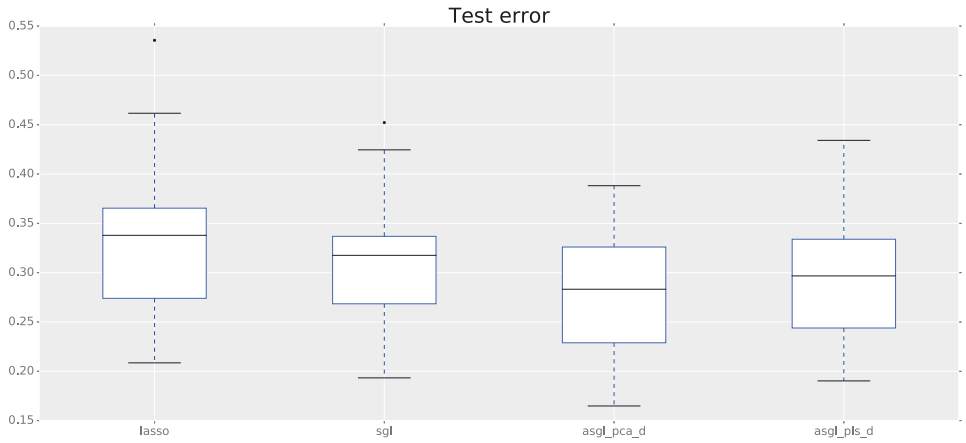



Figure 8: Gene expression data of rat eye disease. 20 random dataset divisions were considered. Box-plot showing the number of significant genes.

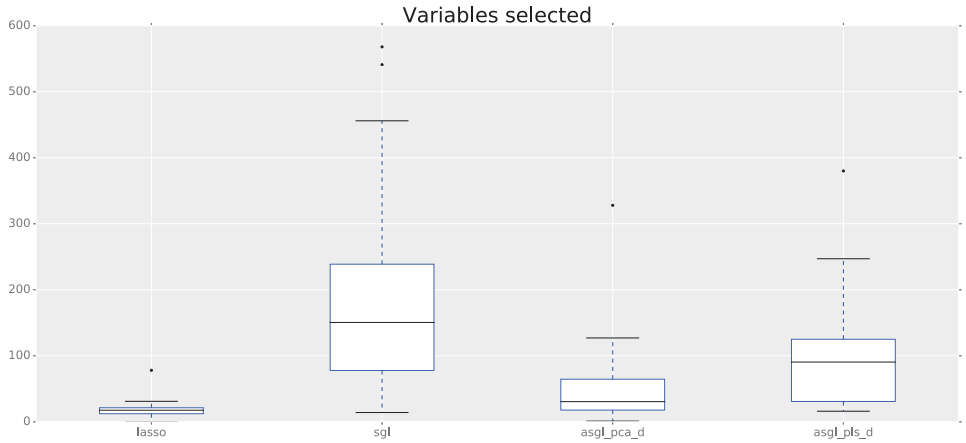
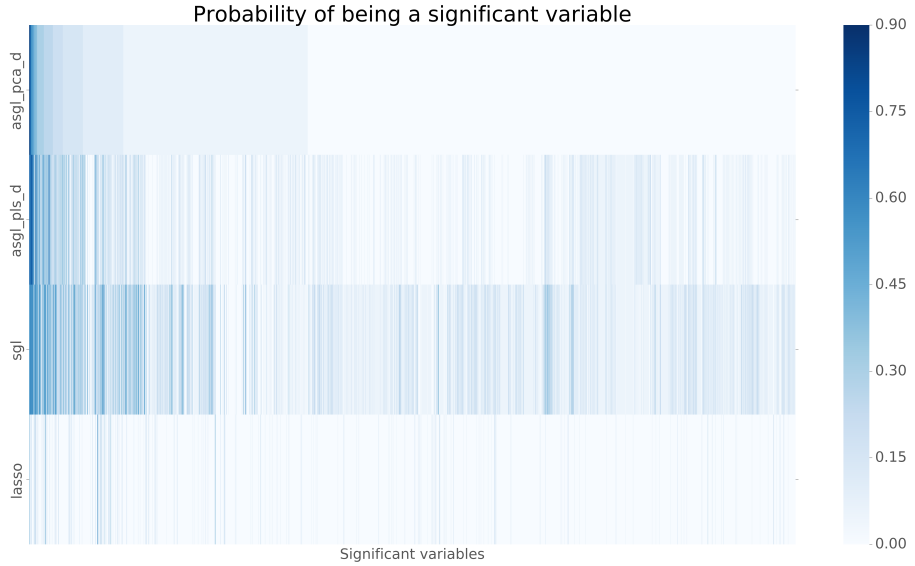


Figure 9: Gene expression data of rat eye disease. 20 random dataset divisions were considered. Heatmap showing the probability of being a significant gene. Each row represents a model and each column represents a gene.




significant. The LASSO is the one offering more sparse solutions, using only 19 variables (in mean) per model. SGL is the one using the largest number of variables, approximately 190, and also the one with the largest variability in this metric. Both ASGL $pca_d$ and ASGL $pls_d$ selected a smaller number of variables than SGL but still larger than LASSO, and they achieve the best prediction results of the four models.

Given that we have the results obtained from 20 repetitions, it is possible to count the number of times each gene has been selected as significant by one of the models in any of the repetitions. Dividing this number by the total number of repetitions, a sort of "probability of being a significant gene" associated to each gene for each model considered is obtained. Out of the 3734 genes in the dataset, 1612 genes were selected at least one time by any of the models in any of the repetitions (the majority being selected by SGL models). Figure 9 shows the probability of being a significant gene for these 1612 variables and for each model. Rows represent the different models considered and columns represent each gene. Genes are sorted based on the probabilities obtained in the ASGL model with $pca_d$ weights.

Considering a probability threshold of 0.5, only 1 gene in the LASSO

Table 5: Gene expression data of rat eye disease. 20 random dataset divisions were considered. Number of genes above the probability threshold for different quantile levels.

| Number of genes above the probability threshold | | | | |
|---|---|---|---|---|
| | $\tau = 0.3$ | $\tau = 0.5$ | $\tau = 0.7$ | Three quantiles |
| LASSO | 0 | 1 | 1 | 0 |
| SGL | 19 | 35 | 17 | 0 |
| ASGL-$pca_d$ | 23 | 9 | 17 | 7 |
| ASGL-$pls_d$ | 41 | 17 | 37 | 9 |

models reach a probability of significance above the threshold, showing no stability on the gene selection along the 20 repetitions, and anticipating problems with possible further biological interpretation of the statistical results. In the case of the SGL model, 35 genes are above the probability threshold, being 0.6 the maximum probability achieved. On the other hand, the ASGL model with $pls_d$ weights includes 17 genes with probabilities above the threshold with a maximum probability value of 0.75, and the ASGL model with $pca_d$ weights has 9 genes above the probability threshold with a maximum probability value of 0.9, showing more stability on the selection along the 20 repetitions and possibly better biological interpretation of the results than the other models.

Results displayed in Table 4 and Figure 9 have been obtained using estimators of the median of the response variable, however, it can be interesting to compare the genes selected at different quantiles. For this reason, the process described above is repeated and LASSO, SGL, ASGL $pls_d$ and ASGL $pca_d$ models are solved for quantile levels $\tau = 0.3$ and $\tau = 0.7$, obtaining probabilities of being a significant gene for each quantile level and each model. Considering a probability threshold of 0.5, Table 5 show the number of genes above the probability threshold for each quantile, and also the number of genes in the same model that have been selected along the different quantile levels.

The LASSO model shows no stability on the variable selection, having only one gene above the threshold for $\tau = 0.5$ and $\tau = 0.7$, and no gene with probability of being significant above 0.5 on the three quantiles simultaneously. The SGL show some stability across the 20 repetitions considering

each quantile independently, but when considering all the quantiles simultaneously it has no gene above the probability threshold. On the other hand, in the case of the ASGL $pls_d$ model, 9 genes had a probability of being significant greater than 0.5 in the 3 quantiles, and in the case of the ASGL $pca_d$ models, 7 genes fulfilled this, showing more robust results than the other estimators.

We conclude that the best results in this real dataset study are provided by the ASGL model with $pca_d$ weights, given that this model is the one with the smallest prediction error and showing great stability on the gene selection.

# 7 Computational aspects

All the simulations and data analysis commented in Sections 5 and 6 were run in a cluster node with two `Intel (R) Xeon(R) CPU E5-2630 v3` (2.4GHz, 20MB Smart Cache) processors, with 32Gb of RAM memory running CentOS 6.5 Final (Rocks 6.1.1 Sand Boa). The computation itself has been developed in Python 2.7.15 (Anaconda Inc.) All the optimization problems have been solved using the CVXPY optimization framework for Python [Diamond and Boyd, 2016] and the open source solver ECOS [Domahidi et al., 2013].

# 8 Conclusion

In this paper the definition of the SGL estimator has been extended to the QR framework. A new estimator for quantile regression based on the usage of adaptive weights, the adaptive sparse group LASSO in quantile regression has also been proposed. Typically in the literature the weights of the adaptive estimators are defined based on non-penalized models, however, this definition limits the usage of the adaptive estimators to low-dimensional scenarios. As a solution to this problem, four weight calculation alternatives that can be used in high-dimensional scenarios when working with adaptive estimators are proposed. We test the performance of the proposed alternatives in a set of synthetic data scenarios that includes high-dimensional and low-dimensional examples, showing that it is a competitive option in both situations. The performance of the proposed work is also studied in

a real dataset including gene expression values of rat eye disease. We compare the results of the proposed work with LASSO, SGL and ASGL with weights from unpenalized models (the last one only in the low-dimensional examples). Based on the synthetic datasets both ASGL $pls_d$ and ASGL $pca_d$ achieve very good results, as described in Section 5, however, when dealing with the real dataset, the ASGL $pca_d$ estimator achieved better results in terms of prediction error and stability of the variables selected. For this reason we conclude that the ASGL $pca_d$ provides the best results among the options proposed in this work.

This work has risen some questions that will require further investigation. One interesting problem is this of the optimization of the hyper-parameters. In this work we make use of grid-search, but it is worth commenting that new hyper-parameter tuning alternatives have appeared in recent years [Laria et al., 2019], and it can be interesting to investigate the usage of this or other options in the optimization of the parameters of the proposed models here.

As commented in Section 3, the general tendency found in the literature when working with adaptive estimators is to work in order to demonstrate that the estimator fulfills the so-called oracle properties. This demonstration is usually based on the usage of the un-penalized model estimator, step that can only be achieved in low-dimensional scenarios. Actually, this requirement can be reformulated, changing the usage of the un-penalized estimator by any other root-n consistent estimator We thus conjecture that the ASGL $pls_d$ and ASGL $pca_d$ estimators, as they are used in this work, may be root-n consistent, fulfilling this way the oracle properties. This conjecture however will require further research.

# 9 Acknowledgments

In this research we have made use of Uranus, a supercomputer cluster located at University Carlos III of Madrid and funded jointly by EU-FEDER funds and by the Spanish Government via the National Projects No. UNC313-4E-2361, No. ENE2009-12213- C03-03, No. ENE2012-33219 and No. ENE2015-68265-P. This research was partially supported by research grants and Project ECO2015-66593-P from Ministerio de Economía, Industria y Competitividad, Project MTM2017-88708-P from Ministerio de Economía y Competi-

tividad, FEDER funds and Project IJCI-2017-34038 from Agencia Estatal de Investigación, Ministerio de Ciencia, Innovación y Universidades.

## References

[Chatterjee et al., 2011] Chatterjee, S., Banerjee, A., Chatterjee, S., and Ganguly, A. R. (2011). Sparse Group Lasso for Regression on Land Climate Variables. In *2011 IEEE 11th International Conference on Data Mining Workshops*, pages 1–8. IEEE.

[Chiang et al., 2006] Chiang, A. P., Beck, J. S., Yen, H.-J., Tayeh, M. K., Scheetz, T. E., Swiderski, R. E., Nishimura, D. Y., Braun, T. A., Kim, K.-Y. A., Huang, J., Elbedour, K., Carmi, R., Slusarski, D. C., Casavant, T. L., Stone, E. M., and Sheffield, V. C. (2006). Homozygosity mapping with SNP arrays identifies TRIM32, an E3 ubiquitin ligase, as a Bardet-Biedl syndrome gene (BBS11). *Proceedings of the National Academy of Sciences*, 103(16):6287–6292.

[Ciuperca, 2017] Ciuperca, G. (2017). Adaptive fused LASSO in grouped quantile regression. *Journal of Statistical Theory and Practice*, 11(1):107–125.

[Ciuperca, 2019] Ciuperca, G. (2019). Adaptive group LASSO selection in quantile models. *Statistical Papers*, 60(1):173–197.

[Diamond and Boyd, 2016] Diamond, S. and Boyd, S. (2016). CVXPY: A Python-Embedded Modeling Language for Convex Optimization. *arXiv:1603.00943*.

[Domahidi et al., 2013] Domahidi, A., Chu, E., and Boyd, S. (2013). ECOS: An SOCP Solver for Embedded Systems. In *European Control Conference (ECC)*.

[Fan and Li, 2001] Fan, J. and Li, R. (2001). Variable Selection via Nonconcave Penalized Likelihood and Its Oracle Properties. *Journal of the American Statistical Association*, 96(456):1348–1360.

[Friedman et al., 2010] Friedman, J., Hastie, T., and Tibshirani, R. (2010). A note on the group lasso and a sparse group lasso. *arXiv:1001.0736v1*.

[Huang et al., 2008] Huang, J., Ma, S., and Zhang, C.-H. (2008). Adaptive Lasso for Sparse High-dimensional Regression. *Statistica Sinica*, 1(374):1–28.

[Koenker, 2005] Koenker, R. (2005). *Quantile Regression*. Cambridge university Press.

[Koenker and Bassett, 1978] Koenker, R. and Bassett, G. (1978). Regression Quantiles. *Econometrica*, 46(1):33–50.

[Laria et al., 2019] Laria, J. C., Aguilera-Morillo, M. C., and Lillo, R. E. (2019). An iterative sparse-group lasso. *Journal of Computational and Graphical Statistics*, :1–21. doi:10.1080/10618600.2019.1573687

[Li and Zhu, 2008] Li, Y. and Zhu, J. (2008). L1- -Norm Quantile Regression. *Journal of Computational and Graphical Statistics*, 17(1):1–23.

[Nardi and Rinaldo, 2008] Nardi, Y. and Rinaldo, A. (2008). On the asymptotic properties of the group lasso estimator for linear models. *Electronic Journal of Statistics*, 2(0):605–633.

[Poignard, 2018] Poignard, B. (2018). Asymptotic theory of the adaptive Sparse Group Lasso. *Annals of the Institute of Statistical Mathematics*.

[Scheetz et al., 2006] Scheetz, T. E., Kim, K.-Y. A., Swiderski, R. E., Philp, A. R., Braun, T. A., Knudtson, K. L., Dorrance, A. M., DiBona, G. F., Huang, J., Casavant, T. L., Sheffield, V. C., and Stone, E. M. (2006). Regulation of gene expression in the mammalian eye and its relevance to eye disease. *Proceedings of the National Academy of Sciences*, 103(39):14429–14434.

[Simon et al., 2013] Simon, N., Friedman, J., Hastie, T., and Tibshirani, R. (2013). A sparse-group lasso. *Journal of Computational and Graphical Statistics*, 22(2):231–245.

[Subramanian et al., 2005] Subramanian, A., Tamayo, P., Mootha, V. K., Mukherjee, S., Ebert, B. L., Gillette, M. A., Paulovich, A., Pomeroy, S. L., Golub, T. R., Lander, E. S., and Mesirov, J. P. (2005). Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences*, 102(43):15545–15550.

[Tibshirani, 1996] Tibshirani, R. (1996). Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288.

[Wang et al., 2012] Wang, L., Wu, Y., and Li, R. (2012). Quantile regression for analyzing heterogeneity in ultra-high dimension. *Journal of the American Statistical Association*, 107(497):214–222.

[Wright et al., 2010] Wright, J., Ma, Y., Mairal, J., Sapiro, G., Huang, T. S., and Yan, S. (2010). Sparse Representation for Computer Vision and Pattern Recognition. *Proceedings of the IEEE*, 98(6):1031–1044.

[Wu and Liu, 2009] Wu, Y. and Liu, Y. (2009). Variable selection in quantile regression. *Statistica Sinica*, 19(2):801–817.

[Yuan and Lin, 2006] Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society. Series B (Methodological)*, 68(1):49–67.

[Zhao et al., 2014] Zhao, W., Zhang, R., and Liu, J. (2014). Sparse group variable selection based on quantile hierarchical Lasso. *Journal of Applied Statistics*, 41(8):1658–1677.

[Zhou and Zhu, 2010] Zhou, N. and Zhu, J. (2010). Group variable selection via a hierarchical lasso and its oracle property. *Statistics and Its Interface*, 3:557–574.

[Zou, 2006] Zou, H. (2006). The Adaptive Lasso and Its Oracle Properties. *Journal of the American Statistical Association*, 101(476):1418–1429.