Working paper

2021-06

Statistics and Econometrics
ISSN 2387-0303

A quantile based dimension reduction technique

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

uc3m | Universidad Carlos III de Madrid

A quantile based dimension reduction technique

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

Abstract

Partial least squares (PLS) is a dimensionality reduction technique used as an alternative to ordinary least squares (OLS) in situations where the data is colinear or high dimensional. Both PLS and OLS provide mean based estimates, which are extremely sensitive to the presence of outliers or heavy tailed distributions. In contrast, quantile regression is an alternative to OLS that computes robust quantile based estimates. In this work, the multivariate PLS is extended to the quantile regression framework, obtaining a theoretical formulation of the problem and a robust dimensionality reduction technique that we call fast partial quantile regression (fPQR), that provides quantile based estimates. An efficient implementation of fPQR is also derived, and its performance is studied through simulation experiments and the chemometrics well known biscuit dough dataset, a real high dimensional example.

keywords: partial-least-squares; quantile-regression; dimension-reduction; outliers; robust.

1 Introduction

Partial least squares (PLS) [Wold, 1973], [Wold et al., 2001] is a dimensionality reduction technique commonly applied to two data blocks (predictors and responses) that works by projecting the available data into a latent structure. The key idea behind PLS is that it can summarize the predictors into a small set of uncorrelated latent variables that have maximal covariance with the responses. PLS has proven to be a versatile alternative to

*Department of Statistics, Universidad Carlos III de Madrid.
†uc3m-Santander Big Data Institute.
‡Department of Applied Statistics and Operational Research, and Quality, Universitat Politècnica de València

ordinary least squares (OLS), obtaining parsimonious models even when dealing with ill-posed multicollinear problems, commonly found in different areas of scientific research such as chemometrics, social science or medicine. See for example [Nguyen and Rocke, 2002], where it is used in a tumor classification problem. In recent years PLS has also received attention when dealing with the increasingly common problem of high dimensional data, in which the number of observations is small and the number of variables is very large. In this regard, Boulesteix and Strimmer [2006] successfully applied PLS to a genomic dataset. Partial least squares is based on the cross-covariance matrix between predictors and response, and on least squares models. Least squares models are known to behave nicely when the errors are normally distributed, but there is no guarantee that the normality will be satisfied in many experimental data problems, where heavy tailed distributions, and even outliers are expected to be found. This makes PLS extremely sensitive to the presence of outliers or non normal data. The solution to this problem has traditionally been centered in robustifying the least squares estimator in which PLS is based, see for example [Serneels et al., 2005] where they make use of a robust M-regression estimator, or [Acitas et al., 2020], where a partial robust adaptive modified maximum likelihood estimator is proposed, among others.

Quantile regression [Koenker and Bassett, 1978] is an important statistical methodology that allows to describe the conditional quantiles of a response given a set of covariates. Fitting the data at a set of quantiles provides a more comprehensive picture of the response distribution than does the mean, and as opposed to least squares, quantile regression is resistant to outliers, and can deal with heavy tailed distributions and heteroscedasticity, the situation when variances depend on some covariates. Specifically, when the center of the distribution is of interest, the least absolute deviation (LAD), also called median regression, a particular case of quantile regression, provides more robust estimators than least squares regression. In recent years many papers have been published extending quantile regression to the high dimensional framework by performing variable selection, see for example Wu and Liu [2009] where an adaptive lasso for quantile regression is introduced, or [Mendez-Civieta et al., 2020], where an adaptive sparse group lasso for quantile regression is proposed. However, to the best of our knowledge there is very little work on quantile based dimension reduction techniques. A well known PLS implementation is given by the NIPALS algorithm [Wold, 1973]. Dodge and Whittaker [2009] extended the NIPALS algorithm for univariate response problems to the quantile regression framework. They proposed a quantile covariance metric based on the quantile regression slope and used this metric to modify the univariate NIPALS, a modification that they called partial quantile regression (PQR). The work from Dodge and Whittaker [2009] lays the foundation for an extension of PLS to the quantile regression framework, however we find some shortcomings in the development of the methodology and the algorithmic implementation that should be addressed. First, it

has no background on what is the optimization problem that their PQR algorithm is solving. Second, it is centered in univariate response problems, providing no solution for multivariate response problems commonly found in fields such as chemometrics. Third, the computation time of their quantile coviariance, key in the algorithmic implementation, grows linearly with the number of variables, making solving high dimensional problems computationally expensive. The main contribution of our work is centered in addressing these problems. We define the optimization problem that the fPQR algorithm solves and study different quantile covariance alternatives [Li et al., 2015], [Choi and Shin, 2018]. We provide an efficient implementation of fPQR, greatly reducing the computation time when compared with that of [Dodge and Whittaker, 2009] while achieving more accurate predictions. We also provide an implementation suitable for multivariate response settings. The result is a methodology that parallels the nice properties of PLS: it is a dimension reduction technique that obtains uncorrelated scores maximizing the quantile covariance between predictors and responses. But additionally, it is also a robust, quantile linked methodology suitable for dealing with outliers, heteroscedastic or heavy tailed datasets. The median estimator of the fPQR algorithm is a robust alternative to PLS, while other quantile levels can provide additional information on the tails of the responses.

The rest of the paper is organized as follows. In Section 2 a brief introduction of the PLS algorithm for multivariate response is provided. Section 3 introduces the fPQR algorithm and studies different options for a quantile covariance metric. Section 4 tests the performance of the proposed fPQR algorithm in three synthetic dataset frameworks studying the quality of the estimated β coefficients and the prediction error. In Section 5, the proposed algorithm is used in a real high dimensional data example. Some computational aspects are briefly commented in Section 6, and the conclusions are provided in Section 7.

2 The PLS model for multivariate response

Let $X \in \mathbb{R}^{n\times m}$ and $Y \in \mathbb{R}^{n\times l}$ be two data matrices, samples drawn from some unknown population following the linear model,

$$\boldsymbol{y}_i = \boldsymbol{x_i} B + \boldsymbol{\varepsilon}_i, \quad i = 1, \ldots, n, \tag{1}$$

where $\boldsymbol{y}_i \equiv (y_{i1}, \ldots, y_{il})$ is the vector containing the response variables for the i-th observation, $\boldsymbol{x}_i \equiv (x_{i1}, \ldots, x_{im})$ contains the predictive variables, $B \in \mathbb{R}^{m\times l}$ is the matrix containing the coefficients from the linear relations, and $\boldsymbol{\varepsilon_i} \equiv (\varepsilon_{i1}, \ldots, \varepsilon_{il})$ is the error term. Without loss of generality, consider that both X and Y are mean centered. The PLS regression

methodology works by assuming the existence of a latent structure,

$$X = TP^t + E; \quad Y = TQ^t + F, \tag{2}$$

where $T \in \mathbb{R}^{n\times h}$ is the scores matrix formed by h (usually being $h \ll m$) linear combinations of the original variables, $P \in \mathbb{R}^{m\times h}$ and $Q \in \mathbb{R}^{l\times h}$ are loadings matrices and $E \in \mathbb{R}^{n\times m}$ and $F \in \mathbb{R}^{n\times l}$ are random error matrices. The aim of PLS regression is precisely to regress the response matrix Y onto the h latent variables, stored in the scores matrix T, defining this way a low-dimensional regression model,

$$\boldsymbol{y}_i = \boldsymbol{t}_i\Gamma + \boldsymbol{\varepsilon}_i^*, \quad i = 1, \ldots, n, \tag{3}$$

where Γ is the matrix of regression coefficients. PLS is an iterative algorithm in which the scores in T are obtained sequantially. There are multiple definitions of the PLS algorithm available in the literature, being NIPALS [Wold, 1973] and SIMPLS [de Jong, 1993] the most frequently used ones. Here a version of NIPALS that will be useful in the implementation of the fPQR algorithm is considered:

Step 1: Define $X_0 = X$ and $Y_0 = Y$.

Step 2: Compute $S_1 = X_0^t Y_0$ the sample covariance matrix.

Step 3: Obtain the eigen decomposition of $S_1 S_1^t$ and take $\boldsymbol{w}_1$ as the eigenvector associated to the largest eigenvalue.

Step 4: Calculate the X score vector as $\boldsymbol{t}_1 = X_1\boldsymbol{w}_1$.

Step 5: Calculate the X loading vector as $\boldsymbol{p}_1 = \dfrac{X_1^t\boldsymbol{t}_1}{\boldsymbol{t}_1^t\boldsymbol{t}_1}$.

Step 6: Calculate the Y loading vector as $\boldsymbol{q}_1 = \dfrac{Y_1^t\boldsymbol{t}_1}{\boldsymbol{t}_1^t\boldsymbol{t}_1}$.

Step 7: Deflat the matrix X_0 from the information already explained by scores $\boldsymbol{t}_1$ and obtain $X_1 = X_0 - \boldsymbol{t}_1\boldsymbol{p}_1^t$.

Step 8: Deflat the matrix Y_1 from the information already explained by scores $\boldsymbol{t}_1$ and obtain $Y_1 = Y_0 - \boldsymbol{t}_1\boldsymbol{q}_1^t$.

Iterate through steps 2-8 until all h components are computed. Observe that the deflation process stated in step 7 ensures that the score matrix T will be orthogonal. Once all the

required components have been computed, the parameter estimates $\hat{\Gamma}$ from equation (3) are obtained solving the low dimensional least squares model,

$$\hat{\Gamma} = \arg\min_{\Gamma} \left\{ \|Y - T\Gamma\|^2 \right\}. \tag{4}$$

Finally, one can project the estimate $\hat{\Gamma}$ back into the original sub-space spawned by X and obtain,

$$\hat{B} = W(P^t W)^{-1}\hat{\Gamma}. \tag{5}$$

PLS is essentially a covariance maximization problem where, at each iteration $a+1$, the objective function being solved is defined as,

$$\boldsymbol{w}_{a+1} = \underset{\boldsymbol{w}, \|\boldsymbol{w}\|=1}{\arg\max} \left\{ \operatorname{cov}(X_a \boldsymbol{w}, Y_a) \operatorname{cov}(X_a \boldsymbol{w}, Y_a)^t \right\}, \tag{6}$$

where $X_0 = X$ and $Y_0 = Y$, and the solution is the eigenvector associated to the largest eigenvalue λ_1,

$$S_a S_a^t \boldsymbol{w_a} = \lambda_1 \boldsymbol{w_a}. \tag{7}$$

Posing PLS as a covariance optimization problem opens the door to the possibility of using alternative covariance definitions. Traditionally, robust versions have been considered in order to obtain robustified PLS algorithms, see for example [Hubert and Branden, 2003]. In this work we are interested in defining not only a robust PLS estimator, but an estimator linked to the quantiles of the response matrix, giving the possibility to study the tails of the response matrix and not just the central behavior. As a solution to this question, a robust quantile based dimension reduction technique that we call fast partial quantile regression (fPQR) is introduced in the next section.

3 Fast partial quantile regression

There are two key steps in the definition of the fPQR methodology. First, the usage of a quantile covariance metric linked to the quantiles, instead of the traditional covariance, that is linked to the mean. As it will be discussed in Section 4, the metric that we consider to be the best alternative was proposed by Li et al. [2015], although other alternatives [Dodge and Whittaker, 2009], [Choi and Shin, 2018] will also be studied along Sections 3.3 and 4. Second, the estimation of the Γ coefficients defined in equation (3). In the PLS algorithm, these coefficients are estimated using ordinary least squares, but in the fPQR algorithm a quantile regression model is used instead ensuring that the $\hat{\Gamma}$ estimates remain linked to the

quantiles of the response matrix Y.

3.1 A quantile covariance

In a very interesting work, Li et al. [2015] extended the usage of autoregressive models to the quantile framework by defining a novel measure suitable for examining the linear relationships between any two random variables for a given quantile $\tau \in (0, 1)$, a measure that they called quantile correlation. Given two random variables Z_1 and Z_2, take Q_{τ,Z_2} as the τ-th quantile of Z_2 and $\mathrm{Q}_{\tau,Z_2}(Z_1)$ as the τ-th quantile of Z_2 conditional to Z_1. Then it is possible to demonstrate that $\mathrm{Q}_{\tau,Z_2}(Z_1)$ is independent of Z_1 if and only if the random variables $I(Z_2 - \mathrm{Q}_{\tau,Z_2} > 0)$ and Z_1 are independent, where $I(\cdot)$ is the indicator function. This fact motivated the definition of the quantile covariance proposed in their work as,

$$\begin{aligned} \mathrm{qcov}_\tau\{Z_1, Z_2\} &= \mathrm{cov}\left\{I\left(Z_2 - \mathrm{Q}_{\tau,Z_2} > 0\right), Z_1\right\} \\ &= E\left\{\psi_\tau\left(Z_2 - \mathrm{Q}_{\tau,Z_2}\right)(Z_1 - EZ_1)\right\}, \end{aligned} \tag{8}$$

where $\psi_\tau(w) = \tau - I(w < 0)$. Being based on a traditional covariance makes this quantile covariance easy and fast to compute. Additionally, although this definition is proposed for random variables, it can be extended to random vectors, making it possible to adapt to the data matrices found in multidimensional problems. Observe however that, opposed to the traditional covariance, this quantile covariance does not enjoy the symmetry property, that is, $\mathrm{qcov}_\tau(Z_1, Z_2) \neq \mathrm{qcov}_\tau(Z_2, Z_1)$. A complete definition of this metric can be found in [Li et al., 2015] where they study a nice relation between this metric and the slope from a quantile regression model, and also the asymptotic properties of the estimator.

3.2 The fPQR algorithm

The objective function that the fPQR algorithm solves is obtained by adapting the objective function from a PLS model as it was defined in equation (6) using the quantile covariance introduced in Section 3.1,

$$\begin{aligned} \boldsymbol{w}_{a+1} &= \underset{\boldsymbol{w},\|\boldsymbol{w}\|=1}{\arg\max} \left\{\mathrm{qcov}_\tau(X_a\boldsymbol{w}, Y_a)^t\, \mathrm{qcov}_\tau(X_a\boldsymbol{w}, Y_a)\right\} \\ &= \underset{\boldsymbol{w},\|\boldsymbol{w}\|=1}{\arg\max} \left\{\boldsymbol{w}^t X_a^t \psi_\tau(Y_a - \mathrm{Q}_{\tau,Y_a})\psi_\tau(Y_a - \mathrm{Q}_{\tau,Y_a})^t X_a\boldsymbol{w}\right\}, \end{aligned} \tag{9}$$

where $\psi_\tau(w) = \tau - I(w < 0)$. The solution to this equation is the eigenvector associated to the largest eigenvalue λ_1,

$$X_a^t \psi_\tau(Y_a - \mathrm{Q}_{\tau,Y_a})\psi_\tau(Y_a - \mathrm{Q}_{\tau,Y_a})^t X_a\boldsymbol{w_a} = \lambda_1 \boldsymbol{w_a}. \tag{10}$$

Based on this idea, the main steps of the fPQR algorithm are defined below,

Step 1: Take $\tau \in (0,1)$ the quantile level of interest.

Step 2: Define $X_0 = X$ and $Y_0 = Y$.

Step 3: Compute $S_{1,\tau} = \text{qcov}_\tau(X_0, Y_0)$ the sample quantile covariance matrix.

Step 4: Obtain the eigen decomposition of $S_{1,\tau}S_{1,\tau}^t$ and take $\boldsymbol{w}_1$ as the eigenvector associated to the largest eigenvalue.

Step 5: Calculate the X score vector as $\boldsymbol{t}_1 = X_1\boldsymbol{w}_1$.

Step 6: Calculate the X loading vector as $\boldsymbol{p}_1 = \dfrac{X_1^t\boldsymbol{t}_1}{\boldsymbol{t}_1^t\boldsymbol{t}_1}$.

Step 7: Calculate the Y loading vector as $\boldsymbol{q}_1 = \dfrac{Y_1^t\boldsymbol{t}_1}{\boldsymbol{t}_1^t\boldsymbol{t}_1}$.

Step 8: Deflat the matrix X_1 from the information already explained by scores $\boldsymbol{t}_1$ and obtain $X_2 = X_1 - \boldsymbol{t}_1\boldsymbol{p}_1^t$.

Step 9: Deflat the matrix Y_1 from the information already explained by scores $\boldsymbol{t}_1$ and obtain $Y_2 = Y_1 - \boldsymbol{t}_1\boldsymbol{q}_1^t$.

Iterate through steps 2-8 until all h components are computed. In order to obtain the parameter estimates $\hat{\Gamma}$ in the PLS algorithm, a least squares model was solved following equation (4), but in the fPQR algorithm this is substituted by a quantile regression model solving,

$$\tilde{\Gamma} = \arg\min_{\boldsymbol{\beta}} \left\{ \frac{1}{n}\sum_{i=1}^{n} \rho_\tau(\boldsymbol{y}_i - \boldsymbol{t}_i^t\Gamma) \right\}, \tag{11}$$

where $\rho_\tau(u) = u(\tau - I(u < 0))$ is the quantile regression loss check function. Using a quantile regression model here ensures that the $\hat{\Gamma}$ estimates remain linked to the quantile of the response matrix Y. Finally, one can project $\hat{\Gamma}$ back into the original sub-space spawned by X as it was done in the PLS models in equation (5). The fPQR is an algorithm that shares many of the benefits of PLS:

- It is a dimension reduction technique suitable for multicollinear or high dimensional data;
- The new scores obtained by the algorithm are orthogonal;
- It maximizes the quantile covariance between predictor and response.

But it also has some additional properties:

- It is a robust methodology, suitable for dealing with outliers or heteroscedastic data;
- It can provide an estimation of the central behavior of the response conditional to the predictors, but additionally can provide an estimation of any other quantile of the response, conditional to the predictors, obtaining a complete view of the distribution of the response.

3.3 Other quantile covariance metrics

In Section 3.2, the fPQR algorithm was defined as an optimization problem where a quantile covariance metric is maximized. Although the metric proposed by Li et al. [2015] was used in the definition of the algorithm, it is possible to consider alternative versions of fPQR based on other quantile covariance metrics. Along this section, two other candidates, defined by [Dodge and Whittaker, 2009] and [Choi and Shin, 2018] are considered, showing their definition and some properties related to the fPQR performance.

3.3.1 A quantile covariance from Dodge and Whittaker [2009]

Take two random variables Z_1 and Z_2 following the linear model,

$$Z_2 = Z_1\beta + \varepsilon. \tag{12}$$

The analytical solution of the ordinary least squares estimator for model (12) is,

$$\hat{\beta} = \mathrm{var}(Z_1)^{-1}\,\mathrm{cov}(Z_1, Z_2). \tag{13}$$

Dodge and Whittaker [2009] take advantage of this fact and define a quantile covariance in terms of the quantile regression estimator, mimicking the relation between the OLS estimator and the traditional covariance displayed in equation (13). Consider the quantile regression estimator,

$$\tilde{\beta} = \arg\min_{\beta} \left\{ \mathrm{E}\,\rho_\tau(Z_2 - \beta Z_1) \right\}, \tag{14}$$

where $\rho_\tau(u) = u(\tau - I(u < 0))$ is the quantile regression loss check function. Then the quantile covariance proposed by Dodge and Whittaker [2009] is obtained as,

$$\mathrm{qcov}^*_\tau(Z_1, Z_2) = \mathrm{var}(Z_1)\tilde{\beta}, \tag{15}$$

where $\tilde{\beta}$ is the quantile regression estimator defined in equation (14). Here the superscript "$*$" differentiates this quantile covariance from the one defined in Section 3.1. There are

some remarks worth mentioning:

- The extension of this quantile covariance to a multidimensional setting is not as straightforward as in the traditional covariance or in the quantile covariance proposed by Li et al. [2015]. This means that given a random vector $U \equiv (U_1, \ldots, U_m)$,

$$\text{qcov}^*_\tau(U, Z_2) \neq (\text{qcov}^*_\tau(U_1, Z_2), \ldots, \text{qcov}^*_\tau(U_m, Z_2)). \tag{16}$$

  This implies that, in order to ensure that the quantile covariance (in the sense of Dodge and Whittaker [2009]) between two random variables remains the same regardless of the computation affecting a random vector or not, it must be computed univariatedly. This way, the computation of the quantile covariance between U and Z_2 requires to solve m univariate quantile regression models, where m is the dimension of U, greatly affecting the computation time as the number of variables increase;

- As happened with the quantile covariance defined by Li et al. [2015], this quantile covariance is not symmetric. This means that $\text{qcov}^*_\tau(Z_1, Z_2) \neq \text{qcov}^*_\tau(Z_2, Z_1)$

Additionally to the quantile covariance described above, the key contribution of Dodge and Whittaker [2009] was the adaptation of the univariate NIPALS algorithm to the quantile regression framework. The main differences between their proposal (PQR) and the work developed here (fPQR) are listed below:

- In the work developed here, the optimization problem that the fPQR algorithm solves is clearly defined, and based on this definition, the algorithm is proposed. Opposed to this, Dodge and Whittaker [2009] simply defined the algorithm as a modification of the univariate PLS NIPALS, without studying the optimization problem;

- The fPQR algorithm allows Y to be a multivariate response matrix while the PQR algorithm is limited to univariate responses;

- As it will be seen in Section 4, the covariance considered in the fPQR algorithm alllows the algorithm to run significantly faster than the PQR algorithm.

3.3.2 A quantile covariance from Choi and Shin [2018]

Given two random variables Z_1 and Z_2, the Pearson correlation between the two variables can be seen as the geometric mean of two OLS slopes, $\beta_{2.1}$ of Z_1 on Z_2 and $\beta_{1.2}$ of Z_2 on Z_1,

$$\text{cor}(Z_1, Z_2) = \text{sign}(\beta_{2.1})\sqrt{\beta_{2.1}\beta_{1.2}}. \tag{17}$$

Based on this idea, Choi and Shin [2018] proposed a quantile correlation coefficient defined as the geometric mean of two quantile regression slopes,

$$\text{qcor}^{**}_{\tau}(Z_1, Z_2) = \text{sign}(\beta_{2.1}(\tau))\sqrt{\beta_{2.1}(\tau)\beta_{1.2}(\tau)}, \tag{18}$$

where the superscript "$**$" is used to differentiate this metric from the ones from [Li et al., 2015] and [Dodge and Whittaker, 2009]. A full review of the properties of this metric can be found in the original paper [Choi and Shin, 2018] but there are some remarks that are worth mentioning:

- As it happened with the quantile covariance defined by Dodge and Whittaker [2009], given a random vector $U \equiv (U_1, \ldots, U_m)$,

$$\text{qcor}^{**}_{\tau}(U, Z_2) \neq (\text{qcor}^{**}_{\tau}(U_1, Z_2), \ldots, \text{qcor}^{**}_{\tau}(U_m, Z_2)). \tag{19}$$

  This implies that in order to ensure consistency of the results when dealing with random vectors, this metric must also be computed univariatedly. The computation of $\text{qcor}^{**}_{\tau}(U, Z_2)$ requires thus to solve $2m$ univariate quantile regression models, where m is the dimension of U, greatly affecting the computation time;

- Opposed to the other quantile metrics under study, this is the only metric that is symmetric, meaning that $\text{qcor}^{**}_{\tau}(Z_1, Z_2) = \text{qcor}^{**}_{\tau}(Z_2, Z_1)$.

Observe that the fPQR algorithm requires a quantile covariance, and not a quantile correlation. Although not defined in the original paper, it is possible to obtain an estimation of a quantile covariance based on equation (18). Observe that,

$$\begin{aligned} \text{qcor}^{**}_{\tau}(Z_1, Z_2) &= \text{sign}\left(\beta_{2,1}(\tau)\right)\sqrt{\beta_{2,1}(\tau)\beta_{1,1}(\tau)} \\ &= \text{sign}\left(\beta_{2,1}(\tau)\right)\sqrt{\frac{\text{qcov}^{*}_{\tau}(Z_1, Z_2)\,\text{qcov}^{*}_{\tau}(Z_2, Z_1)}{\text{var}(Z_1)\,\text{var}(Z_2)}}, \end{aligned} \tag{20}$$

where $\text{qcov}^{*}(\cdot,\cdot)$ refers to the quantile covariance introduced in Section 3.3.1. This way, a symmetric quantile covariance can be defined as,

$$\text{qcov}^{**}_{\tau}(Z_1, Z_2) = \text{sign}(\beta_{2,1}(\tau))\sqrt{\text{qcov}^{*}_{\tau}(Z_1, Z_2)\,\text{qcov}^{*}_{\tau}(Z_2, Z_1)}. \tag{21}$$

4 Numerical simulation

This section shows the performance of the proposed fPQR methodology under different synthetic datasets. The three quantile covariances under study, proposed by [Li et al., 2015],

[Dodge and Whittaker, 2009] and [Choi and Shin, 2018] are compared here. Additionally, the algorithm is compared against PLS, taken as a benchmark model, and the partial robust adaptive modified maximum likelihood estimator (PRAMML), proposed by Acitas et al. [2020], which is a robust PLS alternative for univariate response models. In order to compare the quantile estimation provided by fPQR with the mean estimations from PLS and PRAMML, the quantile level of the fPQR is fixed at $\tau = 0.5$ (the median estimation). For each dataset $\mathbb{D}$, a partition into two disjoint subsets, $\mathbb{D}_{train}$ and $\mathbb{D}_{test}$ is considered. $\mathbb{D}_{train}$ is used for training the models, this is, solving the model equations. $\mathbb{D}_{test}$ is used for testing the models prediction accuracy. The following metrics are computed, where "#" denotes the cardinal of a set:

- $\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\|_2$: the euclidean distance between the estimated coefficients and the true coefficients;
- $\frac{1}{\#\mathbb{D}_{test}} \sum (\hat{y}_i - y_i)^2$: the mean squared error between the estimated response and the true response;
- The execution time of each algorithm measured in seconds.

Remark. These simulations compare the results of the fPQR algorithm with the results from PLS and PRAMML. For this reason, the quantile level is fixed at $\tau = 0.5$ and the metric considered is the mean squared error. However, when dealing with other quantile levels, the mean squared error is not a suitable metric, as it does not take into account the quantile being computed. In such scenarios the following quantile error metric can be used instead,

$$E_\tau = \frac{1}{\#\mathbb{D}_{test}} \sum_{(y_i, \boldsymbol{x_i}) \in \mathbb{D}_{test}} \rho_\tau(y_i - \boldsymbol{x}_i^t \hat{\boldsymbol{\beta}}). \tag{22}$$

4.1 Simulation 1

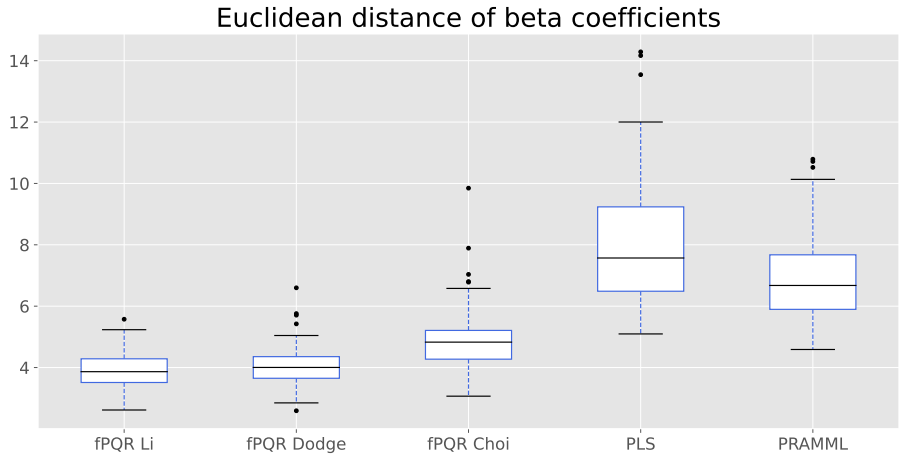
The following simulation scheme is an adaptation taken from Mendez-Civieta et al. [2020]. The idea behind this scheme is to simulate the behavior found in the increasingly common problem of sparse high dimensional data, where the number of variables is very large, and not all the variables affect the response, being some of them just noise. This problem can be found in many different areas of scientific research such as genetics [Boulesteix and Strimmer, 2006] or climate data [Chatterjee et al., 2011], and an interesting solution is the usage of dimension reduction techniques like PLS or the proposed fPQR algorithm. Take the model,

$$\boldsymbol{y} = X\boldsymbol{\beta} + \varepsilon, \tag{23}$$

Table 1: Simulation 1. Sparse high dimensional framework considering a $\chi^2(3)$ error.

	$\left\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right\|$	$\frac{1}{\#\mathbb{D}_{test}}\|\hat{\boldsymbol{y}} - \boldsymbol{y}\|_2^2$	Execution time
fPQR Li	3.88 (0.58)	21.59 (5.13)	0.038 (0.01)
fPQR Dodge	4.05 (0.62)	23.02 (5.98)	38.65 (1.649)
fPQR Choi	4.95 (0.94)	31.48 (11.40)	76.78 (2.716)
PLS	8.03 (2.03)	75.42 (37.21)	0.004 (0.001)
PRAMML	6.64 (1.37)	52.11 (20.66)	0.358 (0.047)

Figure 1: Simulation 1. Mean squared error of β coefficients.



where the predictors matrix X is generated from a standard normal distribution and the error term is generated following a chi squared distribution with 3 degrees of freedom, a distribution known to be heavy tailed and non symmetric. This will favor the usage of robust estimators. Since we are interested in the high dimensional framework, a sample size of $n = 100$ training observations and $m = 100$ predictive variables is considered. Out of the 100 predictive variables, 30 are generated from a standard uniform distribution and the remaining 70 have value 0, meaning that these 70 variables do not affect the response variable and are simply noise in the model. Although in real datasets the number of components in the model should be found based on some sort of cross-validation process, in this simulation it is fixed, taken equal to the number of significant variables, $h = 30$. Additionally, a sample of 500 observations is generated as test set. Observe that this fact does not affect the consideration of the simulation being high dimensional, as the algorithms are trained with a number of observations equal to the number of variables. This data generation process is repeated 100 times, and the results are summarized in terms of the mean value and standard deviation value (shown in parenthesis) of each metric computed.

Results from this simulation scheme are displayed in Table 1 and Figures 1, 2 and 3. In terms of the euclidean distance of the β coefficients, the best results are obtained by the

Figure 2: Simulation 1. Mean squared error of the response variable y.

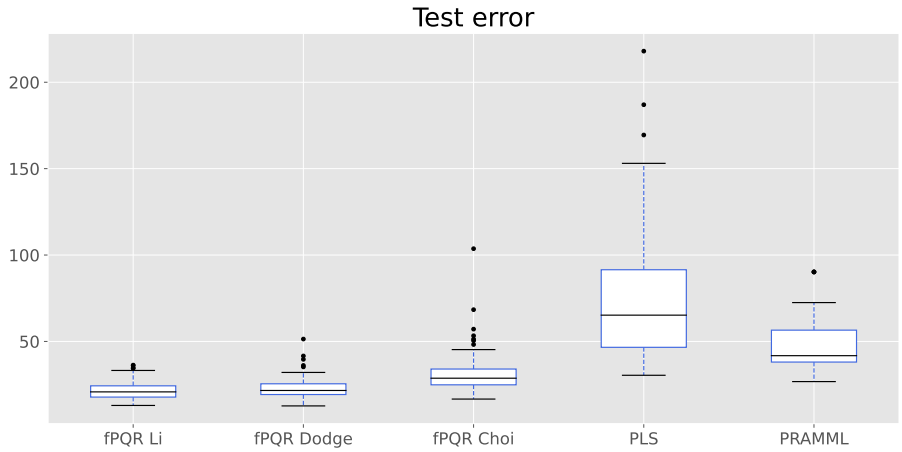
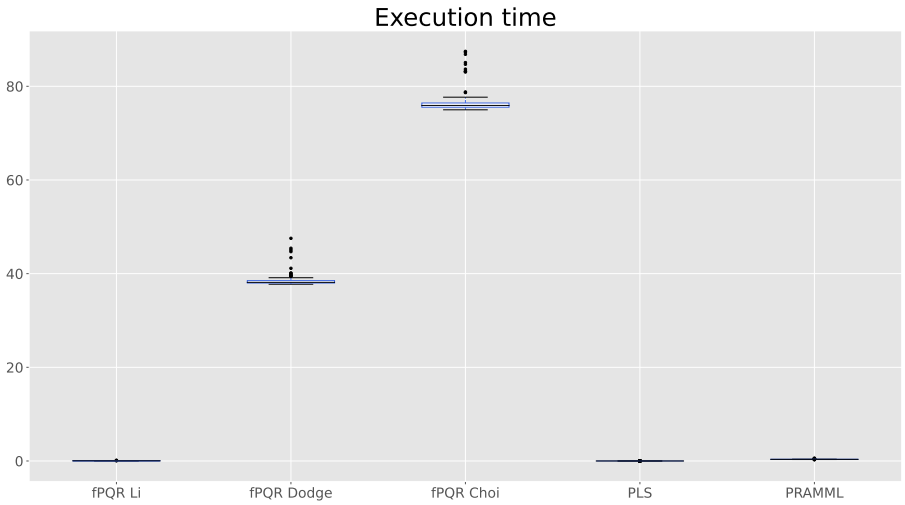


Figure 3: Simulation 1. Execution time measured in seconds.



fPQR Li estimator, followed by the other quantile based alternatives, while PLS obtains the worst results, as expected since the normality assumptions are not met. Observe also that the standard deviation of this metric is smallest in the fPQR Li, indicating more stable results. In terms of prediction accuracy, the best results are obtained also by the fPQR Li algorithm, closely followed by the fPQR Dodge and achieving again the smallest standard deviation values. Finally, regarding the execution time the fastest algorithm was PLS and the second fastest was fPQR Li, while PRAMML took on average 10 times longer than fPQR Li. One can also see the large execution times using fPQR Dodge or fPQR Choi alternatives. This is due to the way these covariances are computed, requiring to solve, at each iteration of the algorithm, $m = 100$ univariate quantile regression models in the case of Dodge metric, and $2m = 200$ models in the case of Choi metric, as it was discussed in Section 3.3.

4.2 Simulation 2

A second simulation is considered where we study the problem of having a multivariate response variable, very common in the field of chemometrics. Take,

$$Y = XB + \varepsilon, \tag{24}$$

where the predictors matrix X of size $n = 100$ and $m = 100$ is generated from a standard normal distribution, and the matrix of coefficients B has size $m = 100$ and $l = 3$. This defines a problem where the response matrix Y has $l = 3$ dimensions. Out of the 100 predictive variables, 30 are generated from a standard uniform distribution and the remaining 70 have value 0, and finally the error term is generated following a chi squared distribution with 3 degrees of freedom. In this simulation, the number of components obtained by the algorithms is taken equal to the number of significant variables, $h = 30$. Additionally, a sample of 500 observations is generated as test set and the simulation is repeated 100 times to ensure the stability of the results. Algorithms PLS and fPQR can deal directly with multivariate response matrices, but PRAMML solves only univariate models, for this reason in this simulation the predictions from PRAMML are obtained by solving $l = 3$ independent univariate models.

Results from this simulation scheme are displayed in Table 2. The best results, both in terms of the euclidean distance and prediction error, are achieved by the fPQR Li algorithm, closely followed by fPQR Dodge. The fPQR Li algorithm also displays the smallest standard deviations, meaning that the results are stable. The PRAMML estimator is outperformed here by all the other algorithms including PLS, probably due to the inability to directly solve multivariate problems, requiring to solve those in a univariate manner. In terms of execution time, the fastest algorithm is PLS, while fPQR Li is the second fastest running 10

Table 2: Simulation 2. Sparse high dimensional framework with multidimensional response, considering a $\chi^2(3)$ error.

	$\left\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right\|$	$\frac{1}{\#\mathbb{D}_{test}}\|\hat{\boldsymbol{y}} - \boldsymbol{y}\|_2^2$	Execution time
fPQR Li	5.16 (0.42)	15.14 (1.85)	0.10 (0.013)
fPQR Dodge	6.11 (0.48)	16.37 (2.18)	116.645 (2.139)
fPQR Choi	6.70 (0.51)	21.55 (3.89)	232.64 (7.088)
PLS	8.61 (1.01)	32.01 (6.55)	0.023 (0.004)
PRAMML	12.06 (1.38)	56.03 (11.74)	1.02 (0.063)

times faster than PRAMML. The fPQR Dodge and Choi algorithms are again the slowest.

4.3 Simulation 3

The last simulation considered takes the scheme from [Serneels et al., 2005] and [Acitas et al., 2020]. Consider the model,

$$\boldsymbol{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon} = TP^t\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \tag{25}$$

where $X = TP^t \in \mathbb{R}^{n\times m}$ is the predictor matrix, $T \in \mathbb{R}^{n\times h}$ is a scores matrix and $P \in \mathbb{R}^{m\times h}$ is a loadings matrix. T and P are generated based on a $N(0,1)$ distribution, and $\boldsymbol{\beta} \in \mathbb{R}^m$ is the vector of true coefficients, generated based on a normal distribution with mean 0 and standard deviation 0.001. Three possible error distributions are considered for $\boldsymbol{\varepsilon} \in \mathbb{R}^n$: a standard normal distribution, a t_1 distribution, which is symmetric as the normal distribution but with heavier tails, and a slash distribution (defined as a standard normal distribution divided by a standard uniform distribution), which is heavy tailed and non symmetric. The number of components in the model is fixed, equal to the dimension of the latent loadings h. This process is repeated 500 times. Two cases are defined based on changes in the number of training observations n, variables m and components h,

- A low dimensional example: $(n, m, h) = (100, 10, 2)$;
- A high dimensional example: $(n, m, h) = (15, 60, 4)$.

Results from this simulation are shown in Tables 3 and 4. In terms of the euclidean distance of the β coefficients, one can see that PRAMML estimator obtains the best results closely followed by fPQR Li and Dodge algorithms, being both competitive alternatives. It is worth remarking the fact that fPQR Li and Dodge outperformed PLS even when considering a normal distribution for the error term, where PLS is expected to excel. Finally, fPQR Choi consistently provides the worst results. The execution time is affected by the number of observations n, variables m and l, and components h, but not by the error distribution, for this reason Table 4 shows the execution time regardless of the error distribution. PLS

Table 3: Simulation 3. Euclidean distance of $\boldsymbol{\beta}$ coefficient estimations under different error distributions.

	$N(0,1)$	t_1	Slash
$(n,m,h) = (100,10,2)$			
fPQR Li	0.19 (0.13)	0.25 (0.15	0.37 (0.23)
fPQR Dodge	0.19 (0.13)	0.26 (0.16)	0.38 (0.24)
fPQR Choi	0.49 (1.37)	3.46 (55.95)	1.69 (5.70)
PLS	0.19 (0.10)	6.23 (22.57)	12.00 (58.51)
PRAMML	0.16 (0.10)	0.23 (0.14)	0.31 (0.19)
$(n,m,h) = (15,60,4)$			
fPQR Li	0.79 (0.33)	1.61 (1.25)	2.21 (1.45)
fPQR Dodge	0.90 (0.40)	1.84 (1.56)	2.49 (1.70)
fPQR Choi	6.74 (42.19)	18.78 (176.08)	28.25 (231.58)
PLS	1.14 (0.42)	14.94 (68.17)	29.55 (183.84)
PRAMML	0.61 (0.31)	1.02 (0.62)	1.42 (0.98)

Table 4: Simulation 3. Execution time

fPQR Li	fPQR Dodge	fPQR Choi	PLS	PRAMML
$(n,m,h) = (100,10,2)$				
0.015	0.27	0.54	0.0006	0.017
$(n,m,h) = (15,60,4)$				
0.017	2.99	5.94	0.0007	0.021

is the fastest algorithm, while fPQR Li is the second fastest closely followed by PRAMML. Results regarding prediction accuracy are not included in this simulation scheme because the error distributions considered generated outliers with very large values, providing predictions where the mean squared error values were very large and very similar regardless of the algorithm.

The three simulations displayed in this section remark the fact that, among the three quantile covariances under study, the best alternative for the fPQR algorithm is the quantile covariance proposed by Li et al. [2015], as it consistently provides the smallest prediction errors and the smallest euclidean distance of the β coefficients. Additionally, it is by far the fastest of the three algorithms, having a computation based on a traditional covariance rather than in solving univariate quantile regression models, as is the case with the other quantile covariances considered. Comparing the fPQR Li algorithm for the median with PLS shows that it outperformed PLS in all the scenarios considered in terms of prediction accuracy and euclidean distance of the β coefficients. When comparing it with robust PLS alternatives like PRAMML, it is worth remarking the fact that fPQR Li can be used to solve multidimensional response problems while PRAMML requires to face this situation by solving univariate models, as discussed in Section 4.2. Additionally, one can see that fPQR Li is a competitive alternative in terms of prediction accuracy and euclidean distance of the β coefficients, providing better estimations in two of the three simulations, and being competitive in the last one. In terms of execution time, fPQR Li also outperformed PRAMML in all the simulations. But the fPQR algorithm has an additional advantage when compared with any PLS based methodology: PLS based methodologies can only obtain estimations for the mean of the response matrix, while fPQR can obtain estimations for different quantile levels. This allows to study not only the central behavior of the response variable, but also the behavior at any other quantile of interest, like the tails of the distribution.

5 Real data analysis: Biscuit data

The biscuit data was first introduced in Osborne et al. [1984]. This dataset contains four response variables, concentration of fat, flour, sucrose and water, of 72 biscuit dough samples, where 40 observations usually define a training set and 32 a prediction set. In this analysis, and following the steps from [Hubert and Branden, 2003], the variable fat was removed because it showed small correlation coefficients with the other constituents and a larger variance. The rest of the response variables show larger correlations and similar variances, and for this, a multivariate analysis is considered. The objective is to predict the values of the three response variables based on NIR spectra measurements taken every 2 nm from 1200 up to 2400. The same preprocessing steps as in [Hubert et al., 2002] and [Hubert and Branden,

Figure 4: Biscuit dataset: NIR spectra of the biscuit dataset.

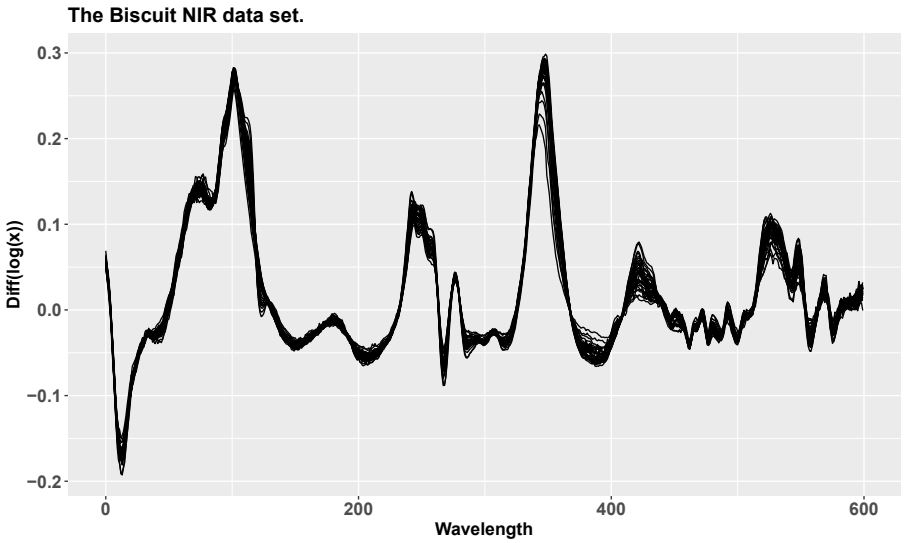


Table 5: Biscuit data: Test mean squared error.

fPQR Li	PLS	PRAMML
0.491	0.614	0.527

2003] were performed, obtaining a NIR spectra prediction matrix of $m = 600$ dimensions, shown in Figure 4, and a response matrix of $l = 3$ dimensions. Though observation 23 is known to be an outlier, it is kept in the dataset.

Using this dataset, a comparison of fPQR Li, PLS and PRAMML estimators is performed. The quantile level is taken as $\tau = 0.5$ so that quantile based results can be compared with the mean based results from PRAMML and PLS, and since the PRAMML estimator solves only univariate models, the predictions from this estimator are obtained by solving 3 independent univariate models. The first step is to select the number of components to be computed. This is done by performing 5-fold cross validation on the training set, and the objective is to minimize the mean squared error of the predictions. Figure 5 shows the CV results, concluding that three is the best number of components for any of the models considered.

The final models are built using the 40 observations from the training set and 3 components, and the mean squared error of the prediction of each model is computed on the test set. Table 5 shows the results. One can see that best result is obtained by fPQR Li, followed by the PRAMML estimator, and PLS obtains the worst result, presumably due to the presence of outliers in the dataset. An additional advantage of fPQR Li is that it can provide estimations for different quantile levels. Take for example observation 41, which is the first one in the test set. This observation has values flour= 16.44, sucrose= 47.65 and water= 12.57, and the median prediction obtained using fPQR Li for $\tau = 0.5$ is flour= 15.68,

Figure 5: Biscuit dataset: CV mean squared error on the number of components.

sucrose= 48.39 and water= 12.82. But one can also calculate an estimation of any other quantile of interest, obtaining this way prediction intervals. For example, the prediction for the 10% percentile of the response is flour= 15.24, sucrose= 47.67 and water= 12.39 for a small biscuit dough given the associated NIR spectra values, while the 90% percentile for a large biscuit dough has values flour= 17.22, sucrose= 48.41 and water= 13.10. The fPQR Li algorithm can thus provide a complete picture of the distribution of the response matrix.

6 Computational aspect

All the simulations and analysis commented in Sections 4 and 5 were run in a computer with an Intel Core i7-10750H CPU (2.6GHz) processor with 32GB RAM memory running the O.S. Windows 10. The computation of the fPQR has been developed in Python 3.8.5 (Anaconda Inc.). The quantile covariance metrics introduced in Section 3.3 required solving quantile regression models. Those were solved using the Python package ASGL, built on top of the CVXPY optimization framework for Python [Diamond and Boyd, 2016] and Mosek solver [ApS, 2021]. The PRAMML estimator was computed using the R package 'rpls' [Filzmoser et al., 2020], as there was no Python implementation for this methodology.

7 Conclusion

In this paper the fast partial quantile regression (fPQR) algorithm has been introduced. This algorithm extends the PLS models to the quantile regression framework. The result is

a dimensionality reduction technique that parallels the nice properties of PLS models but that is linked to the quantiles of the response matrix, being robust to the presence of outliers or heteroscedastic data. As discussed in Section 3, the key idea behind fPQR is the definition of the objective function that it maximizes in terms of a quantile covariance metric, and in this work different metrics are considered [Li et al., 2015], [Dodge and Whittaker, 2009], [Choi and Shin, 2018]. Section 4 studies the performance of the fPQR algorithm using the different quantile metrics in a set of synthetic datasets, concluding that the best results in terms of prediction accuracy, euclidean distance of the β coefficients and execution time are obtained using the quantile covariance defined by Li et al. [2015]. Additionally, the performance of the fPQR algorithm is compared with PLS and PRAMML [Acitas et al., 2020] estimators, showing that, if the median estimation is computed, fPQR is a competitive alternative to other robust PLS algorithms, but additionally, fPQR can obtain estimates for different quantile levels of the response matrix, providing a complete picture of its distribution. The performance of the proposed work is also studied in a real high dimensional dataset containing NIR spectra measurements, where fPQR Li obtains the best prediction accuracy.

8 Acknowledgments

This research was partially supported by research grants and projects PID2020-113961GB-I00 and PID2019-104901RB-I00 from Agencia Estatal de Investigación.

References

S. Acitas, P. Filzmoser, and B. Senoglu. A new partial robust adaptive modified maximum likelihood estimator. *Chemometrics and Intelligent Laboratory Systems*, 204:104068, 2020. ISSN 18733239. doi: 10.1016/j.chemolab.2020.104068. URL `https://doi.org/10.1016/j.chemolab.2020.104068`.

M. ApS. MOSEK Optimizer API for Python 9.3.6, 2021. URL `https://docs.mosek.com/9.3/pythonapi/index.html`.

A.-l. Boulesteix and K. Strimmer. Partial least squares : a versatile tool for the analysis of high-dimensional genomic data. *Briefings in Bioinformatics*, 8(1):32–44, 2006. doi: 10.1093/bib/bbl016.

S. Chatterjee, S. Banerjee, Arindam, and A. R. Ganguly. Sparse Group Lasso for Regression on Land Climate Variables. In *2011 IEEE 11th International Conference on Data Mining*

Workshops, pages 1–8. IEEE, 12 2011. ISBN 978-1-4673-0005-6. doi: 10.1109/ICDMW.2011.155.

J.-E. Choi and D. W. Shin. *Quantile correlation coefficient: a new tail dependence measure*. 2018. ISBN 8223277360. URL `http://arxiv.org/abs/1803.06200`.

S. de Jong. SIMPLS: An alternative approach to partial least squares regression. *Chemometrics and Intelligent Laboratory Systems*, 18(3):251–263, 3 1993. ISSN 0169-7439. doi: 10.1016/0169-7439(93)85002-X.

S. Diamond and S. Boyd. CVXPY: A Python-Embedded Modeling Language for Convex Optimization. *arXiv:1603.00943*, 3 2016.

Y. Dodge and J. Whittaker. Partial quantile regression. *Metrika*, 70:35–57, 2009. ISSN 00261335. doi: 10.1007/s00184-008-0177-4.

P. Filzmoser, S. Acitas, and B. Senoglu. rpls: Robust Partial Least Squares, 2020. URL `https://cran.r-project.org/package=rpls`.

M. Hubert and K. V. Branden. Robust methods for partial least squares regression. *Journal of Chemometrics*, 17(10):537–549, 10 2003. ISSN 0886-9383. doi: 10.1002/cem.822. URL `http://doi.wiley.com/10.1002/cem.822`.

M. Hubert, P. J. Rousseeuw, and S. Verboven. A fast method for robust principal components with applications to chemometrics. *Chemometrics and Intelligent Laboratory Systems*, 60 (1-2):101–111, 2002. ISSN 01697439. doi: 10.1016/S0169-7439(01)00188-5.

R. Koenker and G. Bassett. Regression Quantiles. *Econometrica*, 46(1):33–50, 1 1978. ISSN 00129682. doi: 10.2307/1913643.

G. Li, Y. Li, and C. L. Tsai. Quantile Correlations and Quantile Autoregressive Modeling. *Journal of the American Statistical Association*, 110(509):246–261, 2015. ISSN 1537274X. doi: 10.1080/01621459.2014.892007.

A. Mendez-Civieta, M. C. Aguilera-Morillo, and R. E. Lillo. Adaptive sparse group LASSO in quantile regression. *Advances in Data Analysis and Classification*, 2020. ISSN 18625355. doi: 10.1007/s11634-020-00413-8. URL `https://doi.org/10.1007/s11634-020-00413-8`.

D. V. Nguyen and D. M. Rocke. Tumor classification by partial least squares using microarray gene expression data. *Bioinformatics*, 18(1):39–50, 2002. ISSN 13674803. doi: 10.1093/bioinformatics/18.1.39.

B. G. Osborne, T. Fearn, A. R. Miller, and S. Douglas. Application of near infrared reflectance spectroscopy to the compositional analysis of biscuits and biscuit doughs. *Journal of the Science of Food and Agriculture*, 35(1):99–105, 1984. ISSN 10970010. doi: 10.1002/jsfa.2740350116.

S. Serneels, C. Croux, P. Filzmoser, and P. J. Van Espen. Partial robust M-regression. *Chemometrics and Intelligent Laboratory Systems*, 79(1-2):55–64, 2005. ISSN 01697439. doi: 10.1016/j.chemolab.2005.04.007.

H. Wold. Nonlinear Iterative Partial Least Squares (NIPALS) Modelling: Some Current Developments. In P. R. Krishnaiah, editor, *Multivariate Analysis?III*, pages 383–407. Academic Press, 1973. ISBN 978-0-12-426653-7.

S. Wold, M. Sjöström, and L. Eriksson. PLS-regression: A basic tool of chemometrics. *Chemometrics and Intelligent Laboratory Systems*, 58(2):109–130, 2001. ISSN 01697439. doi: 10.1016/S0169-7439(01)00155-1.

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