UNIVERSITÉ DU QUÉBEC À MONTRÉAL

LA RÉGRESSION ASYMÉTRIQUE QUANTILE ET EXPECTILE EN GRANDE DIMENSION ET LA SÉLECTION DE VARIABLES PAR GROUPES

THÈSE

PRÉSENTÉE

COMME EXIGENCE PARTIELLE

DU DOCTORAT EN MATHÉMATIQUES

PAR

MOHAMED OUHOURANE

AVRIL 2023

REMERCIEMENTS

La réalisation de ce mémoire n'aurait pas été possible sans l'aide précieuse de plusieurs personnes, à qui je voudrais exprimer ma plus sincère gratitude.

Je tiens tout d'abord à remercier chaleureusement mon directeur de recherche, le professeur Karim Oualkacha, pour sa patience, sa disponibilité et surtout ses précieux conseils, qui ont grandement contribué à nourrir ma réflexion.

Je désire également exprimer ma gratitude envers mon co-directeur, le professeur Yi Yang, ainsi que le professeur Abdellah Mkhadri, qui m'ont fourni les outils nécessaires à la réussite de mes études universitaires.

Un grand merci s'adresse également aux professeurs et au personnel administratif du département de mathématiques, en particulier à Madame Isabelle Couture pour son suivi attentif et la mise à jour de mon dossier administratif.

Je souhaite également exprimer toute ma reconnaissance envers les membres du jury qui ont eu l'honneur de lire ma thèse et de l'évaluer. Je leur suis reconnaissant pour leur temps précieux et leurs commentaires pertinents.

Je tiens à exprimer ma profonde gratitude envers les membres de ma famille qui m'ont soutenu tout au long de cette thèse, notamment mes enfants Maria et Yassir, mes oncles Youssef et Lhoussaine, ma mère Aicha, ma tante Meriem, ma sœur Mina, et autres.

Enfin, je tiens à témoigner toute ma reconnaissance envers mes amis et collègues, tels que Brahim, Hassan, Ait Oussouss, Roland, Brigitte, et bien d'autres, pour leur soutien moral et intellectuel inestimable tout au long de mon parcours.

Je voudrais également rendre hommage à la mémoire de mes grands-parents, qui sont malheureusement décédés pendant la période de préparation de ma thèse. Leur soutien indéfectible a toujours été une source de motivation pour moi. À toutes ces personnes, je tiens à exprimer mes plus sincères remerciements, ainsi que mon respect et ma gratitude.

TABLE DES MATIÈRES

TABLE DES FIGURES

LISTE DES TABLEAUX

RÉSUMÉ

L’émergence de plusieurs technologies modernes ont facilité la collecte de données de grande dimension dans plusieurs domaines de la science appliquée, entres autres, dans le domaine de la génétique, la finance et l’économie. De telles données sont sujets à l’hétérogénéité (e.g. sujets très différents, données collectées à partir de plusieurs plate-formes) et à la présence de plusieurs sources de bruits, ce qui rend leur analyse laborieuse. Ainsi, pendant les deux dernières décennies, nous avons assisté à un développement considérable d’outils/modèles statistiques afin d’analyser de telles données. Les modèles de la régression pénalisée ont obtenu une attention particulière dans ce contexte. En effet, la régression est un outil statistique qui a pour but d’analyser la relation entre une variable d’intérêt et des variables informatives via des modèles intuitifs, relativement simples et faciles à interpréter. Les modèles de la régression pénalisée régularisent les données à l’aide de paramètres additionnels introduis dans le modèle afin d’atténuer l’impact du bruit qui est omniprésent dans les données de grande dimension.

Le thème général de cette thèse focalise sur le développement de nouvelles approches dans le cadre de la régression pénalisée en présence des données de grande dimension. Plus précisément, la thèse se concentre sur l’extension des modèles de régression asymétrique, quantile et expectile, avec plusieurs pénalités de sélection de variables par groupe afin de sélectionner des groupes de variables importantes/informatives pour une variable d’intérêt. Nous avons proposé deux nouvelles approches dans ce contexte.

Premièrement, nous avons introduit la régression quantile régularisée avec la pénalité group-Lasso et les pénalités non convexes (group-SCAD et group-MCP) ainsi que leurs approximations locales. L’approche proposée permet de sélectionner les groupes de variables importantes et fournit une estimation de leurs effets sur la variable dépendante/d’intérêt simultanément. Nous avons démontré que le vitesse de convergence de notre approche avec la pénalité group-Lasso est linéaire.

Deuxièmement, nous avons généralisé les pénalités de sélection de variables par bloc aux modèles de la régression des moindres carrés asymétriques, à savoir la régression expectile et la régression expectile couplée. D’un point de vue théorique, nous avons démontré que nos modèles possèdent des propriétés oracles.

Pour les deux approches, nous avons mené des études de simulations exhaustives dans lesquelles les résultats ont montré que nos nouvelles approches ont une performance supérieure par rapport à d’autres méthodes existantes. Finalement, nous avons démontré l’utilité des deux approches en analysant des données réelles de grande dimension.

INTRODUCTION

Le modèle de la régression linéaire et ses extensions sont des méthodes qui cherchent à établir une relation entre une variable dite à expliquer ou dépendante, et une ou plusieurs variables dites explicatives ou indépendantes. Ces modèles sont utilisés dans divers domaines des sciences appliquées, comme la génétique, la biologie, l'économie et la médecine. De façon générale, ces modèles sont conçus pour examiner l'effet des variables explicatives sur le centre (la moyenne) de la distribution de la variable dépendante. Cette modélisation de la moyenne permet d'obtenir de belles propriétés sur l'association entre ces variables dépendante-indépendantes.

Dans de nombreuses situations on cherche à modéliser d'autres caractéristiques (autres positions que le centre) de la distribution de la variable à expliquer. On cherche en particulier à estimer l'effet des variables explicatives sur les queues de la distribution de la variable dépendante. En économie, par exemple, l'association entre le revenu d'un ménage (variable dépendante) et les dépenses pour les loisirs (variable indépendante) peut varier selon la classe sociale du ménage. Ainsi, un ménage pauvre consacre une part moins importante de son revenu pour les loisirs versus une famille riche. Le modèle de régression standard qui modélise le revenu moyen n'est pas adéquat pour capturer l'effet de la classe sociale sur la queue de la distribution de la variable revenu.

Afin d'avoir une explication plus appropriée et riche de la distribution de la variable dépendante, des méthodes de régression ont été proposées pour modéliser non seulement la moyenne, mais aussi d'autres caractéristiques ou localisations de la distribution. Par exemple, la Régression Expectile (RE) [73] modélise les expectiles de la distribution de la variable dépendante et la Régression Quantile (RQ) [52] modélise l'effet des variables explicatives à différents percentiles de la distribution de la variable dépendante. Ces deux méthodes génèrent une description globale et détaillée de l'effet des covariables (variables indépendantes) sur la variable dépendante. En plus, elles prennent en considération l'hétérogénéité dans les données.

Il existe plusieurs définitions de l'hétérogénéité dans la littérature statistique. Dans cette thèse, on s'intéresse particulièrement à l'hétérogénéité causée par les covariables. Dans le cadre des modèles de régression, il y a présence d'une telle hétérogénéité lorsque certaines covariables influencent la variable dépendante de façon non-homogène. Plus précisément, la présence d'hétérogénéité implique l'existence de covariables ayant un effet qui varie en fonction de la localisation sur la distribution conditionnelle de la variable dépendante. On parle dans ce cas d'un effet non-homogène/hétérogène ; on parle également d'un effet hétéroscédastique des covariables. En effet, la présence de ce type d'hétérogénéité révèle automatiquement la présence d'hétéroscédasticité. Cette dernière est présente, dans le cadre des modèles de régression, si la variabilité de la perturbation aléatoire est différente à travers les observations. La présence de l'hétéroscédasticité dans les données est une préoccupation majeure dans l'analyse de régression, car elle invalide l'inférence statistique qui suppose que les erreurs de modélisation ont toutes la même variance. Notons que l'existence de l'hétéroscédasticité peut être également le résultat d'une dépendance inter-sujets comme dans le cas des données familiales ou longitudinales. Elle peut être aussi le résultat de la présence d'observations très différentes lorsque les données sont collectées par plusieurs plateformes. En conséquence, l'hétéroscédasticité est un concept plus général que l'hétérogénéité que nous traitons dans la thèse. De façon générale, les méthodes RQ et RE sont des modèles flexibles et très utiles en présence d'hétéroscédasticité [52, 73].

En 1978, Koenker et Basset [52] ont introduit le concept RQ. Ils ont montré que l'estimateur issu de la méthode RQ est la solution d'un problème d'optimisation qui minimise la somme pondérée des déviations asymétriques en attribuant différents poids aux résidus positifs et négatifs. Ils ont résolu ce problème, qui n'est pas analytique, en le transformant sous forme d'un problème de programmation linéaire. L'algorithme du point intérieur (cf. [51]) ou l'algorithme de majoration-minimisation (MM) [41] sont deux autres techniques d'optimisation alternative pour résoudre ce problème. [52] ont aussi démontré certaines propriétés asymptotiques de l'estimateur RQ.

En présence d'hétérogénéité, l'approche RQ est fréquemment utilisée pour étudier toute la distribution conditionnelle de la variable dépendante en fonction des variables explicatives. La méthode RE ou encore la régression des moindres carrés asymétriques pondérés [1], sont des alternatives à l'approche RQ pour détecter et traiter l'hétérogénéité. Elle consiste à minimiser la somme pondérée des carrés des résidus par l'attribution des pondérations différentes aux résidus négatifs et positifs. Étant donné que la fonction de perte de la méthode RE est continûment différentiable, les estimateurs des paramètres de ce modèle sont faciles à obtenir comparativement à ceux de la RQ puisque le problème d'optimisation sous-jacent est facile à résoudre moyennant un algorithme des moindres carrés pondérés récursif. [73] ont étudié les propriétés asymptotiques de l'estimateur de l'approche RE.

De nos jours, la collecte de données est devenue de plus en plus facile. En particulier, les données d'association pangénomique GWAS (de l'anglais : Genome-Wide Association Study), les données issues des biopuces à ADN et les données biomédicales se distinguent principalement par un nombre de variables dépassant largement le nombre d'observations. La réduction du nombre de variables explicatives de ce type de données a été révolutionnée par l'introduction de la méthode Lasso (de l'anglais : Least absolute shrinkage and selection operator) [90] dans le cadre de la régression linéaire. Celle-ci permet de combiner simultanément la sélection et l'estimation des paramètres du modèle dans une procédure unique et simple par l'ajout d'une pénalité l_1 sur les paramètres du modèle. Dans certaines situations, il est important de sélectionner un groupe de variables dans la mesure où ces dernières sont pertinentes ensemble. Dans le cadre de la régression linéaire, la pénalité groupe Lasso a été introduite par [109] comme une extension de la pénalité Lasso pour répondre de façon appropriée à ce type de situation. Un certain nombre d'insuffisances de ces deux méthodes ont été soulevées par [27], [110] et [12]. Les deux méthodes tendent à introduire trop de variables (ou blocs de variables) explicatives dans le modèle optimal, ou elles exercent un sur-rétrécissement sur les coefficients significatifs. Ces auteurs ont proposé d'autres pénalités non convexes qui possèdent de belles propriétés asymptotiques, comme SCAD (de l'anglais : Smoothly Clipped Absolute Deviation)[27], MCP (de l'anglais : Minimax Concave Penalty)[110], groupe SCAD [10] et groupe MCP [10].

La plupart des méthodes statistiques en grande dimension supposent généralement l'homogénéité sur la structure des données. Or l'existence d'hétérogénéité dans les données de grande dimension a été largement démontrée. Par exemple, [21] ont identifié la présence d'hétéroscédasticité dans les données eQTLs (expression Quantitative Trait Loci) qui est généralement associée aux variations de l'expression des gènes et ont démontré la nécessité de considérer l'hétéroscédasticité dans les modèles statistiques. Le développement de nouveaux modèles afin de tenir compte de l'hétérogénéité en présence de données de grande dimension est devenu un sujet actif en recherche. [61] ont introduit la pénalité l_1 dans le cadre de la méthode RQ. [72] ont généralisé les pénalités non convexes pour l'approche RQ et ont aussi proposé un algorithme d'optimisation de descente par coordonnée pour résoudre la RQ pénalisée. [33] ont proposé les pénalités Lasso et non convexes pour l'approche RE. Ils ont trouvé également des bornes asymptotiques pour l'erreur d'estimation des estimateurs proposés.

Les approches RE et RQ sont des outils importants pour tenir compte de l'hétérogénéité, mais elles peuvent échouer à identifier les variables qui causent l'hétérogénéité. [33] ont aussi proposé la méthode de la régression expectile couplée, nommé COSALES (de l'anglais : COupled Sparse Asymmetric LEast Squares), pour sélectionner les variables importantes impliquées dans l'hétérogénéité. Les propriétés oracles ont été également démontrées pour l'estimateur COSALES.

Cette thèse s'inscrit dans le cadre des travaux récents portant sur la régression linéaire asymétrique, RQ et RE, en grande dimension en tenant compte de l'hétérogénéité due aux covariables. L'objectif de nos travaux de recherche est d'étendre les pénalités groupe Lasso, groupe SCAD et groupe MCP dans le cadre de la régression quantile, la régression expectile et la régression expectile couplée. L'amorce est constituée d'un chapitre préliminaire. Dans ce chapitre, nous essayons d'exposer en détails les méthodes de la régression standard, de la RQ et la RE en petite et en grande dimension. Les chapitres 2 et 3 sont constitués de deux articles scientifiques, en langue anglaise, qui contiennent d'importants résultats théoriques et empiriques sur la régression asymétrique en grande dimension. Ainsi, nous dotons les méthodes RQ, RE et RE couplée de la propriété de la sélection de variables par groupe. Cette propriété peut être très utile dans plusieurs domaines d'applications lorsqu'on suppose que les covariables peuvent agir en groupe pour expliquer la variable réponse et que l'on possède de l'information additionnelle, à priori, pour regrouper les covariables. La performance des trois nouvelles approches est illustrée par une série d'exemples simulés et l'analyse de données réelles. Du point de vue théorique, l'analyse de la convergence linéaire établie pour la RQ au chapitre 2 et des bandes non asymptotiques sont également fournies pour la RE et la RE couplée au chapitre 3. La contribution du chapitre 2 est publiée dans la revue Italienne de statistique ”Statistical Methods & Applications”. La contribution du chapitre 3 sera soumise très prochainement pour publication à la revue ”Electronic Journal of Statistics”. Dans le chapitre 4, nous présentons une illustration exhaustive de nos paquets en R afin d'initier l'utilisateur aux méthodes proposées dans cette thèse. Finalement, nous résumons notre travail et nous suggérons quelques perspectives de cette thèse.

CHAPITRE 1
CHAPITRE PRÉLIMINAIRE

Les modèles de régression consistent à modéliser/expliquer la relation entre une variable aléatoire à expliquer Y et une ou plusieurs variables explicatives x_1, x_2, ..., x_p, appelées aussi prédicteurs/covariables. Le modèle de la régression linéaire est un cas particulier de ces modèles. Il consiste à expliquer l'impact des variables explicatives sur la variable d'intérêt par une relation linéaire entre Y et les covariables $x_j, j = 1, \ldots, p$. Dans ce chapitre, nous allons présenter plusieurs modèles de la régression linéaire en petite et grande dimension.

1.1 La régression linéaire classique

Soit un échantillon de n observations $\{(y_i, x_{i1}, \ldots, x_{ip}), i = 1, \ldots, n\}$. Le modèle de la régression linéaire est de la forme suivante

$$y_i = \beta_1 x_{i1} + \ldots + \beta_p x_{ip} + \epsilon_i = \mathbf{x}_i^\top \boldsymbol{\beta} + \epsilon_i, \;\; i = 1 \ldots n, \tag{1.1}$$

où $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^\top$ est le vecteur des paramètres du modèle qu'on cherche à estimer, $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})^\top$, et ϵ est le terme d'erreur qui est non observé. Si l'ordonnée à l'origine (intercept ou constante) fait partie du modèle (1.1), on peut supposer que la variable $\mathbf{x}_1$ est égale à 1 pour tout i. Sous la forme matricielle, le modèle (1.1) s'écrit comme suit

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon},$$

où $\mathbf{y} = (y_1, \ldots, y_n)^\top$, $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_n)^\top$ et

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix} \equiv \begin{pmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_n^\top \end{pmatrix}. \tag{1.2}$$

Il existe plusieurs méthodes pour estimer $\boldsymbol{\beta}$, mais la plus utilisée est la méthode des moindres carrés ordinaires (MCO) ; elle consiste à minimiser la somme des carrés des résidus. Elle est équivalente à la méthode du maximum de vraisemblance si on suppose que la variable dépendante suit une loi normale de moyenne $\mu = \mathbf{x}^\top \boldsymbol{\beta}$ et de variance constante σ^2. L'estimateur MCO est défini par

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} (y_i - \mathbf{x}_i^\top \boldsymbol{\beta})^2 = \arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2. \tag{1.3}$$

Sous les hypothèses suivantes :

- La matrice $\mathbf{X}$ est de pleine rang ;
- $\mathbb{E}(\boldsymbol{\epsilon}) = \mathbf{0}_n$;
- Les erreurs ϵ_i sont indépendantes (non corrélées) ;
- var$(\boldsymbol{\epsilon}) = \sigma^2 \mathbf{I}$ (homoscédasticité) ;

la forme explicite de l'estimateur MCO, solution du problème (1.3), est donnée par

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}.$$

D'après le théorème de Gauss Markov, l'estimateur $\hat{\boldsymbol{\beta}}$, issu de la méthode MCO, est le meilleur estimateur linéaire sans biais de $\boldsymbol{\beta}$. Autrement dit, aucun autre estimateur linéaire sans biais de $\boldsymbol{\beta}$ n'a une variance plus petit que cet estimateur. Dans ce cas-ci, on dit que l'estimateur MCO est BLUE (de l'anglais, Best Linear Unbiased Estimator).

1.2 La régression robuste

En statistique, la régression robuste est une branche de la modélisation statistique, elle a été développée et introduite pour combler certaines limitations des modèles de la régression classique. En effet, bien que la méthode MCO soit largement utilisée puisqu'elle possède de belles propriétés statistiques si les hypothèses sous-jacentes sont satisfaites, elle peut conduire à des résultats erronés si l'une des hypothèses du théorème de Gauss Markov n'est pas vérifiée, car elle n'est pas robuste aux violations de telles hypothèses. En particulier, la méthode MCO est très sensible aux valeurs aberrantes. Une valeur aberrante est une observation qui diffère considérablement des autres observations. Elle est due souvent soit à une erreur de mesure, soit au fait que la population ait une distribution à queue lourde. Dans le premier cas, on souhaite identifier de telles observations aberrantes et les enlever de l'échantillon sous étude ou utiliser une méthode de régression robuste aux valeurs aberrantes. Tandis que dans le deuxième cas, une valeur aberrante peut être due à un fort aplatissement de la distribution (queue lourde). Dans ce cas, il faut être très prudent dans l'utilisation des modèles de régression linéaire standards. La régression robuste a été proposée pour remédier aux problèmes rencontrés par la régression classique. Par exemple, la régression LAD (de l'anglais, Least Absolute Deviation) est une alternative robuste à la méthode MCO. Elle est équivalente à la méthode du maximum de vraisemblance si les erreurs sont distribuées selon la loi de Laplace [60], au lieu de la loi normale dans le cas de la méthode MCO.

On considère le modèle (1.1) et on suppose que $\epsilon_1, \ldots, \epsilon_n$ sont i.i.d. et de médiane égale à zéro. L'estimateur de la régression LAD est défini alors par

$$\hat{\boldsymbol{\beta}}_M = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} |y_i - \mathbf{x}_i^\top \boldsymbol{\beta}|. \tag{1.4}$$

Contrairement à la méthode MCO où on a une solution explicite pour l'estimateur de β, la solution de l'équation (1.4) n'est pas explicite car la norme l_1 (c-à-d $|\mathbf{t}| = \sum_{i=1}^{p} |t_i|$) n'est pas dérivable en zéro. En conséquence, une solution numérique approximative s'impose. Une transformation de (1.4) en un problème de programmation linéaire permet de trouver la solution [8].

Les méthodes de régression MCO et LAD utilisent des poids symétriques sur les normes l_2 (c-à-d $\|\mathbf{t}\| = \sqrt{\sum_{i=1}^{p} t_i^2}$) et l_1 ; et elles modélisent la moyenne et la médiane de la variable dépendante conditionnelle aux variables indépendantes, respectivement. Cependant, ces méthodes ne reflètent pas une information exhaustive sur la distribution de la variable Y conditionnelle aux prédicteurs $\mathbf{x}$. Par exemple, on ne peut pas modéliser l'effet des covariables sur les petites ou les grandes valeurs de la variable Y. La régression quantile (RQ) et la régression expectile (RE) sont deux méthodes de régression qui utilisent des fonctions de perte asymétrique permettant une modélisation complète de la distribution conditionnelle de Y. Les méthodes de régression expectile et quantile sont des généralisations des méthodes qui modélisent la moyenne et la médiane (MCO et LAD), respectivement. Dans la section suivante, nous introduisons ces deux méthodes de la régression asymétrique.

1.3 La régression quantile et expectile

1.3.1 La régression quantile

On dit qu'un élève obtient une note au τ^e quantile d'un devoir si ce dernier a une meilleure note que $100\tau\%$ du groupe. En particulier, la moitié des élèves ont une note supérieure à celle de l'étudiant médian et le reste (la moitié) ont une note inférieure. Les quartiles, qui constituent des cas particuliers des quantiles, répartissent la population en quatre segments de proportions égales. Sous forme mathématique, soit Y une variable aléatoire de fonction de répartition $F_Y(y) := P(Y \leq y)$, alors le quantile d'ordre $\tau \in (0,1)$ de Y est défini par :

$$q_\tau(Y) = \inf_y \{y : F_Y(y) \geq \tau\}. \tag{1.5}$$

Dans les travaux de Koenker et Bassett [52], les auteurs ont formulé le quantile sous forme d'un problème d'optimisation. Ils ont pu démontrer que le quantile $q_\tau(Y)$ est la solution du problème de minimisation suivant :

$$\min_q \mathbb{E}\big(\rho_\tau(Y - q)\big), \tag{1.6}$$

où la fonction de perte $\rho_\tau(.)$ est donnée par :

$$\rho_\tau(u) = |\tau - \mathbb{1}_{(u \leq 0)}| \cdot |u|, \tag{1.7}$$

et $\mathbb{1}_A$ est la fonction indicatrice définie par

$$\mathbb{1}_A(x) = \begin{cases} 1, & \text{si } x \in A, \\ 0, & \text{si } x \notin A. \end{cases}$$

En effet, nous avons

$$\mathbb{E}\big(\rho_\tau(Y-q)\big) = (\tau-1)\int_{-\infty}^{q}(Y-q)dF_Y(y) + \tau\int_{q}^{+\infty}(Y-q)dF_Y(y).$$

D'après les conditions d'optimalité du premier ordre, le minimum q satisfait

$$\begin{aligned}
0 \quad &= (\tau-1)\Big([(y-q)f(y)]_{-\infty}^{q} + \textstyle\int_{-\infty}^{q} \frac{\partial(y-q)}{\partial y} dF_Y(y)\Big)\\
&= \tau\Big([(y-q)f(y)]_{q}^{-\infty} + \textstyle\int_{q}^{+\infty} \frac{\partial(y-q)}{\partial y} dF_Y(y)\Big)\\
&= (\tau-1)\Big(0 - F_Y(q)\Big) + \tau\Big(0 - (1 - F_Y(q))\Big)\\
&= F_Y(q) - \tau.
\end{aligned}$$

Si la fonction de répartition de la variable Y est absolument continue, la solution de l'équation $F_Y(q) = \tau$ est unique.

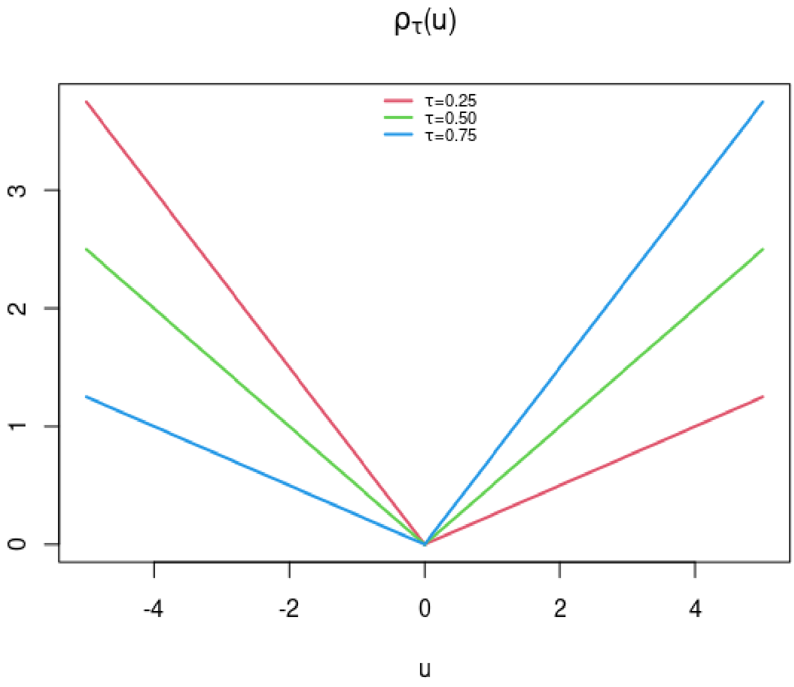


Figure 1.1 – La fonction de perte de quantile pour trois valeurs de $\tau \in \{0.25, 0.50, 0.75\}$.

La Figure (1.1) montre une représentation graphique de la fonction de perte $\rho_\tau(.)$ pour trois valeurs de τ en fonction de u. Pour $\tau = 0.5$, quelque soit la valeur de u, la fonction de perte accorde le même poids à u à gauche et à droite de zéro. Dans le cas $\tau = 0.75$, $\rho_\tau(.)$ accorde un poids élevé (0.75) pour les valeurs positives de u, et un poids moins élevé (0.25) pour les valeurs négatives de u. Inversement, si $\tau = 0.25$, $\rho_\tau(.)$ accorde un poids faible (0.25) pour les valeurs positives de u, et un poids élevé (0.75) pour les valeurs négatives de u.

Étant donné un échantillon empirique $\{y_1, y_2, ..., y_n\}$, la loi des grands nombres peut justifier l'estimation d'une espérance mathématique par la moyenne empirique. Ainsi, le quantile empirique $\hat{q}_\tau(Y)$ de niveau τ est donné par :

$$\hat{q}_\tau(Y) = \arg\min_q \Big(\frac{1}{n}\sum_{i=1}^{n} \rho_\tau(y_i - q)\Big). \tag{1.8}$$

Jusqu'à maintenant, nous avons supposé que la variable aléatoire Y ne dépend pas d'autres variables, et que les équations (1.6) et (1.8) définissent le quantile marginal de Y. En présence de variables explicatives $\mathbf{x}$ de Y, la définition (1.5) peut être généralisée au cas conditionnel [52]. La régression quantile généralise l'idée de la définition du quantile marginal à l'estimation des fonctions quantiles conditionnelles. En effet, l'idée consiste à trouver le quantile conditionnel d'une variable réponse Y en fonction de la valeur des covariables observées $\mathbf{x}$, c.-à-d., pour un quantile τ, le quantile conditionnel de Y sachant $\mathbf{x}$ est donné par :

$$q_\tau(Y|\mathbf{x})(\tau) = \inf_y\{y : F_{Y|\mathbf{x}}(y) \geq \tau\},$$

où $F_{Y|\mathbf{x}}$ est la fonction de répartition de Y sachant $\mathbf{x}$. Par analogie avec la définition du quantile non conditionnel (1.6), $q_\tau(Y|\mathbf{x})$ est la solution du problème de minimisation suivant :

$$\min_q \mathbb{E}\big(\rho_\tau(Y - q)|\mathbf{x}\big),$$

où $\rho_\tau(.)$ est donnée par (1.7).

En régression linéaire standard, on suppose qu'il existe une relation linéaire entre $\mathbb{E}(Y|\mathbf{x})$ et $\mathbf{x}$. Par analogie, en régression quantile linéaire, on suppose qu'il existe une relation linéaire entre $q_\tau(Y|\mathbf{x})$ et $\mathbf{x}$, c-à-d., $q_\tau(Y|\mathbf{x}) = \mathbf{x}^\top \boldsymbol{\beta}_\tau$.

À partir d'un échantillon i.i.d. $\{(y_i, x_{i1}, ..., x_{ip}), i = 1, ..., n\}$, l'estimation du paramètre $\boldsymbol{\beta} = (\beta_1, ..., \beta_p)^\top$ du modèle de régression quantile est obtenue en minimisant le critère suivant :

$$\sum_{i=1}^{n} \rho_\tau(y_i - \mathbf{x}_i^\top \boldsymbol{\beta}). \tag{1.9}$$

La fonction de perte $\rho_\tau(.)$ n'est pas différentiable en zéro, alors la solution du problème (1.9) n'a pas une forme explicite. Différents algorithmes ont été développés dans la littérature afin de résoudre le problème (1.9). Il s'agit, entre autres, de l'algorithme du simplexe [53], l'algorithme du point intérieur [79], et l'algorithme Majoration-Minimisation (MM) [41].

L'algorithme MM connait une popularité croissante dans le domaine de l'apprentissage statistique. En particulier, nous nous avons servi de cet algorithme combiné avec d'autres techniques d'optimisation pour résoudre nos problèmes de régressions asymétriques en grande dimension. Le principe de l'algorithme MM consiste à minimiser une fonction de perte difficile à minimiser en l'approximant par une autre fonction facile à minimiser. Plus précisément, soit $g(\theta)$ une fonction à minimiser que l'on suppose différentiable et bornée inférieurement. Étant donné la valeur de θ à l'itération i, θ^i, à l'itération $(i+1)$, l'algorithme MM cherche une fonction $f(\theta|\theta^i)$ qui majore $g(\theta)$, $\forall \theta$. De plus, au point θ^i les deux fonctions sont égales. Formellement, nous avons

$$f(\theta|\theta^i) \geq g(\theta) \quad \text{et} \quad f(\theta^i|\theta^i) = g(\theta^i).$$

Ainsi, on définit $\hat{\theta}$ à l'itération $i+1$ comme suit

$$\theta^{i+1} = \arg\min_{\theta \in \mathbb{R}} f(\theta|\theta^i).$$

Donc, on peut écrire

$$g(\theta^{i+1}) \leq f(\theta^{i+1}|\theta^i) \leq f(\theta^i|\theta^i) \leq g(\theta^i). \tag{1.10}$$

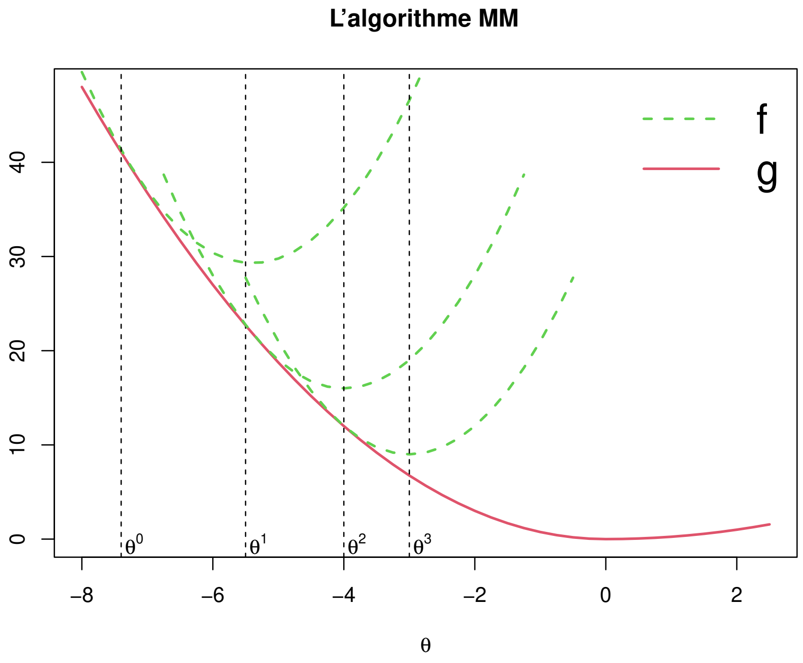


Figure 1.2 – Illustration du fonctionnement de l'algorithme MM pour la fonction $g(\theta)$ en rouge. Les courbes vertes en pointillés décrivent la fonction $f(.|\theta^i)$ à chaque itération de l'algorithme MM.

L'inégalité (1.10) montre que la fonction objective $g(\theta)$ décroît à chaque itération. Si elle est bornée inférieurement, la suite $g(\theta^i)$ convergera. La Figure 1.2 donne une illustration de l'algorithme MM.

Contrairement à la régression classique qui ne décrit que l'effet de $\mathbf{x}$ sur la moyenne de Y, la régression quantile vise à estimer la médiane conditionnelle ou d'autre quantile de la variable à expliquer, Y, étant donné les covariables. Une autre méthode alternative aux quantiles est la régression expectile. En effet, comme les quantiles, les expectiles décrivent aussi toute la distribution conditionnelle de Y sachant $\mathbf{x}$.

1.3.2 La régression expectile

Newey et Powell [73] ont proposé le concept de la régression expectile, qui est relativement moins étudié dans la littérature par rapport au quantile qui a connu un grand succès en pratique à cause de son interprétation facile et de sa robustesse. Avant de présenter ce modèle, nous allons introduire le concept de l'expectile marginal d'une variable aléatoire Y.

L'expectile de niveau $\tau \in (0,1)$ de la variable aléatoire Y est défini par :

$$\mathscr{E}_\tau = \arg\min_{\mathscr{E}} \mathbb{E}\big(\rho_\tau(Y - \mathscr{E})\big),$$

où la fonction de perte $\rho_\tau(.)$ est donnée par :

$$\rho_\tau(u) = |\tau - \mathbb{1}(u \leq 0)| \cdot u^2. \tag{1.11}$$

Si le quantile est une extension de la médiane qui utilise la fonction valeur absolue $|t|$ comme fonction de perte, alors l'expectile est une extension de la moyenne qui utilise la fonction de perte quadratique t^2.

En dérivant par rapport à $\mathscr{E}$, le minimum $\mathscr{E}_\tau$ satisfait :

$$\mathbb{E}\Big((Y - \mathscr{E}_\tau)\psi_\tau(Y - \mathscr{E}_\tau)\Big) = 0, \tag{1.12}$$

où $\psi_\tau(t) = |\tau - \mathbb{1}(t \leq 0)|$ est la fonction qui attribue le poids τ lorsque $Y > \mathscr{E}_\tau$ et $1-\tau$ lorsque $Y \leq \mathscr{E}_\tau$. Après simplification de l'équation (1.12), on obtient une définition plus pertinente de l'expectile [4] :

$$\mathscr{E}_\tau = \mathbb{E}\left(\frac{\psi_\tau(Y - \mathscr{E}_\tau)}{\mathbb{E}(\psi_\tau(Y - \mathscr{E}_\tau))} Y\right). \tag{1.13}$$

L'équation (1.13) montre que l'expectile est une moyenne pondérée, avec une pondération qui dépend de la distribution de la variable aléatoire Y. Si $\tau = 0.5$, la fonction $\psi_\tau(.)$ est constante et dans ce cas nous avons $\mathscr{E}_{0.5} = \mathbb{E}(Y)$.

Étant donné un échantillon observé $\{y_1, y_2, ..., y_n\}$, l'expectile empirique de niveau τ est la solution du problème d'optimisation suivant :

$$\mathscr{E}_\tau = \min_{\mathscr{E}} \frac{1}{n} \sum_{i=1}^{n} \rho_\tau(y_i - \mathscr{E}). \tag{1.14}$$

La solution du problème (1.14) est donnée par :

$$\mathscr{E}_\tau = \sum_{i=1}^{n} \frac{\psi_\tau(y_i - \mathscr{E}_\tau)}{\sum_{i=1}^{n} \psi_\tau(y_i - \mathscr{E}_\tau)} y_i.$$

La Figure 1.3 illustre la fonction de perte $\rho_\tau(.)$ pour trois valeurs de τ. L'effet de la fonction de perte de l'expectile est similaire à celle de quantile. De plus, la fonction de perte expectile est différentiable partout contrairement à la fonction de perte quantile.

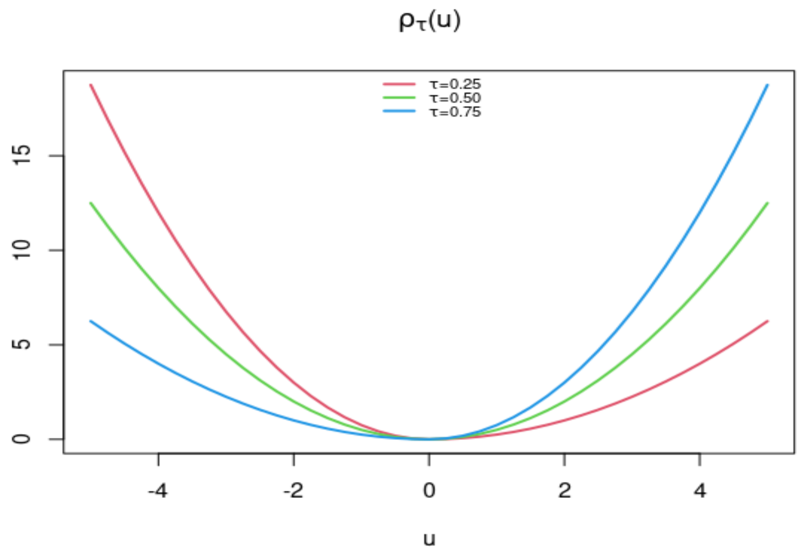


Figure 1.3 – La fonction de perte d'expectile pour trois valeurs de $\tau \in \{0.25, 0.50, 0.75\}$.

Par similarité au quantile conditionnel étudié dans la section précédente, l'expectile conditionnel est une extension naturelle de la moyenne conditionnelle. En présence d'une ou de plusieurs variables explicatives, pour $\tau \in (0,1)$ fixe, l'expectile conditionnel est défini par :

$$\mathscr{E}_\tau(Y|\mathbf{x}) = \arg\min_{\mathscr{E}} \mathbb{E}\big(\rho_\tau(Y - \mathscr{E}(\mathbf{x}))|\mathbf{x}\big). \tag{1.15}$$

L'hypothèse de linéarité entre Y et $\mathbf{x}$ permet de supposer que la fonction inconnue, $\mathscr{E}$, s'écrit comme suit :

$$\mathscr{E}(\mathbf{x}) = \mathbf{x}^\top \boldsymbol{\beta}_\tau.$$

Ainsi, le modèle de la régression expectile linéaire est défini comme suit :

$$y = \mathbf{x}^\top \boldsymbol{\beta} + \epsilon \text{ avec } \mathscr{E}_\tau(\epsilon|\mathbf{x}) = 0. \tag{1.16}$$

L'estimateur des paramètres du modèle (1.16) est la solution du problème de minimisation suivant

$$\hat{\boldsymbol{\beta}}_\tau = \arg\min_{\boldsymbol{\beta}} \frac{1}{n} \sum_{i=1}^{n} \rho_\tau(y_i - \mathbf{x}_i^\top \boldsymbol{\beta}). \tag{1.17}$$

Puisque la fonction de perte expectile $\rho_\tau(.)$ est continûment différentiable, alors les conditions du premier ordre appliquées à l'équation (1.17) donnent l'estimateur suivant :

$$\hat{\boldsymbol{\beta}}_\tau = \big(\sum_{i=1}^{n} \mathbf{x}_i^\top \psi_\tau(\hat{\epsilon}_i) \mathbf{x}_i \big)^{-1} \sum_{i=1}^{n} \mathbf{x}_i \psi_\tau(\hat{\epsilon}_i) y_i, \tag{1.18}$$

où $\hat{\epsilon}_i = y_i - \mathbf{x}_i^\top \hat{\boldsymbol{\beta}}_\tau$. La solution $\hat{\boldsymbol{\beta}}_\tau$ dépend des $\hat{\epsilon}_i$ qui dépendent à leur tour de $\hat{\boldsymbol{\beta}}_\tau$. Alors, pour calculer la valeur de l'estimateur (1.17), on utilise l'algorithme des moindres carrés pondérés récursif qui consiste en une estimation alternée entre $\hat{\boldsymbol{\beta}}_\tau$ dans (1.18) et $\hat{\epsilon}_i = y_i - \mathbf{x}_i^\top \hat{\boldsymbol{\beta}}_\tau$.

1.4 La régression linéaire en grande dimension

Dans de nombreux domaines d'analyse de données réelles, le nombre de variables p est souvent très élevé, voire même supérieur à la taille de l'échantillon n. En génétique par exemple, le nombre de sujets n dans les données d'association pangénomique GWAS (de l'anglais Genome-Wide Association Study) est relativement petit par rapport au nombre de SNPs p (de l'anglais Single-Nucleotide Polymorphism). Ce phénomène est dû au coût élevé de ce type d'expériences et à la rareté de certaines maladies génétiques. La méthode MCO ne permet pas d'estimer les paramètres du modèle (1.1) dans de tels cas car la matrice $\mathbf{X}$ n'est pas de plein rang. D'où la nécessité de réduire la dimension des données et de sélectionner parmi ces variables un sous-ensemble optimal de variables qui expliqueront bien la variable Y. Différents algorithmes qui combinent l'optimisation d'un critère et l'estimation du modèle par MCO sur un sous-ensemble sont disponibles, par exemple une sélection par élimination, "stepwise". Par contre, pour identifier un bon sous-ensemble de variables, nous devons tester 2^p modèles avec cette méthode, ce qui est presque impossible à réaliser pour $p > 15$. Récemment, des méthodes alternatives qui combinent la sélection et l'estimation simultanée de variables dans une seule procédure ont été proposées [90, 118]. Elles introduisent une pénalité sur les coefficients, ce qui a pour effet d'automatiser la sélection de variables en fixant certains coefficients à zéro et en rétrécissant les autres. Dans cette section, nous présentons quelques méthodes de pénalisation en lien direct avec les approches proposées dans cette thèse.

1.4.1 La méthode Lasso : Least Absolute Shrinkage and Selection Operator

L'approche Lasso [90] est la méthode la plus populaire dans le cadre de la régression pénalisée. Cette méthode a donné naissance à une nouvelle branche de la statistique, qui combine la théorie d'optimisation, statistique et informatique. Dans le cadre de la régression linéaire standard, cette approche minimise la somme des carrés résiduels en imposant une contrainte sur la norme l_1 des coefficients. L'estimateur de $\boldsymbol{\beta}$ de la méthode Lasso est la solution $\hat{\boldsymbol{\beta}}$ du problème de minimisation :

$$\begin{cases} \min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \left(y_i - \mathbf{x}_i^\top \boldsymbol{\beta}\right)^2, \\ \text{sous contrainte } \sum_{i=1}^{p} |\beta_i| \leq s. \end{cases} \tag{1.19}$$

Le problème Lasso a été résolu premièrement par des techniques de programmation quadratique sur une grille de valeur du paramètre s. La solution de départ $(s = 0)$ correspond à la solution nulle et la valeur finale correspond à la solution des moindres carrés ordinaires, où la contrainte $\sum_{i=1}^{p} |\beta_i| \leq s$ est inutile, pour une grande valeur de s.

Il est connu que la pénalité l_1 a pour effet de rétrécir continûment la solution et elle fait tendre ces coefficients vers 0 et oblige certains coefficients à être nuls. Ce rétrécissement améliore souvent la prévision du modèle par la réduction de la variabilité des coefficients estimés. En contrepartie, il introduit un léger biais d'estimation. Dans le contexte des données de grande dimension, puisque nous cherchons en général un estimateur qui minimise le risque quadratique moyen (la somme de la variance et le carré du biais). Ainsi, nous pouvons préconiser un estimateur légèrement biaisé avec une réduction importante de la variance. L'estimateur Lasso possède également la propriété de la parcimonie qui le distingue par rapport à d'autres estimateurs biaisés comme la régression Ridge qui introduit une pénalité de type l_2 sur $\boldsymbol{\beta}$ dans l'équation (1.19).

La Figure 1.4 illustre le fonctionnement des méthodes Lasso et Ridge. L'ellipse (en bleu) représente le critère quadratique $(\boldsymbol{\beta}-\boldsymbol{\beta}_{MCO})^\top(\mathbf{X}^\top\mathbf{X})(\boldsymbol{\beta}-\boldsymbol{\beta}_{MCO})$. Ce critère est équivalent au critère de MCO à une constante près. Les régions sous forme de losange (en vert) et de cercle (en noir) correspondent aux zones admissibles $|\beta_1| + |\beta_2| < s_i$ et $\sqrt{\beta_1^2 + \beta_2^2} < s_i$, i=1,2, respectivement. La parcimonie de la solution Lasso dépend de s. En effet, pour $s = s_1$, $\hat{\beta}_1^{Lasso}$ est nul et $\hat{\beta}_2^{Lasso}$ est non nul. Par contre, les deux coefficients ne sont pas nuls pour la solution Ridge ; pour $s = s_2$ les pénalités Lasso et Ridge ne sont pas utiles et la solution optimale est la solution de la méthode MCO.

Le problème (1.19) est convexe, alors on peut écrire aussi sa version lagrangienne

$$\min_{\beta_0,\boldsymbol{\beta}} \sum_{i=1}^{n} \left(y_i - \mathbf{x}_i^\top \boldsymbol{\beta}\right)^2 + \lambda \sum_{i=1}^{p} |\beta_j|, \ \lambda \geq 0, \tag{1.20}$$

où $\lambda \geq 0$ est un paramètre de régularisation. Les deux versions sont équivalentes et pour chaque λ fixe, il existe un s unique qui donne le même estimation et vice-versa [90].

Bien que la méthode Lasso soit intéressante pour la sélection des variables, lorsque le nombre de variables est grand, cette méthode a tendance à sélectionner de nombreuses variables non significatives afin de maintenir un biais relativement petit sur les coefficients des variables pertinentes. Autrement, moins de variables non significatives seront sélectionnées ; ce qui induit un biais (ou rétrécissement) important sur les variables pertinentes. Notons également qu'en pratique, lorsqu'on utilise la validation croisée pour choisir λ, la méthode Lasso a tendance à inclure beaucoup de variables bruits. Cet inconvénient est dû à l'utilisation du même paramètre de régularisation λ pour l'estimation et la sélection des variables simultanément [69, 80]. D'où la nécessité d'introduire d'autres méthodes avec plus de paramètres d'ajustement de rétrécissement comme, adaptative Lasso, SCAD (de l'anglais Smoothly Clipped Absolute Deviation), MCP (de l'anglais Minimax Concave Penalty) afin de stabiliser la procédure de sélection des variables.

1.4.2 La méthode adaptative Lasso

C'est une version pondérée de la méthode Lasso qui a été proposée par [117] ; elle consiste à trouver $\boldsymbol{\beta}$ qui est la solution du problème de minimisation :

$$\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} (y_i - \mathbf{x}_i^\top \boldsymbol{\beta})^2 + \lambda \sum_{j=1}^{p} \hat{w}_j |\beta_j|, \ \lambda \geq 0, \tag{1.21}$$

où $\hat{\mathbf{w}} = (\hat{w}_1, \hat{w}_2, ..., \hat{w}_p)^\top$ est un vecteur de poids estimé par un estimateur initial, qui pénalise différemment les coefficients, et λ est un paramètre de régularisation. Cette pondération permet de réduire le biais sur les variables importantes d'une part ; d'autre part, de garantir la consistance de l'estimateur Lasso en sélection de variables.

La pénalité associée au coefficient β_j est donc $\hat{w}_j\lambda$. Elle est d'autant plus forte lorsque $\hat{w}_j$ a une valeur forte, ce qui conduit la méthode Lasso à annuler plus facilement ce coefficient si nous prenons une forte valeur de $\hat{w}_j$, et vice versa. Pour l'estimateur initial de $\hat{\mathbf{w}}$, le choix usuel de $\hat{w}_j$ est l'inverse de l'estimateur du MCO de β_j s'il est existe (si $p > n$, $\hat{\beta}_{MCO}$ n'existe pas), ou l'inverse de l'estimateur Ridge $\big($c-à-d $\hat{w}_j = (\hat{\beta}_{MCO})_j^{-1}$ ou $\hat{w}_j = (\hat{\beta}_{Ridge})_j^{-1}\big)$.

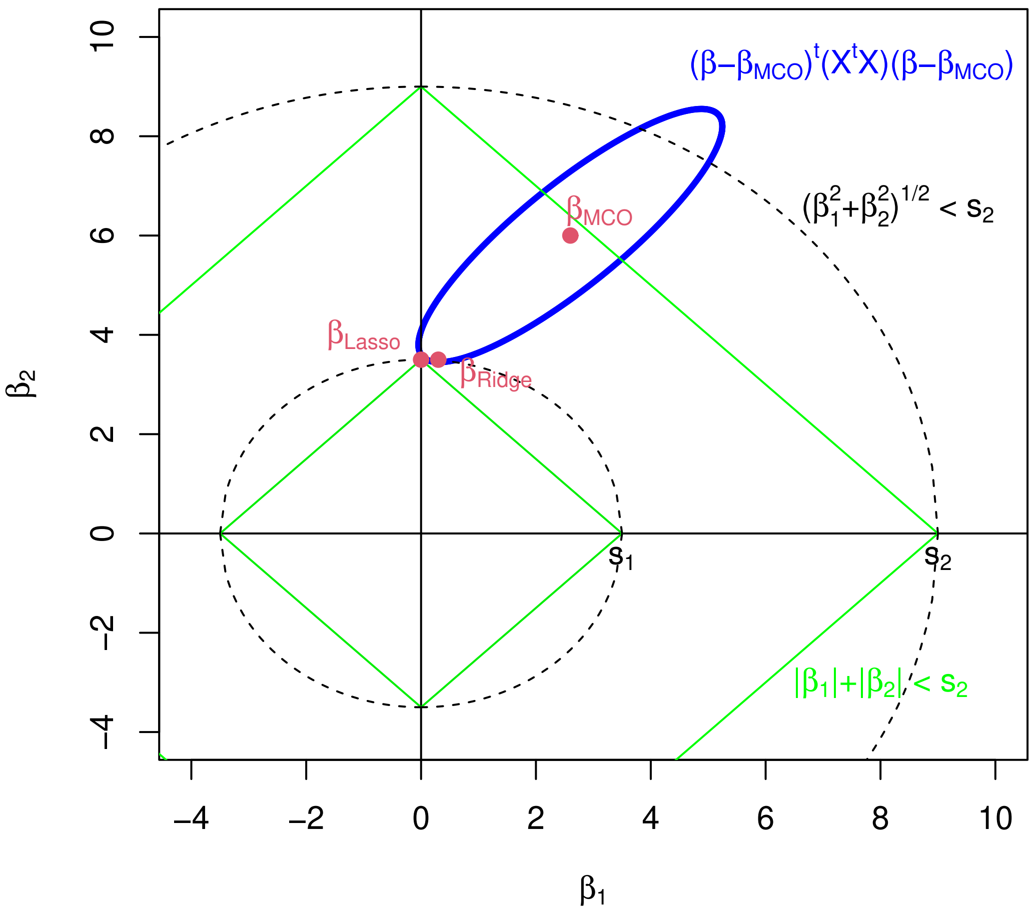


Figure 1.4 – L'ellipse en bleu représente le critère quadratique $(\boldsymbol{\beta}-\boldsymbol{\beta}_{MCO})^\top(\mathbf{X}^\top\mathbf{X})(\boldsymbol{\beta}-\boldsymbol{\beta}_{MCO})$, les zones en forme de carrés et de cercles sont des régions admissibles $|\beta_1|+|\beta_2| \leq s_i$ et $\beta_1^2+\beta_2^2 \leq s_i$ si $i = 1, 2$, respectivement.

1.4.3 Les méthodes SCAD et MCP

Les méthodes SCAD [27] et MCP [110] consistent à remplacer la pénalité l_1 dans le problème (1.20) par les pénalités

$$P_{\lambda,\theta}(t) = \begin{cases} \lambda w_k t \mathbb{1}_{(t \leq \lambda)}, \\ w_k \dfrac{\theta\lambda t - (t^2 + \lambda^2)/2}{\theta - 1} \mathbb{1}_{(\lambda < t \leq \theta\lambda)}, \\ w_k \dfrac{\lambda^2(\theta^2 - 1)}{2(\theta - 1)} \mathbb{1}_{(t > \theta\lambda)}, \end{cases} \tag{1.22}$$

$$P_{\lambda,\theta}(t) = \begin{cases} w_k(\lambda t - \dfrac{t^2}{2\theta}) \mathbb{1}_{(t \leq \theta\lambda)}, \\ w_k \frac{1}{2}\lambda^2\theta \mathbb{1}_{(t > \theta\lambda)}, \end{cases} \tag{1.23}$$

respectivement, avec $\theta > 2$ pour SCAD et $\theta > 1$ pour MCP. Ainsi, les deux méthodes sont la solution du problème d'optimisation suivant :

$$\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} (y_i - \mathbf{x}_i^\top \boldsymbol{\beta})^2 + \sum_{i=1}^{p} P_{\lambda,\theta}(|\beta_j|),\ \lambda \geq 0. \tag{1.24}$$

Les méthodes SCAD et MCP pénalisent les petits coefficients sévèrement comme Lasso, et pénalisent moins les grands coefficients ; ce qui réduit le biais ainsi que le nombre de variables non significatives sélectionnées. La Figure 1.5 illustre le fonctionnement des deux pénalités comparées à la pénalité Lasso. Nous notons que la pénalité $P_{\lambda,\theta}(t)$ SCAD coïncide avec la pénalité de Lasso jusqu'à $|t| = \lambda$, ensuite, $P_{\lambda,\theta}(t)$ fait une transition lisse vers une forme quadratique jusqu'à $|t| = \theta\lambda$, après, elle reste constante pour tout $|t| > \theta\lambda$. Pour bien clarifier le fonctionnement de la pénalité SCAD et sa différence par rapport à la pénalité Lasso, nous considérons le gradient des ces deux pénalités, celui-ci représente le degré de rétrécissement exercé par la pénalité. Pour des petites valeurs de t (c-à-d $|t| \leq \lambda$), les pénalités Lasso et SCAD exerce le même degré de rétrécissement λ sur t (c-à-d $P'_{\lambda,\theta}(t) = \lambda$). Cependant, lorsque $\lambda \leq t \leq \theta\lambda$, la pénalité SCAD réduit continuellement le degré de rétrécissement ; par contre, le rétrécissement de la pénalité Lasso est toujours égal à λ. Le rétrécissement de la pénalité SCAD devient nul lorsque $t \geq \theta\lambda$. Le même raisonnement peut être appliqué quand nous comparons les pénalités Lasso et MCP.

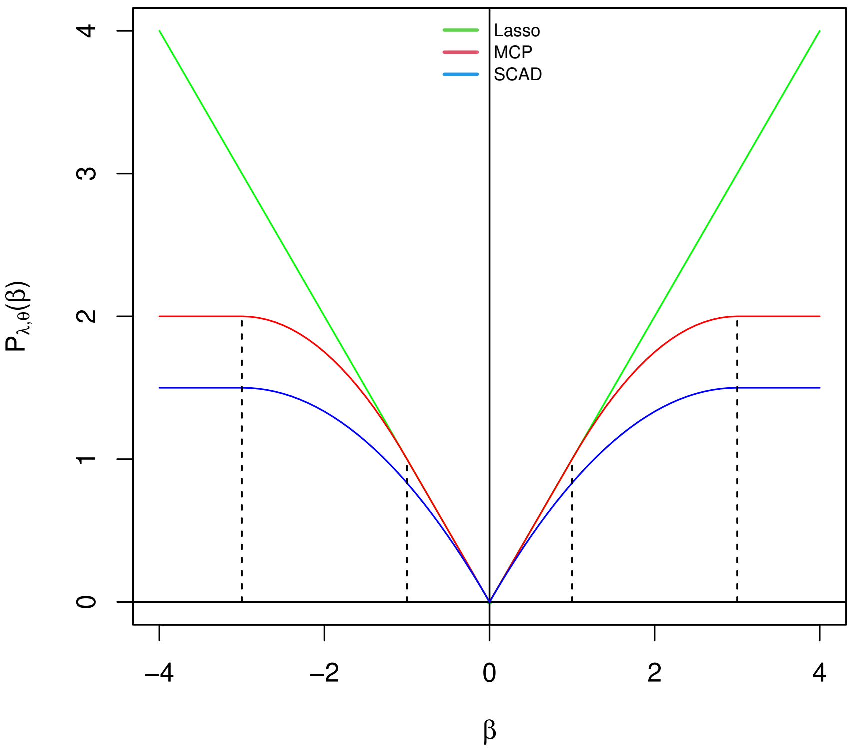


Figure 1.5 – Les pénalités Lasso, SCAD et MCP pour $\lambda = 1$ et $\theta = 3$

1.4.4 La méthode group Lasso

la méthode Lasso a été introduite pour faire la sélection individuelle des variables. Cependant, la sélection par groupe de variables est une caractéristique très désirée dans certaines applications. Par exemple, dans le cadre de l'analyse de la variance (ANOVA) à plusieurs facteurs, chaque facteur peut être exprimé sous forme d'un groupe de variables binaires. La pénalité group Lasso [109] est l'extension de la pénalité Lasso pour la sélection de groupe de variables connu à priori. L'idée est d'avoir une pénalité fournissant une sélection parcimonieuse de groupes (fournis a priori) et non de variables.

Nous supposons que les p variables sont réparties en K groupes qui ne se chevauchent pas $\big(\{1,\ldots,p\} = \cup_{k=1}^{K} I_k \; and \; I_k \cap I_{k'} \; for \; k \neq k'\big)$. Nous notons par $\mathbf{X}_k$ et $\boldsymbol{\beta}_k$ la matrice et le vecteur correspondent aux variables du groupe k et p_k est la taille du groupe k. Ainsi, la méthode group Lasso est la solution du problème d'optimisation :

$$\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} (y_i - \mathbf{x}_i^\top \boldsymbol{\beta})^2 + \lambda \sum_{k=1}^{K} w_k \|\boldsymbol{\beta}_k\|_2, \; \lambda \geq 0, \tag{1.25}$$

où $w_k \geq 0$ est un poids associé au groupe I_k. En pratique, nous prenons $w_k = \sqrt{p_k}$ pour ajuster le degré de pénalisation selon la taille du groupe k. Sinon, les groupes avec un grand nombre de variables ont tendance à être sélectionnés en premier. Si l'ordonnée à l'origine fait parti du modèle, celui-ci sera non pénalisé, alors nous prenons $w_1 = 0$. Si chaque groupe k contient une seule variable, la pénalité group Lasso se réduit à la pénalité Lasso.

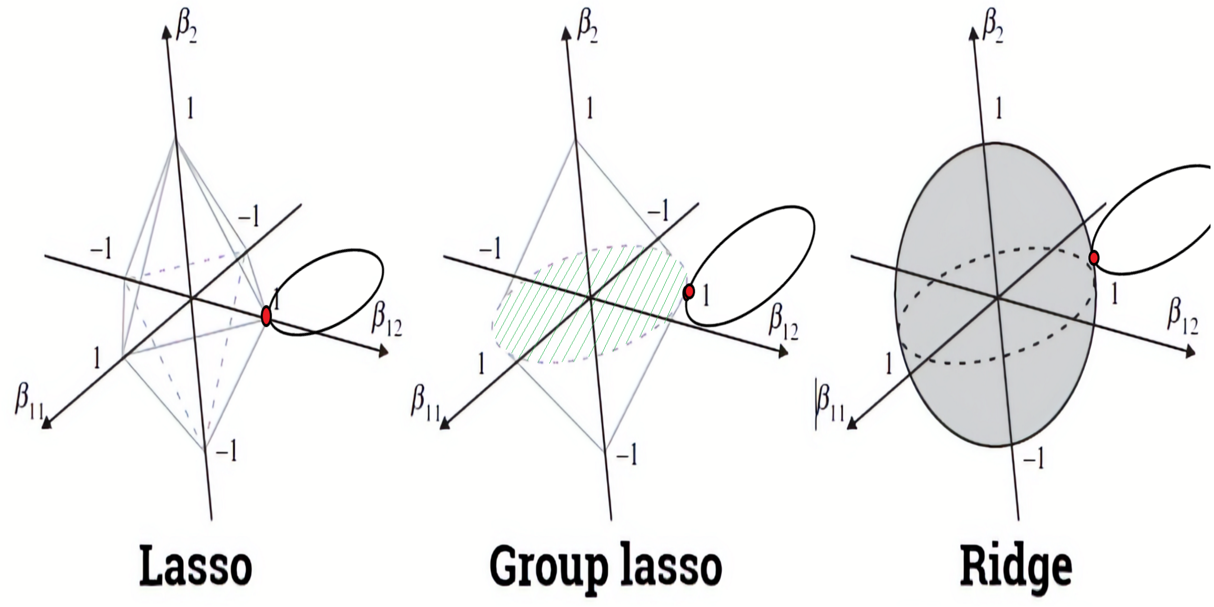


Figure 1.6 – Les pénalités Lasso, Group Lasso et Ridge pour $\lambda = 1$.

La Figure 1.6 représente les pénalités Lasso, group Lasso et Ridge en trois dimensions et l'ellipse du critère de MCO. Pour la méthode group Lasso, les deux premières variables forment le premier groupe $I_1 = \{1, 2\}$, et la troisième variable forme le deuxième groupe $I_2 = \{3\}$. Comme on peut le voir, la forme géométrique du Lasso (losange) ne permet pas de sélectionner tout le groupe I_1, il y a seulement la composante β_{12} qui n'est pas nulle. Parcontre, la forme géométrique du group Lasso (cercle hachuré en vert) permet de sélectionner l'ensemble du groupe I_1. La pénalité Ridge, sous forme d'une sphère, n'annule aucune variable. Notons que cette figure est une version modifiée d'une figure prise du site "https ://towardsdatascience.com/sparse-group-lasso-in-python-255e379ab892".

Group Lasso est une méthode intéressante pour la sélection de variables par groupe. Mais, comme la méthode Lasso, elle possède les mêmes limitations (voir section 1.4.1). Ainsi, les méthodes group SCAD et group MCP ont été introduites pour combler les limitations de group Lasso.

1.4.5 Les méthodes group SCAD et group MCP

Pour doter les méthodes SCAD et MCP de la propriété de sélection de variables par bloc, [10] ont proposé les méthodes group SCAD et group MCP. Elles sont capables de produire un modèle parcimonieux par groupe tout en réduisant le problème du rétrécissement élevé.

Les méthodes group SCAD et group MCP sont la solution du problème d'optimisation suivant :

$$\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} (y_i - \mathbf{x}_i^\top \boldsymbol{\beta})^2 + \sum_{k=1}^{K} P_{\lambda,\theta}(\|\boldsymbol{\beta}_k\|_2), \tag{1.26}$$

où $P_{\lambda,\theta}(.)$ est donnée par l'équation (1.22) pour group SCAD et l'équation (1.23) pour group MCP.

1.4.6 Algorithmes de résolution

Les problèmes (1.20), (1.21) et (1.25) dépendent d'un seul paramètre de lissage λ ; tandis que les problèmes (1.24) et (1.26) dépendent de deux paramètres (λ, θ). En pratique, le paramètre θ est fixé par [10] à $\theta = 3$ pour la pénalité (1.23) et $\theta = 4$ pour la pénalité (1.22). Pour le paramètre $\lambda \geq 0$, on choisi une grille de valeurs, allant d'une valeur maximale $\lambda_{\max}$ pour laquelle tous les coefficients pénalisés sont nuls, jusqu'à $\lambda = 0$ ou une valeur minimale $\lambda_{\min}$ pour laquelle le poids de la pénalité devient négligeable et l'estimateur $\boldsymbol{\beta}$ correspond à l'estimateur MCO si le modèle est identifiable, c-à-d., le nombre de variables sélectionnées est inférieur à n.

Résoudre le problème Lasso a fait l'objet de plusieurs travaux de recherche. Efron *et al.* [24] ont proposé l'algorithme Lars (Least-angle regression) qui permet de calculer efficacement le chemin de solution du Lasso pour $\lambda \in [\lambda_{\min}, \lambda_{\max}]$. Le temps de calcul de cet algorithme est équivalent à celui de la méthode MCO. L'algorithme Lars est aussi utilisé pour résoudre une variété de problèmes similaires au problème Lasso comme : adaptative Lasso [117], Elastic net [119] et relaxed Lasso [69]. L'algorithme de descente par coordonnée CDA (de l'anglais Coordinate Descent Algorithm ; [32]), ou la minimisation alternée en théorie d'optimisation, a connu un grand succès pour trouver la solution optimale du problème d'optimisation Lasso ainsi qu'une vaste variété de méthodes d'apprentissage statistique en grande dimension comme les séparateurs à vaste marge SVM (Support Vector Machine)[104], la régression logistique [32], la régression quantile [72, 33], etc. Il consiste à optimiser chaque paramètre ou bloc de paramètres séparément, tout en fixant les autres, et répéter la procédure jusqu'à convergence. Le fonctionnement de cet algorithme dans le cadre de la régression quantile et expectile sera exposé en détail dans les chapitres 2 et 3.

1.5 La régression quantile et expectile en grande dimension

Les modèles de la régression quantile et de la régression expectile pénalisés peuvent être écrit sous la forme de problème d'optimisation suivant :

$$\arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \rho_\tau(y_i - \mathbf{x}_i^\top \boldsymbol{\beta}) + \sum_{i=1}^{p} P_{\lambda,\theta}(|\beta_i|), \tag{1.27}$$

où $\rho_\tau(.)$ est la fonction de perte quantile/expectile donnée par (1.7) ou (1.11), respectivement. $\lambda \geq 0$ et θ sont des paramètres de régularisation. $P_{\lambda,\theta}(.)$ est la pénalité Lasso ou l'une des pénalités (1.22) et (1.23). Dans le cas du modèle de la régression quantile, la solution de (1.27) n'est pas analytique, car la fonction ρ_τ dans (1.7) et la pénalité l_1 ne sont pas différentiable partout. Ce modèle peut être écrit sous forme d'un problème d'optimisation linéaire. [61] ont proposé un algorithme qui calcule la solution pour la pénalité Lasso. [85] a proposé un algorithme pour la pénalité de type $l_1 + l_2$. Récemment, [72] ont proposé un algorithme alternatif et efficace pour une variété de pénalités, incluant Lasso, adaptive Lasso, MCP, SCAD. Il consiste à approximer la fonction de perte (1.7) par une fonction lisse et différentiable en zéro, qui peut être majorée localement par une forme quadratique. Ce qui permet d'utiliser l'algorithme CDA pour résoudre efficacement le problème de la régression quantile pénalisée.

Le modèle de la régression expectile permet de capturer l'hétérogénéité des variables indépendantes, sans toutefois préciser si l'effet de ces variables est sur la moyenne ou sur la variance de la variable dépendante. La régression expectile couplée [23] est une alternative de la régression expectile qui permet de séparer l'effet des variables hétérogènes. Elle est définie par

$$\arg\min_{\boldsymbol{\beta},\boldsymbol{\phi}} \sum_{i=1}^{n} \rho_{0.5}(y_i - \mathbf{x}_i^\top \boldsymbol{\beta}) + \rho_\tau(y_i - \mathbf{x}_i^\top \boldsymbol{\beta} - \mathbf{x}_i^\top \boldsymbol{\phi}) \tag{1.28}$$

où $\rho_\tau(.)$ est la fonction de perte expectile donnée par (1.11). [33] ont introduit une version pénalisée du modèle (1.28).

Les deux fonctions de perte, expectile et expectile couplée, sont continûment différentiable, mais elles ne sont pas deux fois différentiable en zéro. Pour résoudre ces deux problèmes, [33] ont proposé un algorithme efficace qui consiste à combiner l'algorithme CDA et l'algorithme MM.

La sélection de variables dans le cadre de la régression en grande dimension est en plein expansion. En particulier, les méthodes de la régression asymétrique pénalisées permettent d'obtenir des modèles parcimonieux et de capturer l'hétérogénéité des variables indépendantes en sélectionnant des ensembles de variables qui peuvent différer d'un quantile à l'autre ou d'un expectile à l'autre. Cependant, les méthodes de régression asymétrique en grande dimension ont été conçue pour faire une sélection individuelle des variables. Cette thèse s'inscrit dans le cadre de la sélection de variables par groupe en régression linéaire asymétrique en grande dimension. L'objectif de nos travaux de recherche est l'extension des pénalités de sélection de variables par groupe comme group Lasso, group SCAD et group MCP aux modèles de la régression asymétrique, afin de les doter de la capacité à faire de la sélection par groupe, et de leur permettre de capturer l'effet des groupes de variables hétérogènes.

CHAPITRE 2

GROUP PENALIZED QUANTILE REGRESSION

Dans le chapitre suivant, nous proposons le modèle de la régression quantile avec les pénalités de sélection de variables par groupe. En effet, l'idée de cette contribution consiste à proposer un algorithme efficace pour résoudre notre problème. Cet algorithme vise à approximer la fonction de perte quantile non différentiable par une fonction différentiable, et nous appliquons le principe Majoration-Minimisation sur celle ci à chaque mise à jour d'un bloc de variables dans l'algorithme de descente par bloc de variables. Nous avons montré que notre algorithme a une vitesse de convergence linéaire. En appliquant notre approche à deux jeux de données génétiques, nous avons abouti à des résultats qui concordent avec les résultats de d'autres travaux de recherche. L'analyse a consisté à regrouper les SNPs au sein de chaque gène et effectuer la sélection des gènes dans le cadre de RQ. Notre contribution est publié dans la revue italienne *Statistical Methods & Applications*

2.1 Abstract

Quantile regression models have become a widely used statistical tool in genetics and in the omics fields because they can provide a rich description of the predictors' effects on an outcome without imposing stringent parametric assumptions on the outcome-predictors relationship. This work considers the problem of selecting grouped variables in high-dimensional linear quantile regression models. We introduce a group penalized pseudo quantile regression (GPQR) framework with both group-lasso and group non-convex penalties. We approximate the quantile regression check function using a pseudo-quantile check function. Then, using the majorization-minimization principle, we derive a simple and computationally efficient group-wise descent algorithm to solve group penalized quantile regression. We establish the convergence rate property of our algorithm with the group-lasso penalty and illustrate the GPQR approach performance using simulations in high-dimensional settings. Furthermore, we demonstrate the use of the GPQR method in a gene-based association analysis of data from the Alzheimer's Disease Neuroimaging Initiative study and in an epigenetic analysis of DNA methylation data.

2.2 Introduction

Given the high-dimensional nature of omics experiments (omics refers to genomics, metabolomics, proteomics and transcriptomics), data regularization is becoming a standard approach to better extract relevant predictors for an outcome because there is typically a wild excess of predictors over participants. These top-ranked or selected predictors can be meaningful with respect to having a functional relationship to the trait or outcome. The lasso regularized regression [90] and its generalizations are attractive data-regularization tools for analyzing high-dimensional data.

In many situations, it is reasonable to group predictors so that the predictors belonging to the same group are included or excluded from a model simultaneously. For instance, in genome-wide association studies (GWAS) [116, 58], to understand the underlying biological structure of a complex disease better (e.g. Alzheimer's disease), one might want to group single-nucleotide-polymorphisms (SNPs) within a gene or genes within a biochemical pathway and then exploit group structure effects on a disease. In epigenetics studies, considering the correlations between methylation levels (features) in nearby positions along the genome can lead to better identifying differentially methylated genomic regions between two groups (outcome) [57]. Another attractive motivation of the group-variable selection models is the additive model with polynomial or non-parametric components, whereby each component/group may be expressed as a linear combination of basis functions of the original variables. In this context, the selection of important variables corresponds to the selection of groups of basis functions.

One can achieve group-variable selection by adding group penalties to the regularization-based regression approaches. The group lasso penalty [109, 67], also denoted as an l_1/l_2 penalty, is an extension of the lasso for performing variable selection on (predefined) groups of variables in generalized linear regression models. If each group of predictors reduces to a single predictor, then the group lasso penalty reduces to a standard lasso penalty. Because both [109]'s and [67]'s approaches require orthonormality of the groups, recently, [105] used a quadratic majorization trick within block descent algorithms to relax the predictors' groupwise orthonormality assumption. Moreover, [105] developed efficient algorithms to solve group-lasso penalized regression for a class of loss functions, including ordinary least squares, logistic, and several large margin classifier loss functions.

Like the regular lasso, the group-lasso lacks the selection consistency property because it tends to overly shrink the relevant group of variables. To overcome the over-shrinkage problem and gain selection consistency, [99] have extended the smoothly clipped absolute deviation (SCAD) penalty [27] and the minimax concave penalty (MCP) [110] for group variable selection, however both approaches require the groups to be orthonormal. [12] suggested group-SCAD and group-MCP penalized approaches in the case of ordinary least squares (OLS) regression and a general design matrix (i.e. non-orthonormal groups).

Omics data, however, are heterogeneity prone, and covariates may have different effects on different segments of the conditional distribution of the response. These types of heterogeneity are of interest to many researchers ; however, they tend to be overlooked by using (group) penalized OLS methods that only capture the effects of the covariates on the mean of the conditional distribution.

Quantile regression (QR) assesses how conditional quantiles of the response variable vary with respect to measured covariates[52, 54]. By allowing estimation of the predictor effects in different quantiles, QR provides a more complete picture of the conditional distribution of the response variable than the single estimate of the conditional mean that can be obtained via OLS regression. QR is widely used in genetics and in the omics fields, and [13] provide a good review of its application in these fields.

Many recent studies have focused on penalized QR in high dimensional settings. Earlier in this decade, several authors investigated the theoretical properties of penalized QR, including [5], [97], and [26] for the lasso penalty ; [98], [29], and references therein for the non-convex penalties. More recently, several studies have focused on the computational aspect for solving the penalized QR framework, including [101] and [61] for the lasso penalty ; [78] for non-convex penalties ; [108] for the elastic net penalty ; [46] and [72] for the lasso, SCAD and MCP penalties.

Several authors have introduced penalized QR in the context of both semi- and non-parametric frameworks [76, 112]. [95] proposed a Bayesian semi-parametric QR additive model, where penalized splines are employed for non-parametric components, and the lasso penalty is employed for the parametric components. However, model selection is restricted to the (linear) parametric part of the model, and there is no selection in the non-parametric part of the model. [30] extended the boosting algorithm to a semi-parametric QR, which allows for selection of both linear and nonlinear effects.

Although the theoretical aspect of group penalized QR has recently been addressed by a few authors, computationally efficient algorithms for solving groupwise penalized QR have received less attention in the literature. [48] developed theoretical results for the convergence rate and the oracle property of the group-lasso QR estimator. To estimate the model parameters, the author transformed the group-lasso QR problem to a second order cone programming (SOCP) problem and then used an interior point algorithm to solve it. Anterior point algorithms, however, can be computationally challenging in the presence of high dimensional data [25]. Asymptotic normality of the adaptive group-lasso QR estimator was addressed in [18], for a fixed and divergent number of the groups. [34] proposed a group-lasso penalized QR for the binary response. In [34], a continuous latent variable is considered to govern the binary response, and techniques similar to those used in Bayesian lasso (binary) QR frameworks [44, 56] are employed to develop a Bayesian Gibbs sampling procedure to estimate the model parameters. Because continuous priors are imposed on the regression parameters, sparsity cannot be achieved (i.e. draws from the posterior distributions are never exactly zero), and variable selection needs further manipulation. Finally, although [78] claimed that their R package software, `rn`, performs groupwise penalized QR, the method as described in their manuscript only handles single-variable-selection non-convex penalized QR and no procedure within the `rqPen` R package achieves group selection. In summary, computationally-efficient methods are lacking for group variable selection in QR.

In this work, we develop a unified computationally-efficient framework for solving penalized quantile regression with group-lasso, group-SCAD, and group-MCP penalties. Because one of the biggest challenges in solving QR lies in the non-differentiability of the loss/check function [54, 41], we rely on the pseudo-quantile check functions proposed in [3] and [76], and we use the majorization-minimization principle within block coordinate descent algorithms to solve the groupwise regularized QR problem. We also develop two additional alternative algorithms to solve the group-SCAD and group-MCP penalized QR based on the local linear approximation trick [120]. Our framework, termed group penalized pseudo quantile regression (GPQR), allows for general design matrices. That is, it does not require the predictors to be groupwise orthonormal. The framework is implemented in an R software package, GPQR, which is publicly available in GitHub (https://github.com/ouhourane/GPQR). Moreover, we study the rate of convergence of our framework for the group-lasso penalty.

The remainder of this article proceeds as follows. In Section 2.3 we formulate our GPQR framework, we provide the convergence rate analysis of our algorithm for the group lasso penalty, and we give details about the algorithm's implementation. Evaluation of the performance of our methods through exhaustive simulation studies is considered in Section 2.4. In Section 2.5, the use of the proposed methodology is illustrated in gene-based analyses of two interesting real genetic datasets. We conclude with a discussion section.

2.3 Pseudo quantile regression and group penalizations

Let $\{(y_1, \mathbf{x}_1), \cdots, (y_n, \mathbf{x}_n)\}$ be observed data, where y_i is the observed response and $\mathbf{x}_i = (1, x_{i1}, \ldots, x_{ip})^\top$ is a $(p+1)$-dimensional observed vector of predictors for subject $i = 1, \ldots, n$. We denote by $\mathbf{X}$ the design matrix with n rows and $p+1$ columns. We assume that the predictors $1, X_1, \ldots, X_p$ are put into K groups $(1, 2, 3, \ldots, p+1) = \bigcup_{k=1}^{K} I_k$, where the size of each group is p_k (the cardinality of index set I_k is p_k) and the groups are non-overlapping ($I_k \cap I_{k'} = \emptyset$ for $k \neq k'$). Because the intercept is included, we assume $I_1 = \{1\}$.

The group penalized QR problem can be formulated as

$$\hat{\boldsymbol{\beta}}_\tau = \arg\min_{\boldsymbol{\beta}} \Big(R(\boldsymbol{\beta}) := \frac{1}{n} \sum_{i=1}^{n} \rho_\tau(y_i - \mathbf{x}_i^\top \boldsymbol{\beta}) + \sum_{k=1}^{K} w_k P_\lambda(\|\boldsymbol{\beta}_k\|_2) \Big), \tag{2.1}$$

where $\rho_\tau(u) = |\tau - \mathbb{1}(u \leq 0)| \cdot |u|$, is the so-called check/hinge function [54], and $(\hat{\boldsymbol{\beta}}_\tau)_k$ is the vector of the effects of the predictors belonging to group k on the τth conditional quantile of the response. Hereafter, for ease of notation, we drop the subscript for the vector $\boldsymbol{\beta}_\tau$ when no confusion arises. $P_\lambda(\cdot)$ is the penalty function with two regularization parameters (λ, θ). The parameter w_k is used to adjust for the group sizes in the penalty; a reasonable choice is $w_k = \sqrt{p_k}$ [84]. Because the intercept is not penalized, $w_1 = 0$. In this work, we consider the group Lasso (GLasso), group MCP (GMCP), and group SCAD (GSCAD) penalties which are defined respectively by the penalty function, $P_\lambda(t)$, as follows

$$\lambda t, \qquad (2.2) \qquad \begin{cases} (\lambda t - \dfrac{t^2}{2\theta}) & \text{if } 0 \le t \le \theta\lambda, \\ \frac{1}{2}\lambda^2\theta & \text{if } t \ge \theta\lambda, \end{cases} \qquad (2.3)$$

$$\begin{cases} \lambda t & \text{if } 0 \le t \le \lambda, \\ \dfrac{\theta\lambda t - (t^2 + \lambda^2)/2}{\theta - 1} & \text{if } \lambda \le t \le \theta\lambda, \\ \dfrac{\lambda^2(\theta^2 - 1)}{2(\theta - 1)} & \text{if } t \ge \theta\lambda, \end{cases} \qquad (2.4)$$

where θ is a second tuning parameter of the GMCP and GSCAD penalties, with $\theta > 1$ for GMCP and $\theta > 2$ for GSCAD. Investigation of optimal values of θ has been discussed in the literature and fixed values, such as $\theta = 4$ for GSCAD and $\theta = 3$ for GMCP, have been suggested as suitable for many problems; however, the performance does not improve significantly with θ selected by data driven approaches, [27, 75]. We therefore set θ equal to the recommended values in all our simulations and real data analyses. Hereafter, the subscript θ in the penalty P_λ is removed for notation simplicity.

Solving (2.1) can be very computationally challenging, especially in high-dimensional settings, owing to the non-differentiability of $\rho_\tau(\cdot)$. To overcome this issue, we suggest replacing $\rho_\tau(\cdot)$ in equation (2.1) with one of the following two pseudo-quantile approximation loss functions [72] :

$$\Psi^{(1)}_{\tau,\delta}(u) = \begin{cases} (\tau - 1)(u + \dfrac{\delta}{2}) & \text{if } u < -\delta \\ \dfrac{(1-\tau)u^2}{2\delta} & \text{if } -\delta \le u < 0 \\ 0.5\tau u^2/\delta & \text{if } 0 \le u < \delta \\ \tau(u - 0.5\delta) & \text{if } \delta \le u \end{cases} \qquad (2.5)$$

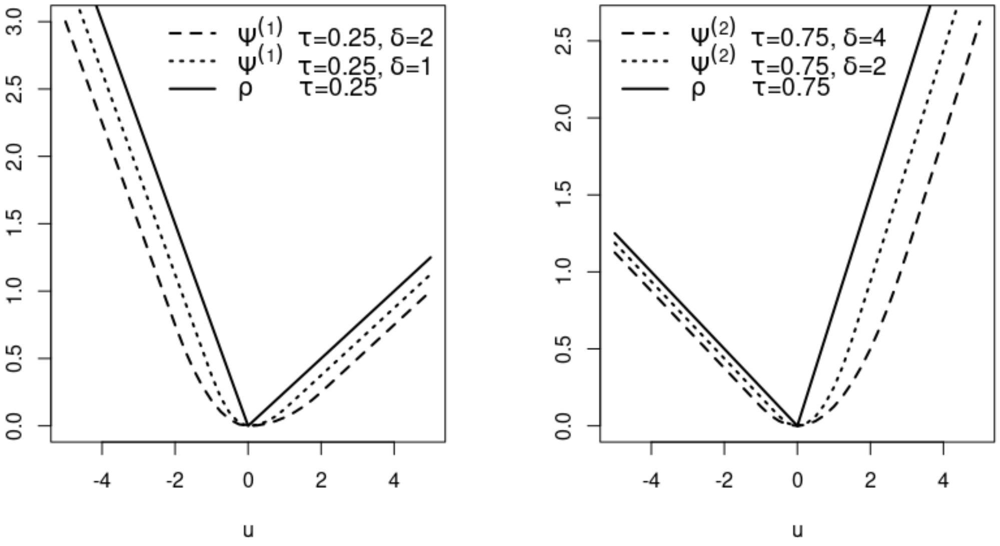


Figure 2.1 – Left panel : the standard quantile function $\rho_{0.25}(.)$ is shown by a solid line and the pseudo quantile function $\Psi^{(1)}_{\tau,\delta}(.)$ for $\tau = 0.25$ and $\delta = \{1, 2\}$ are shown by the dotted and dashed lines, respectively. Right panel : the standard quantile function $\rho_{0.75}(.)$ is shown by a solid line and the pseudo quantile function $\Psi^{(2)}_{\tau,\delta}$ for $\tau = 0.75$ and $\delta = \{2, 4\}$ are shown by the dotted and dashed lines, respectively.

$$\Psi^{(2)}_{\tau,\delta}(u) = \begin{cases} (\tau-1)u - \frac{\delta(1-\tau)^2}{2} & \text{if } u < \dfrac{\tau-1}{\delta^{-1}} \\ \frac{1}{2\delta}u^2 & \text{if } \dfrac{\tau-1}{\delta^{-1}} \leq u \leq \tau\delta \\ \tau u - \frac{\delta\tau^2}{2} & \text{if } u > \tau\delta. \end{cases} \tag{2.6}$$

Hence, the GPQR problem, in its general form, is given by

$$\hat{\boldsymbol{\beta}}(\delta) = \arg\min_{\boldsymbol{\beta}} \Big(R_\delta(\boldsymbol{\beta}) := L(\boldsymbol{\beta}) + \sum_{k=1}^{K} w_k P_\lambda(\|\boldsymbol{\beta}_k\|_2) \Big), \tag{2.7}$$

where $L(\boldsymbol{\beta}) = n^{-1}\sum_{i=1}^{n} \Psi_\tau(y_i - \mathbf{x}_i^\top \boldsymbol{\beta})$ and $\Psi_\tau(\cdot) = \Psi^{(1)}_{\tau,\delta}(\cdot)$ or $\Psi^{(2)}_{\tau,\delta}(\cdot)$ is one of the two pseudo functions (2.5) or (2.6). Figure 2.1 (left panel) illustrates the QR check function $\rho_\tau(\cdot)$ and the pseudo loss function, $\Psi^{(1)}_{\tau,\delta}(\cdot)$, for $\tau = 0.25$ and $\delta = \{1, 2\}$. The right panel contrasts the function $\Psi^{(2)}_{\tau,\delta}(\cdot)$ and $\rho_\tau(\cdot)$ for $\delta = \{2, 4\}$ and $\tau = 0.75$. Actually, when δ becomes small, the two pseudo loss functions become close in shape to the QR check function ; however, both functions are differentiable everywhere and have continuous derivatives.

The pseudo approximation (2.5) is proposed by [43]. It has also been used by [76] and [112] in the context of non-parametric QR. The pseudo approximation (2.6) is introduced by [3]. The first pseudo approximation is given by four intervals ; however, the second pseudo approximation is defined only on three intervals, which leads to a difference in computation times when calculating the gradient in favor of (2.6) [72]. The next proposition provides the theoretical justifications for the success of these two approximations in providing a good solution for the initial problem (2.1).

Proposition 2.1 *For any fixed value of δ, let $\hat{\boldsymbol{\beta}}(\delta)$ be the unique minimizer of $R_\delta(\boldsymbol{\beta})$ in* (2.7)*. Then we have*

$$\inf_{\boldsymbol{\beta}} R(\boldsymbol{\beta}) \leq R(\hat{\boldsymbol{\beta}}(\delta)) \leq \inf_{\boldsymbol{\beta}} R(\boldsymbol{\beta}) + 2\kappa\delta,$$

where $R(\boldsymbol{\beta})$ is the exact group penalized quantile regression loss function defined in (2.1) *and $\kappa = \max(\tau, 1-\tau)/2$ or $\max(\tau^2, (1-\tau)^2)/2$.*

The proof of Proposition 2.1 is detailed in Section 2.7.1 of the Supplementary Material. The above two inequalities are true for all possible values of the tuning parameters (λ for GLasso or (λ, θ) for GMCP/GSCAD). Thus, we can compute the solution of (2.1) for the three group penalties by solving (2.7) with a small value of δ. In fact, as $\delta \to 0$, the QR with original check function $\rho_\tau(.)$ and its pseudo approximations $\Psi_\delta(.)$ are very similar. [72] showed that the convergence speed of pseudo-QR with the lasso penalty is greatly decreased for small values of δ, and therefore, it can be used to control the trade-off between speed and accuracy. For the GPQR framework, we followed [72] and set $\delta = 1$ in all analyses, which is a suitable value of δ to balance between algorithm computational efficiency and model accuracy. This is also the default value of this parameter in their `SQR` R package.

To solve problem (2.7), we propose a groupwise descent algorithm; the details are as follows. Let $\tilde{\boldsymbol{\beta}} = (\tilde{\boldsymbol{\beta}}_1, \dots, \tilde{\boldsymbol{\beta}}_{k-1}, \tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{k+1}, \dots, \tilde{\boldsymbol{\beta}}_K)$ be the current iteration and $\tilde{\boldsymbol{\beta}}_{-k} = (\tilde{\boldsymbol{\beta}}_1, \dots, \tilde{\boldsymbol{\beta}}_{k-1}, \tilde{\boldsymbol{\beta}}_{k+1}, \dots, \tilde{\boldsymbol{\beta}}_K)$ be the current iteration with the k-th group excluded. Suppose we are updating the k-th group of $\boldsymbol{\beta}$, that is, $\boldsymbol{\beta}_k = (\beta_1, , \dots, , \beta_{p_k})^\top$ for some $k \in \{1, , \dots, , K\}$. Furthermore, consider the objective function $R_\delta(\boldsymbol{\beta})$ in (2.7) as a function of the k-th group $\boldsymbol{\beta}_k$, while keeping all the other groups fixed at $\tilde{\boldsymbol{\beta}}_{-k}$, that is, $R_\delta(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k}) := R_\delta(\boldsymbol{\beta})_{\boldsymbol{\beta}_{k'} = \tilde{\boldsymbol{\beta}}_{k'}, 1 \le k' \le K, k' \ne k}$ and $L(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k}) := L(\boldsymbol{\beta})_{\boldsymbol{\beta}_{k'} = \tilde{\boldsymbol{\beta}}_{k'}, 1 \le k' \le K, k' \ne k}$. Thus, at each iteration, we optimize the objective function $R_\delta(\boldsymbol{\beta})$ only in terms of the k-th group variables $\boldsymbol{\beta}_k$, while keeping all the other groups fixed at $\tilde{\boldsymbol{\beta}}_{-k}$ (i.e, $\tilde{\boldsymbol{\beta}}_k^{\text{new}} \leftarrow \arg\min_{\boldsymbol{\beta}_k} R_\delta(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k})$). To solve this problem efficiently for group k, we derive an upper-bound quadratic-form approximation for $R_\delta(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k})$ based on the quadratic majorization property of $L(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k})$ given in the next proposition. Then we minimize the surrogate majorizing quadratic form rather than the actual $R_\delta(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k})$.

Proposition 2.2 *Let $\mathbf{X}_k$ be the sub-matrix of $\mathbf{X}$ corresponding to group k. The quadratic majorization condition is satisfied for both pseudo loss approximations. That is, for all $\boldsymbol{\beta}$ and $\tilde{\boldsymbol{\beta}}$ we have*

$$L(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k}) \le L(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}) + (\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top \nabla_k L(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}) + \frac{1}{2}(\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top \mathbf{H}_k (\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k), \tag{2.8}$$

where $\mathbf{H}_k = \frac{2\boldsymbol{X_k}^\top \boldsymbol{X_k}/n}{\delta/max(\tau, 1-\tau)}$ for $\Psi^{(1)}_{\tau,\delta}(\cdot)$, and $\mathbf{H}_k = \frac{2\boldsymbol{X_k}^\top \boldsymbol{X_k}/n}{\delta}$ for $\Psi^{(2)}_{\tau,\delta}(\cdot)$.

The proof of Proposition 2.2 is detailed in Section 2.7.2 of the Supplementary material.

The upper bound in (2.8) can be further relaxed to get the following upper bound approximation of $R_\delta(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k})$

$$\begin{aligned} R_\delta(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k}) \le Q(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k}) \quad &:= \quad L(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}) + (\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top \nabla_k L(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}) \\ &+ \quad \frac{\gamma_k}{2}(\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top (\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k) + w_k P_\lambda(\|\boldsymbol{\beta}_k\|_2), \end{aligned} \tag{2.9}$$

where γ_k is the largest eigenvalue of the matrix $\mathbf{H}_k$.

Thus, we minimize the quadratic form $Q(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k})$ groupwise, while cycling through groups. The update solution using $Q(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k})$ of (2.9) has a closed form for the three group penalties.

Note, after updating all the groups in a cycle, one can verify that the objective function (2.7) is decreased (i.e., it satisfies the descent property) using the majorization-minimization principle [41, 42]. This assures the convergence of the GPQR algorithms.

Validation of GPQR convergence is carried out through simulation scenarios in Section 2.4.3 to demonstrate that the algorithm solution satisfies the Karush-Kuhn-Tucker (KKT) conditions. The derivation of both the theoretical and numerical KKT conditions of the GPQR algorithm are outlined in Sections 6 and 7 of the Supplementary material, respectively. Our KKT conditions are calculated based on the pseudo QR objective function $R_\delta(\boldsymbol{\beta})$ given in (2.7).

Next, we present the GPQR framework in detail for each group-penalty.

2.3.1 Pseudo QR with group-Lasso penalty

This section gives details of the GPQR algorithm with the GLasso penalty and its convergence rate properties.

In this case, the penalty term $P_\lambda(\|\boldsymbol{\beta}_k\|_2)$ in (2.9) is replaced by the GLasso penalty given in (2.2). By employing the proximal gradient algorithm for $Q(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k})$ to update $\boldsymbol{\beta}_k$, one can write

$$\begin{aligned}
\tilde{\boldsymbol{\beta}}_k^{\text{new}} &= \arg\min_{\boldsymbol{\beta}_k} Q(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k}) \\
&= \arg\min_{\boldsymbol{\beta}_k} \frac{1}{2}\|\boldsymbol{\beta}_k - (\widetilde{\boldsymbol{\beta}}_k - \gamma_k^{-1}\nabla_k L(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}))\|_2^2 + \lambda w_k \gamma_k^{-1}\|\boldsymbol{\beta}_k\|_2 \\
&= \text{prox}_{\lambda w_k \gamma_k^{-1} h}(\widetilde{\boldsymbol{\beta}}_k - \gamma_k^{-1}\nabla_k L(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k})]),
\end{aligned} \tag{2.10}$$

where the proximal mapping of the function $h(\cdot) = \|\cdot\|_2$ is given by

$$\text{prox}_{\lambda h}(\mathbf{u}) = \arg\min_{\mathbf{v}} \lambda h(\mathbf{v}) + \frac{1}{2}\|\mathbf{v} - \mathbf{u}\|_2^2 .$$

The following algorithm gives details of the GPQR with the GLasso penalty.

Algorithm 1: The GPQR algorithm for the GLasso penalty

```
Calculate $\gamma_k$, the maximum eigenvalue of $\mathbf{H}_k$ for $k = 1, \ldots, K$, and initialize $\tilde{\boldsymbol{\beta}}$;
repeat
    for $k = 1, 2, \ldots, K$ do
        $\tilde{\boldsymbol{\beta}}_k^{\text{new}} \leftarrow \text{prox}_{\lambda w_k \gamma_k^{-1} h}(\tilde{\boldsymbol{\beta}}_k - \gamma_k^{-1} \nabla_k L(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}))$
    end
until Convergence of $\tilde{\boldsymbol{\beta}}$;
Return $\tilde{\boldsymbol{\beta}}$;
```

The next theorem provides the convergence rate analysis of the GPQR algorithm with the GLasso penalty.

Théorème 2.3 *The GPQR algorithm with GLasso penalty (Algorithm 1) converges at least linearly to the global solution $\boldsymbol{\beta}^*$.*

The proof of Theorem 2.3 is relegated to Section 2.7.3 of the Supplementary material.

2.3.2 Pseudo QR with GSCAD and GMCP penalties

The nonconvex group penalties, GSCAD and GMCP, are used both to perform group variable-selection and to reduce the bias towards zero introduced by the GLasso. For instance, to understand the effect of the GSCAD penalty (2.4) compared with the GLasso (2.2), let us consider its derivative function, which relies directly on the shrinkage amount of the parameters. For small values of $\|\boldsymbol{\beta}_k\|_2$ (i.e. $\|\boldsymbol{\beta}_k\|_2 \leq \lambda$), GSCAD exercises the same shrinkage on the parameters' effects, as the GLasso does (i.e. $P'_\lambda(\|\boldsymbol{\beta}_k\|_2) = \lambda$). However, the GSCAD penalty continuously reduces the shrinkage for $\lambda \leq \|\boldsymbol{\beta}_k\|_2 \leq \theta\lambda$, and the shrinkage becomes zero when $\|\boldsymbol{\beta}_k\|_2 \geq \lambda\theta$ (i.e. $P'_\lambda(\|\boldsymbol{\beta}_k\|_2) = 0$). A similar reasoning can explain the GMCP penalty effect [11].

The following proposition gives closed form solutions to the update, $\tilde{\boldsymbol{\beta}}_k^{\text{new}}$, in (2.9) when $P_\lambda(\|\boldsymbol{\beta}_k\|_2)$ is given by (2.3) for GMCP, and by (2.4) for the GSCAD.

Proposition 2.4 *Let $Q(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}})$ be the surrogate function given by* (2.9) *and let $P_\lambda(\|\boldsymbol{\beta}_k\|_2)$ be one of the two penalties given in* (2.3) *and* (2.4). *The closed form solutions to* (2.9) *of $\tilde{\boldsymbol{\beta}}_k^{\text{new}}$ for the GPQR algorithm with the*

GMCP and GSCAD penalties are, respectively, given by

$$\tilde{\boldsymbol{\beta}}_k^{\text{new}} \longleftarrow F(\mathbf{Z}_k) = \begin{cases} \frac{1}{\gamma_k - w_k/\theta} \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} S(\|\mathbf{Z}_k\|_2, \lambda w_k), & \textit{if} \ \ \|\mathbf{Z}_k\|_2 \leq \gamma_k \theta \lambda \\ \frac{1}{\gamma_k} \mathbf{Z}_k, & \textit{if} \ \ \|\mathbf{Z}_k\|_2 > \gamma_k \theta \lambda, \end{cases} \tag{2.11}$$

$$\tilde{\boldsymbol{\beta}}_k^{\text{new}} \longleftarrow F(\mathbf{Z}_k) = \begin{cases} \frac{1}{\gamma_k} \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} S(\|\mathbf{Z}_k\|_2, \lambda w_k), & \textit{if} \quad \|\mathbf{Z}_k\|_2 \leq (w_k + \gamma_k)\lambda \\ \frac{1}{\gamma_k - \dfrac{w_k}{\theta - 1}} \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} (\|\mathbf{Z}_k\|_2 - \frac{\lambda w_k \theta}{\theta - 1}), & \textit{if} \quad (w_k + \gamma_k)\lambda < \|\mathbf{Z}_k\|_2 \leq \gamma_k \theta \lambda \\ \frac{1}{\gamma_k} \mathbf{Z}_k, & \textit{if} \quad \|\mathbf{Z}_k\|_2 > \gamma_k \theta \lambda, \end{cases} \tag{2.12}$$

where $\mathbf{Z}_k = \gamma_k \tilde{\boldsymbol{\beta}}_k - \nabla_k L(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k})$*, and S(.) is the soft-threshold operator, defined as*

$$S(\|\mathbf{z}\|_2, \lambda) := \begin{cases} 0, & \textit{if} \ \ \|\mathbf{z}\|_2 \leq \lambda \\ \|\mathbf{z}\|_2 - \lambda, & \textit{if} \ \ \|\mathbf{z}\|_2 > \lambda. \end{cases}$$

The proof of Proposition 2.4 is detailed in Section 2.7.4 of the Supplementary material.

The following algorithm summarizes the steps of the GPQR framework with the GMCP or GSCAD penalty :

Algorithm 2: The GPQR algorithm with the GMCP or GSCAD penalty.

```
Calculate $\gamma_k$, the maximum eigenvalue of $\mathbf{H}_k$ for $k = 1, \ldots, K$, and initialize $\tilde{\boldsymbol{\beta}}$;
repeat
    for $k = 1, 2, \ldots, K$ do
        $\tilde{\boldsymbol{\beta}}_k^{\text{new}} \leftarrow F(-\nabla_k L(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}) + \gamma_k \tilde{\boldsymbol{\beta}}_k)$
    end
    where $F(\cdot)$ is given by (2.11) and (2.12) for GMCP and GSCAD penalties, respectively;
until Convergence of $\tilde{\boldsymbol{\beta}}$;
Return $\tilde{\boldsymbol{\beta}}$.;
```

The convexity of $P_\lambda(t)$ for the GLasso is a crucial property for proving the convergence, at least linearly, of the GPQR in Theorem 2.3. However, this property is not available for the non-convex GMCP and GSCAD penalties.

2.3.3 Pseudo QR with Group Local Linear Approximation penalty

In this section, we propose to extend the local linear approximation (LLA) trick to solve the GPQR with the GMCP and GSCAD penalties to remedy the possible computational weakness of the two nonconvex penalties.

The LLA approximation is based on the first order Taylor expansion of the MCP or SCAD penalty functions around $||\tilde{\boldsymbol{\beta}}_k||_2$. Thus, one can write

$$P_\lambda(||\boldsymbol{\beta}_k||_2) \approx P_\lambda(||\tilde{\boldsymbol{\beta}}_k||_2) + P'_{\lambda,\theta}(||\tilde{\boldsymbol{\beta}}_k||_2)(||\boldsymbol{\beta}_k||_2 - ||\tilde{\boldsymbol{\beta}}_k||_2), \tag{2.13}$$

where $P_\lambda(.)$ is one of the two penalties given in (2.3) and (2.4).

Substituting (2.13) into (2.9) leads to the following update for the GPQR with the group local linear approximation (GLLA) penalty

$$\tilde{\boldsymbol{\beta}}_k^{\text{new}} = \arg\min_{\boldsymbol{\beta}_k} Q(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k}) + \lambda w'_k ||\boldsymbol{\beta}_k||_2, \tag{2.14}$$

where $w'_1 = 0$ and $w'_k = \dfrac{w_k P'_{\lambda,\theta}(||\tilde{\boldsymbol{\beta}}_k||_2)}{\lambda}$ for $k = 2, \ldots, K$. The weight w'_k depends on the penalty function through the first derivative, $P'_{\lambda,\theta}(||\tilde{\boldsymbol{\beta}}_k||_2)$, which is given for the GMCP and GSCAD, respectively, as follows :

$$\begin{cases} \lambda - \dfrac{||\tilde{\boldsymbol{\beta}}_k||_2}{\theta}, & \text{if } ||\tilde{\boldsymbol{\beta}}_k||_2 \leq \theta\lambda \\ 0, & \text{if } ||\tilde{\boldsymbol{\beta}}_k||_2 > \theta\lambda, \end{cases}$$

$$\begin{cases} \lambda, & \text{if } \quad ||\boldsymbol{\beta}_k||_2 \leq \lambda \\ \dfrac{\theta\lambda}{\theta - 1} - \dfrac{||\tilde{\boldsymbol{\beta}}_k||_2}{\theta - 1}, & \text{if } \quad \lambda < ||\boldsymbol{\beta}_k||_2 \leq \theta\lambda \\ 0, & \text{if } \quad ||\boldsymbol{\beta}_k||_2 > \theta\lambda. \end{cases}$$

The problem (2.14) can be solved using a GLasso-type update similar to Algorithm 1 described in Section 2.3.1. Thus, we use the proximal gradient algorithm in (2.10) to solve it.

The details of the GPQR approach with the GLLA penalty is described in the following algorithm.

Algorithm 3: The GPQR algorithm with the GLLA penalty.

```
Initialize $\tilde{\boldsymbol{\beta}}$ and set $(w'_1, \gamma_1) = (0, 0)$;
Calculate $\gamma_k$ as the maximum eigenvalue of $\mathbf{H}_k$ and $w'_k = \lambda^{-1} P'_{\lambda,\theta}(\|\tilde{\boldsymbol{\beta}}_k\|_2)$ for $k = 2, \ldots, K$;
repeat
    for $k = 1, 2, \ldots, K$ do
        $\tilde{\boldsymbol{\beta}}_k^{\mathrm{new}} \leftarrow \mathrm{prox}_{\lambda w'_k \gamma_k^{-1} h}(\tilde{\boldsymbol{\beta}}_k - \gamma_k^{-1} \nabla_k L(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}))$,
    end
    for $k = 2, 3, \ldots, K$ do
        $w_k'^{new} \leftarrow \frac{w_k P'_{\lambda,\theta}(\|\tilde{\boldsymbol{\beta}}_k^{\mathrm{new}}\|_2)}{\lambda}$,
    end
until Convergence of $\tilde{\boldsymbol{\beta}}$;
Return $\tilde{\boldsymbol{\beta}}$.;
```

Note that the GLLA penalty is a convex majorant of the GMCP (or GSCAD) penalty. Thus, for each fixed value of λ, the GLLA allows a search of the solution in a locally convex region, and consequently it may lead to stable and smooth path solutions.

A comparison of the GLLA approximation and the exact GMCP and GSCAD penalties is illustrated in Section 2.7.5 of the Supplementary material. Figure $S.1$ shows that the exact and approximate path solutions of the GPQR algorithm with nonconvex penalties are nearly identical for all values of the tuning parameter λ. This proves the efficiency of the GLLA approximation.

2.3.4 Implementation

In this section we give details about the implementation of the proposed GPQR algorithms.

The intercept term is always included in all our models. Each GPQR model is solved by using a fine grid of λ. We proceeded by choosing $\lambda_{\max}$ which is the smallest λ that allows all groups, $\boldsymbol{\beta}_k,\ (2 \leqslant k \leqslant K)$, to be zero except the intercept. To obtain $\lambda_{\max}$, we first calculated the estimates, $\hat{\beta}_0$, for the null model with only the intercept :

$$\hat{\beta}_0 = \text{arg}\min_{\beta_0} \frac{1}{n}\sum_{i=1}^{n} \Psi_\tau(y_i - \beta_0). \tag{2.15}$$

According to the KKT conditions of (2.15), we derived the following formula :

$$\lambda_{\max} = \max_{k=2,\dots,K} \|\nabla_k L(\hat{\beta}_0, \mathbf{0})\|_2/\omega_k.$$

Let $\lambda_{\min} = \eta\lambda_{\max}$, where $0 < \eta < 1$ is a small number. We generated a sequence of λs by placing 98 evenly spaced points, $\{\lambda^{[l]}\}_{l=2}^{99}$, between $\lambda_{\max}$ and $\lambda_{\min}$ in log-scale and let $\lambda^{[1]} = \lambda_{\max}$ and $\lambda^{[100]} = \lambda_{\min}$.

We also used the warm-start trick in solving the solution paths : the solution of $\widehat{\boldsymbol{\beta}}$ at $\lambda^{[l-1]}$ is taken as the initial value for solving the solution of $\widehat{\boldsymbol{\beta}}$ at $\lambda^{[l]}$.

For computing efficiency at each λ, we used the "strong rule statement" proposed by [91], which screens out group predictors. Let $\hat{\boldsymbol{\beta}}^{[l]}$ be the solution at $\lambda^{[l]}$. For finding the solution $\boldsymbol{\beta}^{[l+1]}$ at $\lambda^{[l+1]}$, we introduced a supplementary screening step to check whether a group k satisfies the following condition :

$$\|\nabla_k L(\widehat{\boldsymbol{\beta}}^{[l]})\|_2 \geq \omega_k(2\lambda^{[l+1]} - \lambda^{[l]}). \tag{2.16}$$

Let $\mathbf{S}$ be the subset of the predictors' groups that are not discarded by condition (2.16) and $\mathbf{S}^c$, its complement. According to the strong rule, at $\lambda^{[l+1]}$, the coefficients of the groups in the set $\mathbf{S}$ are very likely to be active and those of the groups in the complement set $\mathbf{S}^c$ are very likely to be inactive. If this statement is correct, then solving the proposed GPQR models will only require a reduced data set, $(\mathbf{y}, \mathbf{X_S})$, where $\mathbf{X_S}$ is the restricted matrix where the columns are the groups belonging to $\mathbf{S}$. Denote this solution as $\hat{\boldsymbol{\beta}}_{\mathbf{S}}$. Then, one must verify if the strong rule statement is well confirmed at $\lambda^{[l+1]}$ by verifying if $\tilde{\boldsymbol{\beta}}^{[l+1]} = (\hat{\boldsymbol{\beta}}_{\mathbf{S}}, \mathbf{0})$ satisfies the KKT conditions. Following the calculation details in Sections 6 and 7 of the Supplementary material, this means that for the GLasso, GMCP, and GSCAD, any group k from the inactive set, $\mathbf{S}^c$, needs to satisfy the following inequality

$$\|\nabla_k L(\tilde{\boldsymbol{\beta}}^{[l+1]})\|_2 \leq \omega_k \lambda^{[l+1]}.$$

For GLLA, the inactive group, k, needs to verify

$$\|\nabla_k L(\tilde{\boldsymbol{\beta}}^{[l+1]})\|_2 \leq \omega'_k \lambda^{[l+1]},$$

where ω'_k is the weight given in (2.14).

If there are no violations of the strong rule statement, then the solution at $\lambda = \lambda^{[l+1]}$ is $\tilde{\boldsymbol{\beta}}^{[l+1]} = (\hat{\boldsymbol{\beta}}_{\mathbf{S}}, \mathbf{0})$, otherwise we add the subset of the violator groups, denoted as $\mathbf{V}$, into the active set, $\mathbf{S} = \mathbf{S} \cup \mathbf{V}$, and repeat the whole procedure with the reduced data set $(\mathbf{y}, \mathbf{X_S})$.

2.4 Numerical experiments

We conducted simulation studies with four scenarios to illustrate the methodology presented in this work. In the first scenario, we aimed both i) to graphically illustrate key advantages of using group penalized quantile regression approaches to detect heterogeneous effects of predictors, as alternatives to group penalized Least-Square (LS) regression methods, and ii) to compare the proposed approaches with existing penalized QR methods, namely, the regularized Bayesian QR(BQR) method [2], the standard quantile regression with the lasso, SCAD, and MCP penalties [72], and Boosting Additive QR (BAQR) [30]. The LS methods are implemented in the `grpreg` R package [10], with GLasso, GSCAD, and GMCP penalties. The BQR approach is implemented in the `Brq` R package [2]. The BAQR is implemented in the `mboost` R package [36]. The standard penalized quantile regression of [72] uses the pseudo quantile approximation functions ((2.5) and (2.6)) to fit the QR, and is implemented in the `SQR` R package.

The second and the third scenarios targeted evaluation of the proposed approach's performance in terms of computational efficiency and prediction accuracy. The fourth scenario aimed to evaluate the GPQR algorithms convergence based on the numerical KKT conditions derived in Section 2.7.7 of the Supplementary material.

2.4.1 Simulation setting of Scenarios 1, 2 and 3

Setting of Scenario 1. To illustrate key advantages of the proposed method compared with single-variable selection QR methods, we focused on a setting in which the predictors are highly correlated in this scenario. The model is based on an illustration example in [70]. We set the sample size to $n = 100$ observations and $p = 20$ predictors. The predictors $X_j, j = 1, \ldots, 20$, were generated as follows :

— We generated $Z_j, j = 1, \ldots, 11$, following the standard normal distribution ;

— We set $X_j = Z_1 + \epsilon_j, \epsilon_j \sim N(0, 0.1), j = 1, \ldots, 4;$

— $X_j = Z_2 + \epsilon_j, \epsilon_j \sim N(0, 0.1), j = 5, \ldots, 8;$

— $X_j = Z_3 + \epsilon_j, \epsilon_j \sim N(0, 0.1), j = 9, \ldots, 12;$

— $X_j = Z_{j-9}, \ j = 13, \ldots, 20.$

Thus, we set the predictors' effects to be

$$\boldsymbol{\beta} = (\underbrace{3,3,3,3}_{G_1},\underbrace{2,2,2,2}_{G_2},\underbrace{-1,-1,-1,-1}_{G_3},\underbrace{0,\dots,0}_{G_4 - G_{11}})^\top$$

and $\sigma = 3$. The response Y is generated from the following location-scale linear regression model

$$Y = \sum_{j=1}^{20} \boldsymbol{\beta}_j X_j + \Phi(X_{20})\epsilon, \quad \epsilon \sim N(0,3),$$

where $\Phi(.)$ is the cumulative distribution function of the standard normal distribution. Many authors consider that using $\Phi(.)$ in variance simulation generates a model with heteroscedasticity [98, 33]. The predictors X_1, X_2, X_3, and X_4 form group G_1, for which the underlying common factor is Z_1 ; the predictors X_5, X_6, X_7, and X_8 form the second group G_2, for which the underlying common factor is Z_2 ; finally, X_9, X_{10}, X_{11}, and X_{12} form the third group G_3, for which the underlying common factor is Z_3. The within-group correlations are high. An oracle estimator would identify the groups G_1, G_2, and G_3 as the important variables, and variable X_{20} (i.e. $G_{11} = \Phi(X_{20})$) when $\tau \neq 0.5$.

Setting of Scenario 2. This scenario considers an additive model involving both continuous and categorical factors (i.e. groups of predictors) to relate y to the predictors. The model in this scenario is based in part on simulation studies conducted in [109]. We generated 21 independent random variables $Z_1, \dots, Z_{20}$ and W from $N(0,1)$. We set the predictors to be defined as $X_1 = Z_1$ and $X_j = (Z_j + W)/\sqrt{2}$, for $j = 2, \dots, 20$. Furthermore, each predictor $X_j, j = 11, \dots, 20$, was trichotomized as $\tilde{X}_j = 0$ if X_j is smaller than $\Phi^{-1}(1/3)$, $\tilde{X}_j = 1$ if X_j is larger than $\Phi^{-1}(2/3)$, and $\tilde{X}_j = 2$ if X_j is between $\Phi^{-1}(1/3)$ and $\Phi^{-1}(2/3)$. The response was then simulated from the heterogeneous additive model

$$Y = \underbrace{3X_3^3 + X_3^2 + X_3}_{G3} + \underbrace{\frac{1}{3}X_6^3 - X_6^2 + \frac{2}{3}X_6}_{G6} + \underbrace{2\mathbb{1}(\tilde{X}_{11} = 0) + \mathbb{1}(\tilde{X}_{11} = 1)}_{G11} + \Phi(X_1)\epsilon;$$

where $\epsilon \sim N(0,1)$, and $\mathbb{1}(\cdot)$ is the indicator function. In this scenario, each continuous factor was represented by a polynomial of degree 3 and each categorical factor was represented by two levels of its corresponding trichotomized variable. Thus, by construction, we have a total $p = 50$ (i.e. 30 continuous and 20 categorical variables), and we set the sample size to $n = 50$.

Setting of Scenario 3 : In this scenario, we considered an additive model involving continuous factors represented by polynomials of degree 3 to link y to the predictors. The data generation is motivated in part by a simulation study carried out in [78]. First, we simulated $(\tilde{X}_1, \tilde{X}_2, \dots, \tilde{X}_p)^\top$ from a multivariate normal distribution $N_p(\mathbf{0}_p, \boldsymbol{\Sigma})$ with $\boldsymbol{\Sigma} = (\sigma_{jk})_{p\times p}$ and $\sigma_{jk} = 0.5^{|j-k|}$. Second, we set $X_1 = \Phi(\tilde{X}_1)$ and $X_j = \tilde{X}_j$ for $j = 2, \ldots, p$. Third, each variable from {6,12,15,20} was represented through a third-order polynomial. Then, we simulated the response variable from the following regression model :

$$Y = \underbrace{X_6 + X_6^2 + X_6^3}+ \underbrace{\frac{1}{3}X_{12} - X_{12}^2 + \frac{2}{3}X_{12}^3}+ \underbrace{\frac{1}{2}X_{15} - X_{15}^2 + \frac{1}{2}X_{15}^3}$$
$$+ \underbrace{X_{20} + X_{20}^2 + X_{20}^3}+X_1\epsilon,$$

where $\epsilon \sim N(0,1)$. We considered $n = 300$ and $p = 1000$. In this scenario, we considered $\{X_j, X_j^2, X_j^3\}$ as a group when fitting penalized LS and all the proposed models. Thus, the final design matrix consists of $q = 3p = 3000$ variables.

Note, in both Scenarios 2 and 3, X_1 plays the role of the heteroskedastic predictor and does not influence the center of the response conditional distribution. Thus, one of the aims of these two settings is to test the GPQR ability to select X_1 when considering lower and/or upper conditional quantiles (i.e. $\tau \neq 0.5$).

We implemented 100 Monte Carlo replications in each of the last two scenarios. Each replication consists of a training dataset of 300 observations, and a test dataset of 300 observations. The training dataset is used to fit the proposed models and their competitors (at a desired τth quantile) to determine the optimal λ using five-fold cross-validation (CV). For our models, the optimal λ corresponds to the value of λ that gives a small value for the quantile-based prediction errors (QPE_τ) defined as

$$QPE_\tau = \frac{1}{n} \sum_{i \in \text{validation}} \rho_\tau(y_i - \mathbf{x}_i^\top \hat{\boldsymbol{\beta}}).$$

The performance evaluation of the methods, including the LS methods, is computed on the test data sets, and is based on the following statistics :

- False Positive FP : the number of the groups of variables with zero coefficients incorrectly included in the final model ;
- P1 : the proportion of the true active/non-null groups, $\boldsymbol{\beta}_k \neq 0$, that are selected ;
- P2 : the proportion of simulation runs X_1 (or X_{20} in Scenario 1) is selected ;
- AE : the absolute estimate error defined by $\sum_{j=0}^{p} |\hat{\beta}_j - \beta_j|$;

— The quantile-based prediction error (QPE_τ) defined as

$$QPE_\tau = \frac{1}{n}\sum_{i=1}^{n} \rho_\tau(y_i - \mathbf{x}_i^\top \hat{\boldsymbol{\beta}});$$

— The root mean square error (RMSE) defined as

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\text{Quantile}_{Y_i}(\tau|\mathbf{x}_i) - \widehat{\text{Quantile}}_{Y_i}(\tau|\mathbf{x}_i))^2},$$

where $\widehat{\text{Quantile}}_{Y_i}(\tau|\mathbf{x}_i)$ is the estimated value of the true quantile, $\text{Quantile}_{Y_i}(\tau|\mathbf{x}_i)$, of the Y_i conditional on $\mathbf{x}_i$.

The QPE_τ and $RMSE$ statistics have been used recently in [102] for model-prediction evaluation in the expectile regression framework. Note, for the LS methods, QPE is defined as the absolute deviation/prediction error (i.e. $QPE_{0.5}$), and $RMSE = [\sum_{i=1}^{n}(\text{Quantile}_{Y_i}(0.5|\mathbf{x}_i) - \mathbf{x}_i^\top \hat{\boldsymbol{\beta}}_{LS})^2/n]^{1/2}$.

In Scenario 1, two locations were investigated with the quantile-based models, $\tau = 0.5$ and 0.95; in Scenarios 2 and 3, the proposed methods were fitted for three locations/quantiles, $\tau = 0.5, 0.75$ and 0.95.

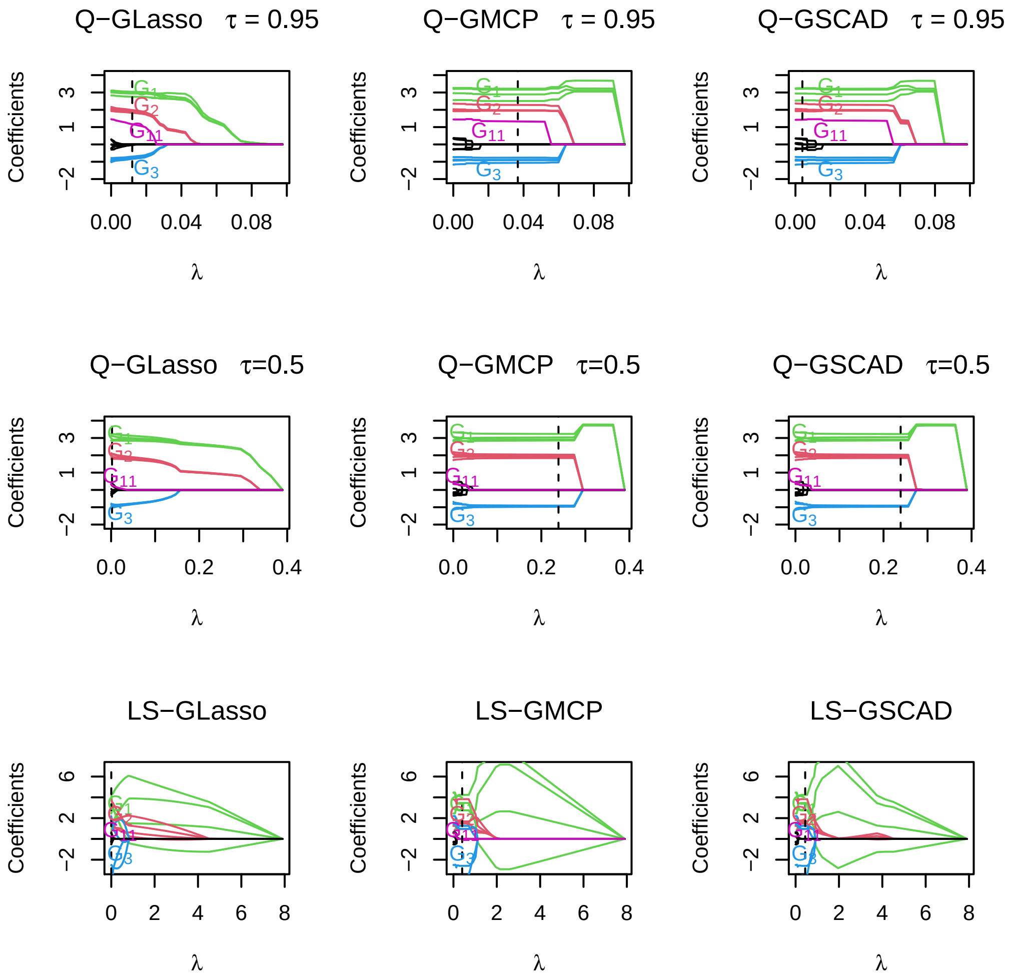


Figure 2.2 – At the top from left to right, the coefficient paths of the penalized quantile regression with the three group penalties (Q-GLasso, Q-GMCP, and Q-GSCAD) and $\tau = 0.95$ are shown as a function of the tuning parameter λ; the vertical dashed line reports λ selected by five-fold CV. The middle from left to right shows the coefficient paths of the penalized quantile regression with $\tau = 0.5$. The bottom row from left to right shows the coefficient paths corresponding to the `grpreg` R package with the same three penalties. The group coefficients G_1, G_2, G_3, and G_{11} are plotted in green, red, blue and pink, respectively. The black line corresponds to the noisy groups of predictors.

2.4.2 Simulation results of Scenarios 1, 2, and 3

In this section we outline and discuss the results of the first three scenarios.

Results of Scenario 1 :

Graphical illustration results (based on one replication) : Figure 2.2 shows the path solutions for the grid on $[\lambda_{\min}, \lambda_{\max}]$ of λ, for the GPQR and LS methods with the GLasso, GSCAD, and GMCP. The GPQR is fitted for two locations, $\tau = \{0.5, 0.95\}$. Figure 2.2 shows that the coefficients' profiles of the GPQR with $\tau \in \{0.50, 0.95\}$ tend to be smooth, however, the LS paths fluctuate widely, and some coefficients are in opposite directions/signs to their true values. This poor behavior of the LS methods is remarkably confirmed by the AE statistic in Table 2.1, which shows substantial bias of the LS parameters' estimators. Furthermore, the heteroskedastic variable $\Phi(X_{20})$ in the scale component, represented by G_{11}, is often recovered when fitting the GPQR model for the 0.95th conditional quantile (pink group) ; this is not the case for the GPQR model with $\tau = 0.5$ and for the LS methods. This shows that the GPQR framework can be useful for detecting heteroskedastic groups of variables.

Numerical results : Table 2.1 outlines the results for averages, over 100 replications, of the six statistics defined earlier. Notice that, the BQR, standard penalized QR, and BAQR methods are designed for individual variable selection, and therefore, they do not enforce the selection of a whole group of variables. Thus, for fair comparison with these three methods in this scenario, the false positive (FP) and P1 statistics were calculated for all methods as the number of predictors with zero coefficients incorrectly included in the final model and the proportion of the true active variables, respectively.

	$\tau = 0.50$										
Stats	LS-GLasso	Q-Lasso	Q-GLasso	LS-GMCP	Q-MCP	Q-GMCP	LS-GSCAD	Q-SCAD	Q-GSCAD	BQR	$BAQR$
FP	5.14	4.68	4.18	1.18	2.1	1.12	1.21	4.8	1.28	**0.40**	0.85
P1	**100%**	98%	**100%**	**100%**	81%	**100%**	**100%**	73%	**100%**	37%	64%
P2	70%	56%	55%	**4**%	28%	12%	15%	66%	14%	12%	12%
AE	16.4	10.0	**1.61**	15.6	10.1	3.84	15.9	23.1	3.8	16.9	14.11
RMSE	0.44	0.76	0.31	0.36	0.51	**0.24**	1.04	0.55	**0.22**	35.26	1.62
QPE_τ	0.68	0.72	0.65	0.66	0.69	**0.64**	0.76	0.69	**0.64**	2.41	1.04
Time	0.09	0.09	0.10	**0.08**	0.09	**0.08**	0.09	0.09	**0.08**	1.09	0.60

	$\tau = 0.95$										
Stats		Q-Lasso	Q-GLasso		Q-MCP	Q-GMCP		Q-SCAD	Q-GSCAD	BQR	$BAQR$
FP		5.35	5.05		1.8	**1.67**		2.57	1.94	2.25	2.34
P1		99%	**100%**		63%	96%		52%	98%	39%	75%
P2		91%	**94**%		60%	67%		66%	79%	90%	53%
AE		14.50	**2.82**		16.76	3.41		24.31	2.89	21.87	18.12
RMSE		7.02	**6.64**		55.45	30.57		61.54	18.29	30.35	30.46
QPE_τ		**0.28**	**0.28**		0.37	0.33		0.39	**0.29**	1.17	0.42
Time		0.22	0.22		**0.10**	**0.10**		0.11	0.11	1.12	6.73

Table 2.1 – Simulation results of FP, P1, P2, AE, RMSE, QPE_τ, and running time (Time) for Scenario 1, based on 100 replications. The results are reported for our GPQR, the group-variable least squares method (LS), and three single-variable methods : the Bayesian quantile regression (BQR) ; the standard quantile regression with lasso (Q-lasso), MCP (Q-MCP), and SCAD (Q-SCAD) penalties ; and boosting quantile regression (BAQR)

Table 2.1 shows that the GPQR outperforms all the other methods for almost all the statistics, except for the FP statistic, for which the BQR and BAQR have the smallest values. By contrast, because in this scenario the predictors are highly correlated, the single-variable selection methods suffer from *unstable* selection of correlated predictors [96]. This is well illustrated in the results of the P1 (especially for the BQR) and AE statistics in Table 2.1. The LS methods perform well in general, in this scenario, and surprisingly detect the heteroskedastic predictor, X_{20}, especially with the GLasso penalty, which has a high value of the P2 statistic (70%). When it is fitted for $\tau = 0.95$, the GPQR approach outperforms the LS for the P2 statistic, which reaches 94% for P2 using the GPQR and GLasso penalty. Yet, the LS results of the AE statistic reveal substantial bias for the model-parameter estimators. We also reported the Time statistic (in seconds) for computational efficiency comparison. The results of this statistic in Table 2.1 show that all the methods have comparable run-times, except for the BQR. This is not surprising because the BQR uses a Markov chain Monte Carlo (MCMC) algorithm to estimate the solution. Moreover, the BQR does not enforce sparsity, and so, it provides non-exact zero coefficient estimates and builds on the parameters' posterior distribution to provide credible intervals for variable selection. The performance of Boosting algorithms is influenced by two principal tuning parameters, including *mstop*, which is the maximum number of iterations the boosting algorithm will run for. Large *mstop* values lead to including more components. Oppositely, smaller *mstop* values lead to excluding more components [66]. Consequently, the FP and P1 statistics are sensitive to *mstop*. In Scenario 1, the optimum value of this parameter is selected via cross-validation.

Note, we have only reported the results of the GPQR with the check function approximation (2.5) in this scenario. The unreported results of the GPQR with check function approximation (2.6) are similar to those presented in Table 2.1.

Results of Scenarios 2 and 3 : The simulation results of the average, over 100 replications, of the FP, P1, P2, AE, RMSE, and QPE_τ statistics are outlined in Tables 2.2 and 2.3 for Scenarios 2 and 3, respectively. The six statistics are calculated for the GPQR approach with all suggested group penalties. In these two scenarios, we reported results for both pseudo check function approximations, (2.5) and (2.6).

							$GLLA$	$GLLA$	$GLLA$	$GLLA$
Stats	$GLasso_{1*}$	$GLasso_2$	$GMCP_1$	$GMCP_2$	$GSCAD_1$	$GSCAD_2$	MCP_1	MCP_2	$SCAD_1$	$SCAD_2$
$\tau = 0.50$										
FP	3.39	2.96	1.21	0.93	**0.74**	0.81	1.06	1.22	0.92	1.16
P1	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
P2	45%	39%	5%	**0**$\%$	**0**$\%$	**0**$\%$	5%	**0**$\%$	**0**$\%$	**0**$\%$
AE	15.73	**15.16**	16.00	15.24	15.93	15.23	15.25	15.24	15.25	15.24
RMSE	0.030	0.031	0.042	0.040	**0.027**	0.028	0.041	0.041	0.028	0.028
QPE_τ	0.29	0.30	0.22	**0.21**	**0.21**	**0.21**	0.22	**0.21**	**0.21**	0.22
$\tau = 0.75$										
FP	2.60	2.64	**1.61**	2.00	1.71	1.70	2.08	2.01	2.02	1.83
P1	100%	100%	98%	98%	100%	100%	100%	100%	100%	100%
P2	76%	81%	43%	31%	42%	39%	50%	47%	38%	46%
AE	**15.24**	15.48	15.90	15.76	15.33	15.52	15.55	15.81	**15.23**	15.44
RMSE	**0.099**	**0.100**	0.114	0.113	0.111	0.109	0.115	0.114	0.113	0.111
QPE_τ	0.184	**0.181**	0.185	0.186	0.185	0.186	0.187	0.185	0.184	0.183
$\tau = 0.95$										
FP	2.42	2.22	1.23	1.12	**1.10**	1.28	1.13	1.20	1.27	1.14
P1	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
P2	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
AE	15.30	**15.11**	15.64	15.53	15.63	15.52	15.65	15.53	15.64	15.51
RMSE	**0.564**	0.566	0.641	0.638	0.620	0.621	0.644	0.643	0.617	0.623
QPE_τ	**0.082**	0.085	0.092	0.093	0.089	0.090	0.088	0.090	0.091	0.088

Table 2.2 – Simulation results of FP, P1, P2, AE, RMSE, and QPE_τ for Scenario 2 based on 100 replications. The six statistics are calculated for the GPQR approach with all suggested group penalties. * Subscripts 1 and 2 indicate that the GPQR is fitted using the pseudo check functions (2.5) and (2.6), respectively.

							$GLLA$	$GLLA$	$GLLA$	$GLLA$
Stats	$GLasso_{1^*}$	$GLasso_2$	$GMCP_1$	$GMCP_2$	$GSCAD_1$	$GSCAD_2$	MCP_1	MCP_2	$SCAD_1$	$SCAD_2$
					$\tau = 0.50$					
FP	4.64	3.92	0.79	0.74	0.88	0.63	0.72	**0.69**	0.75	0.71
P1	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
P2	58%	67%	**0**$\%$	**0**$\%$	**0**$\%$	**0**$\%$	**0**$\%$	**0**$\%$	**0**$\%$	**0**$\%$
AE	14.96	14.77	15.22	14.33	15.21	**14.30**	14.35	14.33	14.32	14.32
RMSE	**0.404**	0.410	0.430	0.428	0.535	0.541	0.439	0.435	0.529	0.530
QPE_τ	0.604	0.625	**0.542**	0.549	0.562	0.566	0.550	0.545	0.571	0.569
					$\tau = 0.75$					
FP	14.6	16.1	2.05	2.12	1.41	**1.37**	1.99	1.87	1.44	1.67
P1	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
P2	70%	73%	17%	38%	30%	57%	20%	53%	27%	63%
AE	**22.9**	23.1	24.1	24.0	24.1	24.1	24.1	24.0	24.0	24.1
RMSE	1.121	1.117	0.723	0.727	**0.681**	0.687	0.724	0.732	0.684	0.678
QPE_τ	0.509	0.512	0.454	**0.451**	0.460	0.465	0.517	0.522	0.466	0.461
					$\tau = 0.95$					
FP	12.23	11.65	2.27	2.32	1.43	1.51	2.36	2.52	**1.37**	1.41
P1	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
P2	93%	90%	91%	**97**$\%$	93%	94%	83%	**97**$\%$	94%	93%
AE	22.3	**22.0**	24.2	24.3	24.3	24.3	24.2	24.3	24.3	24.2
RMSE	10.95	11.17	**4.342**	4.523	4.721	4.635	4.882	4.404	5.14	5.22
QPE_τ	0.247	0.251	0.199	0.209	0.202	0.207	**0.197**	0.200	0.203	0.201

Table 2.3 – Simulation results of FP, P1, P2, AE, RMSE and QPE_τ for scenario 3, based on 100 replications. The six statistics are calculated for the GPQR approach with all suggested group penalties. * Subscripts 1 and 2 indicate that the GPQR is fitted using the pseudo check functions (2.5) and (2.6), respectively.

Tables 2.2 and 2.3 show that all models select the true active groups, with the P_1 statistic always around 100%. By contrast, the FP statistic reveals that the GPQR with the GMCP and GSCAD penalties tends to provide less false positives than the GLasso.

The P2 statistic shows how many times the heterogeneous variable, X_1, is selected in each model fit. For $\tau = 0.5$, it is expected that X_1 will not be selected because it has no effect on the center of y. However, as τ increases, the proportion of selecting X_1 increases for all approaches. For $\tau = 0.75$, P2 ranges between $(17\%, 73\%)$, and when $\tau = 0.95$, P_2 is approximately around 100%.

2.4.3 Checking the KKT conditions

In this section we test the accuracy of the proposed algorithms' solutions by checking their numerical KKT conditions, defined in Section 2.7.7 of the Supplementary material. More precisely, because we are using the majorization-minimization principle to solve (2.7), the aim of this scenario is to evaluate if the minimizer $\widehat{\boldsymbol{\beta}}$, obtained by solving (2.9), satisfies the first-order optimality conditions (i.e. the KKT conditions) for the objective function (2.7). This ensures that the GPQR algorithms converge to the desired solution. Derivation of the KKT conditions is given in more detail in Sections 6 and 7 of the Supplementary material.

Setting of Scenario 4 : We designed this simulation scenario following a numerical example suggested in [105]. First, we simulated q initial predictors, $X_1, X_2, \ldots, X_q$, from a centered multivariate normal distribution with a compound symmetry correlation matrix, $\boldsymbol{\Gamma}$, with $\boldsymbol{\Gamma}_{jj'} = \rho$, for all $j \neq j'$. We then generated the response following the regression model

$$Y = \sum_{j=1}^{q} \left(\frac{2}{3} X_j - X_j^2 + \frac{1}{3} X_j^3 \right) \beta_j + \epsilon,$$

where $\beta_j = (-1)^j \exp\{-(2j-1)/20\}$, the error term ϵ is generated from $N(0, \sigma^2)$, and σ^2 is chosen so that the signal-to-noise ratio (SNR) is 3 (i.e. $SNR = \|\mathbf{X}\boldsymbol{\beta}\|_2/\sqrt{n}\sigma$). We considered $\{X_j, X_j^2, X_j^3\}$ as a group when fitting all the proposed models. Thus, the final design matrix of the predictors has $p = 3q$ columns. In this scenario, we set two values of $q = 1000, 3000$, and we fixed $n = 100$. For all group penalties, we fitted three conditional quantile regression models, with $\tau = 0.50, 0.75, 0.95$.

For all algorithms, we calculated the number of coefficients among p coefficients that violated the KKT condition check at each λ value. This number is then averaged over the 100 λ values. We repeated this process 10 times on 10 independent datasets. Table 2.4 reports the results that are averaged over 100 λ values and averaged over the 10 independent runs.

Table 2.4 shows that all exact group-penalized methods have a zero-violation count, except the GSCAD which has 1 violation. The GPQR with the GLLA penalty also has small violation counts. Thus, one can argue that all the proposed approaches are accurate algorithms that pass the KKT checks without severe violation.

	$(n,p) = (100, 3000)$			$(n,p) = (100, 9000)$		
Method	$\tau = 0.50$	$\tau = 0.75$	$\tau = 0.95$	$\tau = 0.50$	$\tau = 0.75$	$\tau = 0.95$
$GLasso_1$	0	0	0	0	0	0
$GLasso_2$	0	0	0	0	0	0
$GMCP_1$	0	0	0	0	0	0
$GMCP_2$	0	0	0	0	0	0
$GSCAD_1$	0	0	1	0	0	0
$GSCAD_2$	0	0	1	0	0	0
$McpGLLA_1$	8	4	2	10	5	3
$McpGLLA_2$	8	4	2	10	5	3
$ScadGLLA_1$	3	1	1	3	2	1
$ScadGLLA_2$	3	1	1	3	2	1

Table 2.4 – The reported numbers are the average number of coefficients among the p coefficients that violated the KKT condition check using the GPQR with the GLasso, GMCP, GSCAD, and GLLA penalties. Subscripts 1 and 2 indicate that the GPQR is fitted using the check functions (2.5)) and (2.6) respectively. Results are averaged over the λ sequence of 100 values and averaged over 10 independent runs.

2.5 Real data

2.5.1 Gene-based analysis of Alzheimer's Disease Neuroimaging Initiative (ADNI) data

The data used in the preparation of this article were obtained from the ADNI database (`adni.loni.usc.edu`). The ADNI was launched in 2003 as a public-private partnership, led by principal investigator Michael W. Weiner, MD. The ADNI's primary goal is to test whether serial magnetic resonance imaging (MRI), positron emission tomography (PET), other biological markers, and clinical and neuropsychological assessment can be combined to measure the progression of mild cognitive impairment (MCI) and early Alzheimer's disease (AD).

It is known that the pathogenic relevance in AD presents a decrease of the biomarker cerebrospinal fluid amyloid-$\beta 42$ (CSF $A\beta 42$) levels and an increase in the biomarker cerebrospinal fluid total tau (CSF T-tau) levels [59]. Moreover, it is known that individuals with a family history of AD have a higher risk for AD than those without a family history. This reveals that underlying genetic factors may play a key role in AD [37]. In fact, in several GWAS, the two biomarkers CSF $A\beta 42$ and CSF T-tau have been reported to be associated with several SNPs falling within or near the genes *APOE*, *TOMM40* and *APOC1*, located in Chromosome 19 [49].

To illustrate the use of our framework in GWAS, we conducted a gene-based association study using the GPQR approaches in the ADNI cohort. More precisely, we considered the CSF T-tau/$A\beta 42$ ratio as an AD imaging quantitative trait (response) on 442 subjects. As predictors, we used single-nucleotide polymorphisms (SNPs) falling within a genomic region of 629 kilobase pairs located around the three genes of interest (*APOE*, *TOMM40*, and *APOC1*). This region results in $K = 17$ genes/groups with observed genotypes of 1162 SNPs of the ADNI samples. We then assigned the SNPs to genes based on their base-pair coordinates. We used the `biomaRt` R package [22] to extract the genes' start-end genomic coordinates.

This analysis aims to replicate/select the three genes of interest as associated with the response variable, CSF T-tau/$A\beta 42$, using the GPQR framework and compare its performance with that of group penalized LS methods. In our analyses, all models were fitted with 17 penalized genes/groups, and we adjusted for sex, age, and diagnostic without penalization because such covariates are known to be potential confounding factors for AD.

We conducted two analyses for this data. First, we fitted the GPQR and group penalized LS methods for all 442 analyzed subjects with five-fold CV to obtain a better model estimation. In the second analysis, we aimed to evaluate the prediction performance of the methods. Thus, we randomly divided the data into a training sample of two-thirds of the observations and the remainder making up the test data. The model is fitted to the training data and the prediction errors are calculated on the test data. The tuning parameters were selected by five-fold CV on the training data. The whole procedure was repeated 100 times and we reported the empirical distribution of the prediction-errors and model-size statistics using box plots, for all methods. The model-size statistic is defined as the number of significant genes. Figures 2.3, 2.4, and Table 2.5 of the main manuscript summarize the results from the gene-based association study of the ADNI cohort in the center of the response variable (i.e. mean and median).

In Figure 2.3, the LS methods tend to select the null model, whereas the median regressions (QR with $\tau = 0.5$) select a model with a moderate number of significant genes. Interestingly, at least two of the three genes of interest are selected as active groups by the GPQR using the five-fold CV criterion (pink vertical line). This is also in agreement with the results of Figure 2.4 (left panel) which shows the distribution of the model-size statistic for the 100 replications of the second analysis. In Table 2.5, when comparing the methods based on the selection of the genes of interest (*APOE*, *TOMM40*, and *APOC1*), we notice that the GPQR with the GLasso penalty (Q-GLasso) is better than the OLS with the GLasso penalty (LS-GLasso). In fact, the proportions of the three genes detected by the Q-GLasso are significantly larger than the LS-GLasso.

By contrast, the right panel of Figure 2.4 shows an improvement in predictive performance when using the QR approaches. This is also in accordance with the results reported in Figure 2.3 and Table 2.5.

Genes	LS-GLasso	Q-GLasso	LS-GMCP	Q-GMCP	LS-GSCAD	Q-GSCAD
APOC1	8.0	81.6	3.5	23.1	8.0	20.4
TOMM40	0.0	26.5	0.0	0.6	0.0	0.0
APOE	43.4	81.8	19.0	34.7	45.1	20.8

Table 2.5 – The number of times (in $\%$) the genes *APOE*, *TOMM40*, and *APOC1* are selected based on 100 replications, for the ADNI data. The group quantile methods are fitted with $\tau = 0.5$

We implemented further analyses of the ADNI data to investigate the effects of the important three genes

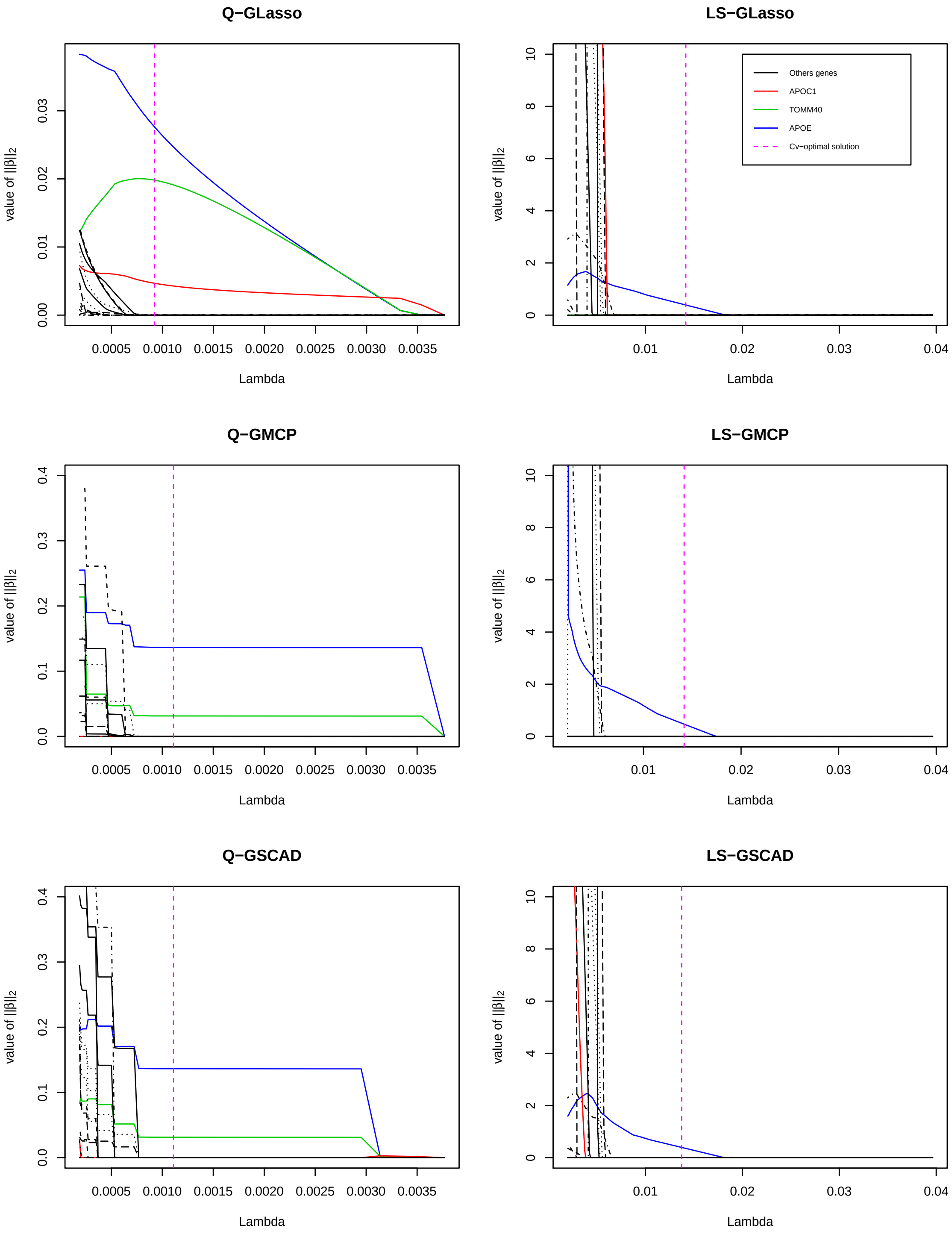


Figure 2.3 – On the left from top to bottom, the L2-norm of the coefficient paths of the Q-GLasso, Q-GMCP, and Q-GSCAD, respectively, with $\tau = 0.5$, are shown as a function of the tuning parameter λ. On the right from top to bottom are the coefficient paths of the same group methods with LS.

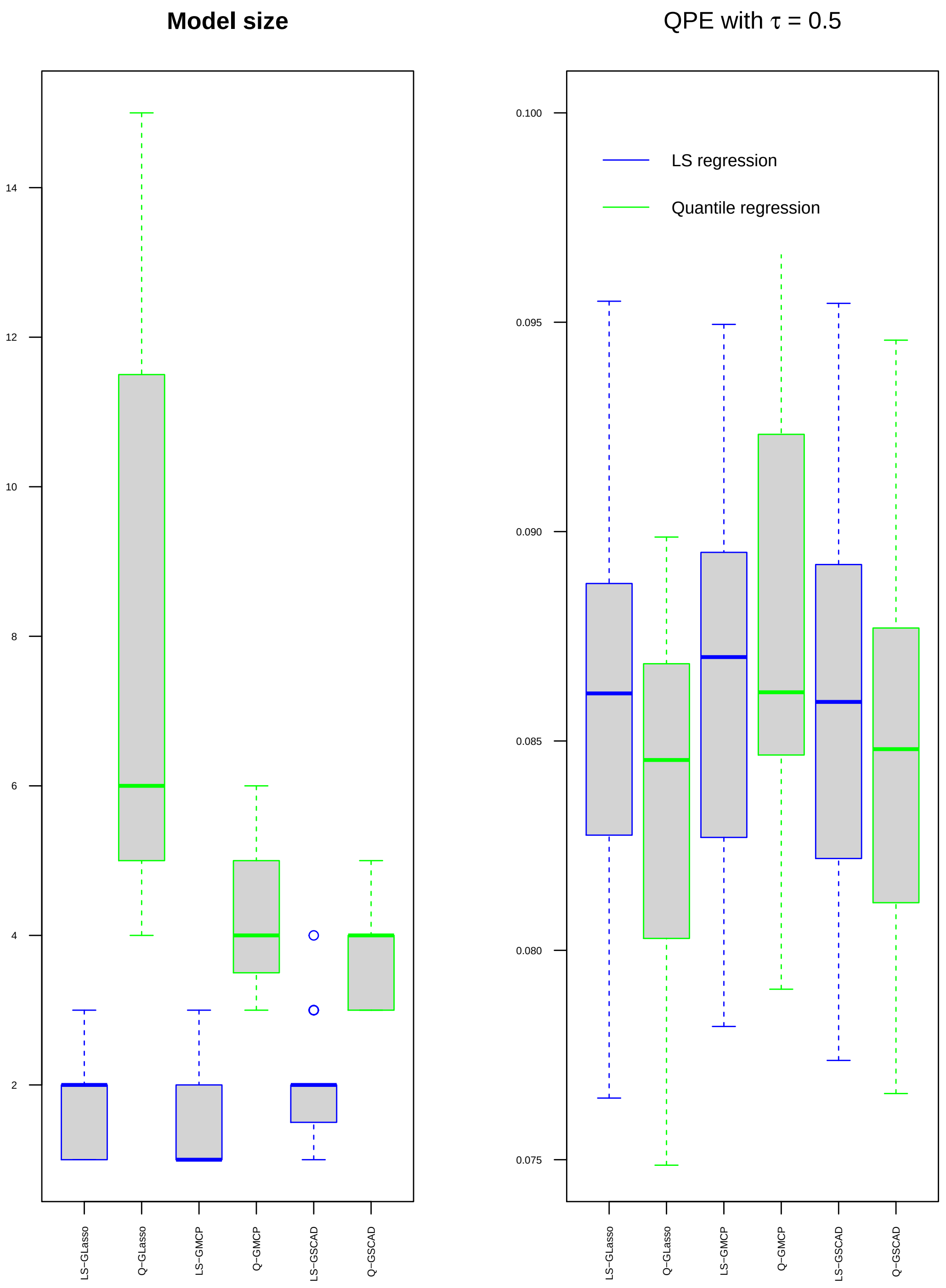


Figure 2.4 – Comparison of the number of selected genes (Model Size) and the prediction accuracy (QPE_τ) based on 100 replications, for the ADNI data. The group quantile methods are fitted with $\tau = 0.5$.

in the lower and upper tails of the conditional distribution of the response variable. Thus, the GPQR model was fitted for four additional locations, $\tau \in \{0.1, 0.25, 0.75, 0.9\}$. Figures $S.2$ and $S.3$ of the Supplementary materials summarizes the results for this analysis.

Figure $S.2$ shows $\|\hat{\boldsymbol{\beta}}_k\|_2$ for the three important genes in the ordinate axis as a function of a grid of values of $\tau \in \{0.1, 0.25, 0.5, 0.75, 0.9\}$. For each value of τ, we used the 5-fold CV procedure to obtain the optimal λ_τ; Figure $S.2$ reports the GPQR solution with optimal λ_τ for all 442 subjects of the ADNI cohort. Although one would expect that the genes' effects could be more important for subjects with higher levels of the response variable (i.e. higher quantiles), the results in Figure $S.2$ show no significant evidence of this expectation. This might be explained by the presence of both relevant and noisy SNPs within the same gene, which can add some estimation instability to the overall gene effect using the GPQR. Sparse-group selection methods, which achieve both group selection and single-variable selection within each group might be suitable in such situations [83, 31].

Figure $S.3$ highlights the results of the L2-norm of the coefficient paths of the Q-GLasso, Q-GMCP, and Q-GSCAD, respectively, as a function of the tuning parameter λ, for $\tau = 0.25$ and 0.75. It shows similar patterns to those in the analysis of the GPQR with the GLasso and $\tau = 0.5$. However, the GPQR with the GMCP (or GSCAD) behaves differently for $\tau = 0.25$ and $\tau = 0.75$. In fact, fitting the GPQR with group MCP/SCAD detects *APOE* and *APOC1* for $\tau = 0.75$; but, when the $0.25-$th quantile model is fitted, it selects *APOE* and *TOMM40*.

For more investigation, we also analyzed the 0.25-th and 0.75-th quantiles, similar to the second analysis of the median/center regression (i.e. we conducted 0.25-th and 0.75-th quantile regression models for 100 random training/test replications of the ADNI cohort). Table $S.1$ of the Supplementary materials summarizes the results for this analysis.

The results are based on 100 random training/test data replications of the ADNI cohort. Each replication consists of a random split of the whole cohort dataset to training (67% observations) and test (33% observations) datasets. The model is fitted to the training data to choose the optimal solution and the tuning parameter using five-fold CV. Then, the prediction performance is evaluated in the test data. Table $S.1$ outlines the average, over 100 replications, of the following three statistics : 1) the number of times (in proportion) the three genes of interest (*APOE*, *TOMM40*, and *APOC1*) were selected, 2) the quantile-based error prediction (QPE_τ), and 3) the model size (Size) statistic.

Table $S.1$ shows that the GPQR behaves relatively differently when looking for the effects of the genes in the different locations of the response conditional-distribution, particularly for the model-size statistic. This table shows that the proportions of the three genes detected for $\tau = 0.25$ are larger than for $\tau = 0.75$.

Table $S.1$ also shows inconsistency in the results of the three penalties for the same location, except for the QPE_τ statistic, which is stable across different specifications. This might be explained, on one hand, by the known sensitivity of the lasso-type penalized regression models to the five-fold assignment used in the CV procedure, [81]. On the other hand, a good tuning parameter choice depends on the unknown parameter σ^2 which is the homogeneous noise variance in linear models [7]. For the ADNI data, more knowledge about the standard deviation is necessary and this needs more data investigation. The conclusion chapter emphasizes this issue and provides tentative solutions.

2.5.2 Gene-based analysis of the DNA methylation data near the *BLK* gene

This section illustrates the GPQR approach performance for binary classification using DNA methylation around the *BLK* gene, located in chromosome 8, to detect differentially methylated regions (DMRs). DMRs refer to genomic regions with significantly different methylation levels between two groups of samples (e.g. : case-controls). The data consists of methylation levels of 5,986 cytosine-guanine dinucleotides (i.e. CpG sites) within a genomic region of 2 million base pairs (i.e. 2 Mb pairs located in Chromosome 8, ranging between positions Chr8-10321522 and Chr8-12391296). The methylation levels in these CpG sites (predictors) are measured in 40 samples using bisulfite sequencing [57]. Each sample corresponds to one of three cell types : B cells (8 samples), T cells (19 samples), and Monocytes (13 samples). These samples are derived from whole blood collected from a cohort of healthy individuals from Sweden. This genomic region is known to be hypomethylated near the *BLK* gene in B-cells, compared to other cell types [35].

We first coded the cell types as $y = \{0, 1\}$ variable, with $y = 1$ corresponding to B-cells and $y = 0$ corresponding to T- and Monocyte-cell types. To build groups of predictors (CpGs sites), we proceeded in a similar way to Section 2.5.1. That is, we extracted the start-end genomic positions of all genes belonging to the 2Mb region. $K = 36$ genes fall within this region. Then, we used prior information about the genomic position of each CpG site and assigned each CpG to a corresponding gene based on its base pair coordinate. More precisely, if the genomic position of a CpG site is between the start and end positions of a gene, we considered that the CpG belongs to this gene/group. The CpG assignment procedure is implemented in the `biomaRt` R package. In total, 4,427 of all the 5,986 CpG sites spread over the 36 genes. The size of the studied groups ranges between 1 and 756, with 398 CpG sites falling between the start-end coordinates of the *BLK* gene.

The $\{0, 1\}$ response variable is then fitted by the group penalized LS and GPQR methods with $\tau = 0.5$.

Given the binary nature of the response variable, we also compared the proposed methods with support vector machine (SVM) and logistic regression with a group lasso penalty. Both methods are implemented in the `gglasso` R package [104].

This analysis aims to test the performance of our methods in detecting the group of CpG sites belonging to the *BLK* gene as a DMR for the $0-1$ response, and to test the power of the GPQR in classification. The classification function is $\mathbb{1}(\text{fitted value} > 0.5)$, where $\mathbb{1}(A)$ is the indicator function which equals 1 if A is true and 0 if A is false.

In Figure 2.5, the x and y axes correspond respectively to the genomic position, say t_j, of the j-th CpG site and the coefficient value $(\hat{\beta}_j)_{1\leq j\leq 4427}$ of the optimal solution that is obtained using five-fold CV. More precisely, each blue dot point in Figure 2.5 represents a pair $(t_j, \hat{\beta}_j)$, for $j = 1, \ldots, 4427$. As we can see, the region around 11.3 Mb with size 150kb (i.e., the region delimited by the two vertical lines) is detected/selected by the quantile, SVM, and logistic regression methods, but not with the LS approach. This region is known to be the DMR between the DNA methylation profiles of the B-cells and T/Mono cells [92]. These results are also in agreement with [57]'s analysis.

In a second analysis of this DNA methylation data, we randomly divided the data into a training sample of 30 observations with the remainder making up the test data. The model is fitted to the training data and the misclassification error rate (MER) is calculated on test data. The MER is defined as the ratio of the number of misclassified observations to the total number of observations. The tuning parameters are selected by five-fold CV on the training data, and τ is fixed to 0.5 for all our algorithms in this analysis. The whole procedure is repeated 100 times. The results of this analysis are shown in Figures 2.6 and 2.7.

In Figure 2.6, the y axis represents the rate of selection of each gene, over 100 replications, of the methylation data analysis. Each segment represents a gene, with large segments corresponding to genes containing a high number of CpG sites (i.e. large genes), and vice versa. The DMR region is always selected by the quantile regression and SVM methods ; it is often selected by logistic regression, but the region is never selected by the LS methods. Furthermore, our proposed quantile approaches and SVM outperform the LS approaches and the logistic regression in terms of classification prediction accuracy. This is illustrated by the results of the MER statistic in Figure 2.7.

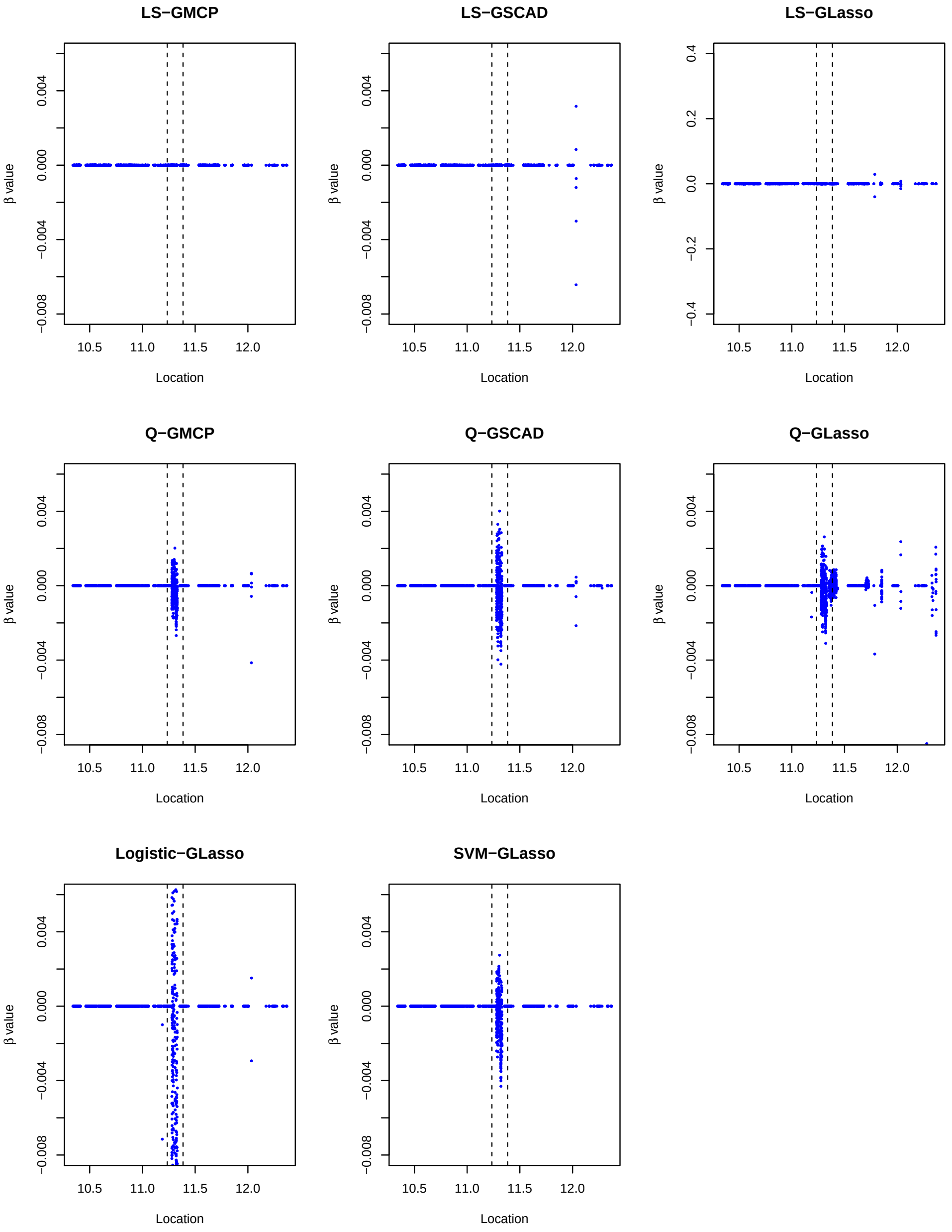


Figure 2.5 – At the top from left to right, the optimal value (five-fold CV) for the regression coefficients of the LS-methods with the three group penalties (GMCP, GSCAD, and GLasso) are shown as a function of the genomic position. The middle from left to right shows the coefficient values of the same group penalties for the quantile regression, with $\tau = 0.5$. The bottom row shows the coefficient value of the SVM and logistic regression with the GLasso penalty. The x and y axes correspond respectively to the genomic position, t_j, of the j-th CpG site and the coefficient value $(\hat{\beta}_j)_{1 \leq j \leq 4427}$ of the optimal solution.

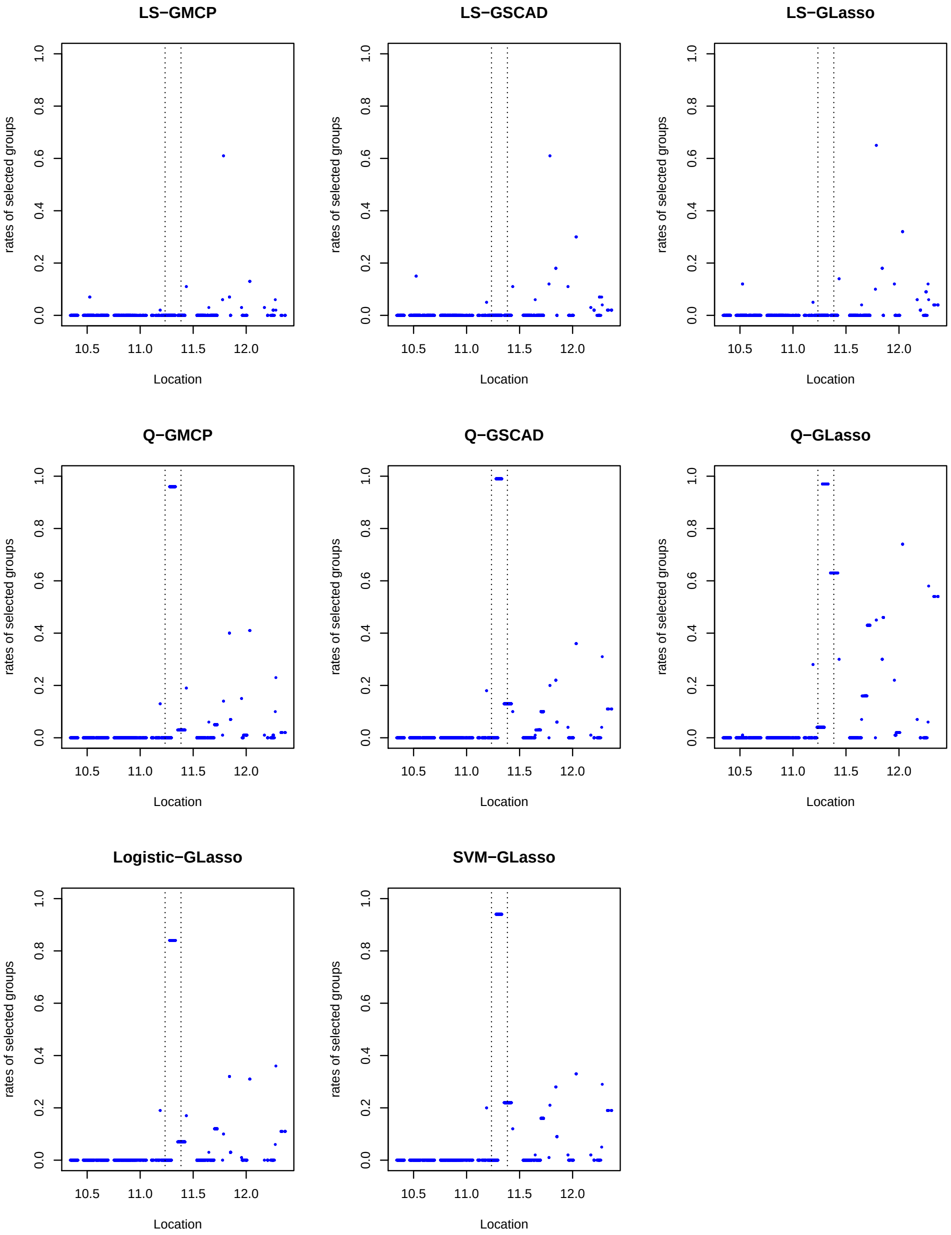


Figure 2.6 – Comparison of the proportion of selected genes for the DNA methylation data. At the top from left to right, the proportion of LS-GMCP, LS-GSCAD, and LS-GLasso are shown as a function of the genomic position. The middle from left to right shows the proportion of the same group penalties for the quantile regression, with $\tau = 0.5$. The bottom row shows the proportion of the SVM and logistic regression with the GLasso penalty. The x and y axes correspond respectively to the genomic position, t_j, of the j-th CpG site and the proportion of non-zero $(\hat{\beta}_j)_{1 \leq j \leq 4427}$.

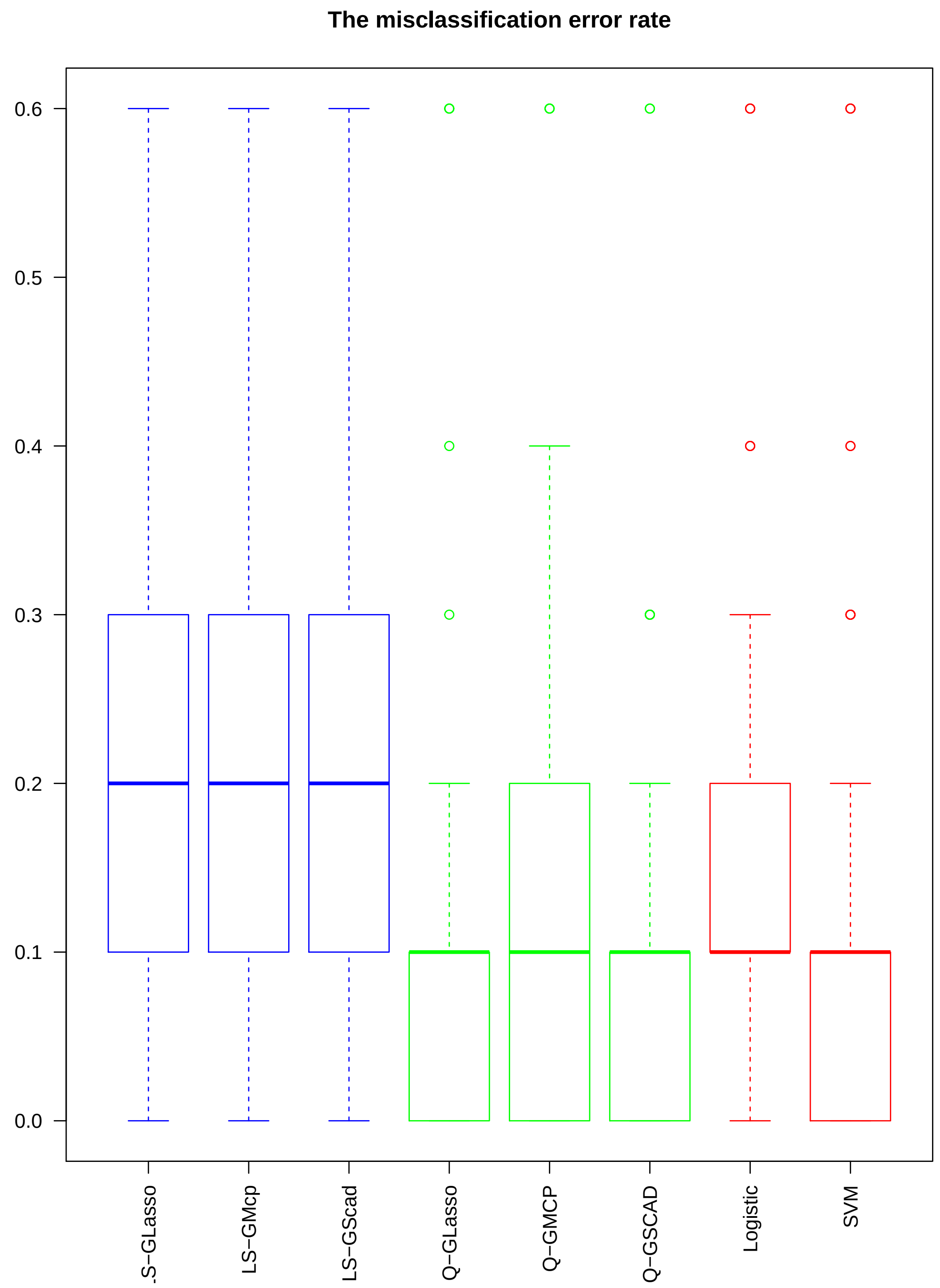


Figure 2.7 – Comparison of the MER for the DNA methylation data. The MER of the GPQR, LS-methods, and logistic/SVM are plotted in blue, green and red, respectively.

2.6 Discussion

In this work, we have proposed a unified and computationally-efficient block descent algorithm for solving the group penalized quantile regression (GPQR) in high-dimensional settings. GPQR fits quantile regression with the most appealing group penalties, namely, the group lasso penalty, the group non-convex penalties (SCAD and MCP) and their local approximations. The GPQR allows for the selection of important (heterogeneous) groups of predictors and provides estimates of their effects on the response simultaneously.

We provided a detailed theoretical justification of the linear convergence rate property of the GPQR with group lasso penalty. Moreover, simulation studies have confirmed that the quantile regression performs better for group variable-selection than single-variable quantile regression methods and group-variable selection least-squares approaches in terms of prediction accuracy, variable selection, and detection of heteroscedasticity.

2.7 Supplementary Material

2.7.1 Proof of Proposition 2.1

From Proposition 1 of [72], we have

$$-\delta\kappa \leq \Psi_\tau(u) - \rho_\tau(u) \leq \delta\kappa \quad \forall u \in \mathbb{R},$$

where the constant $\kappa = sup(\tau, 1-\tau)/2$ or $sup(\tau^2, (1-\tau)^2)/2$. This yields to the following inequalities

$$-\delta\kappa + R(\boldsymbol{\beta}) \leq R_\delta(\boldsymbol{\beta}) \leq \delta\kappa + R(\boldsymbol{\beta}). \tag{2.17}$$

Let $\hat{\boldsymbol{\beta}}$ be the unique minimizer of $R(\boldsymbol{\beta})$ in (1), then we have

$$\begin{aligned}
\inf_{\boldsymbol{\beta}} R(\boldsymbol{\beta}) &\leq R(\hat{\boldsymbol{\beta}}(\delta)) \\
&\overset{(a)}{\leq} R_\delta(\hat{\boldsymbol{\beta}}(\delta)) + \delta\kappa \\
&\overset{(b)}{\leq} R_\delta(\hat{\boldsymbol{\beta}}) + \delta\kappa \\
&\overset{(c)}{\leq} R(\hat{\boldsymbol{\beta}}) + \delta\kappa + \delta\kappa \\
&\leq \inf_{\boldsymbol{\beta}} R(\boldsymbol{\beta}) + 2\delta\kappa.
\end{aligned}$$

Inequality (a) is due to the first inequality in (2.17), inequality (b) is due to $\hat{\boldsymbol{\beta}}(\delta)$ is the minimizer of $R_\delta(\boldsymbol{\beta})$ and inequality (b) is due to the second inequality in (2.17). This ends the proof of Proposition 2.1.

2.7.2 Proof of Proposition 2.2

Preuve. Following [72], we can show that the smooth quantile loss function $\Psi_\tau(.)$ has a Lipschitz continuous derivative $\Psi'_\tau(.)$, i.e.

$$\begin{aligned}
\textit{when} \quad \Psi_\tau = \Psi^{(1)}_{\tau,\delta}: \quad |\Psi'_\tau(u) - \Psi'_\tau(v)| &\leq \frac{\max(\tau, 1-\tau)}{\delta}|u-v| \quad \forall u, v \in \mathbb{R}, \\
\textit{when} \;\; \Psi_\tau = \Psi^{(2)}_{\tau,\delta}: \qquad |\Psi'_\tau(u) - \Psi'_\tau(v)| &\leq \frac{1}{\delta}|u-v| \quad \forall u, v \in \mathbb{R}.
\end{aligned}$$

Thus, we have

$$|\Psi'_\tau(u) - \Psi'_\tau(v)| \leq c|u-v| \quad \forall u, v \in \mathbb{R}, \tag{2.18}$$

where $c = \frac{\max(\tau,1-\tau)}{\delta}$ for $\Psi^{(1)}_{\tau,\delta}$ and $c = \frac{1}{\delta}$ for $\Psi^{(2)}_{\tau,\delta}$.

For $\boldsymbol{\beta}_k$ and $\tilde{\boldsymbol{\beta}}_k$, let $\mathbf{V}_k = \boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k$ and define $g(t) = L(\tilde{\boldsymbol{\beta}}_k + t\mathbf{V}_k, \tilde{\boldsymbol{\beta}}_{-k})$. Thus, we have $g(0) = L(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k})$, $g(1) = L(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k})$.

By the mean value theorem, $\exists a \in (0,1)$ such that

$$g(1) = g(0) + g'(a) = g(0) + g'(0) + (g'(a) - g'(0)). \tag{2.19}$$

Since we have

$$g'(t) = n^{-1}\sum_{i=1}^{n} \mathbf{x}_{i,k}^\top \mathbf{V}_k \Psi'_\tau(y_i - \mathbf{x}_{i,-k}^\top \tilde{\boldsymbol{\beta}}_{-k} - \mathbf{x}_{i,k}^\top \tilde{\boldsymbol{\beta}}_k + t\mathbf{x}_{i,k}^\top \mathbf{V}_k))$$

it follows that $g'(0) = (\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top \nabla_k L(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k})$, and thus, one can write

$$\begin{aligned}
|\, g'(a) - g'(0)| &= |n^{-1}\sum_{i=1}^{n} \mathbf{x}_{ik}^\top \mathbf{V}_k [\Psi'_\tau(y_i - \mathbf{x}_{i,-k}^\top \tilde{\boldsymbol{\beta}}_{-k} - \mathbf{x}_{ik}^\top(\tilde{\boldsymbol{\beta}}_k + a\mathbf{V}_k)) - \Psi'_\tau(y_i - \mathbf{x}_{i,-k}^\top \tilde{\boldsymbol{\beta}}_{-k} - \mathbf{x}_{ik}^\top \tilde{\boldsymbol{\beta}}_k)]| \\
&\leq n^{-1}\sum_{i=1}^{n} |\mathbf{x}_{ik}^\top \mathbf{V}_k| |\Psi'_\tau(y_i - \mathbf{x}_{i,-k}^\top \tilde{\boldsymbol{\beta}}_{-k} - \mathbf{x}_{ik}^\top(\tilde{\boldsymbol{\beta}}_k + a\mathbf{V}_k)) - \Psi'_\tau(y_i - \mathbf{x}_{i,-k}^\top \tilde{\boldsymbol{\beta}}_{-k} - \mathbf{x}_{ik}^\top \tilde{\boldsymbol{\beta}}_k)| \\
&\overset{(a)}{\leq} n^{-1}\sum_{i=1}^{n} |\mathbf{x}_{ik}^\top \mathbf{V}_k| c |a\mathbf{x}_{ik}^\top \mathbf{V}_k| \\
&\leq cn^{-1}\sum_{i=1}^{n} \|\mathbf{x}_{ik}^\top \mathbf{V}_k\|^2 \\
&\leq cn^{-1}\mathbf{V}_k^\top \mathbf{X}_k^\top \mathbf{X}_k \mathbf{V}_k.
\end{aligned}$$

Inequality (a) is due to equation (2.18).Using the last inequality and (2.19) leads to the following inequality

$$\begin{aligned}
L(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k}) \quad \leq \quad & L(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}) + (\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top \nabla_k L(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}) + \\
& \qquad cn^{-1}(\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top \mathbf{X}_k^\top \mathbf{X}_k (\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k).
\end{aligned}$$

This ends the proof of Proposition 2.2. □

2.7.3 The convergence analysis of Algorithm 1 : proof of Theorem 2.3

Some properties of the smooth quantile loss function, $L(\boldsymbol{\beta}) = n^{-1}\mathbf{1}_n^\top \Psi_\tau(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})$, are used in the steps of the Theorem's proof ; they are given first. The smooth quantile check function, Ψ_τ, can be either $\Psi^{(1)}_{\tau,\delta}$ or $\Psi^{(2)}_{\tau,\delta}$ and $\mathbf{1}_n \in \mathbb{R}^n$ denotes the vector of all ones.

Since we have

$$\nabla L(\boldsymbol{\beta}) = -n^{-1}\mathbf{X}^\top \Psi'_\tau(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}),$$

then, using (2.18), it follows that

$$\begin{aligned}\|\nabla L(\boldsymbol{\beta}) - \nabla L(\boldsymbol{\beta}')\| &= n^{-1}\|\mathbf{X}^\top(\Psi'_\tau(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) - \Psi'_\tau(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}'))\| \\ &\leq n^{-1}\|\mathbf{X}\|\|\Psi'_\tau(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) - \Psi'_\tau(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}')\| \\ &\leq cn^{-1}\|\mathbf{X}\|\|\mathbf{X}(\boldsymbol{\beta} - \boldsymbol{\beta}')\| \\ &\leq cn^{-1}\|\mathbf{X}\|^2\|\boldsymbol{\beta} - \boldsymbol{\beta}'\| \\ &\leq \gamma\|\boldsymbol{\beta} - \boldsymbol{\beta}'\|, \qquad \forall \boldsymbol{\beta}, \boldsymbol{\beta}' \in \mathbb{R}^p,\end{aligned}$$

where γ is the largest eigenvalue of $cn^{-1}\mathbf{X}^\top\mathbf{X}$, and $c = \frac{\max(\tau, 1-\tau)}{\delta}$ for $\Psi^{(1)}_{\tau,\delta}(u)$ and $c = \frac{1}{\delta}$ for $\Psi^{(2)}_{\tau,\delta}(u)$. This implies that the gradient of $L(\cdot)$ is uniformly Lipschitz continuous with Lipschitz constant γ. When restricted to each block, we have

$$\nabla_k L(\boldsymbol{\beta}) = -n^{-1}\sum_{i=1}^n \Psi'_\tau(y_i - \mathbf{x}_i^\top\boldsymbol{\beta})\mathbf{x}_{ik} = -n^{-1}\mathbf{X}_k^\top \Psi'_\tau(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}),\ k = 1, \ldots, K.$$

Thus, we have

$$\begin{aligned}\|\nabla_k L(\mathbf{u}_k; \boldsymbol{\beta}_{-k}) - \nabla_k L(\mathbf{v}_k, \boldsymbol{\beta}_{-k})\| &\leq n^{-1}c\|\mathbf{X}_k\|^2\|\mathbf{u}_k - \mathbf{v}_k\| \\ &\leq \gamma_k\|\mathbf{u}_k - \mathbf{v}_k\|, \qquad \forall \mathbf{u}_k, \mathbf{v}_k \in \mathbb{R}^{p_k},\ \forall k \in \{1, \ldots, K\},\end{aligned}$$

where γ_k is the largest eigenvalue of $cn^{-1}\mathbf{X}_k^\top\mathbf{X}_k$. This implies that the gradient of $L(\cdot)$ is block-wise uniformly Lipschitz continuous with Lipschitz constant γ_k.

Moreover, for group k, let $u_k(\cdot; \boldsymbol{\beta}_{-k})$ be the quadratic majorization function of $L(., \boldsymbol{\beta}_{-k})$, at $\boldsymbol{\beta}_k$, defined as follows

$$u_k(\mathbf{v}_k; \boldsymbol{\beta}_{-k}) = L(\boldsymbol{\beta}) + \langle \nabla_k L(\boldsymbol{\beta}),\ \mathbf{v}_k - \boldsymbol{\beta}_k\rangle + \frac{\gamma_k}{2}\|\mathbf{v}_k - \boldsymbol{\beta}_k\|^2.$$

Note that we omit the dependency u_k on $\boldsymbol{\beta}_k$ to ease exposition. The function $u_k(\mathbf{v}_k; \boldsymbol{\beta}_{-k})$satisfies the following conditions

1. $u_k(\boldsymbol{\beta}_k; \boldsymbol{\beta}_{-k}) = L(\boldsymbol{\beta})$;
2. $u_k(\mathbf{v}_k; \boldsymbol{\beta}_{-k}) \geq L(\mathbf{v}_k, \boldsymbol{\beta}_{-k}),\ \text{for}\ \mathbf{v}_k \neq \boldsymbol{\beta}_k$;
3. $\nabla u_k(\boldsymbol{\beta}_k; \boldsymbol{\beta}_{-k}) = \nabla_k L(\boldsymbol{\beta}_k, \boldsymbol{\beta}_{-k}).$

We can verify that $u_k(\cdot; \boldsymbol{\beta}_{-k})$ is strongly convex :

$$\begin{aligned} u_k(\mathbf{u}_k; \boldsymbol{\beta}_{-k}) &\geq u_k(\mathbf{v}_k; \boldsymbol{\beta}_{-k}) + \big\langle \nabla u_k(\mathbf{v}_k; \boldsymbol{\beta}_{-k}), \mathbf{u}_k - \mathbf{v}_k \big\rangle \\ &+ \tfrac{\gamma_k}{2}\|\mathbf{u}_k - \mathbf{v}_k\|^2 \qquad \forall \mathbf{u}_k, \mathbf{v}_k \in \mathbb{R}^{p_k}, \forall k. \end{aligned} \tag{2.20}$$

Further, we have

$$\begin{aligned} \|\nabla u_k(\mathbf{v}_k; \boldsymbol{\beta}_{-k}) - \nabla u_k(\mathbf{v}_k; \boldsymbol{\beta}'_{-k})\| &= \|\nabla_k L(\boldsymbol{\beta}) - \nabla_k L(\boldsymbol{\beta}') + \gamma_k(\boldsymbol{\beta}_k - \boldsymbol{\beta}'_k)\| \\ &\leq \|\nabla_k L(\boldsymbol{\beta}) - \nabla_k L(\boldsymbol{\beta}')\| + \gamma_k \|\boldsymbol{\beta}_k - \boldsymbol{\beta}'_k\| \\ &\leq n^{-1}\|\mathbf{X}_k^\top(\Psi'_\tau(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) - \Psi'_\tau(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}'))\| \\ &+ \gamma_k\|\boldsymbol{\beta}_k - \boldsymbol{\beta}'_k\| \\ &\overset{(a)}{\leq} n^{-1}c\|\mathbf{X}_k\|\|\mathbf{X}\|\|(\boldsymbol{\beta} - \boldsymbol{\beta}')\| + \gamma_k\|\boldsymbol{\beta}_k - \boldsymbol{\beta}'_k\| \\ &\leq G_k\|\boldsymbol{\beta} - \boldsymbol{\beta}'\|, \qquad \forall \mathbf{v}_k \in \mathbb{R}^{p_k},\ \forall k, \boldsymbol{\beta}, \boldsymbol{\beta}' \in \mathbb{R}^{p+1}, \end{aligned} \tag{2.21}$$

where $G_k = \sqrt{\gamma_k}\sqrt{\gamma} + \gamma_k$. Inequality (a) is due to equation (2.18)

The proof of Theorem 2.3 relies on the iteration complexity analysis which is given next. This analysis is divided into three parts : the sufficient descent step, the cost-to-go estimate step, and the local error bound step. Similar techniques can be found in [64], [65], [111], [88] and [38].

Iteration Complexity Analysis. For ease of exposition, let us rewrite (7) as the following unconstrained optimization problem

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^{p+1}} Q(\boldsymbol{\beta}) := \min_{\boldsymbol{\beta} \in \mathbb{R}^{p+1}} L(\boldsymbol{\beta}) + \sum_{k=1}^{K} h_k(\boldsymbol{\beta}_k), \tag{2.22}$$

where $L(\boldsymbol{\beta})$ is the smooth quantile loss function which is smooth convex in $\boldsymbol{\beta} \in \mathbb{R}^{p+1}$ while $h_k(\boldsymbol{\beta}_k) = w_k\lambda\|\boldsymbol{\beta}_k\|$ is nonsmooth convex in $\boldsymbol{\beta}_k$ for each $k = 1, \ldots, K$. We have the following cyclic block-coordinate update of $\boldsymbol{\beta}_k$ by (11)

$$\boldsymbol{\beta}_k := \mathbf{prox}_{\gamma_k^{-1}h_k}(\boldsymbol{\beta}_k - \gamma_k^{-1}\nabla_k L(\boldsymbol{\beta})).$$

The following notation is convenient for this iteration complexity analysis. Let $(\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_K)$ be a K-block partition of the optimization variable $\boldsymbol{\beta}$ (i.e., $\boldsymbol{\beta} = (\boldsymbol{\beta}_1^\top, \ldots, \boldsymbol{\beta}_K^\top)^\top \in \mathbb{R}^{p+1}$, with $\boldsymbol{\beta}_k \in \mathbb{R}^{p_k}$ and $\sum_{k=1}^K p_k = p+1$). Also, denote the subvector of $\boldsymbol{\beta}$ with its kth component removed by $\boldsymbol{\beta}_{-k} = (\boldsymbol{\beta}_1^\top, \ldots, \boldsymbol{\beta}_{k-1}^\top, \boldsymbol{\beta}_{k+1}^\top, \ldots, \boldsymbol{\beta}_K^\top)^\top$ and recover $\boldsymbol{\beta}$ from $\boldsymbol{\beta}_{-k}$ by $\boldsymbol{\beta} = (\boldsymbol{\beta}_k^\top, \boldsymbol{\beta}_{-k}^\top)^\top$. Moreover, in the cyclic coordinate descent algorithm, let $\boldsymbol{\beta}^r$ be the update of $\boldsymbol{\beta}$ after the rth cycle, $r \geq 0$. When updating $\boldsymbol{\beta}_k$ in the $(r+1)$th cycle using the proximal operator (i.e. GPQR Algorithm 1), the following notations are also adopted

$$\begin{aligned}
&\boldsymbol{B}_k^{r+1} = [(\boldsymbol{\beta}_1^{r+1})^\top, \ldots, (\boldsymbol{\beta}_{k-1}^{r+1})^\top, (\boldsymbol{\beta}_k^r)^\top, (\boldsymbol{\beta}_{k+1}^r)^\top, \ldots, (\boldsymbol{\beta}_K^r)^\top]^\top, \ k = 2, \ldots, K, \\
&\boldsymbol{B}_{-k}^{r+1} = [(\boldsymbol{\beta}_1^{r+1})^\top, \ldots, (\boldsymbol{\beta}_{k-1}^{r+1})^\top, (\boldsymbol{\beta}_{k+1}^r)^\top, \ldots, (\boldsymbol{\beta}_K^r)^\top]^\top, \ k = 2, \ldots, K, \\
&\boldsymbol{\beta}_{-k} = [(\boldsymbol{\beta}_1)^\top, \ldots, (\boldsymbol{\beta}_{k-1})^\top, (\boldsymbol{\beta}_{k+1})^\top, \ldots, (\boldsymbol{\beta}_K)^\top]^\top, k = 2, \ldots, K.
\end{aligned}$$

By definition we have $\boldsymbol{B}_1^{r+1} := \boldsymbol{\beta}^r$ and $\boldsymbol{B}_{K+1}^{r+1} := \boldsymbol{\beta}^{r+1}$.

Sufficient Descent.

Consider the proximal gradient method applied to solving the following problem

$$\min_{\boldsymbol{\beta}_k \in \mathbb{R}^{p_k}} Q(\boldsymbol{\beta}_k, \boldsymbol{B}_{-k}^{r+1}) = \min_{\boldsymbol{\beta}_k \in \mathbb{R}^{p_k}} L(\boldsymbol{\beta}_k, \boldsymbol{B}_{-k}^{r+1}) + h_k(\boldsymbol{\beta}_k).$$

By the convexity of $h_k(\cdot)$, there exists $\zeta_k^{r+1} \in \partial h_k(\boldsymbol{\beta}_k^{r+1})$ such that

$$h_k(\boldsymbol{\beta}_k^r) - h_k(\boldsymbol{\beta}_k^{r+1}) \geq \left\langle \zeta_k^{r+1}, \, \boldsymbol{\beta}_k^r - \boldsymbol{\beta}_k^{r+1} \right\rangle, \ \forall \boldsymbol{\beta}_k^r, \tag{2.23}$$

where ∂h_k is is a sub-gradient of h_k.

Using (2.20) and (2.23), one has

$$\begin{aligned}
&Q(\boldsymbol{\beta}_k^r, \boldsymbol{B}_{-k}^{r+1}) - Q(\boldsymbol{\beta}_k^{r+1}, \boldsymbol{B}_{-k}^{r+1}) \\
=&u_k(\boldsymbol{\beta}_k^r; \boldsymbol{B}_{-k}^{r+1}) + h_k(\boldsymbol{\beta}_k^r) - \left(u_k(\boldsymbol{\beta}_k^{r+1}; \boldsymbol{B}_{-k}^{r+1}) + h_k(\boldsymbol{\beta}_k^{r+1})\right) \\
\geq& \left\langle \nabla u_k(\boldsymbol{\beta}_k^{r+1}; \boldsymbol{B}_{-k}^{r+1}), \, \boldsymbol{\beta}_k^r - \boldsymbol{\beta}_k^{r+1} \right\rangle + h_k(\boldsymbol{\beta}_k^r) - h_k(\boldsymbol{\beta}_k^{r+1}) + \frac{\gamma_k}{2} \|\boldsymbol{\beta}_k^r - \boldsymbol{\beta}_k^{r+1}\|^2 \\
\geq& \left\langle \nabla u_k(\boldsymbol{\beta}_k^{r+1}; \boldsymbol{B}_{-k}^{r+1}) + \zeta_k^{r+1}, \, \boldsymbol{\beta}_k^r - \boldsymbol{\beta}_k^{r+1} \right\rangle + \frac{\gamma_k}{2} \|\boldsymbol{\beta}_k^r - \boldsymbol{\beta}_k^{r+1}\|^2 \\
\overset{(a)}{\geq}& \frac{\gamma_k}{2} \|\boldsymbol{\beta}_k^r - \boldsymbol{\beta}_k^{r+1}\|^2.
\end{aligned}$$

Inequality (a) is due to the optimality condition

$$\left\langle \nabla u_k(\boldsymbol{\beta}_k^{r+1}; \boldsymbol{B}_{-k}^{r+1}) + \zeta_k^{r+1}, \, \boldsymbol{\beta}_k^{r+1} - \boldsymbol{\beta}_k^r \right\rangle \leq 0. \tag{2.24}$$

Thus, it follows that

$$Q(\boldsymbol{\beta}^r) - Q(\boldsymbol{\beta}^{r+1}) = \sum_{k=1}^{K} \left[Q(\boldsymbol{\beta}_k^r, \boldsymbol{B}_{-k}^{r+1}) - Q(\boldsymbol{\beta}_k^{r+1}, \boldsymbol{B}_{-k}^{r+1})\right] \geq \frac{\underline{\gamma}}{2}\|\boldsymbol{\beta}^r - \boldsymbol{\beta}^{r+1}\|^2, \tag{2.25}$$

where $\underline{\gamma} = \min_{1\leq k\leq K} \gamma_k$.

Cost-to-go Estimate.

Let $\mathcal{X}^* = \{\boldsymbol{\beta}^* | Q(\boldsymbol{\beta}^*) = \min_{\boldsymbol{\beta}} Q(\boldsymbol{\beta})\}$ be the optimal solution set of problem (2.22). Let $\bar{\boldsymbol{\beta}}^r = (\bar{\boldsymbol{\beta}}_1^r, \ldots, \bar{\boldsymbol{\beta}}_K^r) \in \mathcal{X}^*$ be a point in $\mathcal{X}^*$ such that $\mathrm{d}_{\mathcal{X}^*}(\boldsymbol{\beta}^r) = \min_{\boldsymbol{\beta}\in\mathcal{X}^*} \|\boldsymbol{\beta} - \boldsymbol{\beta}^r\| = \|\bar{\boldsymbol{\beta}}^r - \boldsymbol{\beta}^r\|$.

We have

$$\begin{aligned}
&\left\langle \nabla u_k(\boldsymbol{\beta}_k^{r+1}; \boldsymbol{B}_{-k}^{r+1}),\, \boldsymbol{\beta}_k^{r+1} - \bar{\mathbf{b}}_k^r \right\rangle + \left[h_k(\boldsymbol{\beta}_k^{r+1}) - h_k(\bar{\mathbf{b}}_k^r)\right] \\
\leq &\left\langle \nabla u_k(\boldsymbol{\beta}_k^{r+1}; \boldsymbol{B}_{-k}^{r+1}) + \zeta_k^{r+1},\, \boldsymbol{\beta}_k^{r+1} - \bar{\mathbf{b}}_k^r \right\rangle \\
\leq &0,
\end{aligned} \tag{2.26}$$

where the first inequality is due to the inequality (2.23), and the last inequality, we use the optimality conditions in (2.24).

On the other hand, we also have that

$$\begin{aligned}
Q(\boldsymbol{\beta}^{r+1}) - Q(\bar{\boldsymbol{\beta}}^r) &= L(\boldsymbol{\beta}^{r+1}) - L(\bar{\boldsymbol{\beta}}^r) + \sum_{k=1}^{K} h_k(\boldsymbol{\beta}_k^{r+1}) - \sum_{k=1}^{K} h_k(\bar{\mathbf{b}}_k^r) \\
&\leq \left\langle \nabla L(\boldsymbol{\beta}^{r+1}),\, \boldsymbol{\beta}^{r+1} - \bar{\boldsymbol{\beta}}^r \right\rangle + \sum_{k=1}^{K} h_k(\boldsymbol{\beta}_k^{r+1}) - \sum_{k=1}^{K} h_k(\bar{\mathbf{b}}_k^r) \\
&= \sum_{k=1}^{K} \left\langle \nabla_k L(\boldsymbol{\beta}^{r+1}),\, \boldsymbol{\beta}_k^{r+1} - \bar{\mathbf{b}}_k^r \right\rangle + \sum_{k=1}^{K} \left[h_k(\boldsymbol{\beta}_k^{r+1}) - h_k(\bar{\mathbf{b}}_k^r)\right] \\
&= \sum_{k=1}^{K} \left\langle \nabla_k L(\boldsymbol{\beta}^{r+1}) - \nabla u_k(\boldsymbol{\beta}_k^{r+1}; \boldsymbol{B}_{-k}^{r+1}),\, \boldsymbol{\beta}_k^{r+1} - \bar{\mathbf{b}}_k^r \right\rangle \\
&\quad + \sum_{k=1}^{K} \left\langle \nabla u_k(\boldsymbol{\beta}_k^{r+1}; \boldsymbol{B}_{-k}^{r+1}),\, \boldsymbol{\beta}_k^{r+1} - \bar{\mathbf{b}}_k^r \right\rangle + \sum_{k=1}^{K} \left[h_k(\boldsymbol{\beta}_k^{r+1}) - h_k(\bar{\mathbf{b}}_k^r)\right].
\end{aligned} \tag{2.27}$$

Combine (2.26) and (2.27), we get

$$\begin{aligned}
\big(Q(\boldsymbol{\beta}^{r+1}) - Q(\bar{\boldsymbol{\beta}}^{r})\big)^2 &\le \left(\sum_{k=1}^{K} \big\langle \nabla_k L(\boldsymbol{\beta}^{r+1}) - \nabla u_k(\boldsymbol{\beta}_k^{r+1}; \boldsymbol{B}_{-k}^{r+1}),\, \boldsymbol{\beta}_k^{r+1} - \bar{\mathbf{b}}_k^{r} \big\rangle \right)^2 \\
&\overset{(a)}{\le} \left(\sum_{k=1}^{K} \Big\| \nabla_k L(\boldsymbol{\beta}^{r+1}) - \nabla u_k(\boldsymbol{\beta}_k^{r+1}; \boldsymbol{B}_{-k}^{r+1}) \Big\|^2 \right) \left(\sum_{k=1}^{K} \Big\| \boldsymbol{\beta}_k^{r+1} - \bar{\mathbf{b}}_k^{r} \Big\|^2 \right) \\
&= \left(\sum_{k=1}^{K} \Big\| \nabla_k L(\boldsymbol{\beta}^{r+1}) - \nabla u_k(\boldsymbol{\beta}_k^{r+1}; \boldsymbol{B}_{-k}^{r+1}) \Big\|^2 \right) \Big\| \boldsymbol{\beta}^{r+1} - \bar{\boldsymbol{\beta}}^{r} \Big\|^2 \\
&\overset{(b)}{=} \left(\sum_{k=1}^{K} \Big\| \nabla u_k(\boldsymbol{\beta}_k^{r+1}; \boldsymbol{\beta}_{-k}^{r+1}) - \nabla u_k(\boldsymbol{\beta}_k^{r+1}; \boldsymbol{B}_{-k+1}^{r+1}) \Big\|^2 \right) \Big\| \boldsymbol{\beta}^{r+1} - \boldsymbol{\beta}^{r} + \boldsymbol{\beta}^{r} - \bar{\boldsymbol{\beta}}^{r} \Big\|^2 \\
&\overset{(c)}{\le} \left(\sum_{k=1}^{K} G_k^2 \Big\| \boldsymbol{\beta}^{r+1} - \boldsymbol{B}_{k+1}^{r+1} \Big\|^2 \right) \cdot 2 \left(\Big\| \boldsymbol{\beta}^{r+1} - \boldsymbol{\beta}^{r} \Big\|^2 + \Big\| \boldsymbol{\beta}^{r} - \bar{\boldsymbol{\beta}}^{r} \Big\|^2 \right) \\
&\overset{(d)}{\le} (2 \sum_{k=1}^{K} G_k^2) \Big\| \boldsymbol{\beta}^{r+1} - \boldsymbol{\beta}^{r} \Big\|^2 \left(\Big\| \boldsymbol{\beta}^{r+1} - \boldsymbol{\beta}^{r} \Big\|^2 + \Big\| \boldsymbol{\beta}^{r} - \bar{\boldsymbol{\beta}}^{r} \Big\|^2 \right) \\
&\le G \Big\| \boldsymbol{\beta}^{r+1} - \boldsymbol{\beta}^{r} \Big\|^2 \left(\Big\| \boldsymbol{\beta}^{r+1} - \boldsymbol{\beta}^{r} \Big\|^2 + \mathrm{d}_{\mathcal{X}^*}^2(\boldsymbol{\beta}^{r}) \right), \qquad (2.28)
\end{aligned}$$

where $G = 2K(\sqrt{\overline{\gamma}}\sqrt{\gamma} + \overline{\gamma})$ and $\overline{\gamma} = \max_{1 \le k \le K} \gamma_k$. Inequality (a) in (2.28) is due to the Cauchy-Schwarz inequality, equality (b) is due to that $\nabla_k L(\boldsymbol{\beta}^{r+1}) = \nabla_k L(\boldsymbol{\beta}_k^{r+1}, \boldsymbol{\beta}_{-k}^{r+1}) = \nabla u_k(\boldsymbol{\beta}_k^{r+1}, \boldsymbol{\beta}_{-k}^{r+1})$. In inequalities (c) and (d), we use the inequality (2.21) and $\Big\|\boldsymbol{\beta}^{r+1} - \boldsymbol{B}_{k+1}^{r+1}\Big\| \le \|\boldsymbol{\beta}^{r+1} - \boldsymbol{\beta}^{r}\|$ for all k, respectively.

Local error bound.

Let $\mathbf{d}_{\mathcal{X}^*}(\boldsymbol{\beta}) \equiv \min_{\boldsymbol{\beta}^* \in \mathcal{X}^*} \|\boldsymbol{\beta}^* - \boldsymbol{\beta}\|$. Note that the function $p(\mathbf{z}) = n^{-1}\mathbf{1}_n^\top \Psi_\tau(\mathbf{y} - \mathbf{z})$ is strongly convex in $\mathbf{z} \in \mathbb{R}^n$. We can see that $L(\boldsymbol{\beta}) = p(\mathbf{X}\boldsymbol{\beta})$. It follows from [111] that for any $\xi \ge \min_{\boldsymbol{\beta}} Q(\boldsymbol{\beta})$, there exist $\kappa, \varepsilon > 0$ such that

$$\mathrm{d}_{\mathcal{X}^*}(\boldsymbol{\beta}) \le \kappa \|\boldsymbol{\beta} - \mathbf{prox}_h(\boldsymbol{\beta} - \nabla L(\boldsymbol{\beta}))\|, \qquad (2.29)$$

for all $\boldsymbol{\beta}$ such that $\|\boldsymbol{\beta} - \mathbf{prox}_h(\boldsymbol{\beta} - \nabla L(\boldsymbol{\beta}))\| \le \varepsilon$ and $Q(\boldsymbol{\beta}) \le \xi$.

Now we are ready to prove Theorem 2.3.

Preuve. We first show that there exist some $\sigma > 0$ such that

$$\|\boldsymbol{\beta}^r - \mathbf{prox}_h(\boldsymbol{\beta}^r - \nabla L(\boldsymbol{\beta}^r))\| \le \sigma \|\boldsymbol{\beta}^{r+1} - \boldsymbol{\beta}^r\|, \ \forall r \ge 1. \qquad (2.30)$$

For any $r \ge 1$ and any $1 \le k \le K$, by the optimality of

$$\boldsymbol{\beta}_k^{r+1} := \arg\min_{\boldsymbol{\beta}_k} u_k(\boldsymbol{\beta}_k; \boldsymbol{B}_{-k}^{r+1}) + h_k(\boldsymbol{\beta}_k),$$

we have

$$\boldsymbol{\beta}_k^{r+1} = \mathbf{prox}_{\gamma_k^{-1} h_k}(\boldsymbol{\beta}_k^r - \gamma_k^{-1}\nabla u_k(\boldsymbol{\beta}_k^r; \boldsymbol{B}_{-k}^{r+1})).$$

Let $\overline{\gamma} = \max_{1\le k\le K}\gamma_k$, $\underline{\gamma} = \min_{1\le k\le K}\gamma_k$, $\hat{\gamma}_k = \max(1,\gamma_k)$ and $\tilde{\gamma}_k = \max(1,\gamma_k^{-1})$. It follows from Lemma 4.3 of [47] that

$$\begin{aligned}
&\|\boldsymbol{\beta}_k^r - \mathbf{prox}_{h_k}(\boldsymbol{\beta}_k^r - \nabla_k L(\boldsymbol{\beta}^r))\| \le \hat{\gamma}_k\|\boldsymbol{\beta}_k^r - \mathbf{prox}_{\gamma_k^{-1}h_k}(\boldsymbol{\beta}_k^r - \gamma_k^{-1}\nabla_k L(\boldsymbol{\beta}^r))\| \\
&\le \hat{\gamma}_k\left[\|\boldsymbol{\beta}_k^{r+1} - \mathbf{prox}_{\gamma_k^{-1}h_k}(\boldsymbol{\beta}_k^r - \gamma_k^{-1}\nabla_k L(\boldsymbol{\beta}^r))\| + \|\boldsymbol{\beta}_k^{r+1} - \boldsymbol{\beta}_k^r\|\right] \\
&\le \hat{\gamma}_k\left[|\mathbf{prox}_{\gamma_k^{-1}h_k}(\boldsymbol{\beta}_k^{r+1} - \gamma_k^{-1}\nabla u_k(\boldsymbol{\beta}_k^{r+1}; \boldsymbol{B}_{-k}^{r+1}))\right. \\
&\qquad\left. - \mathbf{prox}_{\gamma_k^{-1}h_k}(\boldsymbol{\beta}_k^r - \gamma_k^{-1}\nabla_k L(\boldsymbol{\beta}^r))| + \|\boldsymbol{\beta}_k^{r+1} - \boldsymbol{\beta}_k^r\|\right] \\
&\le 2\hat{\gamma}_k\|\boldsymbol{\beta}_k^{r+1} - \boldsymbol{\beta}_k^r\| + \hat{\gamma}_k\gamma_k^{-1}\|\nabla u_k(\boldsymbol{\beta}_k^{r+1}; \boldsymbol{B}_{-k}^{r+1}) - \nabla_k L(\boldsymbol{\beta}^r)\| \\
&\le 2\hat{\gamma}_k\|\boldsymbol{\beta}_k^{r+1} - \boldsymbol{\beta}_k^r\| + \hat{\gamma}_k\gamma_k^{-1}\|\nabla_k L(\boldsymbol{B}_k^{r+1}) + \gamma_k(\boldsymbol{\beta}_k^{r+1} - \boldsymbol{\beta}_k^r) - \nabla_k L(\boldsymbol{\beta}^r)\| \\
&\le 3\hat{\gamma}_k\|\boldsymbol{\beta}_k^{r+1} - \boldsymbol{\beta}_k^r\| + \hat{\gamma}_k\tilde{\gamma}_k\|\nabla_k L(\boldsymbol{B}_k^{r+1}) - \nabla_k L(\boldsymbol{\beta}^r)\| \\
&\le 3\hat{\gamma}_k\|\boldsymbol{\beta}_k^{r+1} - \boldsymbol{\beta}_k^r\| + \hat{\gamma}_k\tilde{\gamma}_k\|\nabla L(\boldsymbol{B}_k^{r+1}) - \nabla L(\boldsymbol{\beta}^r)\| \\
&\le 3\hat{\gamma}_k\|\boldsymbol{\beta}_k^{r+1} - \boldsymbol{\beta}_k^r\| + \hat{\gamma}_k\tilde{\gamma}_k\gamma\|\boldsymbol{B}_k^{r+1} - \boldsymbol{\beta}^r\| \\
&\le (3 + \gamma\tilde{\gamma}_k)\hat{\gamma}_k\|\boldsymbol{\beta}^{r+1} - \boldsymbol{\beta}^r\|.
\end{aligned}$$

It follows that

$$\|\boldsymbol{\beta}^r - \mathbf{prox}_h(\boldsymbol{\beta}^r - \nabla L(\boldsymbol{\beta}^r))\| \le (3 + \gamma\tilde{\gamma})\hat{\gamma}K\|\boldsymbol{\beta}^{r+1} - \boldsymbol{\beta}^r\|,$$

where $\hat{\gamma} = \max(1,\bar{\gamma})$ and $\tilde{\gamma} = \max(1,\underline{\gamma}^{-1})$. Therefore, when we take $\sigma = (3 + \gamma\tilde{\gamma})\hat{\gamma}K$, we get the desired result in (2.30). *Note that the sufficient descent property* (2.25) *implies that $\|\boldsymbol{\beta}^{r+1} - \boldsymbol{\beta}^r\| \to 0$ as $r \to \infty$. It follows from* (2.30) *that $\|\boldsymbol{\beta}^r - \mathbf{prox}_h(\boldsymbol{\beta}^r - \nabla L(\boldsymbol{\beta}^r))\| \to 0$ as $r \to \infty$. Thus, by* (2.29) *we have $\mathrm{d}_{\mathcal{X}^*}(\boldsymbol{\beta}^r) \to 0$ as $r \to \infty$. Consequently, using* (2.28), *we have $Q(\boldsymbol{\beta}^r) \to Q^* := \min_{\boldsymbol{\beta}} Q(\boldsymbol{\beta})$, which shows that the GPQR algorithm converges to the global minimum.*

Now, let $c_1 = \underline{\gamma}/2$, $c_2 = \sqrt{G}$, and $\Delta^r = Q(\boldsymbol{\beta}^r) - Q^*$. By the local error bound (2.29) *and the cost-to-go* *estimate* (2.28), *we obtain*

$$
\begin{aligned}
\Delta^{r+1} &\leq c_2\sqrt{\left\|\boldsymbol{\beta}^{r+1}-\boldsymbol{\beta}^{r}\right\|^2\left(\left\|\boldsymbol{\beta}^{r+1}-\boldsymbol{\beta}^{r}\right\|^2+\mathrm{d}^2_{\mathcal{X}^*}(\boldsymbol{\beta}^r)\right)} \\
&\leq c_2\sqrt{\left\|\boldsymbol{\beta}^{r+1}-\boldsymbol{\beta}^{r}\right\|^2\left(\left\|\boldsymbol{\beta}^{r+1}-\boldsymbol{\beta}^{r}\right\|^2+\kappa^2\|\boldsymbol{\beta}^r-\mathbf{prox}_h(\boldsymbol{\beta}^r-\nabla L(\boldsymbol{\beta}^r))\|^2\right)} \\
&\overset{(a)}{\leq} c_2\sqrt{\left\|\boldsymbol{\beta}^{r+1}-\boldsymbol{\beta}^{r}\right\|^2\left(\left\|\boldsymbol{\beta}^{r+1}-\boldsymbol{\beta}^{r}\right\|^2+\kappa^2\sigma^2\|\boldsymbol{\beta}^{r+1}-\boldsymbol{\beta}^r\|^2\right)} \\
&\leq (c_2\sqrt{1+\kappa^2\sigma^2})\|\boldsymbol{\beta}^{r+1}-\boldsymbol{\beta}^r\|^2 \\
&\overset{(b)}{\leq} (c_2\sqrt{1+\kappa^2\sigma^2})c_1^{-1}[Q(\boldsymbol{\beta}^r)-Q(\boldsymbol{\beta}^{r+1})] \\
&= (c_2\sqrt{1+\kappa^2\sigma^2})c_1^{-1}(\Delta^r-\Delta^{r+1}).
\end{aligned}
$$

Inequality (a) is due to (2.30), *and inequality (b) is due to* (2.25). *This implies that*

$$\Delta^{r+1} \leq \frac{c_3}{1+c_3}\Delta^r, \tag{2.31}$$

where $c_3 = (c_2\sqrt{1+\kappa^2\sigma^2})c_1^{-1}$. *We can see from* (2.31) *that* $Q(\boldsymbol{\beta}^r)$ *approaches* Q^* *with at least linear rate of convergence. From* (2.25) *again, this further implies that the sequence* $\{\boldsymbol{\beta}^r\}$ *converges at least linearly.* □

2.7.4 Proof of Proposition 2.4

For Group SCAD penalty

The KKT conditions of the objective function in equation (9) of the main manuscript, with $P_\lambda(\|\boldsymbol{\beta}_k\|_2)$ is given by (4), can be written as

$$-\mathbf{Z}_k + \gamma_k\boldsymbol{\beta}_k + P'_{\lambda,\theta}(\|\boldsymbol{\beta}_k\|_2) = 0,$$

where $\mathbf{Z}_k = -\nabla_k L(\tilde{\boldsymbol{\beta}}) + \gamma_k\tilde{\boldsymbol{\beta}}_k$.

— If $\|\boldsymbol{\beta}_k\|_2 \leq \lambda$ then $-\mathbf{Z}_k + \gamma_k\boldsymbol{\beta}_k + \lambda w_k\mathbf{u} = 0$ where $\mathbf{u}$ is the sub-gradient and $\|\mathbf{u}\|_2 \leq 1$

— If $\boldsymbol{\beta}_k = 0$, then

$$
\begin{aligned}
&\Rightarrow -\mathbf{Z}_k + \lambda w_k\mathbf{u} = 0 \\
&\Rightarrow \|\mathbf{Z}_k\|_2 \leq \lambda w_k.
\end{aligned}
$$

- If $\boldsymbol{\beta}_k \neq 0$, then one has

$$\begin{aligned} -\mathbf{Z}_k + \gamma_k \boldsymbol{\beta}_k + \lambda w_k \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} = 0 \quad & \Rightarrow \quad \|\mathbf{Z}_k\|_2 \leq \gamma_k \|\boldsymbol{\beta}_k\|_2 + \lambda w_k \\ & \Rightarrow \quad \|\mathbf{Z}_k\|_2 \leq \lambda(w_k + \gamma_k). \end{aligned}$$

  Moreover, we have

$$-\mathbf{Z}_k + \gamma_k \boldsymbol{\beta}_k + \lambda w_k \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} = 0 \; (since \; \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} = \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2}),$$

  which implies

$$\boldsymbol{\beta}_k = \frac{1}{\gamma_k} \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} (\|\mathbf{Z}_k\|_2 - \lambda w_k).$$

- If $\lambda \leq \|\boldsymbol{\beta}_k\|_2 \leq \theta\lambda$, then $-\mathbf{Z}_k + \gamma_k \boldsymbol{\beta}_k + \frac{\theta \lambda w_k}{\theta - 1} \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} - \frac{w_k}{\theta-1} \boldsymbol{\beta}_k = 0$. It follows that

$$\mathbf{Z}_k = [\gamma_k + \frac{w_k}{\theta - 1}(\frac{\theta\lambda}{\|\boldsymbol{\beta}_k\|_2} - 1)]\boldsymbol{\beta}_k,$$

  which implies that

$$\|\mathbf{Z}_k\|_2 = (\gamma_k - \frac{w_k}{\theta - 1})\|\boldsymbol{\beta}_k\|_2 + \frac{w_k \lambda \theta}{\theta - 1} \;\; and \;\; \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} = \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2}.$$

  Thus, we have

$$\begin{aligned} & \lambda \leq \|\boldsymbol{\beta}_k\|_2 \leq \theta\lambda \\ & \Rightarrow (\gamma_k - \tfrac{w_k}{\theta-1})\lambda + \lambda w_k \tfrac{\theta}{\theta-1} \leq (\gamma_k - \tfrac{w_k}{\theta-1})\|\boldsymbol{\beta}_k\|_2 + \lambda w_k \tfrac{\theta}{\theta-1} \leq (\gamma_k - \tfrac{w_k}{\theta-1})\theta\lambda + \lambda w_k \tfrac{\theta}{\theta-1} \\ & \Rightarrow \lambda(\gamma_k + w_k) \leq \|\mathbf{Z}_k\|_2 \leq \gamma_k \theta \lambda \\ & \Rightarrow \boldsymbol{\beta}_k = \tfrac{1}{\gamma_k - \frac{w_k}{\theta-1}} \tfrac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} (\|\mathbf{Z}_k\|_2 - \lambda w_k \tfrac{\theta}{\theta-1}). \end{aligned}$$

- If $\|\boldsymbol{\beta}_k\|_2 \geq \theta\lambda$, then $-\mathbf{Z}_k + \gamma_k \boldsymbol{\beta}_k = 0$. This implies that

$$\|\mathbf{Z}_k\|_2 \geq \gamma_k \theta \lambda \quad \text{and} \quad \boldsymbol{\beta}_k = \frac{1}{\gamma_k} \mathbf{Z}_k.$$

To conclude, we have

$$\widehat{\boldsymbol{\beta}}_k = \begin{cases} \frac{1}{\gamma_k} \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} S(\|\mathbf{Z}_k\|_2, \lambda w_k), & \text{if} \quad \|\mathbf{Z}_k\|_2 \leq \lambda(\gamma_k + w_k) \\ \frac{1}{\gamma_k - \dfrac{w_k}{\theta - 1}} \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} (\|\mathbf{Z}_k\|_2 - \frac{\lambda w_k \theta}{\theta - 1}), & \text{if} \quad \lambda(\gamma_k + w_k) < \|\mathbf{Z}_k, \|_2 \leq \gamma_k \theta \lambda \\ \frac{1}{\gamma_k} \mathbf{Z}_k, & \text{if} \quad \|\mathbf{Z}_k\|_2 > \gamma_k \theta \lambda. \end{cases}$$

For Group MCP penalty

Again, the KKT conditions of the objective function in equation (9) of the main manuscript, with $P_\lambda(\|\boldsymbol{\beta}_k\|_2)$ is given by (3), can be written as

$$-\mathbf{Z}_k + \gamma_k \boldsymbol{\beta}_k + P'_{(\lambda,\theta)}(\|\boldsymbol{\beta}_k\|_2) = 0,$$

where $\mathbf{Z}_k = -\nabla_k L(\tilde{\boldsymbol{\beta}}) + \gamma_k \tilde{\boldsymbol{\beta}}_k$.

- If $\|\boldsymbol{\beta}_k\|_2 \leq \theta\lambda$, then $-\mathbf{Z}_k + \gamma_k \boldsymbol{\beta}_k + \lambda \mathbf{u} - \frac{w_k}{\theta}\boldsymbol{\beta}_k = 0$, where $\mathbf{u}$ is the sub-gradient and $\|\mathbf{u}\|_2 \leq 1$.
  - If $\boldsymbol{\beta}_k = 0$, then one has

    $$-\mathbf{Z}_k + \lambda w_k \mathbf{u} = 0,$$

    which implies that

    $$\|\mathbf{Z}_k\|_2 \leq \lambda w_k.$$

  - If $\boldsymbol{\beta}_k \neq 0$, then

    $$-\mathbf{Z}_k + \gamma_k \boldsymbol{\beta}_k + \lambda w_k \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} - \frac{w_k}{\theta}\boldsymbol{\beta}_k = 0,$$

    which implies that

    $$\|\mathbf{Z}_k\|_2 = (\gamma_k - \frac{w_k}{\theta})\|\boldsymbol{\beta}_k\|_2 + \lambda w_k.$$

    Thus,

    $$\begin{aligned}&\|\boldsymbol{\beta}_k\|_2 \leq \theta\lambda \\ &\Rightarrow (\gamma_k - \tfrac{w_k}{\theta})\|\boldsymbol{\beta}_k\|_2 + \lambda w_k \leq (\gamma_k - \tfrac{w_k}{\theta})\theta\lambda + \lambda w_k \\ &\Rightarrow \|\mathbf{Z}_k\|_2 \leq \gamma_k\theta\lambda \\ &\Rightarrow \boldsymbol{\beta}_k = \tfrac{1}{\gamma_k - \frac{w_k}{\theta}} \tfrac{\mathbf{Z}^{(k)}}{\|\mathbf{Z}_k\|_2}(\|\mathbf{Z}_k\|_2 - \lambda w_k).\end{aligned}$$

- If $\|\boldsymbol{\beta}_k\|_2 \geq \theta\lambda$, then we have $-\mathbf{Z}_k + \gamma_k \boldsymbol{\beta}_k = 0$. This implies that

  $$\|\mathbf{Z}_k\|_2 \geq \gamma_k\theta\lambda \quad \text{and} \quad \boldsymbol{\beta}_k = \frac{1}{\gamma_k}\mathbf{Z}_k.$$

To sum up, we have

$$\widehat{\boldsymbol{\beta}}_k = \begin{cases} \frac{1}{\gamma_k - w_k/\theta}\frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} S(\|\mathbf{Z}_k\|_2, \lambda w_k), & \text{if } \|\mathbf{Z}_k\|_2 \leq \gamma_k\theta\lambda \\ \frac{1}{\gamma_k}\mathbf{Z}_k, & \text{if } \|\mathbf{Z}_k\|_2 > \gamma_k\theta\lambda. \end{cases}$$

2.7.5 Solution path comparison of GLLA and GSCAD/GMCP penalties

Illustration of the GPQR approach with GLLA approximation compared to the exact GMCP and GSCAD penalties are given in Figure $S.1$. In this example, we used the smoothed check function $\Psi^{(1)}_{\tau,\delta}(u)$ to approximate the standard quantile check function, with $\delta = 1$. We generated n observations of p-dimensional vector $\mathbf{x}_i, i = 1, \ldots, n$, following a multivariate normal distribution, with $p = 200$ and $n = 100$. We divided the p variables into $K = 191$ groups, and assigned non-zero coefficients to the first three groups and set the 188 coefficients of the remaining 188 groups to be zero :

$$\boldsymbol{\beta} = (\underbrace{3, 3, 3, 3}_{G_1}, \underbrace{2, 2, 2, 2}_{G_2}, \underbrace{-1, -1, -1, -1}_{G_3}, \underbrace{0, \ldots, 0}_{G_4 - G_{191}})^\top.$$

The response $y_i, i = 1, \ldots, n$, is generated from the following linear regression model

$$y_i = \mathbf{x}_i^\top \boldsymbol{\beta} + \epsilon, \quad \epsilon \sim N(0, 1).$$

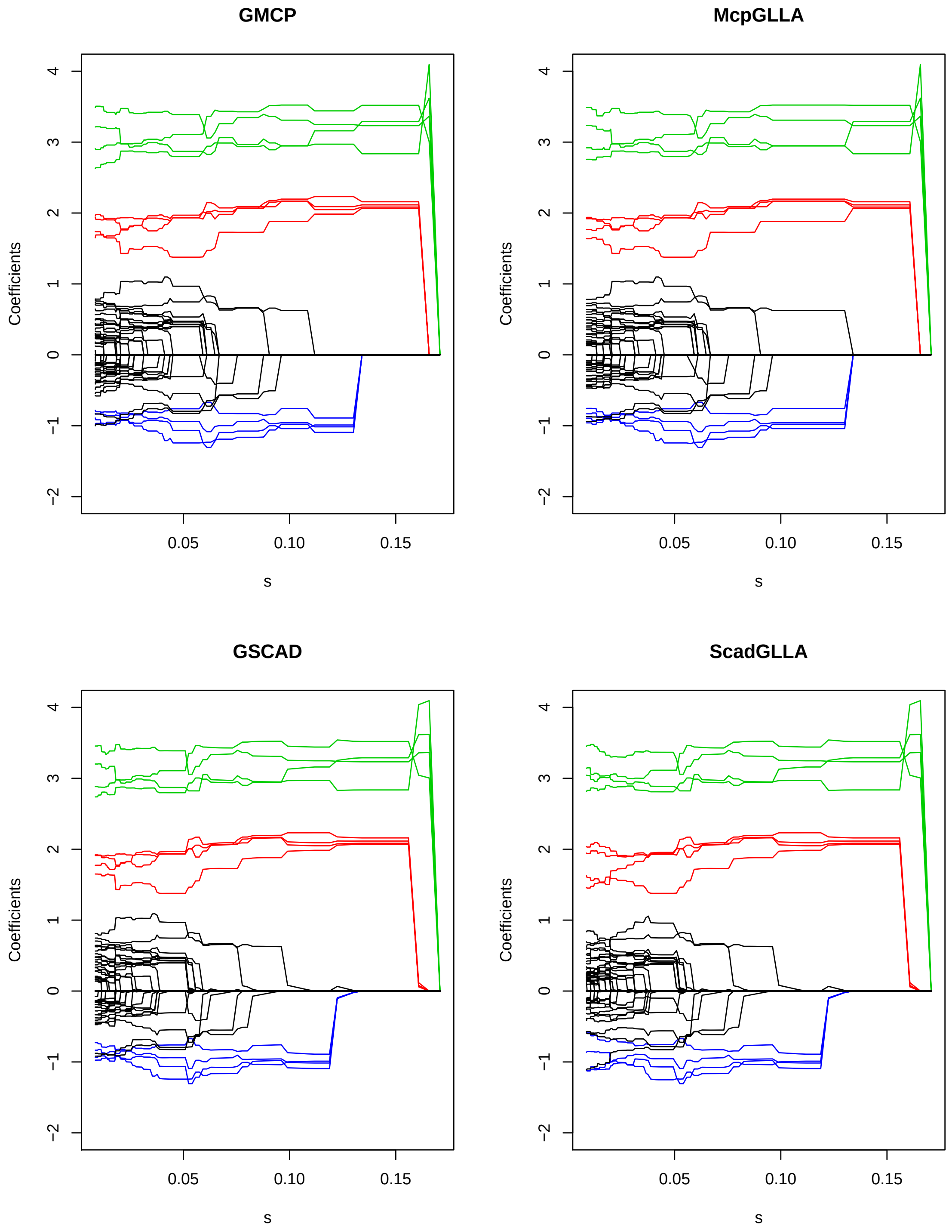


Figure S.1 : In the left, the coefficient paths of the penalized quantile regression with the group penalties (GMCP and GSCAD), and in the right, their GLLA approximations.

2.7.6 Checking the theoretical KKT conditions

In this section, we establish the theoretical KKT conditions of GPQR solution. When our GPQR algorithm converges to the final solution, it must satisfy those conditions, which means that the algorithm converges and finds the right answer.

For GPQR with GLasso penalty, the KKT conditions of the objective function in equation (7) of the main manuscript with $P_\lambda(\|\boldsymbol{\beta}_k\|_2)$ is given by (2) can be written as

$$\nabla_k L(\boldsymbol{\beta}) + \lambda w_k \partial \|\boldsymbol{\beta}_k\|_2 = 0,$$

If $\boldsymbol{\beta}_k = 0$, then we have

$$\nabla_k L(\boldsymbol{\beta}) + \lambda w_k \mathbf{u} = 0,$$

where $\mathbf{u}$ is the sub-gradient of $\|\boldsymbol{\beta}_k\|_2$ and $\|\mathbf{u}\|_2 \leq 1$
which implies

$$\|\nabla_k L(\boldsymbol{\beta})\|_2 \leqslant \lambda w_k. \tag{2.32}$$

If $\boldsymbol{\beta}_k \neq 0$, then we have

$$\nabla_k L(\boldsymbol{\beta}) + \lambda w_k \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} = 0. \tag{2.33}$$

Combining (2.32) and (2.33), we get

$$\begin{cases} \nabla_k L(\boldsymbol{\beta}) + \lambda \omega_k \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} = \mathbf{0}, & if \ \boldsymbol{\beta}_k \neq 0 \\ \|\nabla_k L(\boldsymbol{\beta})\|_2 \leqslant \lambda w_k, & if \ \boldsymbol{\beta}_k = 0. \end{cases}$$

Following the same reasoning as for GLasso and as in Proposition 2.4 , the exact KKT conditions of GMCP, GSCAD and GLLA are given for each solution $\boldsymbol{\beta}_k, \{k = 1, \dots, K\}$ respectively, as

$$\begin{cases} \nabla_k L(\boldsymbol{\beta}) + \lambda \omega_k . \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} - \frac{\boldsymbol{\beta}_k}{\theta} = \mathbf{0}, & if \ \boldsymbol{\beta}_k \neq 0 \ and \ \|\boldsymbol{\beta}_k\|_2 \leqslant \theta\lambda \\ \|\nabla_k L(\boldsymbol{\beta})\|_2 \leqslant \lambda \omega_k, & if \ \boldsymbol{\beta}_k = 0 \ and \ \|\boldsymbol{\beta}_k\|_2 \leqslant \theta\lambda \\ \|\nabla_k L(\boldsymbol{\beta})\|_2 = 0, & if \ \|\boldsymbol{\beta}_k\|_2 > \theta\lambda. \end{cases}$$

$$\begin{cases} \nabla_k L(\boldsymbol{\beta}) + \lambda\omega_k \cdot \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} = \mathbf{0}, & if \;\; \boldsymbol{\beta}_k \neq 0 \;\; and \;\; \|\boldsymbol{\beta}_k\|_2 \leqslant \lambda \\ \|\nabla_k L(\boldsymbol{\beta})\|_2 \leqslant \lambda\omega_k, & if \;\; \boldsymbol{\beta}_k = 0 \;\; and \;\; \|\boldsymbol{\beta}_k\|_2 \leqslant \lambda \\ \nabla_k L(\boldsymbol{\beta}) + \frac{\theta}{\theta-1}\lambda\omega_k \cdot \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} - \frac{\boldsymbol{\beta}_k}{(\theta-1)} = \mathbf{0}, & if \;\; \lambda < \|\boldsymbol{\beta}_k\|_2 \leqslant \theta\lambda \\ \|\nabla_k L(\boldsymbol{\beta})\|_2 = 0, & if \;\; \|\boldsymbol{\beta}_k\|_2 > \theta\lambda. \end{cases}$$

$$\begin{cases} \nabla_k L(\boldsymbol{\beta}) + \lambda\omega_k' \cdot \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} = \mathbf{0}, & if \;\; \boldsymbol{\beta}_k \neq 0 \\ \|\nabla_k L(\boldsymbol{\beta})\|_2 \leqslant \lambda\omega_k', & if \;\; \boldsymbol{\beta}_k = 0. \end{cases}$$

2.7.7 Checking the numerical KKT conditions

The theoretical solution for the GPQR algoithm always passes the KKT condition check defined in the previous section. However, a numerical solution could only approach this theoretical value within certain precision therefore may fail the KKT check. In order to adapt the exact KKT conditions to the numerical solution. Numerically, we declare β_k passes the KKT condition check for GLasso, GMCP, GSCAD and GLLA, respectively if

$$\begin{cases} \|\nabla_k L(\boldsymbol{\beta}) + \lambda\omega_k \cdot \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2}\|_2 \leqslant \epsilon, & if \;\; \boldsymbol{\beta}_k \neq 0 \\ \|\nabla_k L(\boldsymbol{\beta})\|_2 \leqslant \lambda\omega_k + \epsilon, & if \;\; \boldsymbol{\beta}_k = 0, \end{cases}$$

$$\begin{cases} \|\nabla_k L(\boldsymbol{\beta}) + \lambda\omega_k \cdot \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} - \frac{\boldsymbol{\beta}_k}{\theta}\|_2 \leqslant \epsilon, & if \;\; \boldsymbol{\beta}_k \neq 0 \;\; and \;\; \|\boldsymbol{\beta}_k\|_2 \leqslant \theta\lambda \\ \|\nabla_k L(\boldsymbol{\beta})\|_2 \leqslant \lambda\omega_k + \epsilon, & if \;\; \boldsymbol{\beta}_k = 0 \;\; and \;\; \|\boldsymbol{\beta}_k\|_2 \leqslant \theta\lambda \\ \|\nabla_k L(\boldsymbol{\beta})\|_2 \leqslant \epsilon, & if \;\; \|\boldsymbol{\beta}_k\|_2 > \theta\lambda, \end{cases}$$

$$\begin{cases} \|\nabla_k L(\boldsymbol{\beta}) + \lambda\omega_k \cdot \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2}\|_2 \leqslant \epsilon, & if \;\; \boldsymbol{\beta}_k \neq 0 \;\; and \;\; \|\boldsymbol{\beta}_k\|_2 \leqslant \lambda \\ \|\nabla_k L(\boldsymbol{\beta})\|_2 \leqslant \lambda\omega_k + \epsilon, & if \;\; \boldsymbol{\beta}_k = 0 \;\; and \;\; \|\boldsymbol{\beta}_k\|_2 \leqslant \lambda \\ \|\nabla_k L(\boldsymbol{\beta}) + \frac{\theta}{\theta-1}\lambda\omega_k \cdot \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} - \frac{\boldsymbol{\beta}_k}{(\theta-1)}\|_2 \leqslant \epsilon, & if \;\; \lambda < \|\boldsymbol{\beta}_k\|_2 \leqslant \theta\lambda \\ \|\nabla_k L(\boldsymbol{\beta})\|_2 \leqslant \epsilon, & if \;\; \|\boldsymbol{\beta}_k\|_2 > \theta\lambda, \end{cases}$$

$$\begin{cases} \|\nabla_k L(\boldsymbol{\beta}) + \lambda\omega_k' \cdot \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2}\|_2 \leqslant \epsilon, & if \;\; \boldsymbol{\beta}_k \neq 0 \\ \|\nabla_k L(\boldsymbol{\beta})\|_2 \leqslant \lambda\omega_k' + \epsilon, & if \;\; \boldsymbol{\beta}_k = 0. \end{cases}$$

for a small $\epsilon > 0$. In this paper we set $\epsilon = 10^{-4}$

	Q-GLasso	Q-GMCP	Q-GSCAD	Q-GLass	Q-GMCP	Q-GSCAD
Genes		$\tau = 0.25$			$\tau = 0.75$	
APOC1	98.8	97.9	33.7	29.8	7.7	47.9
TOMM40	85.8	29.0	0.0	0.0	0.0	0.0
APOE	94.2	69.2	38.4	14.4	4.8	5.0

	Q-GLasso	Q-GMCP	Q-GSCAD	Q-GLass	Q-GMCP	Q-GSCAD
		$\tau = 0.25$			$\tau = 0.75$	
QPE_τ	0.033	0.032	0.033	0.073	0.075	0.073
Size	13.38	6.61	4.38	3.69	3.46	4.01

Table S.1 : top : comparison of the number of times (in $\%$) the genes *APOE*, *TOMM40* and *APOC1* are selected, based on 100 replications, for ADNI data. bottom : average of the quantile-based error prediction (QPE_τ) and the number of selected groups/genes (Model Size) computed on the 100 runs' test sets. The group quantile methods are fitted with $\tau = 0.25, 0.75$.

2.7.8 ADNI data analysis

In this section we present additional results of the GPQR approach in the gene-based association study of the ADNI cohort. In this analysis we fitted the GPQR model for two additional locations, $\tau = 0.25, 0.75$.

Figure $S.2$ highlights results of the L2-norm of the coefficient paths of Q-GLasso, Q-GMCP and Q-GSCAD respectively, with $\tau = 0.25$, or 0.75, as a function of the tuning parameter λ. The results of Figure $S.2$ obtained by fitting GPQR for all 442 analyzed subjects of the ADNI cohort.

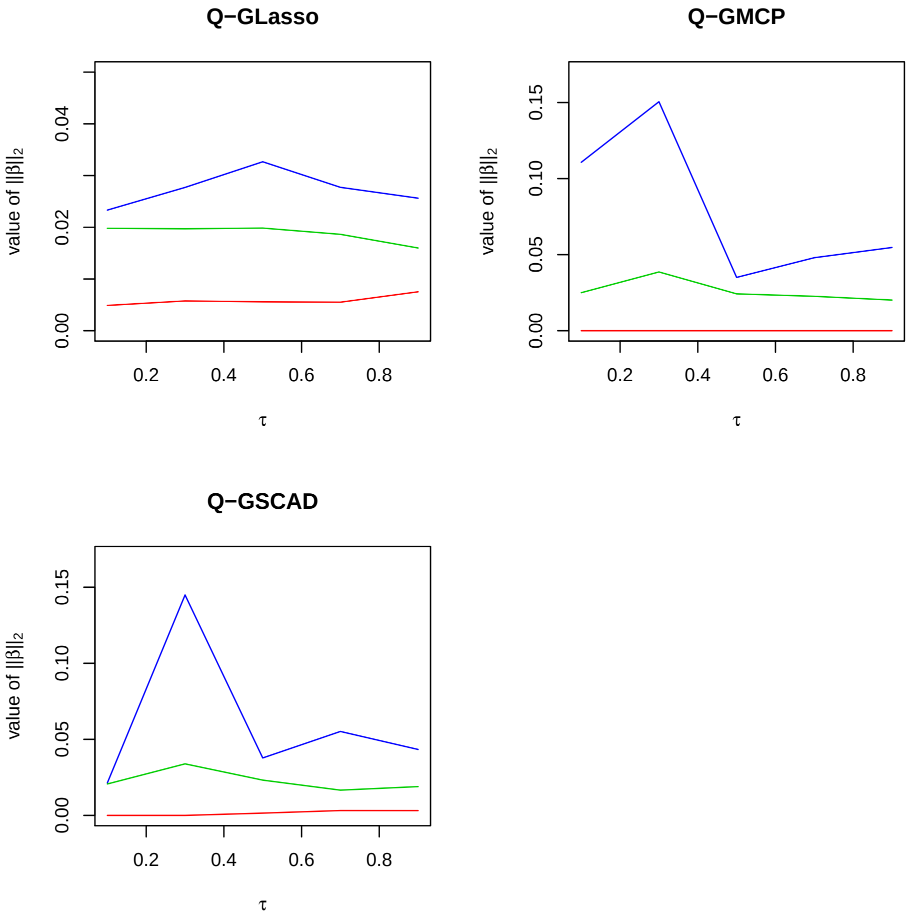


Figure S.2 : L2-norm of the optimal solution coefficients correspond to three important genes are shown as a function of the τ conditional quantile parameter. The genes APOE, TOMM40 APOC1 are plotted in blue, green and red, respectively.

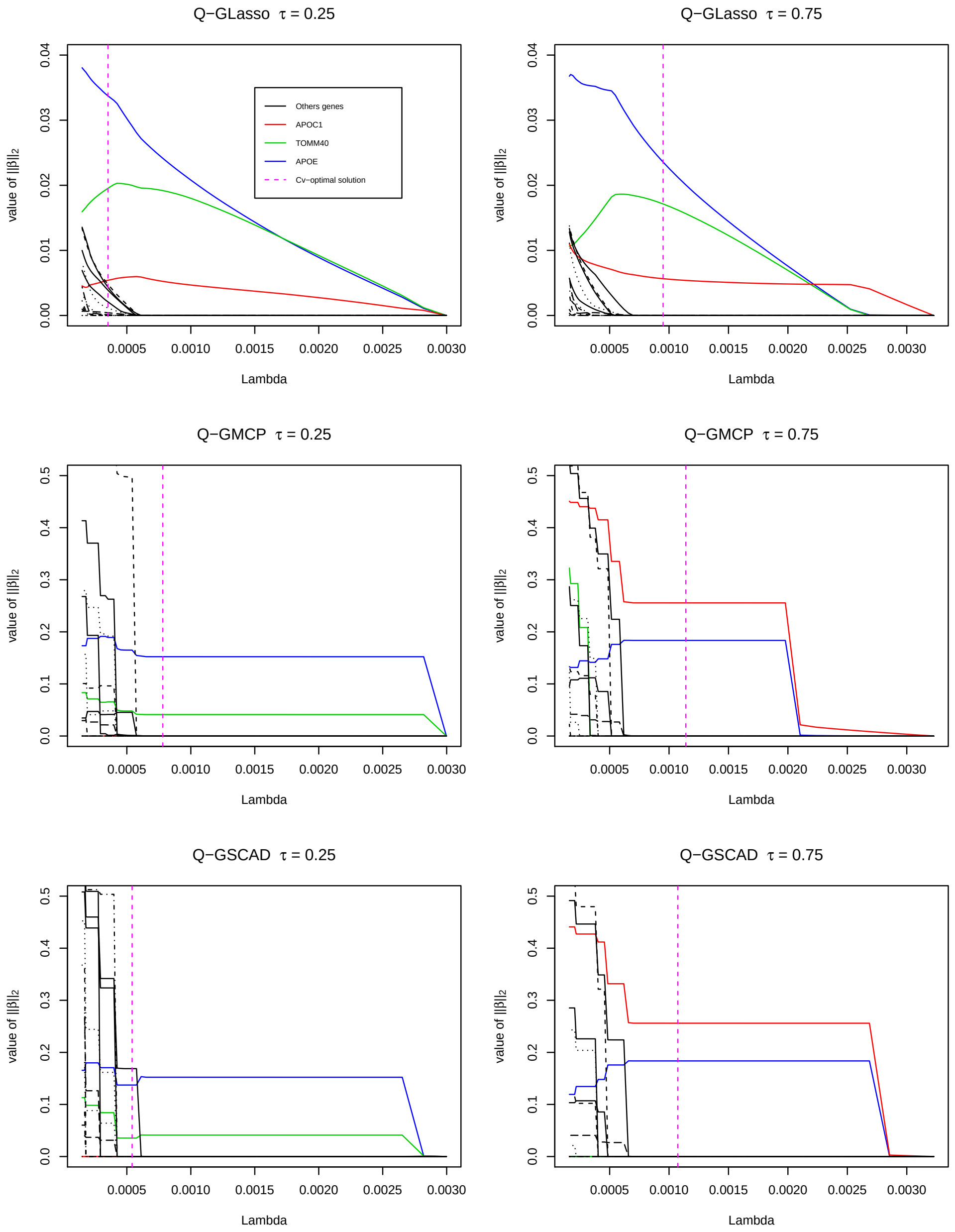


Figure S.3 : At the left and from top to bottom, L2-norm of the coefficient paths of Q-GLasso, Q-GMCP and Q-GSCAD respectively, with $\tau = 0.25$, are shown as a function of a tuning parameter λ. At the right and from top to bottom, the coefficient paths of the same group methods with $\tau = 0.75$.

CHAPITRE 3
GROUP PENALIZED EXPECTILE REGRESSION

Le temps de calcul est un défi majeur en régression quantile en grande dimension. Dans le chapitre 2, nous avons proposé un algorithme de descente par blocs de coordonnées pour résoudre le problème de la régression quantile pénalisée avec des pénalités qui sélectionnent les variables par groupe connu à priori. L'approche GPQR est basée sur l'approximation de la fonction de perte quantile, qui n'est pas différentiable à zéro, par une fonction de perte modifiée qui est différentiable à zéro. Ensuite, en utilisant le principe MM et l'algorithme CDA, nous mettons à jour chaque bloc de coefficients de manière simple et efficace. Dans notre implémentation, nous considérons la pénalité convexes group Lasso et les pénalités non convexes group SCAD et group MCP. Nous établissons la propriété de convergence linaire de l'algorithme GPQR avec la pénalité group Lasso.

L'approche GPQR traite l'hétéroscédasticité en utilisant une fonction de perte non différentiable. Dans ce chapitre, nous traitons celle-ci moyennant une fonction différentiable. La différentiabilité nous a permis de réaliser un développement théorique très intéressant de ces alternatives. La première alternative consiste à doter la régression expectile de la propriété de la sélection des variables par blocs. Elle détecte les groupes de variables hétéroscédastiques sans pouvoir nous dire quels groupes sont importants pour la moyenne conditionnelle et ceux qui le sont pour la variance conditionnelle. La seconde alternative vient répondre au problème de la première. En effet, elle permet de détecter à la fois le sous ensemble des groupes de variables qui a une influence sur la moyenne et celui qui a une influence sur la variance. Nous avons aussi montré l'efficacité de la régression asymétrique par rapport à la régression symétrique sur deux jeux de données biologique et génétique.

3.1 Abstract

The asymmetric least squares regression (or expectile regression) allows estimating unknown expectiles of the conditional distribution of a response variable as a function of a set of predictors and can handle heteroscedasticity issues. High dimensional data, such as omics data, are error prone and usually display heterogeneity. Such heterogeneity is often of scientific interest. In this work, we propose the Group Penalized Expectile Regression (GPER) approach, under high dimensional settings. GPER considers implementation of sparse expectile regression with group Lasso penalty and the group non-convex penalties. However, GPER may fail to tell which groups variables are important for the conditional mean and which groups of variables are important for the conditional scale/variance. To that end, we further propose a COupled Group Penalized Expectile Regression (COGPER) regression which can be efficiently solved by an algorithm similar to that for solving GPER. We establish theoretical properties of the proposed approaches. In particular, GPER and COGPER using the SCAD penalty or MCP is shown to consistently identify the two important subsets for the mean and scale simultaneously. We demonstrate the empirical performance of GPER and COGPER by simulated and real data.

3.2 Introduction

Sparse regression methods, which use penalization techniques for both estimation and variable selection, have been introduced as a mainstream approach for analyzing high-dimensional data. Popular penalized estimators are the l_1-type selectors such as the Lasso [90] and Dantzig estimators [15], and the non-convex penalized estimators such as the Smoothly Clipped Absolute Deviation (SCAD) [27] and the Minimax Concave Penalty (MCP) [110] estimators. L1-type selectors are useful due to their computational efficiency and the non-convex selectors are known to enjoy the oracle property. Several computationally efficient algorithms have also been proposed for computing the non-convex estimators. [120] worked out the Local Linear Approximation (LLA) algorithm, which approximates the non-convex penalties using a series of reweighted l_1 penalization.

In many situations, it is suitable to perform selection of a group/set of predictors sharing a common function (e.g., genes participate in a common biological function or pathway ; methylation levels in nearby positions along the genome present high spatial correlation). Capturing group-variable effects can improve the outcome prediction. Another attractive motivation of the group-variable selection methods is the additive model with polynomial or non-parametric components, thereby each component/group may be expressed as a linear combination of basis functions of the original variables. In this context, the selection of important variables corresponds to the selection of groups of basis functions. [109] have proposed group-Lasso as an extension of the Lasso for achieving group-wise variable selection. Although theoretical consistency can be achieved by the Lasso and group-Lasso estimators if one assumes some regularity assumptions on the design matrix (e.g. the restricted eigenvalue or compatibility conditions [7, 67]), in general, both estimators introduce bias for the model parameter estimation in high dimensions. To reduce bias and achieve oracle properties, [99, 75] have introduced extensions of non-convex penalties (SCAD, MCP) for group-variable selection.

Recent advances in data collection from multi-sources in many areas of research such as genomics, economics, and finance, generate an error accumulation in data pre-processing, and the assumption of homoscedasticity does not hold. To remedy this problem, flexible methods, which incorporate heteroscedasticity in modelling such data, are necessary to take into account the specificity of the collected datasets [98]. In the standard regression, the conditional mean function is explained by a linear combination of the predictors and its estimation results from minimizing a squared error loss function, which assigns equal weights to the residuals. On the opposite, when different weights are assigned to residuals, an exhaustive description of the outcome conditional distribution can be explored. [74] have introduced the Expectile Regression (ER) in which a squared error loss function puts different weights on the residuals, depending on their signs. Like the quantile regression [52], ER is appropriate to detect heteroscedasticity since both methods use an asymmetric loss function to estimate the regression function linking the outcome to the predictors.

Inspired by the success of sparse quantile regression [72], many advances have been made on variable selection in ER under high dimensional settings. For instance, [115] studied penalized ER with the SCAD penalty. [33] developed a unified and efficient algorithm, which fits ER with the Lasso penalty and uses the LLA approximation to handle the non-convex penalties SCAD and MCP. [33] have also established the estimation consistency of the Lasso selector under the restricted eigenvalue condition and the generalized invertability factor (GIF) condition [107, 40], and proved the convergence of LLA algorithm to the oracle estimator in two steps. Moreover, [33] have developed the oracles properties for their proposed estimators under the assumption that the model errors follow a sub-Gaussian distribution. [62] provided asymptotic distributions of penalized expectile regression with SCAD and adaptive Lasso penalties for both independent and identically distributed (i.i.d.) and non-i.i.d. random errors. Furthermore, penalized ER approaches have been introduced in the context of semi- and non-parametric methods where the penalty is used to impose smoothness for non-parametric estimators [45, 86, 103, 106].

When dealing with heteroscedastic high dimensional data, it is of interest to differentiate between predictors that are relevant for the mean and scale/variance of the outcome when they have overlap. In low dimension, [23] have proposed a method which combines both symmetric and asymmetric least squares loss functions to differentiate the effects of the important variables for the mean and the scale simultaneously. Such a resulting combined loss function has also been studied in high dimensions by [33] as an extension of the ER approach. The authors have established theoretical proprieties and efficient algorithms for the proposed estimators, and termed the approach the COupled Sparse Asymmetric LEast Squares regression, COSALES for short.

In this paper, we consider the problem of selecting grouped variables (factors) for accurate prediction in ER and Coupled ER. We extend the computational algorithms and the consistency results of [33] from Lasso and non-convex penalties to group Lasso and group non-convex penalties. First, we propose a bloc coordinate descent (BCD) algorithm for group Lasso and group non-convex penalties, which uses an efficient approach to minimize each sub-problem exactly. Moreover, we propose the group local linear approximation (GLLA) algorithm as an alternative approach for solving ER with the non-convex penalties SCAD and MCP. Yet, we demonstrate that if the GLLA algorithm starts with a reasonable initial estimator, we obtain the oracle estimator in a one-step iteration. Finally, we derive necessary conditions for the consistency of our ER and Coupled ER estimators by adapting both the generalized invertability factor (GIF) and the compatibility condition [14] to the group variable selection context.

The plan of the paper is as follows : in Section 3.3, we briefly review ER and we present our approach termed the Group Penalized Expectile Regression (GPER). In Section 3.4, we present our Coupled GPER framework. Evaluation of the performance of our methods through exhaustive simulation studies is considered in Section 3.5. The use of the proposed methodology is illustrated by analysing real datasets, in Section 3.6.

Discussion is given in Section 3.7. All the proofs are postponed to the appendix.

3.3 Expectile regression and group penalizations

3.3.1 Overview of the unconditional expectile

The τ-mean (or τ-expectile) of a continuous random variable Y is defined as the solution of the following problem

$$\mathscr{E}^\tau(Y) = \underset{\mathscr{E}\in\mathbb{R}}{\arg\min}\, \mathbb{E}\{\rho_\tau(Y - \mathscr{E})\}, \quad \tau \in (0,1), \tag{3.1}$$

where

$$\rho_\tau(u) := |\tau - \mathbb{1}_{(u\leq 0)}|u^2 \tag{3.2}$$

is known as the asymmetric square loss function, which assigns weights τ and $1-\tau$ to positive and negative deviations, respectively.

By equating the first derivative of (3.1) to zero, one has

$$\mathbb{E}\{\psi_\tau(Y - \mathscr{E}^\tau)(Y - \mathscr{E}^\tau)\} = 0, \tag{3.3}$$

where $\psi_\tau(u) := |\tau - \mathbb{1}_{(u\leq 0)}|$ is the check function. The solution of (3.3) leads to a more meaningful definition of the τ-mean, which is given as follows

$$\mathscr{E}^\tau(Y) = \mathbb{E}\left[\frac{\psi_\tau(Y-\mathscr{E}^\tau)}{\mathbb{E}\big[\psi_\tau(Y-\mathscr{E}^\tau)\big]}Y\right].$$

When $\tau = 0.5$, $\psi_{0.5}(u) = 0.5$ and $\mathscr{E}^\tau$ reduces to the mean of Y, (i.e. $\mathscr{E}^{0.5}(Y) = \mathbb{E}[Y]$). Thus, the τ-expectile can be viewed as a generalization of the mean, and like the mean, $\mathscr{E}^\tau$ is a weighted average with random weights. By varying τ, the τ-expectile provides insight at different “locations” of the distribution of Y and thus it is an alternative measure of “locations” of the distribution.

Given a random sample, $\{(y_i)\}_{i=1}^n$, the τ-th empirical expectile

$$\widehat{\mathscr{E}}_\tau = \sum_{i=1}^n \frac{\psi_\tau(y_i - \widehat{\mathscr{E}}_\tau)}{\sum_{i=1}^n \psi_\tau(y_i - \widehat{\mathscr{E}}_\tau)} y_i$$

is the solution that minimizes the empirical loss function

$$\frac{1}{n}\sum_{i=1}^n \rho_\tau(y_i - \mathscr{E}).$$

The extension of the expectile concept to regression has been investigated by [74]. Let $\{(y_1, \mathbf{x}_1), \cdots, (y_n, \mathbf{x}_n)\}$ be an observed data, where y_i is the observed response and $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})^\top$ is a p-dimensional observed vector of predictors for subject $i = 1, \ldots, n$. Note $\mathbf{X}$ the design matrix with n rows and p columns. If an intercept is used in the model, we let the first column of $\mathbf{X}$ be a vector of **1**. The ER model uses the weighted least squares loss $\rho_\tau(u)$ given in (3.2) to assign different weights to negative and positive residuals, and assumes that conditional τ-expectile given the predictors, denoted as $\mathscr{E}^\tau(\mathbf{x}_i)$, is a linear function of $\mathbf{x}_i$ (i.e. $\mathscr{E}^\tau(\mathbf{x}_i) = \mathbf{x}_i^\top \boldsymbol{\beta}_\tau$). This leads to the following estimator of the regression coefficients

$$\hat{\boldsymbol{\beta}}_\tau = \underset{\boldsymbol{\beta}_\tau}{\arg\min} \left(\mathit{\Psi}_\tau(\boldsymbol{\beta}_\tau) := \frac{1}{n} \sum_{i=1}^{n} \rho_\tau(y_i - \mathbf{x}_i^\top \boldsymbol{\beta}_\tau) \right). \tag{3.4}$$

Again, when $\tau = 0.5$, the ER model reduces to the ordinary least squares regression.

Several theoretical properties of the ER model have been established under some assumptions about the random error term of the regression model [74]

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta}_\tau + \boldsymbol{\epsilon}_\tau, \tag{3.5}$$

where $\boldsymbol{\epsilon}_\tau$ is the vector of n independent errors, which satisfies $\mathscr{E}^\tau(\boldsymbol{\epsilon}_\tau|\mathbf{X}) = \mathbf{0}$ for some $\tau \in (0, 1)$. Therefore $\mathscr{E}^\tau(\mathbf{y}|\mathbf{X}) = \mathbf{X}\boldsymbol{\beta}_\tau$, which is to say that the conditional τ-mean of $\mathbf{y}$ is a linear combination of the columns of $\mathbf{X}$. In the ER model, the estimated coefficients $\boldsymbol{\beta}_\tau$ vary as a function of τ, which makes modeling of different "locations" of the conditional distribution possible, and as a consequence heteroscedasticity when it exists, can be investigated by this model. For ease of notation, the subscript in $\boldsymbol{\beta}_\tau$ and $\boldsymbol{\epsilon}_\tau$ is dropped hereafter.

In this work, we focus on the ER model (3.5) with a pre-defined group structure, i.e. we assume that there is a natural grouping of the regression predictors. We assume that the predictors $x_1, x_2 \ldots x_p$ are put into K groups ($\{1, 2, 3 \ldots p\} = \cup_{k=1}^{K} I_k$), such that the size of each group is p_k (the cardinality of index set I_k is p_k) and the groups are non-overlapping ($I_k \cap I_{k'} =$ for $k \neq k'$). This leads to the block representation of $\boldsymbol{\beta} = (\boldsymbol{\beta}_1^\top, ..., \boldsymbol{\beta}_K^\top)^\top$.

In general, the GPER model in high dimensions can be formulated as a minimization problem

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\arg\min} \left(R_\tau(\boldsymbol{\beta}) := \mathit{\Psi}_\tau(\boldsymbol{\beta}) + \sum_{k=1}^{K} w_k P_\lambda(\|\boldsymbol{\beta}_k\|_2) \right), \tag{3.6}$$

with $(\hat{\boldsymbol{\beta}}_k)_{k=1,\ldots,K}$ the sub-vector of $\hat{\boldsymbol{\beta}}$ corresponding to the effects of the predictors belonging to group k for the τ-expectile of the response. $P_\lambda(\cdot)$ is the penalty function with a regularization parameter λ, and w_k

is used to adjust for the group sizes in the penalty. A reasonable choice is $w_k = \sqrt{p_k}$. If the intercept is included in (3.6), then $w_1 = 0$ is taken, which means that the first group is not penalized.

In this work, we consider the group Lasso (GLasso), group MCP (GMCP) and group SCAD (GSCAD) penalties which are defined respectively by the penalty function, $P_\lambda(t)$, as follows

$$\lambda t, \qquad (3.7) \qquad \begin{cases} (\lambda t - \dfrac{t^2}{2\theta}) & \text{if } 0 \le t \le \theta\lambda, \\ \frac{1}{2}\lambda^2\theta & \text{if } t \ge \theta\lambda, \end{cases} \qquad (3.8)$$

$$\begin{cases} \lambda t & \text{if } 0 \le t \le \lambda, \\ \dfrac{\theta\lambda t - (t^2+\lambda^2)/2}{\theta - 1} & \text{if } \lambda \le t \le \theta\lambda, \\ \dfrac{\lambda^2(\theta^2-1)}{2(\theta-1)} & \text{if } t \ge \theta\lambda, \end{cases} \qquad (3.9)$$

where θ is a second tuning parameter of GMCP and GSCAD penalties, with $\theta > 1$ for GMCP and $\theta > 2$ for GSCAD. In this work, we set $\theta = 4$ for GSCAD and $\theta = 3$ for GMCP, which are suggested values for this tuning parameter, for more details about optimal values of θ see [27, 75].

The non-convex penalties (3.8) and (3.9) enjoy the oracle property [27, 28], which means that they achieve the asymptotic equivalence to the ideal non-penalized estimator (oracle estimator) whose coefficients of irrelevant groups of variables equal to zero in advance. That is, the GSCAD and GMCP estimators can perform as well as the oracle estimator if the penalization parameter is appropriately chosen. For the GPER regression, the oracle estimator is defined by

$$\hat{\boldsymbol{\beta}}^{oracle} = \underset{\boldsymbol{\beta} \in \mathbb{R}^p : \boldsymbol{\beta}_{\mathcal{A}^c} = \mathbf{0}}{\arg\min} \; \Psi_\tau(\boldsymbol{\beta}), \qquad (3.10)$$

where $\mathcal{A}$ is the true support set.

The main difficulty of solving the optimization problem (3.6) is that the loss function $\rho_\tau(.)$ in (3.4) does not have the second derivative everywhere. To overcome this problem, we adopt the Majorization-Minimization principle (MM) and the block coordinate descent (BCD) algorithm to find the optimal solution by iteratively minimizing a surrogate function that majorizes the objective function in (3.4), for each group (i.e. block-/group-wise minimization) [103, 77]. In fact, the penalty $\sum_{k=1}^{K} w_k P_\lambda(\|\boldsymbol{\beta}_k\|_2)$ in (3.6) is group-wise separable. This property is used to make group-wise update in each iteration over one group of variables k $(k = 1, \dots, K)$. This technical resolution is detailed next.

3.3.2 GPER algorithm

This section gives details about the group-wise descent algorithm for the expectile regression with GLasso, GMCP and GSCAD penalties.

Let $\tilde{\boldsymbol{\beta}} = (\tilde{\boldsymbol{\beta}}_1, \dots, \tilde{\boldsymbol{\beta}}_{k-1}, \tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{k+1}, \dots, \tilde{\boldsymbol{\beta}}_K)$ be the current iteration and $\tilde{\boldsymbol{\beta}}_{-k} = (\tilde{\boldsymbol{\beta}}_1, \dots, \tilde{\boldsymbol{\beta}}_{k-1}, \tilde{\boldsymbol{\beta}}_{k+1}, \dots, \tilde{\boldsymbol{\beta}}_K)$ be the current iterate with k^{th} group excluded. Assume we are about to update the effects of the k^{th} group $\boldsymbol{\beta}_k = (\beta_1, \dots, \beta_{p_k})^\top$ for some $k \in \{1, \dots, K\}$. Also, consider both the objective function $R_\tau(\boldsymbol{\beta})$ in (3.6) and the ER loss function in (3.4) as functions of the k^{th} group, $\boldsymbol{\beta}_k$, while keeping all other groups fixed at $\tilde{\boldsymbol{\beta}}_{-k}$, i.e., $R_\tau(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k}) = R_\tau(\boldsymbol{\beta})_{\boldsymbol{\beta}_{k'} = \tilde{\boldsymbol{\beta}}_{k'}, 1 \le k' \le K, k' \neq k}$ and $\Psi_\tau(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k}) = \Psi_\tau(\boldsymbol{\beta})_{\boldsymbol{\beta}_{k'} = \tilde{\boldsymbol{\beta}}_{k'}, 1 \le k' \le K, k' \neq k}$. The following proposition summarizes the quadratic majorization property for $R_\tau(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k})$, which leads to solve our problem efficiently for each group k.

Proposition 3.1 *Let $\mathbf{X}_k$ be the sub-matrix of $\mathbf{X}$ corresponding to group k. The quadratic majorization condition is satisfied by the function $\Psi_\tau(.)$. That is, for all $\boldsymbol{\beta}$ and $\tilde{\boldsymbol{\beta}}$ we have*

$$\begin{aligned} R_\tau(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k}) \le \quad & Q(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k}) := \Psi_\tau(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}) + (\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top \nabla_k \Psi_\tau(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}) \\ & + \tfrac{\gamma_k}{2} (\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top (\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k) + w_k P_\lambda(\|\boldsymbol{\beta}_k\|_2), \end{aligned} \tag{3.11}$$

where γ_k is the largest eigenvalue of the matrix $\mathbf{H}_k = c\dfrac{\mathbf{X}_k^\top \mathbf{X}_k}{n}$, with $c = 2\max(1-\tau, \tau)$.

The proof of Proposition (3.1) is detailed in Appendix **A** of section 3.8.1.

Replacing the penalty term $P_\lambda(\|\boldsymbol{\beta}_k\|_2)$ by (3.7), (3.8) or (3.9) in (3.11) leads to a closed form solution of the update, $\tilde{\boldsymbol{\beta}}_k^{\text{new}}$, for the three penalties. The following proposition summarizes these results.

Proposition 3.2 *Let $Q(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k})$ be the surrogate function given by* (3.11) *and let $P_\lambda(\|\boldsymbol{\beta}_k\|_2)$ be one of the tree penalties given in* (3.7), (3.8) *and* (3.9). *The closed form solution to* (3.11) *of $\tilde{\boldsymbol{\beta}}_k^{\text{new}}$ for GPER algorithm* *with GLasso, GMCP and GSCAD penalties is given respectively by*

$$\tilde{\boldsymbol{\beta}}_k^{\mathrm{new}} = F(\mathbf{Z}_k) \longleftarrow \frac{1}{\gamma_k} \frac{S(\|\mathbf{Z}_k\|_2, \lambda w_k)}{\|\mathbf{Z}_k\|_2}\, \mathbf{Z}_k, \tag{3.12}$$

$$\tilde{\boldsymbol{\beta}}_k^{\mathrm{new}} = F(\mathbf{Z}_k) \longleftarrow \begin{cases} \frac{1}{\gamma_k - w_k/\theta} \frac{S(\|\mathbf{Z}_k\|_2, \lambda w_k)}{\|\mathbf{Z}_k\|_2}\, \mathbf{Z}_k, & \textit{if } \|\mathbf{Z}_k\|_2 \leq \gamma_k \theta \lambda \\ \frac{1}{\gamma_k} \mathbf{Z}_k, & \textit{if } \|\mathbf{Z}_k\|_2 > \gamma_k \theta \lambda, \end{cases} \tag{3.13}$$

$$\tilde{\boldsymbol{\beta}}_k^{\mathrm{new}} = F(\mathbf{Z}_k) \longleftarrow \begin{cases} \frac{1}{\gamma_k} \frac{S(\|\mathbf{Z}_k\|_2, \lambda w_k)}{\|\mathbf{Z}_k\|_2}\, \mathbf{Z}_k, & \textit{if} \quad \|\mathbf{Z}_k\|_2 \leq (w_k + \gamma_k)\lambda \\ \frac{S(\|\mathbf{Z}_k\|_2, \frac{\lambda w_k \theta}{\theta - 1})}{(\gamma_k - \dfrac{w_k}{\theta - 1})\|\mathbf{Z}_k\|_2} \mathbf{Z}_k, & \textit{if} \quad (w_k + \gamma_k)\lambda < \|\mathbf{Z}_k\|_2 \leq \gamma_k \theta \lambda \\ \frac{1}{\gamma_k} \mathbf{Z}_k, & \textit{if} \quad \|\mathbf{Z}_k\|_2 > \gamma_k \theta \lambda, \end{cases} \tag{3.14}$$

where $\mathbf{Z}_k = \mathbf{U}_k^\tau + \gamma_k \tilde{\boldsymbol{\beta}}_k$, $\mathbf{U}_k^\tau = -\nabla_k \Psi_\tau(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k})$ *and* $S(.)$ *is the soft-thresholding operator given by*

$$S(z, \lambda) = \begin{cases} z - \lambda \ \textit{ if } \ z > \lambda \\ 0 \ \textit{ if } \ |z| \leq \lambda \\ z + \lambda \ \textit{ if } \ z < -\lambda. \end{cases}$$

The proof of Proposition (3.2) is detailed in Appendix **B** of section 3.8.2.

The following algorithm gives some details about the groupwise descent algorithm for GPER with GLasso, GMCP and GSCAD penalties :

Algorithm 4: The GPER algorithm for GLasso/GMCP/GSCAD penalties

```
Initialize $\tilde{\boldsymbol{\beta}}$;
repeat
    for $k = 1, 2, \ldots, K$ do
        $\tilde{\boldsymbol{\beta}}_k^{\text{new}} \leftarrow F(\mathbf{Z}_k)$
    end
    with $F(\cdot)$ is given by (3.12), (3.13) and (3.14) for GLasso, GMCP and GSCAD penalties, respectively.
until Convergence of $\tilde{\boldsymbol{\beta}}$;
Return $\tilde{\boldsymbol{\beta}}$.
```

3.3.3 ER with Group Local Linear Approximation (GLLA) penalty

Proposition 3.2 allows us to provide an explicit solution for two important special non-convex penalty functions (GMCP and GSCAD). In this section, we propose to extend the local linear approximation trick to solve ER for a more general form of non-convex penalties. We restrict our theory development in Section 3.3.6 for a class of non-convex penalties that satisfy certain conditions. This class includes GMCP and GSCAD.

The GLLA approximation is based on first order Taylor expansion of the non-convex penalty functions around $\|\tilde{\boldsymbol{\beta}}_k\|_2$. Thus, one can write

$$P_\lambda(\|\boldsymbol{\beta}_k\|_2) \approx P_\lambda(\|\tilde{\boldsymbol{\beta}}_k\|_2) + P'_\lambda(\|\tilde{\boldsymbol{\beta}}_k\|_2)(\|\boldsymbol{\beta}_k\|_2 - \|\tilde{\boldsymbol{\beta}}_k\|_2). \tag{3.15}$$

Substituting (3.15) in (3.6) leads to the following GPER problem with the Group LLA (GLLA) penalty

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \left(\Psi_\tau(\boldsymbol{\beta}) + \sum_{k=1}^{K} w'_k \|\boldsymbol{\beta}_k\|_2 \right), \tag{3.16}$$

where $w'_k = w_k P'_\lambda(\|\tilde{\boldsymbol{\beta}}_k\|_2)$ for $k = 1, \ldots, K$. The weight w'_k depends on the non-convex penalty function through the first derivative $P'_\lambda(\|\tilde{\boldsymbol{\beta}}_k\|_2)$. The problem (3.16) can be solved using a GPER-GLasso update similar to the algorithm described in Section 3.3.2.

The details of the GPER approach with GLLA penalty is described in the following algorithm.

Algorithm 5: The GPER algorithm with GLLA penalty

```
Initialize $i = 0$; $\tilde{\boldsymbol{\beta}}^i = \tilde{\boldsymbol{\beta}}^{initial}$;
Compute the weights $\tilde{w}_k^i = w_k P'_\lambda(\|\tilde{\boldsymbol{\beta}}_k^i\|_2)$ for $k = 1, \ldots, K$;
repeat
    — $i \leftarrow i + 1$;
    — Solve the following convex optimization problem for $\hat{\boldsymbol{\beta}}^i$
          $\tilde{\boldsymbol{\beta}}^i = \arg\min_{\boldsymbol{\beta}} \left( \Psi_\tau(\boldsymbol{\beta}) + \sum_{k=1}^{K} \tilde{w}_k^{i-1} \|\boldsymbol{\beta}_k\|_2 \right);$    (3.17)
    — $\tilde{w}_k^i = w_k P'_\lambda(\|\tilde{\boldsymbol{\beta}}_k^i\|_2)$ for $k = 1, \ldots, K$;
until Convergence of $\tilde{\boldsymbol{\beta}}^i$;
Return $\tilde{\boldsymbol{\beta}}$.
```

To solve the problem (3.17), we use Algorithm 4 with GLasso (GPER-GLasso) with $w_k = \tilde{w}_k^{i-1}$ for $k = 1, \ldots, K$. Note that the GLLA penalty is a convex approximation of non-convex penalties (e.g. GMCP, GSCAD). Thus, for each fixed value of λ, GLLA allows a search of the solution in a locally convex region, and consequently it may lead to stable and smooth path solutions.

3.3.4 Implementation

We discuss some techniques used in our implementation to further improve the computational speed of the algorithm GPER. In sparse modelling, the solution is computed by using a descending sequence $(\lambda_m)_{m=1}^M$ of λ values. To generate such a sequence, we set $M-2$ points uniformly (in the log-scale) between the starting and ending points, $\lambda_{\max}$ and $\lambda_{\min}$, where $\lambda_{\max}$ is the smallest λ to let all groups $\boldsymbol{\beta}_k$ to be zero $(2 \leq k \leq K)$, except the intercept. To determine $\lambda_{\max}$, firstly, we initially estimate the intercept β_0 by considering the null model :

$$\widehat{\beta}_0 = \arg\min_{\beta_0} \frac{1}{n} \sum_{i=1}^{n} \rho_\tau(y_i - \beta_0). \tag{3.18}$$

Subsequently, according to the KKT conditions of (3.18), we can obtain the following formula

$$\lambda_{\max} = \max_{k=2,\ldots,K} \frac{\|\nabla_k \Psi_\tau(\widehat{\beta}_0, \mathbf{0})\|_2}{\omega_k}.$$

We take $\lambda_{\min} = \upsilon\lambda_{\max}$ and we set the default value of υ to be 10^{-2} for data with $n > p$ and $\upsilon = 10^{-4}$ for data with $n \leq p$. We also adopt the warm-start trick to implement the solution paths along λ values (i.e. assume that we have already computed the solution $\tilde{\boldsymbol{\beta}}_k^{(m)}$ ($k = 1, \ldots, K$) at λ_m, then $\tilde{\boldsymbol{\beta}}_k^{(m)}$ will be used as the initial value for computing the solution at λ_{m+1} in Algorithm 4. We refer readers to [77, 105] for more details about such computational techniques.

3.3.5 Theory for GPER-GLasso

We assume a fixed design for the covariates. Before presenting our principal theoretical results (Theorems), some notations must be defined and some necessary results must be shown. Let $\mathcal{A} \equiv$ supp$(\boldsymbol{\beta}^*) = \{k = 1, \ldots, K : \boldsymbol{\beta}_k^* \neq 0\}$ and $\mathcal{B} \equiv \{j = 1, \ldots, p : \beta_j^* \neq 0\}$ be the active set of the true vector of parameters $\boldsymbol{\beta}^*$. For any sequence $\{a_i\}_{i\in\mathcal{A}}$, denote $\underline{a}_{\mathcal{A}} = \min_{i\in\mathcal{A}} a_i$ and $\overline{a}_{\mathcal{A}} = \max_{i\in\mathcal{A}} a_i$. For any vector $\mathbf{v} = (\mathbf{v}_1^\top, \ldots, \mathbf{v}_K^\top)^\top \in \mathbb{R}^p$ and an arbitrary index set $I \subset \{1, \ldots, K\}$, we write $\mathbf{v}_I = (\mathbf{v}_k^\top, k \in I)^\top$ and define $\mathbf{X}_I = (\mathbf{X}_k, k \in I)$ to be the sub-matrix consisting of the columns of $\mathbf{X}$ with indices in I. Sub-Gaussian norm ([82]) of a random variable Z is denoted by $\|Z\|_{SG} = \text{sup}_{r\geq 1} r^{-1/2}(E(|Z|^r))^{1/r}$. Let $\overline{c} = \tau \vee (1-\tau) = \max(\tau, 1-\tau)$ and $\underline{c} = \tau \wedge (1-\tau) = \min(\tau, 1-\tau)$. We use $\nabla f(\mathbf{v}) = \partial f(\mathbf{v})/\partial\mathbf{v}$ to represent the gradient of a differentiable function $f : \mathbb{R}^p \to \mathbb{R}$, and we denote $\nabla_I f(\mathbf{v}) = (\partial f(\mathbf{v})/\partial v_k, k \in I)$. The $\ell_{2,1}$-norm and $\ell_{2,\infty}$-norm of $\mathbf{v}$ are defined by $\|\mathbf{v}\|_{2,1} = \sum_{k=1}^K \|\mathbf{v}_k\|_2$ and $\|\mathbf{v}\|_{2,\infty} = \max_{1\leq k\leq K} \|\mathbf{v}_k\|_2$. Denote s be the number of no null groups for the true coefficients $\boldsymbol{\beta}^*$, $s_{\mathcal{A}} = \sum_{k\in\mathcal{A}} p_k$ the number of variables in the set $\mathcal{A}$, $\overline{p}_m = \max_{1\leq k\leq K} p_k$ and $\overline{p}_{\mathcal{A}} = \max_{k\in\mathcal{A}} p_k$. Let $\lambda_{\min}(.)$ and $\lambda_{\max}(.)$ are two functions that return the smallest and largest eigenvalues of a symmetric matrix respectively, and define $\overline{\rho} = \max_{1\leq k\leq K} \rho_k$ and $\underline{\rho} = \min_{1\leq k\leq K} \rho_k$, where $\rho_k = \lambda_{\max}(n^{-1}\mathbf{X}_k^\top\mathbf{X}_k)$. Finally, let $\rho_{\min} = \lambda_{\min}(n^{-1}\mathbf{X}_{\mathcal{B}}^\top\mathbf{X}_{\mathcal{B}})$ and $\rho_{\max} = \lambda_{\max}(n^{-1}\mathbf{X}_{\mathcal{B}}^\top\mathbf{X}_{\mathcal{B}})$. We assume $\rho_{\min} > 0$, thereby the important variables are not linearly dependent. Define $[a]^+ = \max(0, a)$ for any $a \in \mathbb{R}$.

Let $\mathcal{C}_3 = \{\boldsymbol{\delta} \in \mathbb{R}^p, \|\boldsymbol{\delta}_{\mathcal{A}^c}\|_{2,1} \leq 3\|\boldsymbol{\delta}_{\mathcal{A}}\|_{2,1}\}$ be a cone in $\mathbb{R}^p$. To study the estimation accuracy of the GPER-Lasso, we impose the following conditions on the design matrix $\mathbf{X}$ and the random errors $\boldsymbol{\epsilon}$.

— (C1) The columns of $\mathbf{X}$ are normalizable, that is, $M_0 = \max_{1\leq j\leq p} \dfrac{\|X_j\|_2}{\sqrt{n}} \in (0, \infty)$;

— (C2) The random errors ϵ_i are i.i.d. sub-Gaussian random variables satisfying $\mathscr{E}^\tau(\epsilon_i) = 0$, for $i = 1, \ldots, n$;

— (C3) $\kappa = inf_{\boldsymbol{\delta}\in\mathcal{C}_3} \dfrac{\|\mathbf{X}\boldsymbol{\delta}\|_2^2}{n\|\boldsymbol{\delta}\|_{2,1}} \in (0, \infty)$;

- (C4) $\varrho = inf_{\boldsymbol{\delta} \in \mathcal{C}_3} \dfrac{\|\mathbf{X}\boldsymbol{\delta}\|_2^2}{n\|\boldsymbol{\delta}_{\mathcal{A}}\|_{2,1}\|\boldsymbol{\delta}\|_{2,\infty}} \in (0, \infty)$.

The consistency of the group Lasso estimator has been extensively studied in the literature under some conditions [67, 14]. Condition (C3) is known as the restricted eignvalue condition and has been frequently assumed in the literature to study the group Lasso [68]. Condition (C4) is an extension of the generalized invertibility factor (GIF) condition for group variables [107]. Both conditions $(C3)$ and $(C4)$ are crucial assumptions to establish estimation consistency of the GPER-Lasso, for high-dimensional data.

Théorème 3.3 *Assume the true vector of coefficients $\boldsymbol{\beta}^*$ in* (3.5) *is s-sparse (s is the number of no null groups) and assume the conditions (C1)-(C2). Let $\hat{\boldsymbol{\beta}}^{\text{GLasso}}$ be any optimal solution to GPER-Lasso problem. Then with probability at least $1-p^*$, we have $\|\hat{\boldsymbol{\beta}}^{\text{GLasso}} - \boldsymbol{\beta}^*\|_{2,1} \leq 3\lambda^{\text{GLasso}}(4\kappa\underline{c})^{-1}$ if condition (C3) holds, and $\|\hat{\boldsymbol{\beta}}^{\text{GLasso}} - \boldsymbol{\beta}^*\|_{2,\infty} \leq 3\lambda^{\text{GLasso}}(4\underline{c}\varrho)^{-1}$ if condition (C4) holds,*
where

$$p^* = 2p\,\textit{exp}\bigg(- \frac{Cn(\lambda^{\text{GLasso}})^2}{4K_0^2 M_0^2 \overline{p}_m}\bigg), \tag{3.19}$$

$\nu_0 = var(\Psi_\tau'(\epsilon_i))$, $K_0 = \|\Psi_\tau'(\epsilon_i)\|_{SG}$ and $C > 0$ is an absolute constant.

The proof of Theorem 3.3 is detailed in Appendix **D** of section 3.8.4.

3.3.6 Theory for non-convex penalized GPER

To give a unified theoretical analysis of GMCP and GSCAD, we assume that the penalty $P_\lambda(t)$ is a general folded concave penalty function defined on $t \in (-\infty, \infty)$ satisfying (see [29],[33]) :

1. (P1) $P_\lambda(t) = P_\lambda(-t)$;
2. (P2) $P_\lambda(t)$ is non-decreasing, concave in $t \in [0, \infty)$ and $P_\lambda(0) = 0$;
3. (P3) $P_\lambda(t)$ is differentiable in $t \in (0, \infty)$;
4. (P4) $P_\lambda'(t) \geq a_1\lambda$ for $t \in (0, a_2\lambda]$ and $P_\lambda'(0) := P_\lambda'(0+) \geq a_1\lambda$;
5. (P5) $P_\lambda'(t) = 0$ for $t \in [a\lambda, \infty)$ with some prespecified constant $a > a_2$.

The parameters a_1 and a_2 are fixed constants characterising the penalty function. One can verify that a

corresponds to θ in equations (3.8) and (3.9) for both penalties GMCP and GSCAD, respectively, and $a_1 = a_2 = 1$ for GSCAD, and $a_1 = 1 - \theta^{-1}$, $a_2 = 1$ for GMCP.

In the following theorem, we show that the solution given by GLLA Algorithm 5 with any non-convex penalty satisfying the above conditions (1)-(5), enjoys the oracle property. Assume we have a sufficient signal strength in the nonzero components of $\boldsymbol{\beta}^*$. That is, assume $(A1)$ min$_{k \in \mathcal{A}} \|\boldsymbol{\beta}_k^*\|_2 > (a+1)\lambda$. Our result is outlined next.

Théorème 3.4 *Assume in model* (3.5) *the vector of the true coefficients* $\boldsymbol{\beta}^*$ *is s-sparse and satisfies assumption (A1). Assume conditions (C1)-(C2) hold and take* $\hat{\boldsymbol{\beta}}_{GLasso}$ *as the initial value in Algorithm 5. Let* $a_0 = 1 \wedge a_2$. *Take* $\lambda \geq 3\lambda^{\text{GLasso}}(4\kappa\underline{c}a_0)^{-1}$ *when (C3) holds, or* $\lambda \geq 3\lambda^{\text{GLasso}}(4\underline{c}\varrho a_0)^{-1}$ *when (C4) holds, or take* $\lambda \geq 3\lambda^{\text{GLasso}} a_0^{-1}\big((4\underline{c}\varrho)^{-1} \wedge (4\kappa\underline{c})^{-1}\big)$ *when both (C3) and (C4) hold. The GPER-GLLA estimator converges to* $\hat{\boldsymbol{\beta}}^{oracle}$ *after two iterations with probability at least* $1 - p_1 - p_2 - p_3$, *where* $p_1 = p^*$ *is given by* (3.19),

$$p_2 = 2(p - s_{\mathcal{A}})\, \textit{exp}\Big(-\frac{Cn\lambda^2 a_1^2}{4K_0^2 M_0^2 \overline{p}_m}\Big) + \Gamma(Q_1\lambda, n, s_{\mathcal{A}}, K_0, M_0, \rho_{\max}, \nu_0),$$

and

$$p_3 = \Gamma(2\underline{c}\rho_{\min} R \overline{p}_{\mathcal{A}}^{-1}; n, s_{\mathcal{A}}, K_0, M_0, \rho_{\max}, \nu_0),$$

where $Q_1 = \dfrac{a_1 \underline{c} \rho_{\min}}{2\overline{c} M_0 \rho_{\max}^{1/2} \overline{p}_m^{-1/2}}$, $\nu_0 = var(\Psi_\tau'(\epsilon_i))$, $R =$ *min*$_{k \in \mathcal{A}} \|\boldsymbol{\beta}_k^*\|_2 - a\lambda > 0$, K_0 *is defined in Theorem 3.3,* $\Gamma(x; n, s, K, M, \rho, \nu)$ *is given by*

$$\Gamma(x; n, s, K, M, \rho, \nu) = 2\textit{exp}\bigg(-\frac{C\nu^2[(n^{1/2}x - \nu\rho^{1/2}s^{1/2})]^+}{K^4\rho}\bigg) \wedge 2s\textit{exp}\bigg(-\frac{Cnx^2}{K^2M^2s}\bigg),$$

and $C > 0$ *is an absolute constant.*

The proof of Theorem 3.4 is detailed in Appendix **D** of section 3.8.4.

3.3.7 Some solution paths of GPER methods

Our motivation for introducing the GPER approach is illustrated in Figure 3.1 below.

We adopted the illustration example of [71]. We generated one dataset of $n = 50$ observations and five initial predictors, say $\tilde{X}_k$ $(k = 1, \ldots, K = 5)$, from a multivariate standard normal distribution with the correlation among the predictors was set to be equal to 0.5. We computed a cubic B-spline basis (W_k^1, W_k^2, W_k^3) from each predictor $\tilde{X}_k,\ k = 1, \ldots, 5$. Then we set $X_k^j = W_k^j$, for $j = 1, 2, 3$ and $k = 1, \ldots, 5$, which leads to 15 predictors X_k^j that are clustered in $K = 5$ groups, i.e. $G_k = \{X_k^1, X_k^2, X_k^3\}$, for $k = 1, \ldots, 5$.

The response $\mathbf{y}$ is generated as :

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \Phi(\tilde{X}_1)\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim N(0, 1),$$

where $\Phi(\cdot)$ is the cumulative distribution function of the univariate standard normal distribution. Using $\Phi(\cdot)$ in the term of the variance in the simulations is considered by many authors to generate a model with heteroscedasticity [98, 33].

In this illustration example, we consider two scenarios. In the first scenario, we considered that G_2 and G_3 have an effect on the mean of the outcome and G_1 has an effect only on the scale. Thus, $\boldsymbol{\beta}$ is defined as

$$\boldsymbol{\beta} = (\underbrace{0, 0, 0}_{G_1}, \underbrace{2, 2, 2}_{G_2}, \underbrace{-1, -1, -1}_{G_3}, \underbrace{0, 0, 0}_{G_4}, \underbrace{0, 0, 0}_{G_5}).$$

The second scenario is similar to the first one, except that G_1 has an effect on both the mean and scale (i.e. overlapping effect). That is, $\boldsymbol{\beta}$ is given by

$$\boldsymbol{\beta} = (\underbrace{1, 1, 1}_{G_1}, \underbrace{2, 2, 2}_{G_2}, \underbrace{-1, -1, -1}_{G_3}, \underbrace{0, 0, 0}_{G_4}, \underbrace{0, 0, 0}_{G_5}).$$

Figure 3.1 shows the results of the coefficient profiles as a function of λ values for GPER-GLasso, at different locations. In the first two panels (from the left to right), we show major advantages of using group penalized expectile regression approaches when τ is different than 0.5 ($\tau \neq 0.5$) for detecting heteroscedasticity when the groups of variables have an effect only on the scale. Indeed, GPER-GLasso selected the Group G_1 (blue color) for $\tau = 0.95$, but it does not for $\tau = 0.5$, whcih means that G_1 is detected as a heteroscedastic group. However, in the second scenario (two last panels of Figure 3.1), the effect of G_1 overlaps for the mean and scale. In this case, GPER-GLasso selected G_1 for both values of $\tau = 0.5$ and 0.95, and thus, one cannot answer the question if G_1 is a heteroscedastic group or not. This is the main motivation to introduce the COupled (Group) Expectile Regression for analyzing the heteroscedasticity in high-dimensional settings.

3.4 COupled Group Penalized Expectile Regression : COGPER

3.4.1 Methodology : COGPER general algorithm

We consider the following linear scale model for analyzing heteroscedasticity

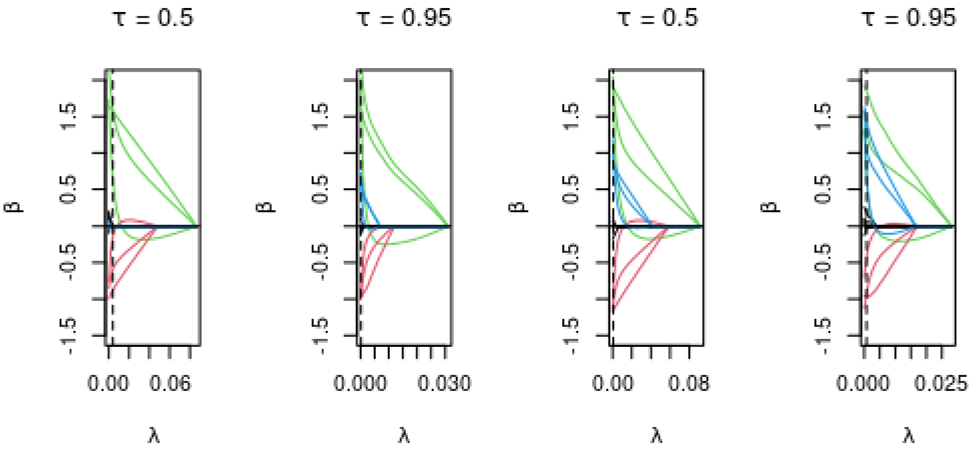


Figure 3.1 – From left to right, the coefficients' profiles corresponding to LS-GLasso (GPER-GLasso with $\tau = 0.5$) and GPER-GLasso with $\tau = 0.95$ for the first and the second scenario respectively are plotted as a function of the tuning parameter λ. The dashed vertical lines report selected optimal λ using 5-fold CV. The group coefficients of G_1, G_2 and G_3 are plotted in blue, green and red colors, respectively. The black color corresponds to the noisy groups of predictors G_4 and G_5.

$$y_i = \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{x}_i^\top \boldsymbol{\gamma} \epsilon_i,$$

where ϵ_i are i.i.d. random errors, and we assume that $\mathbb{E}(\epsilon_i) = 0$. The unknown parameters to be estimated are the p-dimensional vectors $\boldsymbol{\beta}$ and γ, corresponding to the effect of the covariates on the mean and the scale of the response variable, respectively. We suppose that $\mathbf{x}_i^\top \gamma > 0$ for all i. This model has been studied by many authors in standard regression [23, 55]. It has been proposed by [33] in high dimension to select important variables that have an effect on both the mean and the scale functions. Let $e_\tau = \mathscr{E}^\tau(\epsilon_1)$ be the τ-mean of the random error for $\tau \in (0, 1)$, then the τ-mean of y_i given $\mathbf{x}_i$ is $\mathscr{E}^\tau(y_i|\mathbf{x}_i) = \mathbf{x}_i^\top(\boldsymbol{\beta} + \gamma e_\tau)$. Let $\mathcal{A}_1 \equiv supp(\boldsymbol{\beta}^*) = \{k : \boldsymbol{\beta}_k^* \neq 0\}$ and $\mathcal{A}_2 \equiv supp(\boldsymbol{\gamma}^*) = \{k : \boldsymbol{\gamma}_k^* \neq 0\}$ be the active sets of $\boldsymbol{\beta}^*$ and of $\boldsymbol{\gamma}^*$, respectively. Then, when we take $\boldsymbol{\phi} = \gamma e_\tau$, we will deal with $\boldsymbol{\phi}$ instead of γ, and if $e_\tau \neq 0$ we have $supp(\boldsymbol{\gamma}^*) \equiv supp(\boldsymbol{\phi}^*)$.

We rely again on [33] and develop the COupled Group Expectile Regression (COGPER) method, which estimates and selects the relevant groups of variables that have effect on the mean and scale simultaneously.

The COGPER model is defined as follows

$$(\widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\phi}}) = \underset{(\boldsymbol{\beta},\boldsymbol{\phi})\in\mathbb{R}^{2p}}{\arg\min}\; S_\tau(\boldsymbol{\beta}, \boldsymbol{\phi}) + \sum_{k=1}^{K} w_k P_{\lambda_1}(\|\boldsymbol{\beta}_k\|_2) + \sum_{k=1}^{K} u_k P_{\lambda_2}(\|\boldsymbol{\phi}_k\|_2), \tag{3.20}$$

where

$$S_\tau(\boldsymbol{\beta}, \boldsymbol{\phi}) = \mathit{\Psi}_{0.5}(\boldsymbol{\beta}) + \mathit{\Psi}_\tau(\boldsymbol{\beta}, \boldsymbol{\phi}), \tag{3.21}$$

with $\mathit{\Psi}_{0.5}(\boldsymbol{\beta})$ is given by (3.4) and

$$\mathit{\Psi}_\tau(\boldsymbol{\beta}, \boldsymbol{\phi}) = \frac{1}{n}\sum_{i=1}^{n} \rho_\tau(y_i - \mathbf{x}_i^\top \boldsymbol{\beta} - \mathbf{x}_i^\top \boldsymbol{\phi}).$$

The penalties $P_{\lambda_1}(.)$ and $P_{\lambda_2}(.)$ could be one of the penalties GLasso, GMCP or GSCAD defined in (3.7), (3.8) and (3.9), respectively. The scalars w_k and u_k are known weights for each group, and can be defined in a similar way as in the GPER approach to control for the group size, for instance. In this work, we set $w_k = u_k = \sqrt{p_k}$.

Notice that the non-convex penalties, GMCP and GSCAD, enjoy the oracle property. For the COGPER approach, the oracle estimators of $\boldsymbol{\beta}$ and $\boldsymbol{\phi} = \boldsymbol{\gamma} e_\tau$ are given by

$$(\hat{\boldsymbol{\beta}}^{oracle}, \hat{\boldsymbol{\phi}}^{oracle}) = \underset{(\boldsymbol{\beta},\boldsymbol{\phi})\in\mathbb{R}^{2p}:\boldsymbol{\beta}_{\mathcal{A}_1^c}=\mathbf{0},\boldsymbol{\phi}_{\mathcal{A}_2^c}=\mathbf{0}}{\arg\min} S_\tau(\boldsymbol{\beta}, \boldsymbol{\phi}). \tag{3.22}$$

where $\mathcal{A}_1$ and $\mathcal{A}_2$ are the true support set of $\boldsymbol{\beta}$ and $\boldsymbol{\phi}$ respectively.

To solve the problem (3.20), we proceed in a similar way as in Section 3.3. That is, we focus on updating one group at a time ($\boldsymbol{\beta}_k$ or $\boldsymbol{\phi}_k$). We majorize each loss function in the right-hand side of (3.21) by a quadratic surrogate function. Then, for each group k ($k = 1, \ldots, K$), we obtain two upper bound approximations for updating $\boldsymbol{\beta}_k$ and $\boldsymbol{\phi}_k$, respectively, as follows

$$\begin{aligned} Q_1(\boldsymbol{\beta}_k|\tilde{\boldsymbol{\beta}}_{-k}, \tilde{\boldsymbol{\phi}}) \;\; := \;\; & S_\tau(\tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\phi}}) - 2(\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top \mathbf{U}_k^{0.5} + \\ & 2\gamma_k(\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top(\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k) - (\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top \mathbf{U}_k^\tau + \\ & 2c\gamma_k(\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top(\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k) + P_{\lambda_1}(\|\boldsymbol{\beta}_k\|_2), \end{aligned} \tag{3.23}$$

and

$$\begin{aligned} Q_2(\boldsymbol{\phi}_k|\tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\phi}}_{-k}) \;\; := \;\; & S_\tau(\tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\phi}}) - (\boldsymbol{\phi}_k - \tilde{\boldsymbol{\phi}}_k)^\top \mathbf{U}_k^\tau \\ & +2c\gamma_k(\boldsymbol{\phi}_k - \tilde{\boldsymbol{\phi}}_k)^\top(\boldsymbol{\phi}_k - \tilde{\boldsymbol{\phi}}_k) + P_{\lambda_2}(\|\boldsymbol{\phi}_k\|_2), \end{aligned} \tag{3.24}$$

where γ_k is the largest eigenvalue of the matrix $\mathbf{H}_k = \mathbf{X}_k^\top \mathbf{X}_k, c = 2\max(\tau, 1-\tau), \mathbf{U}_k^{0.5} = -\nabla_k \Psi_{0.5}(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k})$ and $\mathbf{U}_k^\tau = -\nabla_{\boldsymbol{\beta}_k} \Psi_\tau(\tilde{\boldsymbol{\beta}}_k; \tilde{\boldsymbol{\beta}}_{-k}, \tilde{\boldsymbol{\phi}}) = -\nabla_{\boldsymbol{\phi}_k} \Psi_\tau(\tilde{\boldsymbol{\phi}}_k; \tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\phi}}_{-k})$.

Proposition 3.5 *Let $Q_1(\boldsymbol{\beta}_k | \tilde{\boldsymbol{\beta}}_{-k}, \tilde{\boldsymbol{\phi}})$ and $Q_2(\boldsymbol{\phi}_k | \tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\phi}}_{-k})$ be the surrogate loss functions given by* (3.23) *and* (3.24). *Let $P_{\lambda_1}(\|\boldsymbol{\beta}_k\|_2)$ and $P_{\lambda_2}(\|\boldsymbol{\phi}_k\|_2)$ be one of the three penalties given in* (3.7), (3.8) *and* (3.9). *The closed form solutions to* (3.23) *and* (3.24) *of $(\tilde{\boldsymbol{\beta}}_k^{(new)}, \tilde{\boldsymbol{\phi}}_k^{(new)})$ for COGPER-GLasso, COGPER-GMCP and COGPER-GSCAD are, respectively, given by*

$$\tilde{\boldsymbol{\beta}}_k^{(new)} = F(\mathbf{Z}_k) \longleftarrow \frac{1}{2(1+c)\gamma_k} \frac{S(\|\mathbf{Z}_k\|_2, \lambda_1 w_k)}{\|\mathbf{Z}_k\|_2} \mathbf{Z}_k$$

$$\tilde{\boldsymbol{\phi}}_k^{(new)} = G(\mathbf{W}_k) \longleftarrow \frac{1}{2c\gamma_k} \frac{S(\|\mathbf{W}_k\|_2, \lambda_2 u_k)}{\|\mathbf{W}_k\|_2} \mathbf{W}_k$$

$$\tilde{\boldsymbol{\beta}}_k^{(new)} = F(\mathbf{Z}_k) \longleftarrow \begin{cases} \frac{1}{2(1+c)\gamma_k - 1/\theta} \frac{S(\|\mathbf{Z}_k\|_2, \lambda_1 w_k)}{\|\mathbf{Z}_k\|_2} \mathbf{Z}_k, \\ \qquad \text{if } \|\mathbf{Z}_k\|_2 \leq 2(1+c)\gamma_k \theta \lambda_1 w_k \\ \frac{1}{2(1+c)\gamma_k} \mathbf{Z}_k, \\ \qquad \text{if } \|\mathbf{Z}_k\|_2 > 2(1+c)\gamma_k \theta \lambda_1 w_k \end{cases}$$

$$\tilde{\boldsymbol{\phi}}_k^{(new)} = G(\mathbf{W}_k) \longleftarrow \begin{cases} \frac{S(\|\mathbf{W}_k\|_2, \lambda_2 w_k)}{2c\gamma_k - 1/\theta} \frac{1}{\|\mathbf{W}_k\|_2} \mathbf{W}_k, & \text{if } \|\mathbf{W}_k\|_2 \leq 2c\gamma_k \theta \lambda_2 u_k \\ \frac{1}{2c\gamma_k} \mathbf{W}_k & \text{if } \|\mathbf{W}_k\|_2 > 2c\gamma_k \theta \lambda_2 u_k, \end{cases}$$

$$\tilde{\boldsymbol{\beta}}_k^{(new)} = F(\mathbf{Z}_k) \longleftarrow \begin{cases} \frac{S(\|\mathbf{Z}_k\|_2, \lambda_1 w_k)}{2(1+c)\gamma_k \|\mathbf{Z}_k\|_2} \mathbf{Z}_k, \\ \qquad \text{if} \quad \|\mathbf{Z}_k\|_2 \leq (1 + 2(1+c)\gamma_k)\lambda_1 w_k \\ \frac{S(\|\mathbf{Z}_k\|_2, \frac{\lambda_1 w_k \theta}{\theta - 1})}{\|\mathbf{Z}_k\|_2 (2(1+c)\gamma_k - \frac{1}{\theta - 1})} \mathbf{Z}_k, \\ \qquad \text{if} \quad (1 + 2(1+c)\gamma_k)\lambda_1 w_k < \|\mathbf{Z}_k\|_2 \leq 2(1+c)\gamma_k \theta \lambda_1 w_k \\ \frac{1}{2(1+c)\gamma_k} \mathbf{Z}_k \\ \qquad \text{if} \quad \|\mathbf{Z}_k\|_2 > 2(1+c)\gamma_k \theta \lambda_1 w_k, \end{cases}$$

$$\tilde{\boldsymbol{\phi}}_k^{(new)} = G(\mathbf{W}_k) \longleftarrow \begin{cases} \frac{1}{2c\gamma_k}\frac{S(\|\mathbf{W}_k\|_2,\lambda_2 u_k)}{\|\mathbf{W}_k\|_2}\mathbf{W}_k, \\ \qquad \textit{if} \quad \|\mathbf{W}_k\|_2 \leq (1+2c\gamma_k)\lambda_2 u_k \\ \frac{1}{2c\gamma_k - \dfrac{1}{\theta-1}}\frac{S(\|\mathbf{W}_k\|_2,\frac{\lambda_2 u_k \theta}{\theta-1})}{\|\mathbf{W}_k\|_2}\mathbf{W}_k, \\ \qquad \textit{if} \quad (1+2c\gamma_k)\lambda_2 u_k < \|\mathbf{W}_k\|_2 \leq 2c\gamma_k\theta\lambda_2 u_k \\ \frac{1}{2c\gamma_k}\mathbf{W}_k \quad \textit{if} \quad \|\mathbf{W}_k\|_2 > 2c\gamma_k\theta\lambda_2 u_k, \end{cases}$$

where $\mathbf{Z}_k = \mathbf{U}_k^{0.5} + \mathbf{U}_k^{\tau} + 2(1+c)\gamma_k\widetilde{\boldsymbol{\beta}}_k$ *and* $\mathbf{W_k} = \mathbf{U}_k^{\tau} + 2c\gamma_k\widetilde{\boldsymbol{\phi}}_k$.

The proof of Proposition (3.5) is detailed in Appendix **C** of section 3.8.3.

The following algorithm summarizes the steps of the COGPER framework with GLasso, GMCP and GSCAD penalties.

Algorithm 6: The COGPER algorithm for GLasso/GMCP/GSCAD penalties

```
Initialize $(\tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\phi}})$;
repeat
    for $k = 1, 2, \ldots, K$ do
        $\tilde{\boldsymbol{\beta}}_k^{\mathrm{new}} \leftarrow F(\mathbf{Z}_k)$
    end
    for $k = 1, 2, \ldots, K$ do
        $\tilde{\boldsymbol{\phi}}_k^{\mathrm{new}} \leftarrow G(\mathbf{W}_k)$
    end
    with $F(\cdot)$ and $G(\cdot)$ are given by proposition (3.5) for GLasso, GMCP and GSCAD penalties;
until Convergence of ($\tilde{\boldsymbol{\beta}}$, $\tilde{\boldsymbol{\phi}}$);
Return ($\tilde{\boldsymbol{\beta}}$, $\tilde{\boldsymbol{\phi}}$).
```

3.4.2 Coupled expectile regression with GLLA penalty

Extension of the GLLA trick to solve coupled ER for a more general form of non-convex penalties can be done in a same way as described in Section 3.3.3. Our theoretical contribution in Section 3.4.5 is focuced on a class of non-convex penalties. This class includes GMCP and GSCAD.

Using the first order Taylor expansion of the non-convex penalty functions around $\|\tilde{\boldsymbol{\beta}}_k\|_2$ and $\|\tilde{\boldsymbol{\phi}}_k\|_2$ as defined in (3.15) leads to the following COGPER problem with the Group LLA (GLLA) penalty

$$(\widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\phi}}) = \underset{(\boldsymbol{\beta},\boldsymbol{\phi})\in\mathbb{R}^{2p}}{\arg\min} \big(S_\tau(\boldsymbol{\beta}, \boldsymbol{\phi}) + \sum_{k=1}^{K} w'_k \|\boldsymbol{\beta}_k\|_2 + \sum_{k=1}^{K} u'_k \|\boldsymbol{\phi}_k\|_2\big), \tag{3.25}$$

where $(w'_k, u'_k) = (w_k P'_{\lambda_1}(\|\tilde{\boldsymbol{\beta}}_k\|_2), u_k P'_{\lambda_2}(\|\tilde{\boldsymbol{\phi}}_k\|_2))$ for $k = 1, \ldots, K$. The weights w'_k and u'_k depend on the non-convex penalty function through the first derivative $P'_\lambda(.)$. The problem (3.25) can be solved using a COGPER-GLasso update similar to Algorithm 6. The details of the COGPER approaches with GLLA penalty is given in the next algorithm.

```
**Algorithm 7:** The COGPER algorithm with GLLA penalty
Initialize $i = 0$; $(\tilde{\boldsymbol{\beta}}^i, \tilde{\boldsymbol{\phi}}^i) = (\tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\phi}})^{initial}$.
Compute the weights $(\tilde{w}^i_k, \tilde{u}^i_k) = (w_k P'_{\lambda_1}(\|\tilde{\boldsymbol{\beta}}^i_k\|_2), u_k P'_{\lambda_2}(\|\tilde{\boldsymbol{\phi}}^i_k\|_2))$ for $k = 1, \ldots, K$;
repeat
    — $i \leftarrow i + 1$;
    — Solve the following convex optimization problem for $(\tilde{\boldsymbol{\beta}}^i, \tilde{\boldsymbol{\phi}}^i)$
        $(\tilde{\boldsymbol{\beta}}^i, \tilde{\boldsymbol{\phi}}^i) = \underset{(\boldsymbol{\beta},\boldsymbol{\phi})\in\mathbb{R}^{2p}}{\arg\min} \left(S_\tau(\boldsymbol{\beta}, \boldsymbol{\phi}) + \sum_{k=1}^{K} \tilde{w}^{i-1}_k \|\boldsymbol{\beta}_k\|_2 + \sum_{k=1}^{K} \tilde{u}^{i-1}_k \|\boldsymbol{\phi}_k\|_2\right);$   (3.26)
    — calculate $(\tilde{w}^i_k, \tilde{u}^i_k) = (w_k P'_{\lambda_1}(\|\tilde{\boldsymbol{\beta}}^i_k\|_2), u_k P'_{\lambda_2}(\|\tilde{\boldsymbol{\phi}}^i_k\|_2)$ for $k = 1, \ldots, K$;
until convergence of $(\tilde{\boldsymbol{\beta}}^i, \tilde{\boldsymbol{\phi}}^i)$;
Return $(\tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\phi}})$.
```

To solve the problem (3.26), we use Algorithm 6 with GLasso (COGPER-GLasso) with $(w_k, u_k) = (\tilde{w}^{i-1}_k, \tilde{w}^{i-1}_k)$ for $k = 1, \ldots, K$.

3.4.3 Implementation

To obtain the solution path of COGPER with the tuning parameters (λ_1, λ_2), one can choose a (relatively small) grid of values for λ_1 and then compute a grid of values of λ_2 covering the entire range, and vice versa. But, the resulting coefficients' path solution might not be smooth with several successive jumps. To remedy this problem, we follow [33] in their implementation and set a common tuning parameter in solving the problem (3.21) for the two penalties, as follows

$$(\widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\phi}}) = \underset{\boldsymbol{\beta}, \boldsymbol{\phi}}{\arg\min} \left(\nu \mathit{\Psi}_{0.5}(\boldsymbol{\beta}) + \mathit{\Psi}_{\tau}(\boldsymbol{\beta}, \boldsymbol{\phi}) \right) + \sum_{k=1}^{K} w_k P_{\lambda}(\boldsymbol{\beta}_k) + \sum_{k=1}^{K} u_k P_{\lambda}(\boldsymbol{\phi}_k), \tag{3.27}$$

where ν is an additional weight parameter for the mean loss function, which compensates for the use of the common tuning parameter for both the mean and scale coefficients. In our implementation we set the default value of $\nu = 1$ as in the `SALES` R package [33], but other values of ν can also be investigated.

The implementation of problem (3.27) has the advantage of allowing smooth path solutions for both $\boldsymbol{\beta}$ and $\boldsymbol{\phi}$. To calculate $\lambda_{\max}$, we first obtain estimates of the intercepts $(\hat{\beta}_0, \hat{\phi}_0)$ through the null model wit all the groups' coefficients are set to be zero

$$(\hat{\beta}_0, \hat{\phi}_0) = \underset{\beta_0, \phi_0}{\arg\min} \left(S_{\tau}^{\nu}(\beta_0, \phi_0) := \nu \mathit{\Psi}_{0.5}(\beta_0, \mathbf{0}) + \mathit{\Psi}_{\tau}(\beta_0, \mathbf{0}, \phi_0, \mathbf{0}) \right).$$

According to KKT conditions, we have

$$\lambda_{\max} = \max \left\{ \max_{k=2,..,K} \|(\nabla_{\boldsymbol{\beta}_k} S_{\tau}^{\nu}(\hat{\beta}_0, \hat{\phi}_0))\|_2 / w_k, \ \max_{k=2,..,K} \|(\nabla_{\boldsymbol{\phi}_k} S_{\tau}^{\nu}(\hat{\beta}_0, \hat{\phi}_0))\|_2 / u_k \right\}.$$

Let $\lambda_{\min} = \eta \lambda_{\max}$, where $\eta = 0.001$ if $n \leq p$; otherwise, $\eta = 0.05$. We take $M - 2 = 98$ points uniformly in log-scale between $\lambda_{\min}$ and $\lambda_{\max}$. This sequence is denoted by $[\lambda^m]_{m=1}^{M}$. We use the warm-start and the strong rule tricks to speed up our code ; see Section 3.3.4 and [77] for more details.

3.4.4 Theory for COGPER-GLasso

For the COGPER-GLasso approach, let $\mathcal{A}_1 \equiv$ supp$(\boldsymbol{\beta})$ and $\mathcal{A}_2 \equiv$ supp$(\boldsymbol{\phi})$ be the active group of $\boldsymbol{\beta}^*$ and of $\boldsymbol{\phi}^*$ respectively. Let $\mathcal{A}_0 = (\mathcal{A}_1, \mathcal{A}_2')$, where $\mathcal{A}_2' = \{k + K : \boldsymbol{\phi}_k^* \neq 0\}$. For $N \geq 1$, define $\xi_N = \{\boldsymbol{\delta} \in \mathbb{R}^{2p} : \|\boldsymbol{\delta}_{\mathcal{A}_0^c}\|_{2,1} \leq N \|\boldsymbol{\delta}_{\mathcal{A}_0}\|_{2,1}\}$, $\underline{\lambda}^{\text{GLasso}} = \lambda_1^{\text{GLasso}} \wedge \lambda_2^{\text{GLasso}} = \min(\lambda_1^{\text{GLasso}}, \lambda_2^{\text{GLasso}})$ and $\overline{\lambda}^{\text{GLasso}} = \lambda_1^{\text{GLasso}} \vee \lambda_2^{\text{GLasso}} = \max(\lambda_1^{\text{GLasso}}, \lambda_2^{\text{GLasso}})$ and $\tilde{N} = \overline{\lambda}^{\text{GLasso}} / \underline{\lambda}^{\text{GLasso}}$. For $k = 1, 2$, denote $\rho_{k,max} = \lambda_{\max}(n^{-1} \mathbf{X}_{\mathcal{A}_k}^{\top} \mathbf{X}_{\mathcal{A}_k})$, $\rho_{k,min} = \lambda_{\min}(n^{-1} \mathbf{X}_{\mathcal{A}_k}^{\top} \mathbf{X}_{\mathcal{A}_k})$, $\phi_{\min} = \rho_{k,min} \wedge \rho_{k,max}$, $\phi_{\max} = \rho_{k,min} \vee \rho_{k,max}$, and we assume $\phi_{\min} > 0$. Let $\boldsymbol{I}_2$ be a 2×2 identity matrix and $\otimes$ denotes the Kronecker product. To establish an error bound for the COGPER-GLasso estimator, the following conditions on the design matrix, $\mathbf{X}$, and the random errors, $\boldsymbol{\epsilon}$, are imposed :

- (C1') The columns of $\mathbf{X}$ are normalizable, that is, $M_0 = \max_{1\leq j\leq p} \dfrac{\|X_j\|_2}{\sqrt{n}} \in (0,\infty)$;
- (C2') $M_1 = \|\mathbf{X}\boldsymbol{\phi}^*\|_\infty \in (0,\infty)$;
- (C3') The random errors ϵ_i are i.i.d. mean-zero sub-Gaussian random variables ;
- (C4') $\overline{\kappa} = \kappa(3\tilde{N}) \in (0,\infty)$ where $\kappa = inf_{\boldsymbol{\delta}\in\xi_N} \dfrac{\boldsymbol{\delta}^\top[\boldsymbol{I}_2 \otimes (n^{-1}\mathbf{X}^\top\mathbf{X})]\boldsymbol{\delta}}{n\|\boldsymbol{\delta}\|_{2,1}^2}$;
- (C5') $\overline{\varrho} = \varrho(3\tilde{N}) \in (0,\infty)$ where $\varrho = inf_{\boldsymbol{\delta}\in\xi_N} \dfrac{\boldsymbol{\delta}^\top[\boldsymbol{I}_2 \otimes (n^{-1}\mathbf{X}^\top\mathbf{X})]\boldsymbol{\delta}}{n\|\boldsymbol{\delta}_{\mathcal{A}_0}\|_{2,1}\|\boldsymbol{\delta}\|_{2,\infty}}$.

As Theorem 3.3, both conditions $(C'4)-(C'5)$ are crucial assumptions to establish the estimation consistency of the COGPER-GLasso estimator.

Théorème 3.6 *Suppose the true parameter vectors $\boldsymbol{\beta}^*$ and $\boldsymbol{\phi}^*$ are respectively s_1-sparse and s_2-sparse and assume conditions (C1')–(C3') hold. Let $\hat{\boldsymbol{\beta}}$ and $\hat{\boldsymbol{\phi}}$ be optimal solutions of COGPER-GLasso. Then, with probability at least $1-\pi^*$*

$$\left\| \begin{pmatrix} \hat{\boldsymbol{\beta}} \\ \hat{\boldsymbol{\phi}} \end{pmatrix} - \begin{pmatrix} \boldsymbol{\beta}^* \\ \boldsymbol{\phi}^* \end{pmatrix} \right\|_{2,1} \leq \frac{(3/2)\overline{\lambda}^{\mathrm{GLasso}}}{c_0\overline{\kappa}}$$

if the condition (C4') holds, and

$$\left\| \begin{pmatrix} \hat{\boldsymbol{\beta}} \\ \hat{\boldsymbol{\phi}} \end{pmatrix} - \begin{pmatrix} \boldsymbol{\beta}^* \\ \boldsymbol{\phi}^* \end{pmatrix} \right\|_{2,\infty} \leq \frac{(3/2)\overline{\lambda}^{\mathrm{GLasso}}}{c_0\underline{\varrho}}$$

if the condition (C5') holds, where

$$\pi^* = 2p\,\mathit{exp}\bigg(-\frac{Cn(\lambda_1^{\mathrm{GLasso}})^2}{4(K_1+K_2)^2M_0^2M_1^2\overline{p}_m}\bigg) + 2p\,\mathit{exp}\bigg(-\frac{Cn(\lambda_2^{\mathrm{GLasso}})^2}{4K_2^2M_0^2M_1^2\overline{p}_m}\bigg)$$

$c_0 = 2^{-1}[(1+\underline{c}) - (1+16\underline{c}^2)^{1/2}]$, $K_1 = \|\epsilon_i\|_{SG}$, $K_2 = \|S'_\tau(\epsilon_i - e_\tau)\|_{SG}$, and $C > 0$ is an absolute constant.

The proof of Theorem 3.6 is detailed in Appendix **D** of section 3.8.4.

3.4.5 Theory for non-convex penalized COGPER

In this section we investigate the theoretical properties of the COGPER approach with the non-convex penalties. More precisely, in the next theorem, we show that the solution given by Algorithm 7 converges to the oracle estimator in two steps. To do this, assume the following additional assumption

(A2) $\min_{k\in\mathcal{A}_1}\|\boldsymbol{\beta}_k^*\| > (a+1)\lambda_1$ and $\min_{k\in\mathcal{A}_2}\|\boldsymbol{\phi}_k^*\| > (a+1)|e_\tau|^{-1}\lambda_2$.

Théorème 3.7 *Suppose that $\boldsymbol{\beta}^*$ and $\boldsymbol{\phi}^*$ are respectively s_1-sparse and s_2-sparse. Take $\hat{\boldsymbol{\beta}}^{\mathrm{GLasso}}$ and $\hat{\boldsymbol{\phi}}^{\mathrm{GLasso}}$ as the initial values and assume conditions (C1')–(C3') hold. Take $\lambda \geq (3/2)(a_0c_0\overline{\kappa})^{-1}\overline{\lambda}^{\mathrm{GLasso}}$ when (C4') holds, or take $\lambda \geq (3/2)(a_0c_0\overline{\varrho})^{-1}\overline{\lambda}^{\mathrm{GLasso}}$ when (C5') holds, or take $\lambda \geq (3/2)\overline{\lambda}^{\mathrm{GLasso}}(a_0c_0)^{-1}\big(\kappa^{-1}\wedge\varrho^{-1}\big)$ when (C4') and (C5') hold. The COGPER-GLLA algorithm converges to the oracle estimators $(\hat{\boldsymbol{\beta}}^{oracle}, \hat{\boldsymbol{\phi}}^{oracle})$ in two iterations with probability at least $1-\pi_1-\pi_2-\pi_3$, where $\pi_1=\pi^*$ is given in Theorem 3.6, and*

$$
\begin{aligned}
\pi_2 = 2(p-s_{\mathcal{A}_1})\,\textit{exp}\bigg(-\frac{Cn\lambda^2a_1^2}{4M_0^2M_1^2(K_1+K_2)^2\overline{p}^2_{\mathcal{A}_1^c}}\bigg)\\
+\,2(p-s_{\mathcal{A}_2})\,\textit{exp}\bigg(-\frac{Cn\lambda^2a_1^2}{4M_0^2M_1^2K_2^2\overline{p}^2_{\mathcal{A}_2^c}}\bigg)\\
+\,\Gamma(Q_2\lambda/2;n,s_{\mathcal{A}_1},K_1+K_2,M_0,M_1,M_1^2\rho_{1,max},\nu_1)\\
+\,\Gamma(Q_2\lambda/2;n,s_{\mathcal{A}_2},K_2,M_0M_1,M_1^2\rho_{2,max},\nu_2),
\end{aligned}
$$

$$
\begin{aligned}
\pi_3 =&\Gamma\Big(c_0\phi_{\min}\frac{\overline{R}}{2\overline{p}_k};n,s_{\mathcal{A}_1},K_1+K_2,M_0,M_1,M_1^2\rho_{1,max},\nu_1\Big)\\
&+\Gamma\Big(c_0\phi_{\min}\frac{\overline{R}}{2\overline{p}_k};n,s_{\mathcal{A}_2},K_2,M_0M_1,M_1^2\rho_{2,max},\nu_2\Big),
\end{aligned}
$$

where $Q_2=\frac{a_1}{(1+2\overline{c})M_0\phi_{\max}^{1/2}}$, $\overline{R}=(1+a)\lambda_1\vee\lambda_2$, $\nu_1=var(\epsilon_i+\Psi_\tau'(\epsilon_i-\mathscr{E}_\tau))$, $\nu_2=var(\Psi_\tau'(\epsilon_i-\mathscr{E}_\tau))$, C, c_0, K_1, and K_2 are given in Theorem 3.6, and the function $\Gamma(.)$ is defined in Theorem 3.4.

The proof of Theorem 3.7 is detailed in Appendix **D** of section 3.8.4.

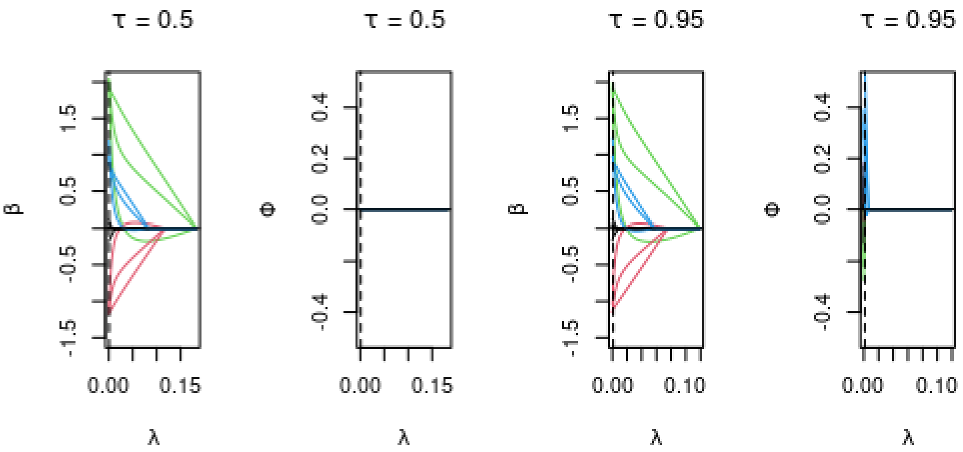


Figure 3.2 – From left to right, the coefficient profiles corresponding to $(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\phi}})$ obtained using COGPER-GLasso for $\tau \in (0.5, 0.95)$, respectively, are plotted as a function of the tuning parameter λ. The dashed line indicates the optimal value of λ using 5-fold CV. The group coefficients G_1, G_2 and G_3 are plotted in blue, green and red colors, respectively. The black color corresponds to the noisy groups G_4 and G_5.

3.4.6 Some solution paths of COGPER method

The motivation for the introduction of COGPER approach is illustrated in Figure 3.2. This figure was provided using the same dataset that is generated under the second model/scenario of the simulation study of Section 3.3.7. Recall that under this model, G_1 was generated with effect on both the mean and scale of the response variable.

Figure 3.2 shows that COGPER-GLasso has a tendency to select the groups of variables that have effect on the conditional τ-mean for $\tau \in (0.5, 0.95)$. Furthermore, the heteroscedastic effect of group G_1 in the scale function is often selected as non-null effect when fitting COGPER for 0.95th conditional mean (Blue-color group in the right panel), but it is not the case for $\tau = 0.5$. This shows that the COGPER not only can be used to detect the heteroscedastic group G_1, but can also estimates the amount of the heteroscedastic effect $\hat{\boldsymbol{\phi}}_1$ and separates it from the mean function effect $\hat{\boldsymbol{\beta}}_1$.

3.5 Numerical experiments

3.5.1 Simulation setting

We carried out a simulation study to illustrate the utility of the proposed approaches. We adapted the examples 1 and 2 of [33] to the additive model context, in which the response is modeled as a sum of functions of the covariates. Two scenarios were considered in the simulations. In both scenarios, we considered fitting an additive model of continuous factors represented by with B-splines basis functions. This means that the effect of the factors is represented through nonlinear functions. Our simulation results are based on a one independent dataset. This data is used to fit models and to select the tuning parameter using the 5-fold cross-validation (5-fold CV). We selected the regularization parameter by minimizing the CV error defined as

$$\frac{1}{n_{\text{validation}}} \sum_{i \in \text{validation}} \rho_\tau(y_i - \mathbf{x}_i^\top \hat{\boldsymbol{\beta}})$$

and

$$\frac{1}{n_{\text{validation}}} \sum_{i \in \text{validation}} \rho_{0.5}(y_i - \mathbf{x}_i^\top \hat{\boldsymbol{\beta}}) + \rho_\tau(y_i - \mathbf{x}_i^\top \hat{\boldsymbol{\beta}} - \mathbf{x}_i^\top \hat{\boldsymbol{\phi}})$$

for GPER and COGPER, respectively.

The first scenario was considered in [98] for the sparse quantile regression and in [33] for expectile regression. The predictors were generated in three steps. First, from the multivariate normal distribution $N(\mathbf{0}, \Sigma)$ with $\Sigma = (0.5^{|i-j|})_{K \times K}$, we draw n-dimensional samples from $(Z_1, \ldots, Z_K)$, where $K = 50$ and $n = 300$. Second, for each variable $Z_k, k = 1, \ldots, K$, we derived a cubic B-spline basis (W_k^1, W_k^2, W_k^3). In the third step, we set $X_1^l = \Phi(W_1^l)$ and $X_k^l = W_k^l$ for $k = 2, 3, \ldots, K$ and $l = 1, 2, 3$, where $\Phi(.)$ is the standard normal CDF. Thus, the design matrix is $300 \times (50 * 3)$ and is defined as $\mathbf{X} = [X_k^l]_{l,k}$, $k = 1, \ldots, 50$, and $l = 1, 2, 3$. The response variable is then simulated from the following linear heteroscedastic model :

$$Y = \underbrace{Z_6}_{G_6} + \underbrace{Z_{12}}_{G_{12}} + \underbrace{Z_{15}}_{G_{15}} + \underbrace{Z_{20}}_{G_{20}} + \underbrace{\Phi(Z_1)}_{G_1^\phi} \epsilon,$$

where $\epsilon \sim N(0, 1)$. Our aim is to select the active variables Z_i through their representation by the cubic B-spline sets/groups.

We compared GPER at two locations $\tau \in \{0.5, 0.85\}$ for the penalties GLasso, GMCP and GSCAD. We computed four statistics, over 100 datasets replication :

— $|\hat{\mathcal{A}}|$: the average number of nonzero group variables $\hat{\boldsymbol{\beta}}_k \neq 0$ for $k = 1, \ldots, p$.

- p_a : proportion of the event $\mathcal{A} \subset \hat{\mathcal{A}}$, where $\mathcal{A}$ is the true active set of $\boldsymbol{\beta}^*$. When $\tau = 0.5$, $\mathcal{A} = \{G_6, G_{12}, G_{15}, G_{20}\}$ and when $\tau = 0.85$, $\mathcal{A} = \{G_1, G_6, G_{12}, G_{15}, G_{20}\}$.
- p_1 : proportion of the event that $\{1\} \subset \hat{\mathcal{A}}$.

τ	Penalty	$\lvert\hat{\mathcal{A}}\rvert$	p_a	p_1
0.50	GLasso	11.02	100%	22%
	GMCP	6.38	100%	18%
	GSCAD	4.86	100%	8%
	GLLA-GMCP	6.24	100%	15%
	GLLA-GSCAD	4.89	100%	11%
0.85	GLasso	15.34	100%	94%
	GMCP	6.26	100%	86%
	GSCAD	6.36	100%	84%
	GLLA-GMCP	6.31	100%	87%
	GLLA-GSCAD	5.01	100%	89%

Table 3.1 – Simulation results of $|\hat{\mathcal{A}}|$, p_a and p_1 for example 1 based on 100 replications. The three statistics are calculated for GPER approach with all suggested group penalties.

From Table 3.1, one can see that GPER approach with the three penalties selects the true active groups, with the p_a statistic equals to 100%. On the other hand, the Size statistic $|\hat{\mathcal{A}}|$ reveals that GPER with GMCP and GSCAD has tendency to provide more accurate sparse models compared to GLasso. The statistic p_1 shows how many times the heteroscedastic group variable, represented by Z_1, is selected in each model fit. For $\tau = 0.5$, it is expected that Z_1 will be not selected since it has no effect on the center of y (p_1 is less than 22%). However, for $\tau = 0.85$, the proportion of selecting Z_1 is greater than 84%, for GPER with all penalties. The GPER approach detects the effect of heteroscedastic variable Z_1, but, it can not estimate its effect value. In order to do that, we take $\tau = 0.85$ for easy separation of the conditional mean and scale functions. Based on 100 independent runs, the following statistics are computed to evaluate the estimation performance and the sparsity recovery of the COGPER estimators :

- $|\hat{\mathcal{A}}_1|$, $|\hat{\mathcal{A}}_2|$: the average number of nonzero group variables for $\hat{\boldsymbol{\beta}}$ and for $\hat{\boldsymbol{\phi}}$, respectively, i.e., $\hat{\mathcal{A}}_1 = \{k, \hat{\boldsymbol{\beta}}_k \neq 0\}$ and $\hat{\mathcal{A}}_2 = \{k, \hat{\boldsymbol{\phi}}_k \neq 0\}$;
- p_{a_1}, p_{a_2} : proportion of the event $\mathcal{A}_1 \subset \hat{\mathcal{A}}_1, \mathcal{A}_2 \subset \hat{\mathcal{A}}_2$, where $\mathcal{A}_1$ and $\mathcal{A}_2$ are the active group sets of $\boldsymbol{\beta}^*$ and $\boldsymbol{\phi}^*$, respectively. In this first scenario, we have $\mathcal{A}_1 = \{G_6, G_{12}, G_{15}, G_{20}\}$ and $\mathcal{A}_2 = \{G_1^{\boldsymbol{\phi}}\}$.

τ	Penalty	$\lvert\hat{\mathcal{A}}_1\rvert$	$\lvert\hat{\mathcal{A}}_2\rvert$	p_{a_1}	p_{a_2}
0.85	GLasso	19.22	14.66	100%	98%
	GMCP	4.34	3.21	100%	90%
	GSCAD	4.04	2.96	100%	94%
	GLLA-GMCP	4.39	3.13	100%	92%
	GLLA-GSCAD	4.41	2.89	100%	95%

Table 3.2 – Simulation results of $|\hat{\mathcal{A}}_1|$, $|\hat{\mathcal{A}}_2|$, p_{a_1} and p_{a_2} for Scenario 1, based on 100 replications. The four statistics are calculated for the COGPER approach with all suggested group penalties.

In Table 3.2, the statistic p_{a_1} is always equals to 100% for COGPER with all penalties ; i.e., all groups of variables that have effect on the mean are selected. On the other hand, the statistic p_{a_2} shows how many times the heterogeneous group G_1, estimated as $\hat{\boldsymbol{\phi}}_1$, is selected. Thus, one can notice that p_{a_2} is always greater than 90%. This shows that the COGPER approach can be used to detect the effect of the heteroscedastic groups, and can also estimate the amount of the heteroscedastic effect and separate it from the effect on the mean function.

In the first example, the active sets of the true groups of variables do not overlap, so the GPER can detect active groups of variables in the scale. In the second example, we conducted a simulation scenario in which the active set of groups for the mean overlaps with the active set for the scale. More precisely, the procedure for generating the predictors and the groups in this scenario was similar to the first scenario, however we generated the response variable from the following linear heteroscedastic model

$$Y = \underbrace{Z_2}_{G_2} + \underbrace{Z_5}_{G_5} + \underbrace{Z_{10}}_{G_{10}} + \underbrace{Z_{15}}_{G_{15}} + (\underbrace{\Phi(Z_1)}_{G_1^{\phi}} + \underbrace{\Phi(Z_5)}_{G_5^{\phi}})\epsilon,$$

where $\epsilon \sim N(0,1)$. This means that the active set of groups for the mean, $\mathcal{A}_1 = \{G_2, G_5, G_{10}, G_{15}\}$, overlaps with the active set of groups for the scale, $\mathcal{A}_2 = \{G_1^{\boldsymbol{\phi}}, G_5^{\boldsymbol{\phi}}\}$. The group G_1 has effect only on the scale, but G_5 has effect on both the mean and the scale (i.e. an overlapping group effect). In this scenario we set $K = 400$ and $n = 300$. Thus, the design matrix $\mathbf{X}$ has 1200 columns and 300 rows. All results are based on 100 data replications. Table 3.3 shows the results of COGPER for this scenario, based on the four statistics $|\hat{\mathcal{A}}_1|$, $|\hat{\mathcal{A}}_2|$, p_{a_1}, and p_{a_2}, defined earlier.

τ	Penalty	$\lvert\hat{\mathcal{A}}_1\rvert$	$\lvert\hat{\mathcal{A}}_2\rvert$	p_{a_1}	p_{a_2}
0.85	GLasso	8.50	8.10	100%	95%
	GMCP	4.44	2.21%	100%	87%
	GSCAD	6.13	5.32	100%	93%
	GLLA-GMCP	4.52	3.07%	100%	94%
	GLLA-GSCAD	5.96	2.99	100%	95%

Table 3.3 – Simulation results of $|\hat{\mathcal{A}}_1|$, $|\hat{\mathcal{A}}_2|$, p_{a_1} and p_{a_2} for Scenario 2, based on 100 replications. The four statistics are calculated for the COGPER approach with all suggested group penalties.

From Table 3.3, one can derive similar conclusions for COGPER as in Table 3.2. The statistic p_{a_1} is always equals to 100% for COGPER with all penalties. The statistic p_{a_2} shows how many times the estimated effect of the heterogeneous groups on the scale (i.e. G_1^{ϕ} and G_5^{ϕ}) have non-zero values. This statistic is greater than 87% for COGPER with all penalties. This confirms good performance of the COGPER approach in disentangling the heterogeneous overlapping group.

3.5.2 Checking KKT condition

Since we are updating each group at a time and cycling between groups until convergence, in this section, we numerically demonstrate that the proposed algorithms satisfy the KKT conditions, which means that the algorithms converge and find the right solution.

The KKT conditions are given in Appendix **E** of section 3.8.5, for each method and each penalty. Here, we design the simulation model by using a modified simulation model in [109]. We simulate first the initial matrix of predictors $X_k, (k = 1, \ldots, K)$ from multivariate normal distribution with correlation $\rho = 0.5$ among the columns in the design matrix. Then, we considered $\{X_k, X_k^2, X_k^3\}$ as a group when fitting the two models GPER and COGPER, so the final predictor matrix has the number of variables $p = 3K$. The response variable was generated as follows

$$Y = \sum_{k=1}^{K} \left(\frac{2}{3} X_k - X_k^2 + \frac{1}{3} X_k^3 \right) \beta_k + \epsilon, \; \epsilon \sim N(0, \sigma^2),$$

where σ is chosen so that the signal-to-noise ratio (SNR) is 3 (i.e. $SNR = \|\mathbf{X}\boldsymbol{\beta}\|_2/\sqrt{n}\sigma$) and $\beta_k = (-1)^k \text{exp}(-(2k-1)/20)$. We considered two values for $K = 1000, 3000$, and we set $n = 100$.

Table (3.4) shows that all group-penalized expectile methods have zero violation count for the first scenario ($K = 1000$) and has also small violation counts for the second scenario ($K = 3000$). Thus, one can argue that all the proposed approaches are accurate algorithms that pass KKT checks without sever violation.

3.6 Real data

3.6.1 The Birth weight data

This dataset was collected by the Medical Center in Springfield, Massachusetts. It was used as an illustration example for demonstrating various aspects of regression modeling [39, 93]. It was also used to illustrate both the group penalized least squares and quantile regression models [109, 34]. The dataset records the birth weights of 189 babies in kilograms and eight predictors concerning their mothers. Among the eight predictors, two are continuous (mother's age in years and mother's weight in pounds at the last menstrual period), and six are categorical : mother's race with three levels (white, black or other), smoking status during pregnancy (yes = 1 or no = 0), number of previous premature labours with three levels (0, 1 or 2 or more), history of hypertension (yes=1 or no=0), presence of uterine irritability (yes=1 or no=0), number of physician visits during the first trimester with four levels (0, 1, 2 or 3 or more).

A preliminary analysis conducted in [93] suggests that non-linear effects of both mother's age and weight may exist. Thus, in our analysis the two continuous variables were represented through two third-order polynomials, i.e. the two continuous variables were considered as groups of three predictors. The categorical variables were considered as groups using dummy variables. So, each categorical predictor of l levels is represented by $l-1$ dummy variables. In summary, we have a total $p = 16$ predictors (i.e. 6 continuous and 10 dummy variables) that are grouped in $K = 8$ groups. A preliminary analysis of this data can be found in the `grpreg` R package.

The goal of this study is to identify the risk factors associated with the baby birth weight response variable.

		GPER			
				$GLLA-$	$GLLA-$
τ	$GLasso$	$GMCP$	$GSCAD$	$GMCP$	$GSCAD$
		$n = 100,\ \ K = 1000$			
0.50	0.00	0.01	0.01	0.01	0.01
0.85	0.00	0.00	0.00	0.00	0.00
		$n = 100,\ \ K = 3000$			
0.50	0.04	0.03	0.05	0.04	0.04
0.85	0.10	0.07	0.05	0.11	0.09
		COGPER			
		$n = 100,\ \ K = 1000$			
0.50	0.76	0.64	0.62	0.69	0.72
0.85	2.34	2.09	2.13	3.01	2.06
		$n = 100,\ \ K = 3000$			
0.50	1.10	1.11	1.12	1.10	1.12
0.85	3.01	2.91	3.23	3.31	2.93

Table 3.4 – Reported numbers are the average number of groups among K groups of variables that violated the KKT conditions check using GPER-GLasso, GPER-GMCP, GPER-GSCAD, COGPER-GLasso, COGPER-GMCP and COGPER-GSCAD. Results are averaged over the λ sequence of 100 values and averaged over 10 independent runs.

In particular, we aim to explore the effect of the mother smoking during pregnancy, which is known to have a heterogeneous effect on the baby birth weight [89]. An ANOVA analysis can be investigated to evaluate differences in the birth weight of the babies between the two groups : smoking versus non-smoking mothers. An F-value of 7.038 leads to a p-value of 0.00867 ; we can conclude that the mother's smoking status is significantly associated with the baby birth weight. However, ANOVA is a mean-based test. Thus, we adjusted both GPER and COGPER for three values of $\tau = 0.2, 0.5, 0.8$, with aim to capture the heterogeneous effect of the mother smoking in different expectiles/locations of the conditional distribution of the response variable.

We conducted two analyses for this data. Firstly, we fitted the GPER and COGPER with the group Lasso penalty for all 189 babies with 5-fold CV to obtain the optimal models for the three locations ($\tau = 0.2, 0.4, 0.8$). Figure 3.3 shows the coefficient path solutions of GPER and COGPER for this analysis. The solution of the optimal models is indicated by vertical lines, which indicate the optimal values of λ for each model fit. From this figure, one can notice that both GPER and COGPER tend to select three important groups of variables : mother's race (green), smoking status (blue) and uterine irritability (red) for all $\tau \in \{0.2, 0.5, 0.8\}$. The coefficient values corresponding to smoking-status effects are estimates of the total effect on both mean and scale functions when using GPER. This cannot tell us if this predictor has an overlapping effect (i.e. if this predictor is also relevant or not to the scale function). Interestingly, Figure 3.3 (bottom right panel) shows that COGPER selects the scale coefficient, $\hat{\phi}$, corresponding to smoking-status as a non zero effect, for $\tau = 0.8$ (blue path solution). This indicates that smoking-status might be a heterogeneous overlapping predictor. COGPER provides also estimates of the scale effect and thus it distinguishes it from the mean function effect.

In the second analysis, we randomly divided the data into a training sample of two-thirds observations and the remainder making up a test data. For the three values of τ, both GPER and COGPER were fitted to the training data to obtain the parameter estimates of the optimal models, where the optimal λ values were selected by 5-fold CV. The performance of the methods in this analysis is based on the following statistics, which are calculated on the test data :

— the estimates of the effects of the three variables that have been selected as relevant in the first analysis : Smoking status during pregnancy (blue), mother's race (green) and presence of uterine irritability (red). These estimates are calculated based on the optimal model of the training data analysis.

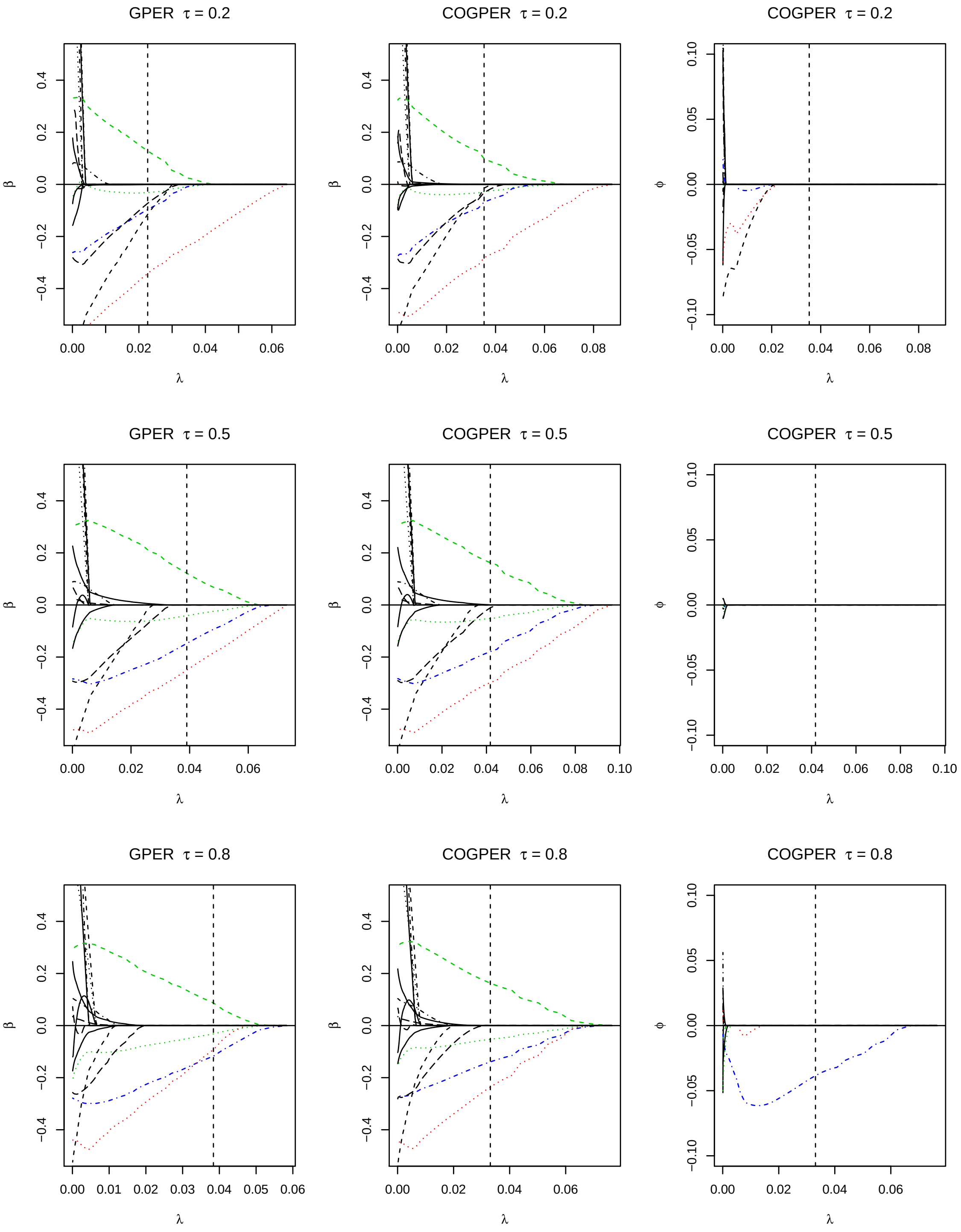


Figure 3.3 – At the left and from top to bottom, the coefficient paths of GPER with $\tau = 0.2$, 0.5 and 0.8 respectively, are shown as a function of the tuning parameter. At the middle and right columns, the coefficients paths $\boldsymbol{\beta}$ and $\boldsymbol{\phi}$ of COGPER respectively with $\tau \in \{0.2, 0.5, 0.8\}$.

— the model-size statistic (MS), which is defined as the number of selected groups for GPER. For COGPER, two MS statistics are needed : MS_1 to count the relevant groups for the mean function (i.e. $\hat{\boldsymbol{\beta}}_k \neq 0$), and MS_2 for counting the relevant groups for the scale function (i.e. $\hat{\boldsymbol{\phi}}_k \neq 0$). The MS statistic estimation is also based on the results of the optimal model of the training data analysis.

— the expectile-based prediction error (EPE), which is calculated on the test data, and is defined as

$$EPE_{\text{gper}} = \frac{1}{n_{\text{test}}} \sum_{i \in \text{test}} \rho_\tau(y_i - \mathbf{x}_i^\top \hat{\boldsymbol{\beta}})$$

for GPER, and

$$EPE_{\text{cogper}} = \frac{1}{n_{\text{test}}} \sum_{i \in \text{test}} (y_i - \mathbf{x}_i^\top \hat{\boldsymbol{\beta}})^2 + \rho_\tau(y_i - \mathbf{x}_i^\top \hat{\boldsymbol{\beta}} - \mathbf{x}_i^\top \hat{\boldsymbol{\phi}})$$

for COGPER.

The whole procedure was repeated 100 times, and we reported the empirical distribution (boxplots) of the aforementioned statistics.

Figure 3.4 highlights the results of the second analysis of GPER and COGPER for the first statistic. That is, based on 100 replications, this figure reports the empirical distribution of the point estimates $\hat{\boldsymbol{\beta}}_k$'s of the three important variables for the optimal model, which is obtained using 5-fold CV in the analysis of the training datasets. The empirical distribution of the estimates of the predictors' effects through the 100 replications demonstrates the consistency of both approaches to select the three variables as relevant predictors, in particular for $\tau \in \{0.5, 0.8\}$. Interestingly, Figure 3.4 (bottom middle panel) shows also that the distribution of the estimates of the smoking-status effect on the variance is non-null when fitting COGPER for the location $\tau = 0.8$.

Figure 3.5 shows the results of the second analysis of GPER and COGPER for the MS and EPE statistics. The top, middle, and right panels in the left of Figure 3.5, which report the empirical distribution of the MS statistic for both GPER (MS) and COGPER (MS_1 and MS_2), confirms also the results of Figure 3.4. In fact, for both methods the average, over 100 replication, of the MS statistic equals to 3 (i.e. the average over 100 run, of the number of active groups at each run, for which $\hat{\boldsymbol{\beta}}_k \neq 0$, is approximately equals to 3). This corresponds to the three variables that have been declared as relevant in the first analysis. The MS_2 distribution of COGPER confirms also the presence of an overlapping group, which corresponds to the heterogeneous effect of the mother smoking-status predictor. Finally, in terms of prediction, the EPE statistic results of Figure 3.5 (right two panels) show that the better fit for both models seems to be at location

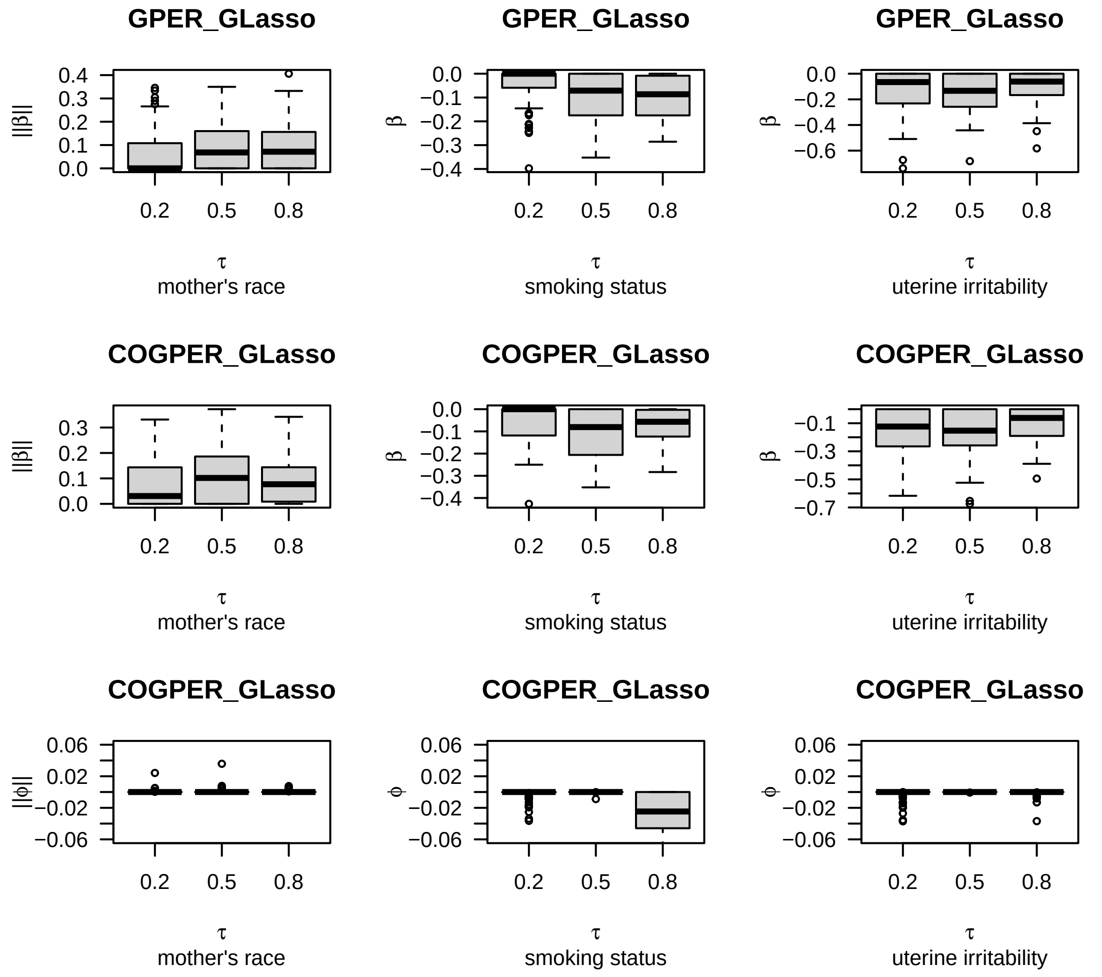


Figure 3.4 – From left to right column, the box-plot of the coefficient values for mother's race, smoking status and uterine irritability respectively. The GPER and COGPER are fitted with $\tau \in \{0.2, 0.5, 0.8\}$.

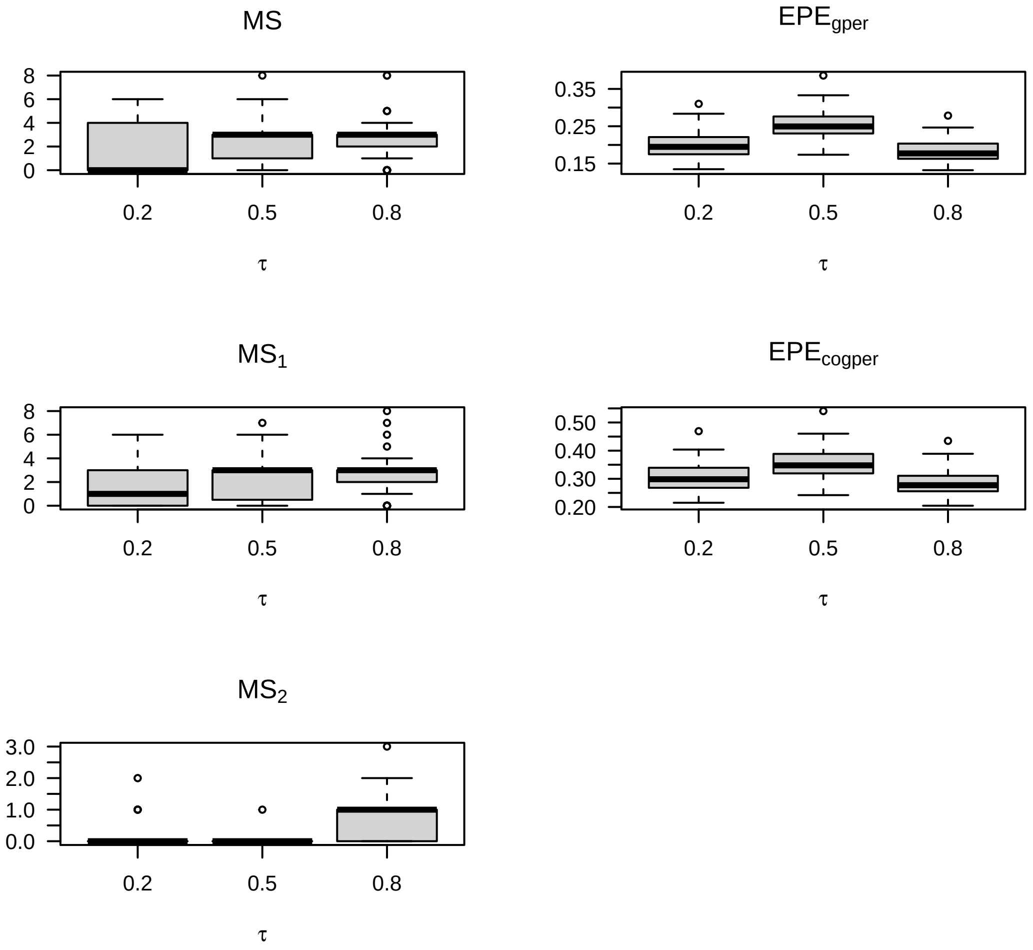


Figure 3.5 – Comparison of the number of selected groups (Model size) and the expectile-based prediction error (EPE), based on 100 replications, for the birth weight dataset. The GPER and COGPER with GLasso penalty are fitted for three locations $\tau \in \{0.2, 0.5, 0.8\}$.

$\tau = 0.8$. This might again emphasize the usefulness of both methods for allowing flexible exploration of the response–predictors relationship. All these results are also in agreements with several studies that have revealed a strong and heterogeneous relationship between mother smoking-status and baby birth weight [87, 100, 17].

3.6.2 Gene-based analysis of DNA methylation data near *BLK* gene

The data considered in this analysis consists of methylation levels of 5,986 CpG sites (i.e. predictors) within a genomic region with around 2 Millions base pairs (Chr8 :10321522-12391296), measured on 40 samples using bisulfite sequencing [57]. Each sample corresponds to one of three cell types : B cells (8 samples), T cells (19 samples), and Monocytes (13 samples). This cell type is considered as the response variable, where $y = 1$ corresponds to B-cells and $y = 0$ corresponds to T- and Monocyte-cell types. The 40 samples are obtained from whole blood collected on a cohort of healthy individuals from Sweden. The studied genomic region is known to be hypomethylated near the *BLK* gene in B-cells, compared to other cell types [35].

We proceeded as follows in order to form groups of predictors (CpGs sites) : (1) we extracted all genes belongings to the 2Mb region and their start-end genomic positions using `biomart` R package. We obtained $K = 36$ genes fall within this region in total. (2) We used prior information about the genomic position of each CpG site and assigned each CpG to a corresponding gene/group based on its base pair coordinate. Specifically, we considered that a CpG belongs to a gene/group if its genomic position is between the start and end positions of that gene. In total, 4,427 of all the 5,986 CpG sites spread over the $K = 36$ genes. The size of the studied groups ranges between 1 and 756, with 398 CpG sites falling between the start-end coordinates of the BLK gene.

This analysis aims to validate the performance of our methods on detecting the group of CpG sites belonging to the *BLK* gene as a Differentially Methylated Region (DMR) for the $0 - 1$ response variable, and to test the power of GPER in classification. The classification function is $\mathbb{1}(\text{fitted value} > 0.5)$, where $\mathbb{1}(A)$ is the indicator function which equals 1 if A is true and 0 if A is false.

Notice that the analysis results of this dataset using GPER with the non-convex penalties, GMCP and GSCAD, were very similar. Thus, we reported only the results of GPER with GMCP penalty.

In Figure 3.6, the x-axis and the y-axis correspond respectively to the genomic position of the CpGs and the coefficient values of the optimal solutions chosen by 5-fold CV. We can observe that the region around 11.3 Mb with size 150kb is significantly detected/selected by the group expectile methods with $\tau = 0.95$ but not with the group least square methods (GPER with $\tau = 0.5$). This region is known as a DMR between DNA methylation profiles of B-cells and T/Mono cells [92]. This observation is consistent with our analysis of this data using group quantile regression [77] and a study conducted by [57].

A second analysis of this DNA methylation data aims to show the advantages of the proposed group penalized expectile regression approaches for classification. This analysis emphasises GPER and COGPER utility when predicting the sample cell type (i.e. observation’s class) from a tail location of the distribution (i.e.

$\tau = 0.95$), in comparison with group penalized least-squares regression ($\tau = 0.5$) and two well-known group supervised classification methods : Group Support Vector Machine (GSVM) and group logistic regression (GLogit). The latter are both implemented in the `gglasso` R package [105]. We randomly divided the data into a training sample of 30 observations with the remainder making up a test data. The model is fitted to the training data and the misclassification error rate (MER) is calculated on the test data. The MER is defined as the ratio of the number of misclassified observations to the total number of observations in the test data. The tuning parameters are selected by 5-fold CV on the training data, and we adjust our models at two different locations $\tau \in \{0.5, 0.95\}$ in this analysis. The whole procedure is executed 100 times. The results of this analysis are presented in Figure 3.7.

In Figure 3.7, the DMR region is 100% selected by the GPER approach with both GLasso and GMCP penalties, for the location $\tau = 0.95$. However, it is selected with a rate less than 80% with the group least squares (GPER with $\tau = 0.5$), GSVM and GLogit methods. In terms of the MSE performance (last panel of Figure 3.7), the fit of GPER with GLasso at the tail of the response variable ($\tau = 0.95$) gives the best the best classification error. This, again, confirms the utility of GPER for the classification regression framework.

Of note, the analysis of this DNA methylation data using COGPER does not reveal any overlapping predictor (i.e. $\hat{\phi}_k = 0$, for all k). The conclusions of COGPER in terms of selecting the BLK gene as a DMR were relatively similar to those GPER ; but COGPER seemed to be less consistent compared to GPER in this DNA methylation analysis (results not reported here). We decided to not report COGPER to provide a clear summary analysis of this data and to avoid results redundancy.

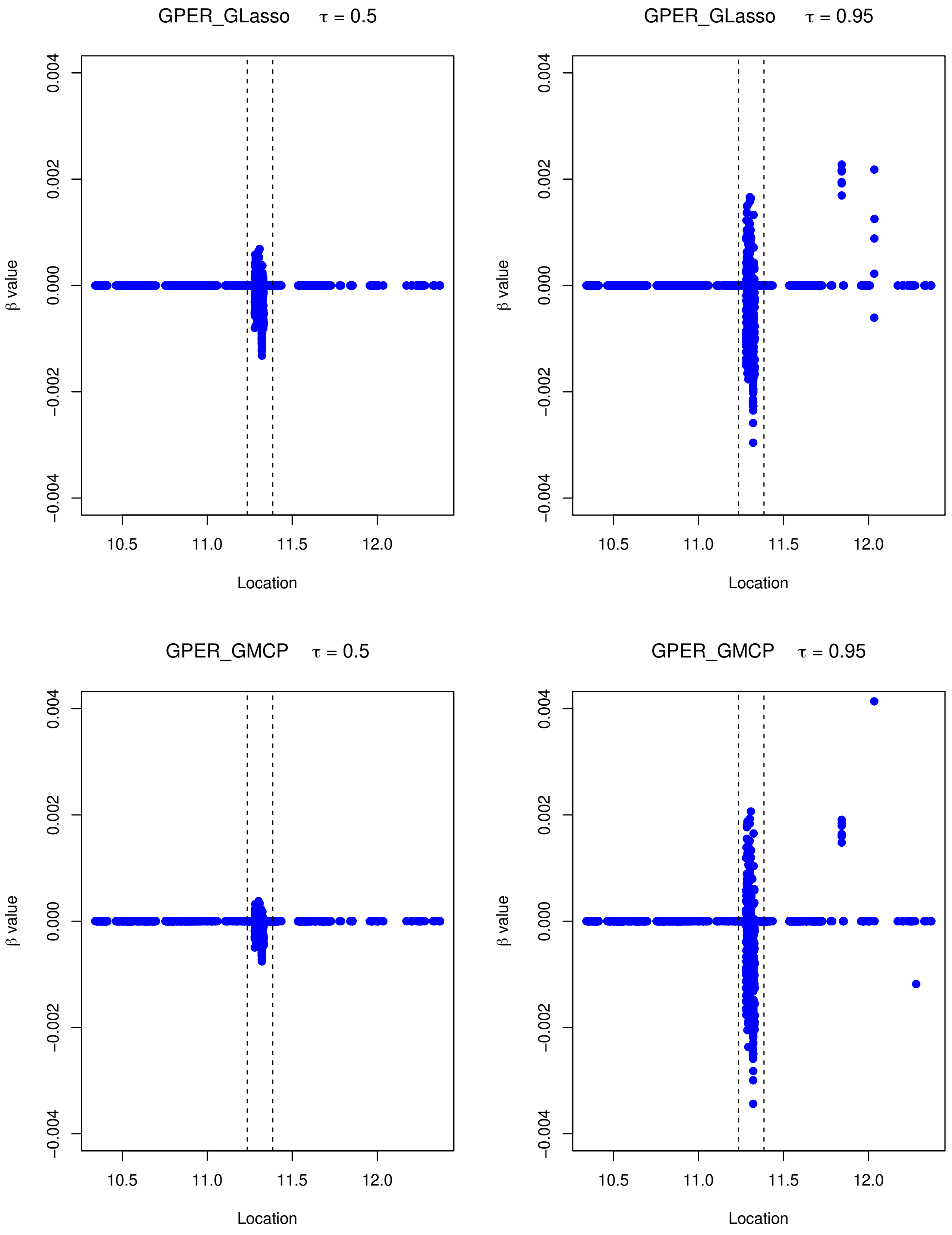


Figure 3.6 – At the left and from top to bottom, the optimal value (5-fold CV) for regression coefficients of GLasso and GMCP respectively, with $\tau = 0.5$, are shown as a function of a real genomic position. At the right and from top to bottom, the coefficients value of the same methods with $\tau = 0.95$.

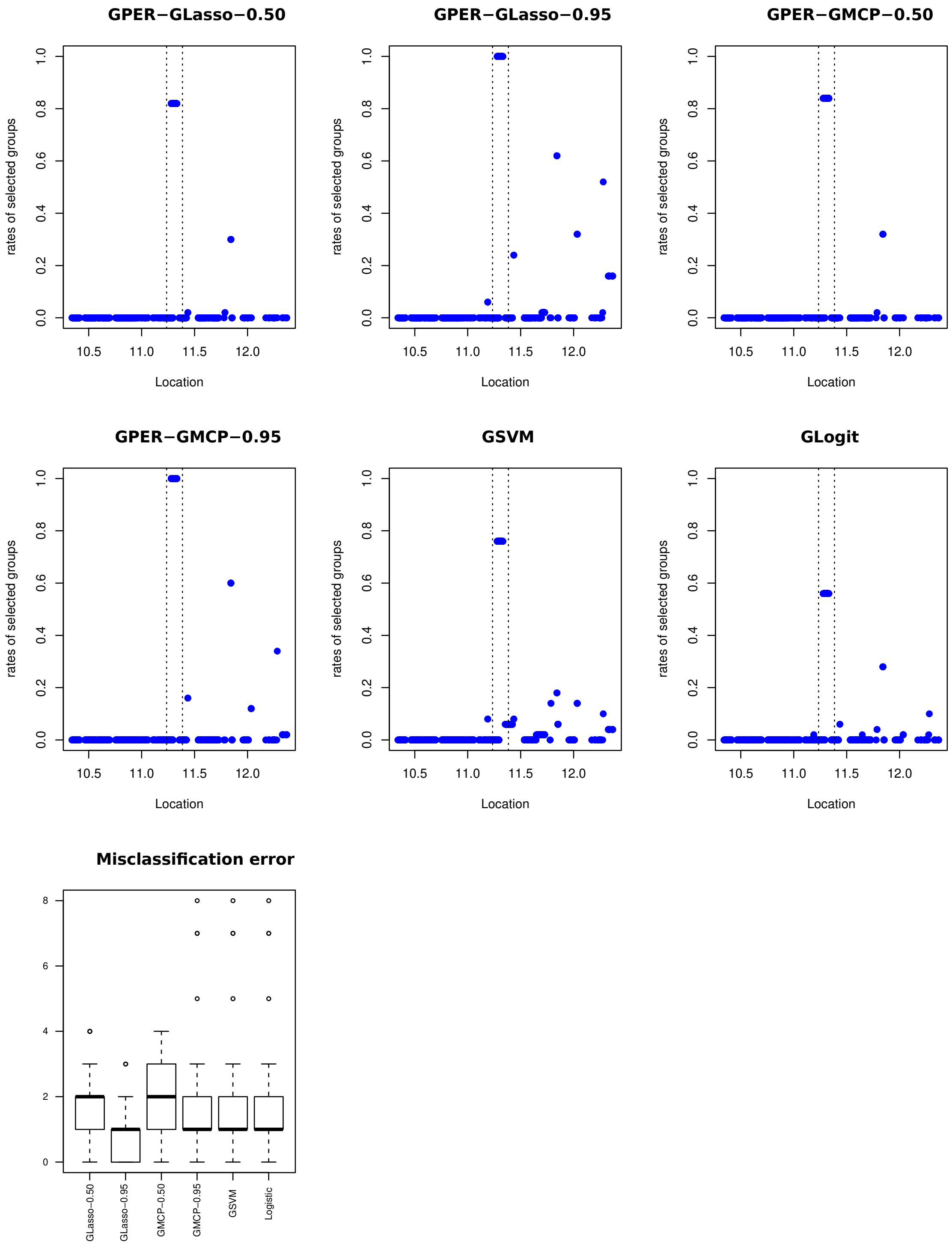


Figure 3.7 – Comparison of the proportion of selected genes for the DNA methylation data. At the top from left to right, the proportion of GPER-GLasso for $\tau \in \{0.50, 0.95\}$ and GPER-GMCP with $\tau = 0.50$ are shown as a function of the genomic position. The middle from left to right shows the proportion of GPER-GMCP with $\tau = 0.50$, GSVM and GLogit. The bottom row shows the misclassification error for all these methods cited above. The x and y axes correspond respectively to the genomic position, t_j, of the j-th CpG site and the proportion of non-zero $(\hat{\beta}_j)_{1 \leq j \leq 4427}$).

3.7 Discussion

In this paper, we have proposed the group penalized expectile regression approaches (GPER and COGPER) for selection of grouped variables. Both approaches, (CO)GPER, handle most known group penalties in the literature, namely, group Lasso, group MCP, group SCAD, and group LLA penalties. (CO)GPER are implemented in computationally-efficient groupwise-majorizatoin-descent algorithms. We have showed theoretically that, under some regularity conditions, our proposed methods enjoy the consistency property for the group Lasso penalty, and we have proved the convergence of (CO)-GPER-GLLA algorithms to the oracle estimator in two steps for the non-convex penalties. The results from our simulation studies have shown that the proposed methods provide appropriate sparse group-variable selection and accurate estimation.

It is well known that the asymmetric squared loss function in (CO)GPER might be sensitive to outliers either in the response and/or in the covariates. Recently, [114] have developed a robust expectile regression approach for ultrahigh dimensional heavy-tailed heterogeneous data. The proposed loss function in [114] differs from the asymmetric squared loss function of (CO)GPER only in the tail, where it is peace-wise linear in the extremes to down-weight the outliers. [114] have also provided attractive theoretical results for their proposed estimator. The extension of our framework to robust expectile regression in the presence of group structure among the covariates could be an interesting avenue to explore.

Asymmetric regression models have been increasingly investigated in the last decade. The extremile regression [19, 20] is a new attractive asymmetric least squares analog of quantile and expectile regression, which is found to be a useful descriptor of the tail of the response distribution, especially for long-tailed distributions. Although the extremile loss function is defined through a power transformation of the cumulative distribution function of the response variable, the extremile regression estimator has a closed form and can be calculated using an iterative reweighted least squares algorithm. This makes it computationally very attractive. Investigating penalized extremile regression in high-dimensional settings might be an interesting avenue of research in penalized asymmetric regression.

3.8 Appendixes

3.8.1 Appendix **A** : proof of Proposition 3.1

Preuve.

Notice first that the expectile loss function $\Psi_\tau(.)$ has a Lipschitz continuous derivative $\Psi'_\tau(.)$. That is, one can verify that

$$|\Psi'_\tau(u) - \Psi'_\tau(v)| \leq c|u - v| \quad \forall u, v \in \mathbb{R}, \tag{3.28}$$

where $c = 2\max(\tau, 1-\tau)$.

For $\boldsymbol{\beta}_k$ and $\tilde{\boldsymbol{\beta}}_k$, let $\mathbf{V}_k = \boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k$ and define $h(t) = \Psi_\tau(\tilde{\boldsymbol{\beta}}_k + t\mathbf{V}_k, \tilde{\boldsymbol{\beta}}_{-k})$.

Then, we have $h(0) = \Psi_\tau(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k})$ and $h(1) = \Psi_\tau(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k})$.

By the mean value theorem, there exits $a \in (0, 1)$ such that

$$h(1) = h(0) + h'(a) = h(0) + h'(0) + (h'(a) - h'(0)). \tag{3.29}$$

Noticed that

$$h'(t) = n^{-1}\sum_{i=1}^{n} \mathbf{x}_{i,k}^\top \mathbf{V}_k \Psi'_\tau(y_i - \mathbf{x}_{i,-k}^\top \tilde{\boldsymbol{\beta}}_{-k} - \mathbf{x}_{i,k}^\top(\tilde{\boldsymbol{\beta}}_k + t\mathbf{V}_k)),$$

which leads to

$$h'(0) = (\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top \nabla_k \Psi_\tau(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}),$$

and

$$\begin{aligned}
|\, h'(a) - h'(0)| = |n^{-1}\sum_{i=1}^{n} \mathbf{x}_{i,k}^\top \mathbf{V}_k [&\Psi'_\tau(y_i - \mathbf{x}_{i,-k}\tilde{\boldsymbol{\beta}}_{-k} - \mathbf{x}_{i,k}(\tilde{\boldsymbol{\beta}}_k + a\mathbf{V}_k)) - \\
&\Psi'_\tau(y_i - \mathbf{x}_{i,-k}\tilde{\boldsymbol{\beta}}_{-k} - \mathbf{x}_{i,k}\tilde{\boldsymbol{\beta}}_k)]| \\
\leq n^{-1}\sum_{i=1}^{n} |\mathbf{x}_{i,k}^\top \mathbf{V}_k||&\Psi'_\tau(y_i - \mathbf{x}_{i,-k}\tilde{\boldsymbol{\beta}}_{-k} - \mathbf{x}_{i,k}(\tilde{\boldsymbol{\beta}}_k + a\mathbf{V}_k)) - \\
&\Psi'_\tau(y_i - \mathbf{x}_{i,-k}\tilde{\boldsymbol{\beta}}_{-k} - \mathbf{x}_{i,k}\tilde{\boldsymbol{\beta}}_k)| \\
\overset{(a)}{\leq} n^{-1}\sum_{i=1}^{n} |\mathbf{x}_{i,k}^\top \mathbf{V}_k| c |a\mathbf{x}_{i,k}^\top \mathbf{V}_k|& \\
\leq cn^{-1}\sum_{i=1}^{n} |\mathbf{x}_{i,k}^\top \mathbf{V}_k|^2& \\
\leq cn^{-1}\mathbf{V}_k^\top \mathbf{x}_k^\top \mathbf{x}_k \mathbf{V}_k.&
\end{aligned}$$

Inequality (a) is due to the equation (3.28). *Plugging the last inequality into* (3.29), *we have*

$$\begin{aligned}\mathit{\Psi}_\tau(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k}) \quad &\leq \quad \mathit{\Psi}_\tau(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}) + (\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top \nabla_k \mathit{\Psi}_\tau(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}) + \\ &\qquad cn^{-1}(\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top \mathbf{x}_k^\top \mathbf{x}_k (\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k).\end{aligned}$$

Thus, we have

$$\begin{aligned}R_\tau(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k}) \quad &= \quad \mathit{\Psi}_\tau(\boldsymbol{\beta}_k, \tilde{\boldsymbol{\beta}}_{-k}) + P_\lambda(\|\boldsymbol{\beta}_k\|_2) \\ &\leq \quad \mathit{\Psi}_\tau(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}) + (\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top \nabla_k \mathit{\Psi}_\tau(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}) + \\ &\qquad cn^{-1}(\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top x_k^\top \mathbf{x}_k (\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k) + P_\lambda(\|\boldsymbol{\beta}_k\|_2) \\ &\leq \quad \mathit{\Psi}_\tau(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}) + (\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top \nabla_k \mathit{\Psi}_\tau(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}) + \\ &\qquad \frac{\gamma_k}{2}(\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k)^\top (\boldsymbol{\beta}_k - \tilde{\boldsymbol{\beta}}_k) + P_\lambda(\|\boldsymbol{\beta}_k\|_2),\end{aligned}$$

where γ_k *is the largest eigenvalue of the matrix* $2\max(1-\tau, \tau)\dfrac{\mathbf{x}_k^\top \mathbf{x}_k}{n}$.

This ends the proof of Proposition 3.1. □

□

3.8.2 Appendix **B** : proof of Proposition 3.2

For GSCAD penalty :

The KKT conditions of the objective function in equation (3.11) of the main manuscript can be written as

$$-\mathbf{Z}_k + \gamma_k \boldsymbol{\beta}_k + \frac{\partial P_\lambda(\|\boldsymbol{\beta}_k\|_2)}{\partial \boldsymbol{\beta}_k} = 0,$$

where $\mathbf{Z}_k = -\nabla_k \Psi_\tau(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}) + \gamma_k \tilde{\boldsymbol{\beta}}_k$.

- If $\|\boldsymbol{\beta}_k\|_2 \leq \lambda$, then $-\mathbf{Z}_k + \gamma_k \boldsymbol{\beta}_k + \lambda w_k \mathbf{u} = 0$,

  where $\mathbf{u}$ is the sub-gradient and $\|\mathbf{u}\|_2 \leq 1$.

  - If $\boldsymbol{\beta}_k = 0$, then we have

$$-\mathbf{Z}_k + \lambda w_k \mathbf{u} = 0,$$

which implies

$$\|\mathbf{Z}_k\|_2 \leq \lambda w_k.$$

— If $\boldsymbol{\beta}_k \neq 0$, then we have

$$-\mathbf{Z}_k + \gamma_k \boldsymbol{\beta}_k + \lambda w_k \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} = 0.$$

Applying the l_2 norm to the least equality, we have

$$\|\mathbf{Z}_k\|_2 = \gamma_k \|\boldsymbol{\beta}_k\|_2 + \lambda w_k,$$

which implies

$$\|\mathbf{Z}_k\|_2 \leq \lambda(w_k + \gamma_k).$$

Moreover, we have

$$-\mathbf{Z}_k + \gamma_k \boldsymbol{\beta}_k + \lambda w_k \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} = 0 \text{ (since } \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} = \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2}).$$

Then, we obtain

$$\boldsymbol{\beta}_k = \frac{1}{\gamma_k} \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} (\|\mathbf{Z}_k\|_2 - \lambda w_k).$$

— If $\lambda \leq \|\boldsymbol{\beta}_k\|_2 \leq \theta\lambda$, then

$$-\mathbf{Z}_k + \gamma_k \boldsymbol{\beta}_k + \frac{\theta \lambda w_k}{\theta - 1} \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} - \frac{w_k}{\theta - 1} \boldsymbol{\beta}_k = 0.$$

It follows that

$$\mathbf{Z}_k = [\gamma_k + \frac{w_k}{\theta - 1} (\frac{\theta \lambda}{\|\boldsymbol{\beta}_k\|_2} - 1)] \boldsymbol{\beta}_k,$$

which implies that

$$\|\mathbf{Z}_k\|_2 = (\gamma_k - \frac{w_k}{\theta - 1}) \|\boldsymbol{\beta}_k\|_2 + \frac{w_k \lambda \theta}{\theta - 1} \tag{3.30}$$

and

$$\frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} = \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2}.$$

Thus, combining the condition $\lambda \leq \|\boldsymbol{\beta}_k\|_2 \leq \theta\lambda$ and equation (3.30), we get

$$\lambda(\gamma_k + w_k) \leq \|\mathbf{Z}_k\|_2 \leq \gamma_k \theta \lambda$$

and

$$\boldsymbol{\beta}_k = \frac{1}{\gamma_k - \frac{w_k}{\theta - 1}} \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} (\|\mathbf{Z}_k\|_2 - \lambda w_k \frac{\theta}{\theta - 1}).$$

- If $\|\boldsymbol{\beta}_k\|_2 \geq \theta\lambda$, then we have

  $$-\mathbf{Z}_k + \gamma_k \boldsymbol{\beta}_k = 0,$$

  which implies

  $$\|\mathbf{Z}_k\|_2 \geq \gamma_k \theta \lambda$$

  and

  $$\boldsymbol{\beta}_k = \frac{1}{\gamma_k} \mathbf{Z}_k.$$

This ends the proof of Proposition 3.2 for GSCAD penalty. □

For GMCP penalty

Again, the KKT conditions of the objective function in equation (3.11) of the main manuscript can be written as

$$-\mathbf{Z}_k + \gamma_k \boldsymbol{\beta}_k + \frac{\partial P_\lambda(\|\boldsymbol{\beta}_k\|_2)}{\partial \boldsymbol{\beta}_k} = 0,$$

where $\mathbf{Z}_k = -\nabla_k \Psi_\tau(\tilde{\boldsymbol{\beta}}_k, \tilde{\boldsymbol{\beta}}_{-k}) + \gamma_k \tilde{\boldsymbol{\beta}}_k$.

- If $\|\boldsymbol{\beta}_k\|_2 \leq \theta\lambda$, then we have

  $$-\mathbf{Z}_k + \gamma_k \boldsymbol{\beta}_k + \lambda \mathbf{u} - \frac{w_k}{\theta} \boldsymbol{\beta}_k = 0,$$

  where $\mathbf{u}$ is the sub-gradient and $\|\mathbf{u}\|_2 \leq 1$.

  - If $\boldsymbol{\beta}_k = 0$, then we obtain

    $$-\mathbf{Z}_k + \lambda w_k \mathbf{u} = 0.$$

    Thus, we have

    $$\|\mathbf{Z}_k\|_2 \leq \lambda w_k.$$

  - If $\boldsymbol{\beta}_k \neq 0$, then we get

    $$-\mathbf{Z}_k + \gamma_k \boldsymbol{\beta}_k + \lambda w_k \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} - \frac{w_k}{\theta} \boldsymbol{\beta}_k = 0.$$

Applying the l_2 norm to the last equality, we obtain

$$\|\mathbf{Z}_k\|_2 = (\gamma_k - \frac{w_k}{\theta})\|\boldsymbol{\beta}_k\|_2 + \lambda w_k.$$

Combining the condition $\|\boldsymbol{\beta}_k\|_2 \leq \theta\lambda$ and the last two equations, we have

$$\|\mathbf{Z}_k\|_2 \leq \gamma_k \theta \lambda$$

and

$$\boldsymbol{\beta}_k = \frac{1}{\gamma_k - \frac{w_k}{\theta}} \frac{\mathbf{Z}^{(k)}}{\|\mathbf{Z}_k\|_2}(\|\mathbf{Z}_k\|_2 - \lambda w_k).$$

— If $\|\boldsymbol{\beta}_k\|_2 \geq \theta\lambda$, then we have $-\mathbf{Z}_k + \gamma_k \boldsymbol{\beta}_k = 0$. This implies that

$$\|\mathbf{Z}_k\|_2 \geq \gamma_k \theta \lambda \quad \text{and} \quad \boldsymbol{\beta}_k = \frac{1}{\gamma_k}\mathbf{Z}_k.$$

This ends the proof of Proposition 3.2 for GMCP penalty. □

3.8.3 Appendix **C** : proof of Proposition 3.5

The KKT conditions of the objective functions in (3.23) and (3.24) of the main manuscript can be written as :

$$-\mathbf{Z}_k + 2(1+c)\gamma_k \boldsymbol{\beta}_k + \frac{\partial P_{\lambda_1}(\|\boldsymbol{\beta}_k\|_2)}{\partial \boldsymbol{\beta}_k} = 0$$

and

$$-\mathbf{W}_k + 2c\gamma_k \boldsymbol{\phi}_k + \frac{\partial P_{\lambda_2}(\|\boldsymbol{\phi}_k\|_2)}{\partial \boldsymbol{\phi}_k} = 0,$$

where $Z_k = \mathbf{U}_k^{0.5} + \mathbf{U}_k^{\tau} + 2(1+c)\gamma_k \widetilde{\boldsymbol{\beta}}_k$ and $\mathbf{W_k} = \mathbf{U}_k^{\tau} + 2c\gamma_k \widetilde{\boldsymbol{\phi}}_k$.

For GLasso penalty

We have

$$-\mathbf{Z}_k + 2(1+c)\gamma_k \boldsymbol{\beta}_k + \lambda_1 \omega_k \mathbf{u} = 0$$

and

$$-\mathbf{W}_k + 2c\gamma_k \boldsymbol{\phi}_k + \lambda_2 u_k \mathbf{v} = 0,$$

where $\mathbf{u}$ and $\mathbf{v}$ are the sub-gradient of $P_{\lambda_1}(.)$ at $\boldsymbol{\beta}_k$ and $P_{\lambda_2}(.)$ at $\boldsymbol{\phi}_k$ repectively. So, we have $\|\mathbf{u}\|_2 \leq 1$, $\|\mathbf{v}\|_2 \leq 1$.

- If $\boldsymbol{\beta}_k = 0$ and $\boldsymbol{\phi}_k = 0$, then we get

$$-\mathbf{Z}_k + \lambda_1 \omega_k \mathbf{u} = 0, \text{ and } -\mathbf{W}_k + \lambda_2 u_k \mathbf{v} = 0,$$

  which implies that

$$\|\mathbf{Z}_k\|_2 \leq \lambda_1 \omega_k, \text{ and } \|\mathbf{W}_k\|_2 \leq \lambda_2 u_k.$$

- If $\boldsymbol{\beta}_k \neq 0$ and $\boldsymbol{\phi}_k \neq 0$, then

$$-\mathbf{Z}_k + 2(1+c)\gamma_k \boldsymbol{\beta}_k + \lambda_1 \omega_k \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} = 0$$

  and

$$-\mathbf{W}_k + 2c\gamma_k \boldsymbol{\phi}_k + \lambda_2 u_k \frac{\boldsymbol{\phi}_k}{\|\boldsymbol{\phi}_k\|_2} = 0.$$

  Moreover, we have

$$-\mathbf{Z}_k + 2(1+c)\gamma_k \boldsymbol{\beta}_k + \lambda_1 w_k \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} = 0 \text{ (since } \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} = \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2})$$

  and

$$-\mathbf{W}_k + 2c\gamma_k \boldsymbol{\phi}_k + \lambda_2 u_k \frac{\mathbf{W}_k}{\|\mathbf{W}_k\|_2} = 0 \text{ (since } \frac{\mathbf{W}_k}{\|\mathbf{W}_k\|_2} = \frac{\boldsymbol{\phi}_k}{\|\boldsymbol{\phi}_k\|_2}),$$

  which implies

$$\boldsymbol{\beta}_k = \frac{1}{2(1+c)\gamma_k} \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} (\|\mathbf{Z}_k\|_2 - \lambda_1 w_k)$$

  and

$$\boldsymbol{\phi}_k = \frac{1}{2c\gamma_k} \frac{\mathbf{W}_k}{\|\mathbf{W}_k\|_2} (\|\mathbf{W}_k\|_2 - \lambda_2 u_k).$$

For GSCAD penalty

- If $\|\boldsymbol{\beta}_k\|_2 \leq \lambda_1$ and $\|\boldsymbol{\phi}_k\|_2 \leq \lambda_2$, then

$$-\mathbf{Z}_k + 2(1+c)\gamma_k \boldsymbol{\beta}_k + w_k \lambda_1 \mathbf{u} = 0$$

  and

$$-\mathbf{W}_k + 2c\gamma_k \boldsymbol{\phi}_k + u_k \lambda_2 \mathbf{v} = 0,$$

where $\mathbf{u}$ and $\mathbf{v}$ are the sub-gradient of $P_{\lambda_1}(.)$ at $\boldsymbol{\beta}_k$ and $P_{\lambda_2}(.)$ at $\boldsymbol{\phi}_k$ repectively. So, we have $\|\mathbf{u}\|_2 \leq 1$, $\|\mathbf{v}\|_2 \leq 1$.

— If $\boldsymbol{\beta}_k = 0$ and $\boldsymbol{\phi}_k = 0$, then we obtain

$$-\mathbf{Z}_k + \lambda_1 \omega_k \mathbf{u} = 0 \ \text{ and } \ -\mathbf{W}_k + \lambda_2 u_k \mathbf{v} = 0,$$

which implies that

$$\|\mathbf{Z}_k\|_2 \leq \lambda_1 \omega_k \ \text{ and } \ \|\mathbf{W}_k\|_2 \leq \lambda_2 u_k.$$

— If $\boldsymbol{\beta}_k \neq 0$ and $\boldsymbol{\phi}_k \neq 0$, then we obtain

$$-\mathbf{Z}_k + 2(1+c)\gamma_k \boldsymbol{\beta}_k + \lambda_1 \omega_k \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} = 0$$

and

$$-\mathbf{W}_k + 2c\gamma_k \boldsymbol{\phi}_k + \lambda_2 u_k \frac{\boldsymbol{\phi}_k}{\|\boldsymbol{\phi}_k\|_2} = 0.$$

Applying the l_2 norm to the last two equations, we get

$$\|\mathbf{Z}_k\|_2 = 2(1+c)\gamma_k \boldsymbol{\beta}_k + \lambda_1 \omega_k$$

and

$$\|\mathbf{W}_k\|_2 = 2c\gamma_k \boldsymbol{\phi}_k + \lambda_2 u_k,$$

which implies

$$\|\mathbf{Z}_k\|_2 \leq (1 + 2(1+c)\gamma_k)\lambda_1 \omega_k$$

and

$$\|\mathbf{W}_k\|_2 \leq (1 + 2c\gamma_k)\lambda_2 u_k.$$

Moreover, we have

$$-\mathbf{Z}_k + 2(1+c)\gamma_k \boldsymbol{\beta}_k + \lambda_1 w_k \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} = 0 \text{ (since } \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} = \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2})$$

and

$$-\mathbf{W}_k + 2c\gamma_k \boldsymbol{\phi}_k + \lambda_2 u_k \frac{\mathbf{W}_k}{\|\mathbf{W}_k\|_2} = 0 \text{ (since } \frac{\mathbf{W}_k}{\|\mathbf{W}_k\|_2} = \frac{\boldsymbol{\phi}_k}{\|\boldsymbol{\phi}_k\|_2}),$$

which implies

$$\boldsymbol{\beta}_k = \frac{1}{2(1+c)\gamma_k} \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} (\|\mathbf{Z}_k\|_2 - \lambda_1 w_k),$$

$$\boldsymbol{\phi}_k = \frac{1}{2c\gamma_k} \frac{\mathbf{W}_k}{\|\mathbf{W}_k\|_2} (\|\mathbf{W}_k\|_2 - \lambda_2 u_k).$$

— If $\lambda_1 \leq \|\boldsymbol{\beta}_k\|_2 \leq \theta\lambda_1$ and $\lambda_2 \leq \|\boldsymbol{\phi}_k\|_2 \leq \theta\lambda_2$, then we get

$$-\mathbf{Z}_k + 2(1+c)\gamma_k\boldsymbol{\beta}_k + \frac{\theta\lambda_1 w_k}{\theta-1}\mathbf{u} - \frac{w_k}{\theta-1}\boldsymbol{\beta}_k = 0$$

and

$$-\mathbf{W}_k + 2c\gamma_k\boldsymbol{\phi}_k + \frac{\theta\lambda_2 u_k}{\theta-1}\mathbf{v} - \frac{u_k}{\theta-1}\boldsymbol{\phi}_k = 0,$$

where $\mathbf{u}$ and $\mathbf{v}$ are the sub-gradient of $P_{\lambda_1}(.)$ at $\boldsymbol{\beta}_k$ and $P_{\lambda_2}(.)$ at $\boldsymbol{\phi}_k$ repectively. So, we have $\|\mathbf{u}\|_2 \leq 1$, $\|\mathbf{v}\|_2 \leq 1$.

— If $\boldsymbol{\beta}_k = 0$ and $\boldsymbol{\phi}_k = 0$, then we have

$$-\mathbf{Z}_k + \frac{\theta\lambda_1 w_k}{\theta-1}\mathbf{u} = 0 \text{ and } -\mathbf{W}_k + \frac{\theta\lambda_2 u_k}{\theta-1}\mathbf{v} = 0,$$

which implies that

$$\|\mathbf{Z}_k\|_2 \leq \frac{\theta\lambda_1 w_k}{\theta-1}\mathbf{u} \text{ and } \|\mathbf{W}_k\|_2 \leq \frac{\theta\lambda_2 u_k}{\theta-1}\mathbf{v} = 0.$$

— If $\boldsymbol{\beta}_k \neq 0$ and $\boldsymbol{\phi}_k \neq 0$, then we obtain

$$-\mathbf{Z}_k + 2(1+c)\gamma_k\boldsymbol{\beta}_k + \frac{\theta\lambda_1 w_k}{\theta-1}\frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} - \frac{w_k}{\theta-1}\boldsymbol{\beta}_k = 0$$

and

$$-\mathbf{W}_k + 2c\gamma_k\boldsymbol{\phi}_k + \frac{\theta\lambda_2 u_k}{\theta-1}\frac{\boldsymbol{\phi}_k}{\|\boldsymbol{\phi}_k\|_2} - \frac{u_k}{\theta-1}\boldsymbol{\phi}_k = 0,$$

which implies that

$$\|\mathbf{Z}_k\|_2 = (2(1+c)\gamma_k - \frac{w_k}{\theta-1})\|\boldsymbol{\beta}_k\|_2 + \frac{w_k\lambda_1\theta}{\theta-1} \text{ (since } \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} = \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2}) \tag{3.31}$$

and

$$\|\mathbf{W}_k\|_2 = (2c\gamma_k - \frac{u_k}{\theta-1})\|\boldsymbol{\phi}_k\|_2 + \frac{u_k\lambda_2\theta}{\theta-1} \text{ (since } \frac{\mathbf{W}_k}{\|\mathbf{W}_k\|_2} = \frac{\boldsymbol{\phi}_k}{\|\boldsymbol{\phi}_k\|_2}). \tag{3.32}$$

For $\boldsymbol{\beta}_k$, combining the condition $\lambda_1 \leq \|\boldsymbol{\beta}_k\|_2 \leq \theta\lambda_1$ and equation (3.31), we obtain

$$\lambda_1 w_k(\gamma_k 2(1+c)+1) \leq \|\mathbf{Z}_k\|_2 \leq 2(1+c)\gamma_k\theta\lambda w_k.$$

If $\|\mathbf{Z}_k\|_2 \geq \frac{w_k\lambda_1\theta}{\theta-1}$, then we have

$$\boldsymbol{\beta}_k = \frac{1}{2(1+c)\gamma_k - \frac{w_k}{\theta-1}}\frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2}(\|\mathbf{Z}_k\|_2 - \lambda_1 w_k\frac{\theta}{\theta-1}).$$

For $\boldsymbol{\phi}_k$, combining the condition $\lambda_2 \leq \|\boldsymbol{\phi}_k\|_2 \leq \theta\lambda_2$ and equation (3.32), we obtain

$$\lambda_2 u_k(2c\gamma_k+1) \leq \|\mathbf{W}_k\|_2 \leq 2c\gamma_k\theta\lambda_2 u_k.$$

If $\|\mathbf{W}_k\|_2 \geq \frac{u_k \lambda_2 \theta}{\theta - 1}$, then we have

$$\boldsymbol{\phi}_k = \frac{1}{2c\gamma_k - \frac{u_k}{\theta-1}} \frac{\mathbf{W}_k}{\|\mathbf{W}_k\|_2}(\|\mathbf{W}_k\|_2 - \lambda_2 u_k \frac{\theta}{\theta - 1}).$$

— If $\|\boldsymbol{\beta}_k\|_2 \geq \theta\lambda_1$ and $\|\boldsymbol{\phi}_k\|_2 \geq \theta\lambda_2$,

then we have

$$-\mathbf{Z}_k + 2(1+c)\gamma_k\boldsymbol{\beta}_k = 0 \text{ and } \mathbf{W}_k + 2c\gamma_k\boldsymbol{\beta}_k = 0.$$

This implies that

$$\|\mathbf{Z}_k\|_2 \geq 2(1+c)\gamma_k\theta\lambda_1 w_k \quad \text{and} \quad \boldsymbol{\beta}_k = \frac{1}{2(1+c)\gamma_k}\mathbf{Z}_k$$

and

$$\|\mathbf{W}_k\|_2 \geq 2c\gamma_k\theta\lambda_2 u_k \quad \text{and} \quad \boldsymbol{\phi}_k = \frac{1}{2c\gamma_k}\mathbf{W}_k.$$

For GMCP penalty

— If $\|\boldsymbol{\beta}_k\|_2 \leq \theta\lambda_1$ and $\|\boldsymbol{\phi}_k\|_2 \leq \theta\lambda_2$, then we get

$$-\mathbf{Z}_k + 2(1+c)\gamma_k\boldsymbol{\beta}_k + \lambda_1 w_k \mathbf{u} - \frac{w_k}{\theta}\boldsymbol{\beta}_k = 0$$

and

$$-\mathbf{W}_k + 2c\gamma_k\boldsymbol{\phi}_k + \lambda_2 u_k \mathbf{v} - \frac{u_k}{\theta}\boldsymbol{\phi}_k = 0,$$

where $\mathbf{u}$ and $\mathbf{v}$ are the sub-gradient of $P_{\lambda_1}(.)$ at $\boldsymbol{\beta}_k$ and $P_{\lambda_2}(.)$ at $\boldsymbol{\phi}_k$ repectively. So, we have $\|\mathbf{u}\|_2 \leq 1$, $\|\mathbf{v}\|_2 \leq 1$.

— If $\boldsymbol{\beta}_k = 0$ and $\boldsymbol{\phi}_k = 0$, then we have

$$-\mathbf{Z}_k + \lambda_1 w_k \mathbf{u} = 0 \text{ and } -\mathbf{W}_k + \lambda_2 u_k \mathbf{v} = 0,$$

which implies that

$$\|\mathbf{Z}_k\|_2 \leq \lambda_1 w_k \mathbf{u} \text{ and } \|\mathbf{W}_k\|_2 \leq \lambda_2 u_k \mathbf{v} = 0.$$

— If $\boldsymbol{\beta}_k \neq 0$ and $\boldsymbol{\phi}_k \neq 0$, then

$$-\mathbf{Z}_k + 2(1+c)\gamma_k\boldsymbol{\beta}_k + \lambda_1 w_k \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} - \frac{w_k}{\theta}\boldsymbol{\beta}_k = 0$$

and

$$-\mathbf{W}_k + 2c\gamma_k\boldsymbol{\phi}_k + \lambda_2 u_k\frac{\boldsymbol{\phi}_k}{\|\boldsymbol{\phi}_k\|_2} - \frac{u_k}{\theta}\boldsymbol{\phi}_k = 0,$$

which implies that

$$\|\mathbf{Z}_k\|_2 = (2(1+c)\gamma_k - \frac{w_k}{\theta})\|\boldsymbol{\beta}_k\|_2 + w_k\lambda_1 \text{ (since } \frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2} = \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2}) \tag{3.33}$$

and

$$\|\mathbf{W}_k\|_2 = (2c\gamma_k - \frac{u_k}{\theta})\|\boldsymbol{\phi}_k\|_2 + u_k\lambda_2 \text{ (since } \frac{\mathbf{W}_k}{\|\mathbf{W}_k\|_2} = \frac{\boldsymbol{\phi}_k}{\|\boldsymbol{\phi}_k\|_2}). \tag{3.34}$$

For $\boldsymbol{\beta}_k$, combining the condition $\|\boldsymbol{\beta}_k\|_2 \leq \theta\lambda_1$ and equation (3.33), we obtain

$$\|\mathbf{Z}_k\|_2 \leq 2(1+c)\gamma_k\theta\lambda w_k.$$

If $\|\mathbf{Z}_k\|_2 \geq w_k\lambda_1$, then we have

$$\boldsymbol{\beta}_k = \frac{1}{2(1+c)\gamma_k - \frac{w_k}{\theta}}\frac{\mathbf{Z}_k}{\|\mathbf{Z}_k\|_2}(\|\mathbf{Z}_k\|_2 - \lambda_1 w_k).$$

For $\boldsymbol{\phi}_k$, combining the condition $\|\boldsymbol{\phi}_k\|_2 \leq \theta\lambda_2$ and equation (3.34), we obtain

$$\|\mathbf{W}_k\|_2 \leq 2c\gamma_k\theta\lambda_2 u_k.$$

If $\|\mathbf{W}_k\|_2 \geq u_k\lambda_2$, then we have

$$\boldsymbol{\phi}_k = \frac{1}{2c\gamma_k - \frac{u_k}{\theta}}\frac{\mathbf{W}_k}{\|\mathbf{W}_k\|_2}(\|\mathbf{W}_k\|_2 - \lambda_2 u_k).$$

– If $\|\boldsymbol{\beta}_k\|_2 \geq \theta\lambda_1$ and $\|\boldsymbol{\phi}_k\|_2 \geq \theta\lambda_2$, then we have

$$-\mathbf{Z}_k + 2(1+c)\gamma_k\boldsymbol{\beta}_k = 0 \text{ and } -\mathbf{W}_k + 2c\gamma_k\boldsymbol{\beta}_k = 0,$$

which implies

$$\|\mathbf{Z}_k\|_2 \geq 2(1+c)\gamma_k\theta\lambda_1 w_k \quad \text{and} \quad \boldsymbol{\beta}_k = \frac{1}{2(1+c)\gamma_k}\mathbf{Z}_k$$

and

$$\|\mathbf{W}_k\|_2 \geq 2c\gamma_k\theta\lambda_2 u_k \quad \text{and} \quad \boldsymbol{\phi}_k = \frac{1}{2c\gamma_k}\mathbf{W}_k.$$

3.8.4 Appendix **D** : proof of Theorems 3.3, 3.4, 3.6 and 3.7

Let us state two Lemmas ; the first lemma is on the properties of the expectile loss function $\Psi_\tau(.)$ and coupled loss function $S_\tau(.)$. The second lemma deals with sub-Gaussian random variables.

Lemma 1

(1) For any $\boldsymbol{\beta}, \boldsymbol{\delta} \in \mathbb{R}^p$, $2\underline{c}\|\mathbf{X}\boldsymbol{\delta}\|_2^2/n \leq \langle \nabla\Psi_\tau(\boldsymbol{\beta}+\boldsymbol{\delta}) - \nabla\Psi_\tau(\boldsymbol{\beta}), \boldsymbol{\delta}\rangle$.
(2) Let $\boldsymbol{\epsilon} = (\epsilon_i, 1 \leq i \leq n)^\top$ and $\boldsymbol{\eta} = (\eta_i, 1 \leq i \leq n)^\top$, where $\eta_i = S'_\tau(\epsilon_i - e_\tau)$.

For $\boldsymbol{\theta}, \boldsymbol{\delta} \in \mathbb{R}^{2p}$, we have

$$n^{-1}c_0\|(\boldsymbol{I}_2 \otimes \mathbf{X})\boldsymbol{\delta}\|_2^2/n \leq \langle \nabla S_\tau(\boldsymbol{\theta}+\boldsymbol{\delta}) - \nabla S_\tau(\boldsymbol{\theta}), \boldsymbol{\delta}\rangle,$$

where $\boldsymbol{I}_2$ is a 2×2 is the identity matrix, and $c_0 = 2^{-1}[(1+\underline{c}) - (1+16\underline{c}^2)^{1/2}] > 0.$

Proof of Lemma 1

The parts (1) and (2) of Lemma 1 follow from the part (1) of Lemma 4 and the part (1) of Lemma 6 in [33], respectively.

Lemma 2

Suppose that $Z_1, \ldots, Z_n \in \mathbb{R}$ are i.i.d sub-Gaussian random variables. Let $\boldsymbol{Z} = (Z_1, \ldots, Z_n)^\top$, $K = \|\boldsymbol{Z}\|_{SG}$, $\boldsymbol{Z}^+ = \max(\boldsymbol{Z}, \mathbf{0})$ and $\boldsymbol{Z}^- = \min(-\boldsymbol{Z}, \mathbf{0})$.

(1) If $\mathbb{E}(\boldsymbol{Z}) = 0$, then there exists an absolute constant C such that for any $\boldsymbol{a} = (a_1, \ldots, a_n)^\top \in \mathbb{R}^n$ and any $t \geq 0$, we have

$$P(|\boldsymbol{a}^\top \boldsymbol{Z}| \geq t) \leq 2\text{exp}\big(-\frac{Ct^2}{K^2\|\boldsymbol{a}\|_2^2}\big).$$

(2) For any $a_1, a_2 \in \mathbb{R}$, the random variable $a_1\boldsymbol{Z}^+ + a_2\boldsymbol{Z}^-$ is sub-Gaussian

(3) Let $\boldsymbol{A}$ be a fixed $m \times n$ matrix. If $\mathbb{E}(\boldsymbol{Z}) = 0$ and var$(\boldsymbol{Z}) = 1$, then there exists an absolute constant $C > 0$ such that for any $t \geq 0$

$$P\big(|\|\boldsymbol{A}\boldsymbol{Z}\|_2 - \|\boldsymbol{A}\|_F| \geq t\big) \leq 2exp\bigg(-\frac{Ct^2}{K^2\|\boldsymbol{A}\|_2^2}\bigg),$$

where $\|\boldsymbol{A}\|_F$ and $\|\boldsymbol{A}\|_2$ represent the Frobenius and l_2 norms of matrix $\boldsymbol{A}$, respectively.

Proof of Lemma 2

The part (1) follows from Proposition 5.10 of [94], and the parts (2) and (3) follow from the parts (4) and (2) of Lemma 3 of [33]

Proof of Theorem 3.3

Let $\hat{\boldsymbol{\delta}} = \hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*$ and $z_\infty^* = \|\nabla\Psi_\tau(\boldsymbol{\beta}^*)\|_{2,\infty}$, then $\hat{\boldsymbol{\beta}}$ satisfies the KKT conditions

$$\nabla_k\Psi_\tau(\boldsymbol{\beta}_k, \boldsymbol{\beta}_{-k}) + \mathbf{u}_k = 0, \quad \text{for } k = 1, \ldots, K,$$

where

$$\mathbf{u}_k = \begin{cases} \lambda^{\text{GLasso}}\dfrac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} & \text{for } \boldsymbol{\beta}_k \neq 0, \\ \mathbf{u}_k, \quad \|\mathbf{u}_k\|_2 \in [-\lambda^{\text{GLasso}}, \lambda^{\text{GLasso}}], & \text{for } \boldsymbol{\beta}_k = 0. \end{cases}$$

It follows that

$$\langle\hat{\boldsymbol{\beta}}_k, \mathbf{u}_k\rangle = \lambda^{\text{GLasso}}\|\hat{\boldsymbol{\beta}}_k\|_2, \ \forall k = 1, \ldots, K. \tag{3.35}$$

Lemma 1 and Holder's inequality lead to

$$
\begin{aligned}
0 \le 2\underline{c}\|X\hat{\boldsymbol{\delta}}\|_2^2/n &\le \langle \nabla\Psi_\tau(\hat{\boldsymbol{\beta}}) - \nabla\Psi_\tau(\boldsymbol{\beta}^*), \hat{\boldsymbol{\delta}}\rangle \\
&= \sum_{k=1}^{K} \langle \nabla\Psi_\tau(\hat{\boldsymbol{\beta}}_k) - \nabla\Psi_\tau(\boldsymbol{\beta}_k^*), \hat{\boldsymbol{\delta}}_k\rangle \\
&= \sum_{k\in\mathcal{A}} \langle \nabla\Psi_\tau(\hat{\boldsymbol{\beta}}_k) - \nabla\Psi_\tau(\boldsymbol{\beta}_k^*), \hat{\boldsymbol{\delta}}_k\rangle + \sum_{k\in\mathcal{A}^c} \langle \nabla\Psi_\tau(\hat{\boldsymbol{\beta}}_k) - \nabla\Psi_\tau(\boldsymbol{\beta}_k^*), \hat{\boldsymbol{\delta}}_k\rangle \\
&\overset{(a)}{=} \sum_{k\in\mathcal{A}} \langle -\mathbf{u}_k - \nabla\Psi_\tau(\boldsymbol{\beta}_k^*), \hat{\boldsymbol{\delta}}_k\rangle + \sum_{k\in\mathcal{A}^c} \langle -\mathbf{u}_k, \hat{\boldsymbol{\beta}}_k\rangle + \sum_{k\in\mathcal{A}^c} \langle -\nabla\Psi_\tau(\boldsymbol{\beta}_k^*), \hat{\boldsymbol{\beta}}_k\rangle \\
&\le \sum_{k\in\mathcal{A}} \| -\mathbf{u}_k - \nabla\Psi_\tau(\boldsymbol{\beta}_k^*)\|_2 \|\hat{\boldsymbol{\delta}}_k\|_2 - \sum_{k\in\mathcal{A}^c} \lambda^{\mathrm{GLasso}} \|\hat{\boldsymbol{\beta}}_k\|_2 \\
&+ \sum_{k\in\mathcal{A}^c} \| -\nabla\Psi_\tau(\boldsymbol{\beta}_k^*)\|_2 \|\hat{\boldsymbol{\beta}}_k\|_2 \\
&\le \sum_{k\in\mathcal{A}} \| -\mathbf{u}_k\|_2 \|\hat{\boldsymbol{\delta}}_k\|_2 + \sum_{k\in\mathcal{A}} \| -\nabla\Psi_\tau(\boldsymbol{\beta}_k^*)\|_2 \|\hat{\boldsymbol{\delta}}_k\|_2 - \lambda^{\mathrm{GLasso}} \sum_{k\in\mathcal{A}^c} \|\hat{\boldsymbol{\beta}}_k\|_2 \\
&+ \sum_{k\in\mathcal{A}^c} \| -\nabla\Psi_\tau(\boldsymbol{\beta}_k^*)\|_2 \|\hat{\boldsymbol{\beta}}_k\|_2.
\end{aligned}
$$

Equality (a) is due to equation (3.35) and $\hat{\boldsymbol{\delta}}_k = \hat{\boldsymbol{\beta}}_k$ for $k \in \mathcal{A}^c$.

From the last inequality, we get

$$
0 \le 2\underline{c}\|X\hat{\boldsymbol{\delta}}\|_2^2/n \le (z_\infty^* + \lambda^{\mathrm{GLasso}})\|\hat{\boldsymbol{\delta}}_{\mathcal{A}}\|_{2,1} + (z_\infty^* - \lambda^{\mathrm{GLasso}})\|\hat{\boldsymbol{\delta}}_{\mathcal{A}^c}\|_{2,1}. \tag{3.36}
$$

Under the event $\xi_N = \{z_\infty^* \le \lambda^{\mathrm{GLasso}}/2\}$, we have

$$
\|\hat{\boldsymbol{\delta}}_{\mathcal{A}^c}\|_{2,1} \le \frac{z_\infty + \lambda^{\mathrm{GLasso}}}{-z_\infty + \lambda^{\mathrm{GLasso}}} \|\hat{\boldsymbol{\delta}}_{\mathcal{A}}\|_{2,1} \le 3\|\hat{\boldsymbol{\delta}}_{\mathcal{A}}\|_{2,1},
$$

which implies that $\hat{\boldsymbol{\delta}} \in \mathcal{C}_3$ satisfies the condition (C3). It follows that

$$
2\underline{c}\kappa\|\hat{\boldsymbol{\delta}}\|_{2,1}^2 \le 2\underline{c}\|X\hat{\boldsymbol{\delta}}\|_2^2/n \le \frac{3}{2}\lambda^{\mathrm{GLasso}}\|\hat{\boldsymbol{\delta}}_{\mathcal{A}}\|_{2,1} \le \frac{3}{2}\lambda^{\mathrm{GLasso}}\|\hat{\boldsymbol{\delta}}\|_{2,1}.
$$

Thus, one has

$$\|\hat{\boldsymbol{\delta}}\|_{2,1} \leq 3\lambda^{\text{GLasso}}(4\kappa\underline{c})^{-1}.$$

Similarly, by condition (C4) and equation (3.36), we deduce

$$2n\underline{c}\varrho\|\hat{\boldsymbol{\delta}}\|_{2,1}\|\hat{\boldsymbol{\delta}}\|_{2,\infty} \leq 2\underline{c}\|\mathbf{X}\hat{\boldsymbol{\delta}}\|_2^2/n \leq \frac{3}{2}\lambda^{\text{GLasso}}\|\hat{\boldsymbol{\delta}}_{\mathcal{A}}\|_{2,1} \leq \frac{3}{2}\lambda^{\text{GLasso}}\|\hat{\boldsymbol{\delta}}\|_{2,1};$$

then

$$\|\hat{\boldsymbol{\delta}}\|_{2,\infty} \leq 3\lambda^{\text{GLasso}}(4\underline{c}\varrho)^{-1}.$$

Thus, we have

$$\begin{aligned} P\Bigg(\big[\|\hat{\boldsymbol{\delta}}\|_{2,1} \leq 3\lambda^{\text{GLasso}}(4\kappa\underline{c})^{-1}\big] \cap \big[\|\hat{\boldsymbol{\delta}}\|_{2,\infty} \leq 3\lambda^{\text{GLasso}}(4\underline{c}\varrho)^{-1}\big]\Bigg) &\geq P(z_\infty^* \leq \lambda^{\text{GLasso}}/2) \\ &\geq 1 - P\Bigg(\|\nabla\Psi_\tau(\boldsymbol{\beta}^*)\|_{2,\infty} \geq \lambda^{\text{GLasso}}/2\Bigg). \end{aligned} \tag{3.37}$$

Developing the last term of (3.37), we have

$$\begin{aligned} P\Bigg(\|\nabla\Psi_\tau(\boldsymbol{\beta}^*)\|_{2,\infty} \geq \lambda^{\text{GLasso}}/2\Bigg) &\leq \sum_{k=1}^{K} P\Bigg(\|\nabla\Psi_\tau(\boldsymbol{\beta}_k^*)\|_2 \geq \lambda^{\text{GLasso}}/2\Bigg) \\ &\leq \sum_{k=1}^{K} P\Bigg(\|\frac{\mathbf{x}_k^\top}{n}\boldsymbol{Z}\|_2 \geq \lambda^{\text{GLasso}}/2\Bigg) \\ &\leq \sum_{k=1}^{K} P\Bigg(\|\frac{\mathbf{x}_k^\top}{\sqrt{n}}\boldsymbol{Z}\|_\infty \geq \frac{\sqrt{n}\lambda^{\text{GLasso}}}{2\sqrt{p_k}}\Bigg) \\ &\leq \sum_{k=1}^{K} p_k \text{max}_{1\leq j\leq p_k} P\Bigg(|\frac{\mathbf{x}_k^\top}{\sqrt{n}}\boldsymbol{Z}| \geq \frac{\sqrt{n}\lambda^{\text{GLasso}}}{2\sqrt{p_k}}\Bigg). \end{aligned}$$

Note that $Z_i = 2\tau\epsilon_i^+ - 2(1-\tau)\epsilon_i^-$, where $\epsilon^+ = \text{max}(\epsilon, 0)$, $\epsilon^- = \text{max}(-\epsilon, 0)$. It follows by part (2) of

Lemma 2 and $\mathscr{E}^{\tau}(\epsilon_i) = 0$ that Z_i are i.i.d sub-Gaussian random variables. Now by part (1) of Lemma 2 we have

$$\begin{aligned} P\biggl(\|\nabla\Psi_\tau(\boldsymbol{\beta}_k)\|_{2,\infty} \geq \lambda^{\text{GLasso}}/2\biggr) &\leq \sum_{k=1}^{K} 2p_k \,\text{exp}(-\frac{Cn(\lambda^{\text{GLasso}})^2}{4K_0^2 M_0 p_k}) \\ &\leq 2p \,\text{exp}(-\frac{Cn(\lambda^{\text{GLasso}})^2}{4K_0^2 M_0^2 \overline{p}_m}). \end{aligned} \tag{3.38}$$

From (3.37) and (3.38) we deduce

$$P\biggl((\|\hat{\boldsymbol{\delta}}\|_{2,1} \leq 3\lambda^{\text{GLasso}}(4\kappa\underline{c})^{-1}) \cap (\|\hat{\boldsymbol{\delta}}\|_{2,\infty} \leq 3\lambda^{\text{GLasso}}(4\underline{c}\varrho)^{-1})\biggr) \geq 1 - 2p \,\text{exp}\biggl(-\frac{Cn(\lambda^{\text{GLasso}})^2}{4K_0^2 M_0^0 \overline{p}_m}\biggr).$$

This ends the proof of Theorem 3.3. □ **Proof of Theorem 3.4** Let $\hat{\boldsymbol{\beta}}^{(0)} = \hat{\boldsymbol{\beta}}^{\text{GLasso}}$, under the condition $\|\hat{\boldsymbol{\beta}}^{(0)} - \boldsymbol{\beta}^*\|_{2,\infty} \leq a_0\lambda$ and by assumptions (A1) and $a_0 = 1 \wedge a_2$, we have

for $k \in \mathcal{A}^c$

$$\begin{aligned} \|\hat{\boldsymbol{\beta}}_k^{(0)}\|_2 &\leq \|\hat{\boldsymbol{\beta}}^{(0)} - \boldsymbol{\beta}^*\|_{2,\infty} \;\; for\; k \in \mathcal{A}^c \; (\boldsymbol{\beta}_k^* = 0) \\ &\leq a_0\lambda \\ &\leq a_2\lambda. \end{aligned} \tag{3.39}$$

For $k \in \mathcal{A}$

$$\begin{aligned} \|\hat{\boldsymbol{\beta}}_k^{(0)}\|_2 &\geq \text{min}_{k\in\mathcal{A}} \|\boldsymbol{\beta}_k^*\|_2 - \|\hat{\boldsymbol{\beta}}^{(0)} - \boldsymbol{\beta}^*\|_{2,\infty} \\ &> (1+a)\lambda - a_0\lambda \\ &> a\lambda. \end{aligned}$$

By the last inequality and under property (P5), we have $P_\lambda(\|\hat{\boldsymbol{\beta}}_k^{(0)})\|_2 = 0$ for all $k \in \mathcal{A}$. Then, $\hat{\boldsymbol{\beta}}^{(1)}$ is solution to the following problem

$$\widehat{\boldsymbol{\beta}}^{(1)} = \arg\min_{\boldsymbol{\beta}} \big(\Psi_\tau(\boldsymbol{\beta}) + \sum_{k\in\mathcal{A}^c} P'_\lambda(\|\boldsymbol{\beta}_k^{(0)}\|_2)\|\boldsymbol{\beta}_k\|_2\big). \tag{3.40}$$

By properties $(P3)$ and $(P4)$ and inequation (3.39), the inequality $P'_\lambda(\|\boldsymbol{\beta}_k^{(0)}\|_2) \geq a_1\lambda$ holds for $k \in \mathcal{A}^c$. Under the event $\{\|\nabla_k\Psi_\tau(\hat{\boldsymbol{\beta}}^{oracle})\|_2 < a_1\lambda, \forall k \in \mathcal{A}^c\}$, we demonstrate that $\hat{\boldsymbol{\beta}}^{oracle}$ is the unique global solution to (3.40). Indeed, from the convexity of Ψ_τ we obtain

$$\begin{aligned}\Psi_\tau(\boldsymbol{\beta}) &\geq \Psi_\tau(\hat{\boldsymbol{\beta}}^{oracle}) + \sum_{k=1}^{K}\langle\nabla_k\Psi_\tau(\hat{\boldsymbol{\beta}}^{oracle}), \boldsymbol{\beta}_k - \hat{\boldsymbol{\beta}}_k^{oracle}\rangle;\\ &\overset{(a)}{=} \Psi_\tau(\hat{\boldsymbol{\beta}}^{oracle}) + \sum_{k\in\mathcal{A}^c}\langle\nabla_k\Psi_\tau(\hat{\boldsymbol{\beta}}^{oracle}), \boldsymbol{\beta}_k - \hat{\boldsymbol{\beta}}_k^{oracle}\rangle.\end{aligned} \tag{3.41}$$

Equality (a) is due to $\nabla_k\Psi_\tau(\hat{\boldsymbol{\beta}}^{oracle}) = 0$ for all $k \in \mathcal{A}$ (KKT conditions of problem (3.10)). Using the inequality (3.41) leads to the following inequality

$$\begin{aligned}&\Big(\Psi_\tau(\boldsymbol{\beta}) + \sum_{k\in\mathcal{A}^c} P'_\lambda(\|\boldsymbol{\beta}_k^{(0)}\|_2)\|\boldsymbol{\beta}_k\|_2\Big) - \Big(\Psi_\tau(\hat{\boldsymbol{\beta}}^{oracle}) + \sum_{k\in\mathcal{A}^c}\langle\nabla_k\Psi_\tau(\hat{\boldsymbol{\beta}}^{oracle}), \hat{\boldsymbol{\beta}}_k^{oracle}\rangle\Big)\\ &\overset{(a)}{\geq} \sum_{k\in\mathcal{A}^c}\Big(P'_\lambda(\|\boldsymbol{\beta}_k^{(0)}\|_2 - \|\nabla_k\Psi_\tau(\hat{\boldsymbol{\beta}}^{oracle})\|_2\Big)\|\boldsymbol{\beta}_k\|_2\\ &\overset{(b)}{\geq} \sum_{k\in\mathcal{A}^c}\Big(a_1\lambda - \|\nabla_k\Psi_\tau(\hat{\boldsymbol{\beta}}^{oracle})\|_2\Big)\|\boldsymbol{\beta}_k\|_2\\ &\geq 0.\end{aligned}$$

Inequalities (a) and (b) are due to the fact that $\langle\nabla_k\Psi_\tau(\hat{\boldsymbol{\beta}}^{oracle}), \hat{\boldsymbol{\beta}}_k\rangle \geq -\|\nabla_k\Psi_\tau(\hat{\boldsymbol{\beta}}^{oracle})\|_2\|\boldsymbol{\beta}_k\|_2$ and the condition (P4). Combining the last inequality with the uniqueness of the solution of problem (3.10), we conclude that $\hat{\boldsymbol{\beta}}^{oracle}$ is the unique solution to (3.40). Hence $\hat{\boldsymbol{\beta}}^{(1)} = \hat{\boldsymbol{\beta}}^{oracle}$. We start the second iteration of GLLA algorithm with the initial value $\hat{\boldsymbol{\beta}}^{oracle}$ solution of the problem (3.40) at the first iteration.

Let $\hat{\boldsymbol{\beta}}$ be the solution to the convex optimization problem in the second iteration of the GLLA algorithm. Under the event $\{\min_{k\in\mathcal{A}}\|\hat{\boldsymbol{\beta}}_k^{oracle}\|_2 > a\lambda\}$, we have $P'_\lambda(\|\hat{\boldsymbol{\beta}}_k^{oracle}\|_2) = 0,\ \forall k \in \mathcal{A}$ (condition (P5)). So, we obtain

$$\widehat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \Bigg(\varPsi_\tau(\boldsymbol{\beta}) + \sum_{k \in \mathcal{A}^c} P'_\lambda(\hat{\boldsymbol{\beta}}_k^{oracle}) \|\boldsymbol{\beta}_k\|_2 \Bigg). \tag{3.42}$$

Using $\hat{\boldsymbol{\beta}}_k^{oracle} = 0,\ \forall k \in \mathcal{A}^c$ and condition $(P4)$, we have $P'_\lambda(\|\hat{\boldsymbol{\beta}}_k^{oracle}\|_2) = P'_\lambda(0) \geq a_1\lambda$. Hence, the problem (3.42) is very similar to (3.40). We deduce that $\hat{\boldsymbol{\beta}}^{oracle}$ is the unique solution to (3.42) under the event $\{\|\nabla_{\mathcal{A}}\varPsi_\tau(\hat{\boldsymbol{\beta}}^{oracle})\|_{2,\infty} < a_1\lambda\}$. Then, under the assumption of Theorem 3.4, the probability that Algorithm 5 initialized by $\hat{\boldsymbol{\beta}}^{\mathrm{GLasso}}$ given by Theorem 3.3 converges to $\hat{\boldsymbol{\beta}}^{oracle}$ after two iterations is at least $1 - p_1 - p_2 - p_3$, where

$$\begin{aligned} p_1 &= P(\|\hat{\boldsymbol{\beta}}^{\mathrm{GLasso}} - \boldsymbol{\beta}^*\|_{2,\infty} > a_0\lambda), \\ p_2 &= P(\{\|\nabla_{\mathcal{A}^c}\varPsi_\tau(\hat{\boldsymbol{\beta}}^{oracle})\|_{2,\infty} \geq a_1\lambda\}), \\ p_3 &= P(\{\text{min}_{k\in\mathcal{A}}\|\hat{\boldsymbol{\beta}}_k^{oracle}\|_2 \leq a\lambda\}). \end{aligned}$$

Let $\hat{\boldsymbol{\delta}} = \hat{\boldsymbol{\beta}}^{\mathrm{GLasso}} - \boldsymbol{\beta}^*$. By the assumption $\lambda \geq 3\lambda^{\mathrm{GLasso}} a_0^{-1}\big((4\underline{c}\varrho)^{-1} \wedge (4\kappa\underline{c})^{-1}\big)$ and Theorem 3.3, we immediately get

$$\begin{aligned} p_1 &\leq P(\|\hat{\boldsymbol{\delta}}\|_{2,\infty} > 3\lambda^{\mathrm{GLasso}}\big((4\underline{c}\varrho)^{-1} \wedge (4\kappa\underline{c})^{-1}\big)) \\ &\leq P\Bigg(\|\hat{\boldsymbol{\delta}}\|_{2,1} > 3\lambda^{\mathrm{GLasso}}(4\kappa\underline{c})^{-1}\Bigg) \vee P\Bigg(\|\hat{\boldsymbol{\delta}}\|_{2,\infty} > 3\lambda^{\mathrm{GLasso}}(4c_0\varrho)^{-1}\Bigg) \\ &\leq p^*. \end{aligned}$$

To establishes the bound for p_2, we have

$$\begin{aligned} p_2 &= P(\{\|\nabla_{\mathcal{A}^c}\varPsi_\tau(\hat{\boldsymbol{\beta}}^{oracle})\|_{2,\infty} \geq a_1\lambda\}) \\ &\leq P(\{\|\nabla_{\mathcal{A}^c}\varPsi_\tau(\boldsymbol{\beta}^*)\|_{2,\infty} \geq a_1\lambda/2\}) \\ &\quad + P(\{\|\nabla_{\mathcal{A}^c}\varPsi_\tau(\boldsymbol{\beta}^*) - \nabla_{\mathcal{A}^c}\varPsi_\tau(\hat{\boldsymbol{\beta}}^{oracle})\|_{2,\infty} \geq a_1\lambda/2\}). \end{aligned} \tag{3.43}$$

By the same reasoning as in (3.38), we deduce

$$P(\{\|\nabla_{\mathcal{A}^c}\Psi_\tau(\boldsymbol{\beta}^*)\|_{2,\infty} \geq a_1\lambda/2\}) \leq 2(p - s_{\mathcal{A}})\ \text{exp}\big(-\frac{Cn\lambda^2a_1^2}{4K_0^2M_0^2\overline{p}_m}\big). \tag{3.44}$$

Developing the second term of (3.43), we have

$$\begin{aligned}
&P\bigg(\{\|\nabla_{\mathcal{A}^c}\Psi_\tau(\boldsymbol{\beta}^*) - \nabla_{\mathcal{A}^c}\Psi_\tau(\hat{\boldsymbol{\beta}}^{oracle})\|_{2,\infty} \geq a_1\lambda/2\}\bigg) \\
&\qquad\qquad \leq P\bigg(\{\max_{k\in\mathcal{A}^c} p_k^{1/2}\|\nabla_k\Psi_\tau(\boldsymbol{\beta}^*) - \nabla_k\Psi_\tau(\hat{\boldsymbol{\beta}}^{oracle})\|_\infty \geq a_1\lambda/2\}\bigg) \\
&\qquad\qquad \leq P\bigg(\|\nabla_{\mathcal{A}^c}\Psi_\tau(\boldsymbol{\beta}^*) - \nabla_{\mathcal{A}^c}\Psi_\tau(\hat{\boldsymbol{\beta}}^{oracle})\|_\infty \geq \frac{a_1\lambda}{2\overline{p}_m^{1/2}}\bigg).
\end{aligned} \tag{3.45}$$

Let $\mathbf{d} = (d_i, i = 1\ldots, n)^\top$ with $d_i = \Psi_\tau'(y_i - \mathbf{x}_i^\top\hat{\boldsymbol{\beta}}^{oracle}) - \Psi_\tau'(y_i - \mathbf{x}_i^\top\boldsymbol{\beta}^*)$. Using the Cauchy-Schwarz inequality and Lemma 2 of [33], we have

$$\begin{aligned}
&\|\nabla_{\mathcal{A}^c}\Psi_\tau(\boldsymbol{\beta}^*) - \nabla_{\mathcal{A}^c}\Psi_\tau(\hat{\boldsymbol{\beta}}^{oracle})\|_\infty \\
&\qquad\qquad = n^{-1}\max_{k\in\mathcal{A}^c}|\sum_i^n d_kx_{ik}| \\
&\qquad\qquad \leq n^{-1}\max_{k\in\mathcal{A}^c}\|\mathbf{d}\|_2\|X_k\|_2 \\
&\qquad\qquad \leq (2\overline{c}M_0)[(\hat{\boldsymbol{\beta}}_{\mathcal{A}}^{oracle} - \boldsymbol{\beta}_{\mathcal{A}}^*)^\top(n^{-1}\mathbf{X}_{\mathcal{A}}^\top\mathbf{X}_{\mathcal{A}})(\hat{\boldsymbol{\beta}}_{\mathcal{A}}^{oracle} - \boldsymbol{\beta}_{\mathcal{A}}^*)]^{1/2} \\
&\qquad\qquad \leq 2\overline{c}M_0\rho_{\max}^{1/2}\|\hat{\boldsymbol{\beta}}^{oracle} - \boldsymbol{\beta}^*\|_2.
\end{aligned} \tag{3.46}$$

Combining (3.45) and (3.46), it follows from Lemma 3 and Lemma 4 of [33] that

$$\begin{aligned}
&P\bigg(\{\|\nabla_{\mathcal{A}^c}\Psi_\tau(\boldsymbol{\beta}^*) - \nabla_{\mathcal{A}^c}\Psi_\tau(\hat{\boldsymbol{\beta}}^{oracle})\|_{2,\infty} \geq a_1\lambda/2\}\bigg) \\
&\qquad\qquad \leq P\bigg(\|\hat{\boldsymbol{\beta}}^{oracle} - \boldsymbol{\beta}^*\|_2 \geq \frac{a_1\lambda}{4\overline{c}M_0\rho_{\max}^{1/2}\overline{p}_m^{1/2}}\bigg) \\
&\qquad\qquad \overset{(a)}{\leq} P\bigg(\|n^{-1}\mathbf{X}_{\mathcal{A}}^\top\xi\|_2 \geq \frac{a_1\underline{c}\rho_{\min}}{2\overline{c}M_0\rho_{\max}^{1/2}\overline{p}_m^{1/2}}\lambda\bigg) \\
&\qquad\qquad = P\bigg(\|n^{-1}\mathbf{X}_{\mathcal{A}}^\top\xi\|_2 \geq Q_1\lambda\bigg) \\
&\qquad\qquad \overset{(b)}{\leq} \Gamma(Q_1\lambda, n, s_{\mathcal{A}}, K_0, M_0, \rho_{\max}, \nu_0).
\end{aligned} \tag{3.47}$$

The inequalities (a) and (b) are due to the Lemmas 4(2) and 3(3) of [33] respectively. Combining (3.43), (3.44) and (3.47), we immediately get the upper bound for p_2

$$p_2 = 2(p - s_{\mathcal{A}})\,\text{exp}\big(-\frac{Cn\lambda^2 a_1^2}{4K_0^2 M_0^2 \overline{p}_m}\big) + \Gamma(Q_1\lambda, n, s_{\mathcal{A}}, K_0, M_0, \rho_{\max}, \nu_0). \tag{3.48}$$

To derive the upper bound for p_3, let $R = \text{min}_{k\in\mathcal{A}}\|\boldsymbol{\beta}_k^*\|_2 - a\lambda > 0$. Then, we have

$$\begin{aligned}
P(\mathcal{E}_3^c) &\le P(\{\text{min}_{k\in\mathcal{A}}\|\hat{\boldsymbol{\beta}}_k^{oracle}\|_2 \le a\lambda\}) \\
&\le P(\max_{k\in\mathcal{A}}\|\hat{\boldsymbol{\beta}}_k^{oracle} - \boldsymbol{\beta}_k^*\|_2 > R) \\
&\le P(\max_{k\in\mathcal{A}} p_k^{1/2}\|\hat{\boldsymbol{\beta}}_k^{oracle} - \boldsymbol{\beta}_k^*\|_\infty > R) \\
&\le P(\|\hat{\boldsymbol{\beta}}_{\mathcal{A}}^{oracle} - \boldsymbol{\beta}_{\mathcal{A}}^*\|_\infty > \frac{R}{\overline{p}_{\mathcal{A}}}) \\
&\le P\bigg(\|n^{-1}\mathbf{X}_{\mathcal{A}}^\top \xi\|_2 \ge 2\underline{c}\rho_{\min} R\overline{p}_{\mathcal{A}}^{-1}\bigg) \\
&\overset{(a)}{\le} \Gamma(2\underline{c}\rho_{\min} R\overline{p}_{\mathcal{A}}^{-1}, n, s_{\mathcal{A}}, K_0, M_0, \rho_{\max}, \nu_0),
\end{aligned} \tag{3.49}$$

where the inequality (a) is due to Lemma 3(3) of [33].

This ends the proof of Theorem 3.4. □

Proof of Theorem 3.6 Let $\boldsymbol{\delta}_1 = \hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*$, $\boldsymbol{\delta}_2 = \hat{\boldsymbol{\phi}} - \boldsymbol{\phi}^*$, $\hat{\boldsymbol{\theta}} = (\hat{\boldsymbol{\beta}}^\top, \hat{\boldsymbol{\phi}}^\top)^\top$, $\hat{\boldsymbol{\delta}} = (\hat{\boldsymbol{\delta}}_2^\top, \hat{\boldsymbol{\delta}}_2^\top)^\top$, $z_{1\infty}^* = \|\partial S_\tau(\boldsymbol{\theta}^*)/\partial\boldsymbol{\beta}_k\|_{2,\infty}$ and $z_{2\infty}^* = \|\partial S_\tau(\boldsymbol{\theta}^*)/\partial\boldsymbol{\phi}_k\|_{2,\infty}$. By Lemma 1 and similar arguments in the proof of Theorem 3.3, we get

$$
\begin{aligned}
0 \leq n^{-1}c_0\|(\boldsymbol{I}_2 \otimes \mathbf{X})\hat{\boldsymbol{\delta}}\|_2^2/n \leq{}& \langle \nabla S_\tau(\hat{\boldsymbol{\theta}}) - \nabla S_\tau(\boldsymbol{\theta}^*), \hat{\boldsymbol{\delta}} \rangle \\
={}& \sum_{k=1}^{K} \langle \nabla S_\tau(\hat{\boldsymbol{\theta}}_k) - \nabla S_\tau(\boldsymbol{\theta}_k^*), \hat{\boldsymbol{\delta}}_k \rangle \\
={}& \sum_{k\in\mathcal{A}_1} \langle \nabla S_\tau(\hat{\boldsymbol{\theta}}_k) - \nabla S_\tau(\boldsymbol{\theta}_k^*), \hat{\boldsymbol{\delta}}_k \rangle + \sum_{k\in\mathcal{A}_1^c} \langle \nabla S_\tau(\hat{\boldsymbol{\theta}}_k) - \nabla S_\tau(\boldsymbol{\theta}_k^*), \hat{\boldsymbol{\delta}}_k \rangle \\
&+ \sum_{k\in\mathcal{A}_2} \langle \nabla S_\tau(\hat{\boldsymbol{\theta}}_k) - \nabla S_\tau(\boldsymbol{\theta}_k^*), \hat{\boldsymbol{\delta}}_k \rangle + \sum_{k\in\mathcal{A}_2^c} \langle \nabla S_\tau(\hat{\boldsymbol{\theta}}_k) - \nabla S_\tau(\boldsymbol{\theta}_k^*), \hat{\boldsymbol{\delta}}_k \rangle \\
\leq{}& (z_{1\infty}^* + \lambda_1^{\text{GLasso}})\|(\hat{\boldsymbol{\delta}}_1)_{\mathcal{A}_1}\|_{2,1} + (z_{1\infty}^* - \lambda_1^{\text{GLasso}})\|(\hat{\boldsymbol{\delta}}_1)_{\mathcal{A}_1^c}\|_{2,1} \\
&+ (z_{2\infty}^* + \lambda_2^{\text{GLasso}})\|(\hat{\boldsymbol{\delta}}_2)_{\mathcal{A}_2}\|_{2,1} + (z_{2\infty}^* - \lambda_2^{\text{GLasso}})\|(\hat{\boldsymbol{\delta}}_2)_{\mathcal{A}_2^c}\|_{2,1}.
\end{aligned}
\tag{3.50}
$$

Under the event $\xi_1 = \{z_{1\infty}^* \leq \lambda_1^{\text{GLasso}}/2\}$ and $\xi_2 = \{z_{2\infty}^* \leq \lambda_2^{\text{GLasso}}/2\}$, it follows from the later inequality that

$$
\begin{aligned}
(-z_{1\infty}^* + \lambda_1^{\text{GLasso}})\|(\hat{\boldsymbol{\delta}}_1)_{\mathcal{A}_1^c}\|_{2,1} + (-z_{2\infty}^* + \lambda_2^{\text{GLasso}})\|(\hat{\boldsymbol{\delta}}_2)_{\mathcal{A}_2^c}\|_{2,1} \leq (z_{1\infty}^* + \lambda_1^{\text{GLasso}})\|(\hat{\boldsymbol{\delta}}_1)_{\mathcal{A}_1}\|_{2,1}+ \\
(z_{2\infty}^* + \lambda_2^{\text{GLasso}})\|(\hat{\boldsymbol{\delta}}_2)_{\mathcal{A}_2}\|_{2,1},
\end{aligned}
$$

which implies that

$$
\begin{aligned}
2^{-1}\lambda_1^{\text{GLasso}}\|(\hat{\boldsymbol{\delta}}_1)_{\mathcal{A}_1^c}\|_{2,1} + 2^{-1}\lambda_2^{\text{GLasso}}\|(\hat{\boldsymbol{\delta}}_2)_{\mathcal{A}_2^c}\|_{2,1} \leq{}& (3/2)\lambda_1^{\text{GLasso}}\|(\hat{\boldsymbol{\delta}}_1)_{\mathcal{A}_1}\|_{2,1} \\
&+ (3/2)\lambda_2^{\text{GLasso}}\|(\hat{\boldsymbol{\delta}}_2)_{\mathcal{A}_2}\|_{2,1}.
\end{aligned}
$$

Thus, we have

$$
\begin{aligned}
2^{-1}\underline{\lambda}^{\text{GLasso}}\|\hat{\boldsymbol{\delta}}_{\mathcal{A}_0^c}\|_{2,1} \leq{}& 2^{-1}\lambda_1^{\text{GLasso}}\|(\hat{\boldsymbol{\delta}}_1)_{\mathcal{A}_1^c}\|_{2,1} + 2^{-1}\lambda_2^{\text{GLasso}}\|(\hat{\boldsymbol{\delta}}_2)_{\mathcal{A}_2^c}\|_{2,1} \\
\leq{}& (3/2)\lambda_1^{\text{GLasso}}\|(\hat{\boldsymbol{\delta}}_1)_{\mathcal{A}_1}\|_{2,1} + (3/2)\lambda_2^{\text{GLasso}}\|(\hat{\boldsymbol{\delta}}_2)_{\mathcal{A}_2}\|_{2,1} \\
\leq{}& (3/2)\overline{\lambda}^{\text{GLasso}}\|\hat{\boldsymbol{\delta}}_{\mathcal{A}_0}\|_{2,1}.
\end{aligned}
$$

Then, we have $\hat{\boldsymbol{\delta}} \in \boldsymbol{\xi}_{3\tilde{N}}$. Now under conditions $(C3)' - (C4)'$ we have from (3.50)

$$
\begin{aligned}
c_0\overline{\kappa}\|\hat{\boldsymbol{\delta}}\|_{2,1}^2 \leq n^{-1}c_0\|(\boldsymbol{I}_2 \otimes \mathbf{X})\hat{\boldsymbol{\delta}}\|_2^2 \leq{}& (3/2)\overline{\lambda}^{\text{GLasso}}\|\hat{\boldsymbol{\delta}}_{\mathcal{A}_0}\|_{2,1} \\
\leq{}& (3/2)\overline{\lambda}^{\text{GLasso}}\|\hat{\boldsymbol{\delta}}\|_{2,1};
\end{aligned}
$$

then, one has

$$\|\hat{\boldsymbol{\delta}}\|_{2,1} \leq (3/2)\overline{\lambda}^{\text{GLasso}}(c_0\overline{\kappa})^{-1}.$$

Similarly, by condition $(C5)'$

$$c_0\varrho\|\hat{\boldsymbol{\delta}}_{\mathcal{A}_0}\|_{2,1}\|\hat{\boldsymbol{\delta}}\|_{2,\infty} \leq n^{-1}c_0\|(\boldsymbol{I}_2 \otimes \mathbf{X})\hat{\boldsymbol{\delta}}\|_2^2 \leq (3/2)\overline{\lambda}^{\text{GLasso}}\|\hat{\boldsymbol{\delta}}_{\mathcal{A}_0}\|_{2,1};$$

thus,

$$\|\hat{\boldsymbol{\delta}}\|_{2,\infty} \leq (3/2)\overline{\lambda}^{\text{GLasso}}(c_0\varrho)^{-1}.$$

It follows that under event ξ_1 and ξ_2, we have $\|\hat{\boldsymbol{\delta}}\|_{2,\infty} \leq (3/2)(c_0\varrho)^{-1}\overline{\lambda}^{\text{GLasso}}$ and $\|\hat{\boldsymbol{\delta}}\|_{2,1} \leq (3/2)(c_0\kappa)^{-1}\overline{\lambda}^{\text{GLasso}}$. By Lemma 6 of [33], ϵ_i and $\eta = S_\tau'(\epsilon_i - e_\tau)$ are both mean zero sub-Gaussian random variables with $K_1 = \|\epsilon_i\|_{SG}$ and $K_2 = \|\eta_i\|_{SG}$. It follows that $\epsilon_i + \eta_i$ is also sub-Gaussian and we have $\|\epsilon_i + \eta_i\|_{SG} \leq K_1 + K_2$. Since $M_1 = \|\mathbf{X}\boldsymbol{\theta}^*\|_\infty$, we get

$$
\begin{aligned}
&P\bigg(\big[\|\hat{\boldsymbol{\delta}}\|_{2,1} \le (3/2)(c_0\kappa)^{-1}\overline{\lambda}^{\mathrm{GLasso}}\big] \cap \big[\|\hat{\boldsymbol{\delta}}\|_{2,\infty} \le (3/2)(c_0\varrho)^{-1}\overline{\lambda}^{\mathrm{GLasso}}\big]\bigg)\\
&\qquad \ge P(\xi_1 \cap \xi_2)\\
&\qquad \ge 1 - P(\xi_1^c) - P(\xi_2^c)\\
&\qquad = 1 - P\bigg(\|n^{-1}\mathbf{X}^\top \boldsymbol{W}(\boldsymbol{\epsilon}+\boldsymbol{\eta})\|_{2,\infty} \ge \lambda_1^{\mathrm{GLasso}}/2\bigg) - P\bigg(\|n^{-1}\mathbf{X}^\top \boldsymbol{W}\boldsymbol{\eta}\|_{2,\infty} \ge \lambda_2^{\mathrm{GLasso}}/2\bigg)\\
&\qquad \ge 1 - \sum_{k=1}^{K} P\bigg(\|n^{-1}\mathbf{x}_k^\top \boldsymbol{W}(\boldsymbol{\epsilon}+\boldsymbol{\eta})\|_{2} \ge \lambda_1^{\mathrm{GLasso}}/2\bigg) - \sum_{k=1}^{K} P\bigg(\|n^{-1}\mathbf{x}_k^\top \boldsymbol{W}\boldsymbol{\eta}\| \ge \lambda_2^{\mathrm{GLasso}}/2\bigg)\\
&\qquad \ge 1 - \sum_{k=1}^{K} P\bigg(\|n^{-1}\mathbf{x}_k^\top \boldsymbol{W}(\boldsymbol{\epsilon}+\boldsymbol{\eta})\|_{\infty} \ge \lambda_1^{\mathrm{GLasso}}/(2\sqrt{p_k})\bigg)\\
&\qquad\qquad - \sum_{k=1}^{K} P\bigg(\|n^{-1}\mathbf{x}_k^\top \boldsymbol{W}\boldsymbol{\eta}\|_{\infty} \ge \lambda_2^{\mathrm{GLasso}}/(2\sqrt{p_k})\bigg)\\
&\qquad \ge 1 - \sum_{k=1}^{K} 2p_k \exp\bigg(- \frac{Cn(\lambda_1^{\mathrm{GLasso}})^2}{4(K_1+K_2)^2 M_0^2 M_1^2 p_k}\bigg) -\\
&\qquad\qquad\qquad\qquad \sum_{k=1}^{K} 2p_k \exp\bigg(- \frac{Cn(\lambda_2^{\mathrm{GLasso}})^2}{4K_2^2 M_0^2 M_1^2 p_k}\bigg)\\
&\qquad \ge 1 - 2p\exp\bigg(- \frac{Cn(\lambda_1^{\mathrm{GLasso}})^2}{4(K_1+K_2)^2 M_0^2 M_1^2 \overline{p}_m}\bigg) - 2p\exp\bigg(- \frac{Cn(\lambda_2^{\mathrm{GLasso}})^2}{4K_2^2 M_0^2 M_1^2 \overline{p}_m}\bigg).
\end{aligned}
$$

This ends the proof of Theorem 3.6. □ **Proof of Theorem 3.7** From Lemma 7 of [33], the restriction of $S_\tau(\boldsymbol{\beta}, \boldsymbol{\phi})$ to the set $\mathcal{S} = \{\boldsymbol{\beta}, \boldsymbol{\phi} \in \mathbb{R}^{2p} : \boldsymbol{\beta}_{\mathcal{A}_1^c} = \mathbf{0}, \boldsymbol{\phi}_{\mathcal{A}_2^c} = \mathbf{0}\}$ is strongly convex. Hence, the oracle estimators ($\hat{\boldsymbol{\beta}}^{oracle}$, $\hat{\boldsymbol{\phi}}^{oracle}$) are the unique solutions of problem (3.22).

Under the event

$$
\mathcal{E}_1 = \{\|\hat{\boldsymbol{\beta}}^{\mathrm{GLasso}} - \boldsymbol{\beta}^*\|_{2,\infty} \le a_0\lambda_1;\ \|\hat{\boldsymbol{\phi}}^{\mathrm{GLasso}} - \boldsymbol{\phi}^*\|_{2,\infty} \le a_0\lambda_2\}
$$

and assumption (A2), we have

$$
\min_{k\in\mathcal{A}_1} \|\hat{\boldsymbol{\beta}}_k^{\mathrm{GLasso}}\|_2 \ge \min_{k\in\mathcal{A}_1} \|\boldsymbol{\beta}_k^*\|_2 - \|\boldsymbol{\beta}^{\mathrm{GLasso}} - \boldsymbol{\beta}^*\|_{2,\infty} > a\lambda_1,
$$

which implies that $P'_\lambda(\|\hat{\boldsymbol{\beta}}_k^{\mathrm{GLasso}}\|_2) = 0$ for $k \in \mathcal{A}_1$.

We have also

$$
\|\hat{\boldsymbol{\beta}}_{\mathcal{A}_1^c}^{\mathrm{GLasso}}\|_{2,\infty} \le \|\boldsymbol{\beta}^{\mathrm{GLasso}} - \boldsymbol{\beta}^*\|_{2,\infty} \le a_2\lambda_1,
$$

implying that

$$P'_\lambda(\|\hat{\boldsymbol{\beta}}_k^{\mathrm{GLasso}}\|_2) \geq a_1\lambda_1 \text{ for } k \in \mathcal{A}_1^c.$$

A similar argument is used to show that $P'_\lambda(\|\hat{\boldsymbol{\phi}}_k^{\mathrm{GLasso}}\|_2) = 0$ for $k \in \mathcal{A}_2$ and $P'_\lambda(\|\hat{\boldsymbol{\phi}}_k^{\mathrm{GLasso}}\|_2) \geq a_1\lambda_2$ for $k \in \mathcal{A}_2^c$.

Now, let $\hat{\boldsymbol{\beta}}^1$ and $\hat{\boldsymbol{\phi}}^1$ be the update after the first iteration of the GLLA algorithm, then under $\mathcal{E}_1$, ($\hat{\boldsymbol{\beta}}^1$ $\hat{\boldsymbol{\phi}}^1$) is minimizers of

$$L_\tau(\boldsymbol{\beta}, \boldsymbol{\phi}) := S_\tau(\boldsymbol{\beta}, \boldsymbol{\phi}) + \sum_{k \in \mathcal{A}_1^c} P'_\lambda(\|\hat{\boldsymbol{\beta}}_k^{\mathrm{GLasso}}\|_2)\|\boldsymbol{\beta}_k\|_2 + \sum_{k \in \mathcal{A}_2^c} P'_\lambda(\|\hat{\boldsymbol{\phi}}_k^{\mathrm{GLasso}}\|_2)\|\boldsymbol{\phi}_k\|_2. \tag{3.51}$$

By definition of the oracle estimators, $\partial S_\tau(\hat{\boldsymbol{\beta}}^{oracle}, \hat{\boldsymbol{\phi}}^{oracle})/\partial \boldsymbol{\beta}_k = 0$ for $k \in \mathcal{A}_1$ and $\partial S_\tau(\hat{\boldsymbol{\beta}}^{oracle}, \hat{\boldsymbol{\phi}}^{oracle})/\partial \boldsymbol{\phi}_k = 0$ for $k \in \mathcal{A}_2$. Also $\hat{\boldsymbol{\beta}}_k^{oracle} = 0$ for $k \in \mathcal{A}_1^c$ and $\hat{\boldsymbol{\phi}}_k^{oracle} = 0$ for $k \in \mathcal{A}_2^c$.

It follows from convexity of $S_\tau(\boldsymbol{\beta}, \boldsymbol{\phi})$ that

$$\begin{aligned} S_\tau(\boldsymbol{\beta}, \boldsymbol{\phi}) &\geq S_\tau(\hat{\boldsymbol{\beta}}^{oracle}, \hat{\boldsymbol{\phi}}^{oracle}) + \sum_{k=1}^{K} \langle \nabla_k S_\tau(\hat{\boldsymbol{\beta}}^{oracle}, \hat{\boldsymbol{\phi}}^{oracle}), \boldsymbol{\delta}_k - \hat{\boldsymbol{\delta}}_k^{oracle} \rangle; \\ &= S_\tau(\hat{\boldsymbol{\beta}}^{oracle}, \hat{\boldsymbol{\phi}}^{oracle}) + \sum_{k \in \mathcal{A}_1^c} \langle \nabla_k S_\tau(\hat{\boldsymbol{\beta}}^{oracle}, \hat{\boldsymbol{\phi}}^{oracle}), \boldsymbol{\beta}_k - \hat{\boldsymbol{\beta}}_k^{oracle} \rangle \\ &\quad + \sum_{k \in \mathcal{A}_2^c} \langle \nabla_k S_\tau(\hat{\boldsymbol{\beta}}^{oracle}, \hat{\boldsymbol{\phi}}^{oracle}), \boldsymbol{\phi}_k - \hat{\boldsymbol{\phi}}_k^{oracle} \rangle \end{aligned} \tag{3.52}$$

Combining (3.51) and (3.52), we have

$$
\begin{aligned}
& L_\tau(\boldsymbol{\beta},\boldsymbol{\phi}) - L_\tau(\hat{\boldsymbol{\beta}}^{oracle},\hat{\boldsymbol{\phi}}^{oracle}) \\
& \overset{(a)}{\geq} \sum_{k\in\mathcal{A}_1^c}\left(P'_{\lambda_1}(\|\boldsymbol{\beta}_k^{(0)}\|_2 - \|\nabla_k S_\tau(\hat{\boldsymbol{\beta}}^{oracle},\hat{\boldsymbol{\phi}}^{oracle})\|_2\right)\|\boldsymbol{\beta}_k\|_2 \\
& \quad + \sum_{k\in\mathcal{A}_2^c}\left(P'_{\lambda}(\|\boldsymbol{\phi}_k^{(0)}\|_2 - \|\nabla_k S_\tau(\hat{\boldsymbol{\beta}}^{oracle},\hat{\boldsymbol{\phi}}^{oracle})\|_2\right)\|\boldsymbol{\phi}_k\|_2 \\
& \geq \sum_{k\in\mathcal{A}_1^c}\left(a_1\lambda_1 - \|\nabla_k S_\tau(\hat{\boldsymbol{\beta}}^{oracle},\hat{\boldsymbol{\phi}}^{oracle})\|_2\right)\|\boldsymbol{\beta}_k\|_2 \\
& \qquad + \sum_{k\in\mathcal{A}_2^c}\left(a_1\lambda_2 - \|\nabla_k S_\tau(\hat{\boldsymbol{\beta}}^{oracle},\hat{\boldsymbol{\phi}}^{oracle})\|_2\right)\|\boldsymbol{\phi}_k\|_2 \\
& \overset{(b)}{\geq} 0.
\end{aligned}
$$

Inequality (a) is due to

$$\langle\nabla_k S_\tau(\hat{\boldsymbol{\beta}}^{oracle},\hat{\boldsymbol{\phi}}^{oracle}),\hat{\boldsymbol{\beta}}_k^{oracle}\rangle \geq -\|\nabla_k S_\tau(\hat{\boldsymbol{\beta}}^{oracle},\hat{\boldsymbol{\phi}}^{oracle})\|_2\|\boldsymbol{\beta}_k\|_2$$

and

$$\langle\nabla_k S_\tau(\hat{\boldsymbol{\beta}}^{oracle},\hat{\boldsymbol{\phi}}^{oracle}),\hat{\boldsymbol{\phi}}_k^{oracle}\rangle \geq -\|\nabla_k S_\tau(\hat{\boldsymbol{\beta}}^{oracle},\hat{\boldsymbol{\phi}}^{oracle})\|_2\|\boldsymbol{\phi}_k\|_2.$$

Inequality (b) is true under the conditions $\mathcal{E}_2 = \{\|\nabla_k S_\tau(\hat{\boldsymbol{\beta}}^{oracle},\hat{\boldsymbol{\phi}}^{oracle})\|_2 < a_1\lambda_1, \forall k\in\mathcal{A}_1^c;\ \|\nabla_k S_\tau(\hat{\boldsymbol{\beta}}^{oracle},\hat{\boldsymbol{\phi}}^{oracle})\|_2 <$ $a_1\lambda_2, \forall k\in\mathcal{A}_2^c\}$.

Combining the last inequality with the uniqueness of the solution of problem (3.22), we conclude that $(\hat{\boldsymbol{\beta}}^{oracle},\hat{\boldsymbol{\phi}}^{oracle})$ is the unique solution to (3.25).

Hence $(\hat{\boldsymbol{\beta}}^{(1)},\hat{\boldsymbol{\phi}}^{(1)}) = (\hat{\boldsymbol{\beta}}^{oracle},\hat{\boldsymbol{\phi}}^{oracle})$. We start the second iteration of GLLA algorithm with the initial value $(\hat{\boldsymbol{\beta}}^{oracle},\hat{\boldsymbol{\phi}}^{oracle})$ solution of the problem (3.25) at the first iteration. Let $(\hat{\boldsymbol{\beta}},\hat{\boldsymbol{\phi}})$ be the solution to the convex optimization problem in the second iteration of the GLLA algorithm. Under the condition

$$\mathcal{E}_2 = \{\min_{k\in\mathcal{A}_1}\|\hat{\boldsymbol{\beta}}_k^{oracle}\|_2 > a\lambda_1, \min_{k\in\mathcal{A}_2}\|\hat{\boldsymbol{\phi}}_k^{oracle}\|_2 > a\lambda_2\},$$

we obtain

$$P'_{\lambda_1}(\|\hat{\boldsymbol{\beta}}_k^{oracle}\|_2) = 0,\ \forall k\in\mathcal{A}_1 \text{ and } P'_{\lambda_2}(\|\hat{\boldsymbol{\phi}}_k^{oracle}\|_2) = 0,\ \forall k\in\mathcal{A}_2.$$

We have also

$$(\hat{\boldsymbol{\beta}}_k^{oracle}, \hat{\boldsymbol{\phi}}_{k'}^{oracle}) = \mathbf{0},\ \forall (k,k') \in \mathcal{A}_1^c \times \mathcal{A}_2^c.$$

Then, by property $(P5)$, we have

$$P'_{\lambda_1}(\hat{\boldsymbol{\beta}}_k^{oracle}) = P'_{\lambda_1}(0) \geq a_1\lambda_1 \text{ and } P'_{\lambda_2}(\hat{\boldsymbol{\phi}}_k^{oracle}) = P'_{\lambda_2}(0) \geq a_1\lambda_2.$$

Hence, optimization problem in the second iteration becomes

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta},\boldsymbol{\phi}}{\arg\min}\left(S_\tau(\boldsymbol{\beta},\boldsymbol{\phi}) + \sum_{k\in\mathcal{A}_1^c} P'_{\lambda_1}(\hat{\boldsymbol{\beta}}_k^{oracle})\|\boldsymbol{\beta}_k\|_2 + \sum_{k\in\mathcal{A}_2^c} P'_{\lambda_2}(\hat{\boldsymbol{\phi}}_k^{oracle})\|\boldsymbol{\phi}_k\|_2\right). \tag{3.53}$$

The problem (3.53) is very similar to (3.51), thus, we deduce that $(\hat{\boldsymbol{\beta}}^{oracle}, \hat{\boldsymbol{\phi}}^{oracle})$ is the unique solution to (3.53) under the event

$$\mathcal{E}_3 = \{\|\nabla_{\mathcal{A}_1} S_\tau(\hat{\boldsymbol{\beta}}^{oracle}, \hat{\boldsymbol{\phi}}^{oracle})\|_{2,\infty} < a_1\lambda_1;\ \|\nabla_{\mathcal{A}_2} S_\tau(\hat{\boldsymbol{\beta}}^{oracle}, \hat{\boldsymbol{\phi}}^{oracle})\|_{2,\infty} < a_1\lambda_2\}.$$

Then, under the assumption of Theorem 3.7, the probability that Algorithm 7 initialized by $(\hat{\boldsymbol{\beta}}^{\mathrm{GLasso}}, \hat{\boldsymbol{\phi}}^{\mathrm{GLasso}})$ given by Theorem 3.6 converges to $(\hat{\boldsymbol{\beta}}^{oracle}, \hat{\boldsymbol{\phi}}^{oracle})$ after two iterations is at least $1 - P(\mathcal{E}_1^c) - P(\mathcal{E}_2^c) - P(\mathcal{E}_3^c)$.

By the assumption of Theorem 3.7, we immediately get

$$\begin{aligned}
P(\mathcal{E}_1^c) &\leq P(\|\hat{\boldsymbol{\delta}}\|_{2,\infty} > a_0\lambda)\\
&\leq P\left(\|\hat{\boldsymbol{\delta}}\|_{2,\infty} > (3/2)\overline{\lambda}^{\mathrm{GLasso}}\left((c_0\overline{\kappa})^{-1} \wedge (c_0\overline{\varrho})^{-1}\right)\right)\\
&\leq P\left(\|\hat{\boldsymbol{\delta}}\|_{2,1} > (3/2)\overline{\lambda}^{\mathrm{GLasso}}(c_0\overline{\kappa})^{-1}\right) \vee P\left(\|\hat{\boldsymbol{\delta}}\|_{2,\infty} > (3/2)\overline{\lambda}^{\mathrm{GLasso}}(c_0\overline{\varrho})^{-1}\right)\\
&\leq \pi_1.
\end{aligned} \tag{3.54}$$

To establish the bound for $P(\mathcal{E}_2^c)$, we have

$$
\begin{aligned}
P(\mathcal{E}_2^c) \le\ & P(\{\|\nabla_{\mathcal{A}_1^c} S_\tau(\hat{\boldsymbol{\beta}}^{oracle}, \hat{\boldsymbol{\phi}}^{oracle})\|_{2,\infty} \ge a_1\lambda_1\} \cup \\
& \qquad \{\|\nabla_{\mathcal{A}_2^c} S_\tau(\hat{\boldsymbol{\beta}}^{oracle}, \hat{\boldsymbol{\phi}}^{oracle})\|_{2,\infty} \ge a_1\lambda_2\}) \\
\le\ & P(\{\|\nabla_{(\mathcal{A}_1\cup\mathcal{A}_2)^c} S_\tau(\hat{\boldsymbol{\beta}}^{oracle}, \hat{\boldsymbol{\phi}}^{oracle})\|_{2,\infty} \ge a_1\lambda\}) \\
\le\ & P(\{\|\nabla_{(\mathcal{A}_1\cup\mathcal{A}_2)^c} S_\tau(\boldsymbol{\beta}^*, \boldsymbol{\phi}^*)\|_{2,\infty} \ge a_1\lambda/2\}) \\
& + P(\{\|\nabla_{(\mathcal{A}_1\cup\mathcal{A}_2)^c} S_\tau(\hat{\boldsymbol{\beta}}^{oracle}, \hat{\boldsymbol{\phi}}^{oracle}) - \nabla_{(\mathcal{A}_1\cup\mathcal{A}_2)^c} S_\tau(\boldsymbol{\beta}^*, \boldsymbol{\phi}^*)\|_{2,\infty} \ge a_1\lambda/2\}).
\end{aligned}
\tag{3.55}
$$

Using (3.38), we deduce that

$$
\begin{aligned}
& P(\{\|\nabla_{(\mathcal{A}_1\cup\mathcal{A}_2)^c} S_\tau(\boldsymbol{\beta}^*, \boldsymbol{\phi}^*)\|_{2,\infty} \ge a_1\lambda/2\}) \\
& \quad \le P(\{\|n^{-1}\mathbf{X}_{\mathcal{A}_1^c}^\top W(\boldsymbol{\epsilon}+\boldsymbol{\eta})\|_{2,\infty} \ge a_1\lambda/2\}) + P(\{\|n^{-1}\mathbf{X}_{\mathcal{A}_2^c}^\top W\boldsymbol{\eta}\|_{2,\infty} \ge a_1\lambda/2\}) \\
& \quad \le P(\{\|n^{-1}\mathbf{X}_{\mathcal{A}_1^c}^\top W(\boldsymbol{\epsilon}+\boldsymbol{\eta})\|_{\infty} \ge a_1\lambda/(2\overline{p}_{\mathcal{A}_1^c})\}) + P(\{\|n^{-1}\mathbf{X}_{\mathcal{A}_2^c}^\top W\boldsymbol{\eta}\|_{\infty} \ge a_1\lambda/(2\overline{p}_{\mathcal{A}_2^c})\}) \\
& \quad \le 2(p - s_{\mathcal{A}_1}) \,\text{exp}\bigg(- \frac{Cn\lambda^2 a_1^2}{4M_0^2M_1^2(K_1+K_2)^2\overline{p}_{\mathcal{A}_1^c}^2}\bigg) + 2(p - s_{\mathcal{A}_2}) \,\text{exp}\bigg(- \frac{Cn\lambda^2 a_1^2}{4M_0^2M_1^2K_2^2\overline{p}_{\mathcal{A}_2^c}^2}\bigg),
\end{aligned}
\tag{3.56}
$$

where $s_{\mathcal{A}_1} = \sum_{k\in\mathcal{A}_1} p_k$ and $s_{\mathcal{A}_2} = \sum_{k\in\mathcal{A}_2} p_k$.

Let **d** $= (d_i, i = 1\ldots,n)^\top$ with $d_i = \rho'_\tau(y_i - \mathbf{x}_i^\top\hat{\boldsymbol{\beta}}^{oracle} - \mathbf{x}_i^\top\hat{\boldsymbol{\phi}}^{oracle}) - \rho'_\tau(y_i - \mathbf{x}_i^\top\boldsymbol{\beta}^* - \mathbf{x}_i^\top\boldsymbol{\phi}^*)$. It follows that

$$
\begin{aligned}
& P\bigg(\{\|\nabla_{(\mathcal{A}_1\cup\mathcal{A}_2)^c}\mathit{\Psi}_\tau(\boldsymbol{\beta}^*) - \nabla_{(\mathcal{A}_1\cup\mathcal{A}_2)^c}\mathit{\Psi}_\tau(\hat{\boldsymbol{\beta}}^{oracle})\|_{2,\infty} \ge a_1\lambda/2\}\bigg) \\
& \qquad \le P\bigg(\{\max_{k\in(\mathcal{A}_1\cup\mathcal{A}_2)^c} p_k\|\nabla_k\mathit{\Psi}_\tau(\boldsymbol{\beta}^*) - \nabla_k\mathit{\Psi}_\tau(\hat{\boldsymbol{\beta}}^{oracle})\|_\infty \ge a_1\lambda/2\}\bigg) \\
& \qquad \le P\bigg(\|\nabla_{(\mathcal{A}_1\cup\mathcal{A}_2)^c}\mathit{\Psi}_\tau(\boldsymbol{\beta}^*) - \nabla_{(\mathcal{A}_1\cup\mathcal{A}_2)^c}\mathit{\Psi}_\tau(\hat{\boldsymbol{\beta}}^{oracle})\|_\infty \ge \frac{a_1\lambda}{2\overline{p}_{(\mathcal{A}_1\cup\mathcal{A}_2)^c}}\bigg).
\end{aligned}
\tag{3.57}
$$

We have then

$$
\begin{aligned}
&\|\nabla_{(\mathcal{A}_1\cup\mathcal{A}_2)^c}\mathit{\Psi}_\tau(\boldsymbol{\beta}^*) - \nabla_{(\mathcal{A}_1\cup\mathcal{A}_2)^c}\mathit{\Psi}_\tau(\hat{\boldsymbol{\beta}}^{oracle})\|_\infty \\
&\qquad \le M_0(\|\mathbf{X}(\hat{\boldsymbol{\beta}}^{oracle} - \boldsymbol{\beta}^*)\|_2 + \|\mathbf{b}\|_2)/\sqrt{n} \\
&\qquad \le M_0\Bigg[(1+2\overline{c})\|\mathbf{X}_{\mathcal{A}_1}(\hat{\boldsymbol{\beta}}^{oracle}_{\mathcal{A}_1} - \boldsymbol{\beta}^*_{\mathcal{A}_1})\|_2 + (2\overline{c})\|\mathbf{X}_{\mathcal{A}_2}(\hat{\boldsymbol{\phi}}^{oracle}_{\mathcal{A}_2} - \boldsymbol{\phi}^*_{\mathcal{A}_2})\|_2\Bigg]/\sqrt{n} \\
&\qquad \le (1+2\overline{c})M_0\phi_{\max}^{1/2}\|\hat{\boldsymbol{\theta}}^{oracle} - \boldsymbol{\theta}^*\|_2.
\end{aligned}
\tag{3.58}
$$

By Lemma 3 and Lemma 6 of [33], we get

$$
\begin{aligned}
&P\Bigg(\{\|\nabla_{(\mathcal{A}_1\cup\mathcal{A}_2)^c}\mathit{\Psi}_\tau(\boldsymbol{\beta}^*) - \nabla_{(\mathcal{A}_1\cup\mathcal{A}_2)^c}\mathit{\Psi}_\tau(\hat{\boldsymbol{\beta}}^{oracle})\|_{2,\infty} \ge a_1\lambda/2\}\Bigg) \\
&\qquad \le P\Bigg(\|\hat{\boldsymbol{\theta}}^{oracle} - \boldsymbol{\theta}^*\|_2 \ge \frac{a_1\lambda}{(1+2\overline{c})M_0\phi_{\max}^{1/2}}\Bigg) \\
&\qquad \le P\Bigg(\Bigg\|\frac{1}{n}\begin{pmatrix}\mathbf{X}_{\mathcal{A}_1}^\top W(\boldsymbol{\epsilon}+\boldsymbol{\eta}) \\ \mathbf{X}_{\mathcal{A}_2}^\top W\boldsymbol{\eta}\end{pmatrix}\Bigg\|_2 \ge Q_2\lambda\Bigg) \\
&\qquad \le P\Bigg(\Bigg\|\frac{1}{n}\mathbf{X}_{\mathcal{A}_1}^\top W(\boldsymbol{\epsilon}+\boldsymbol{\eta})\Bigg\|_2 \ge Q_2\lambda/2\Bigg) + P\Bigg(\Bigg\|\frac{1}{n}\mathbf{X}_{\mathcal{A}_2}^\top W\boldsymbol{\eta}\Bigg\|_2 \ge Q_2\lambda/2\Bigg) \\
&\qquad \le \Gamma(Q_2\lambda/2, n, s_{\mathcal{A}_1}, K_1+K_2, M_0, M_1, M_1^2\rho_{1,max}, \nu_1) + \\
&\qquad\qquad\qquad \Gamma(Q_2\lambda/2, n, s_{\mathcal{A}_2}, K_2, M_0M_1, M_1^2\rho_{2,max}, \nu_2).
\end{aligned}
\tag{3.59}
$$

Combining (3.57), (3.58), and (3.59), it follows from Lemma 3 and Lemma 4 of [33] that

$$
\begin{aligned}
\pi_2 = {} & 2(p - s_{\mathcal{A}_1})\,\textsf{exp}\Bigg(-\frac{Cn\lambda^2a_1^2}{4M_0^2M_1^2(K_1+K_2)^2\overline{p}^2_{\mathcal{A}_1^c}}\Bigg) + 2(p - s_{\mathcal{A}_2})\,\textsf{exp}\Bigg(-\frac{Cn\lambda^2a_1^2}{4M_0^2M_1^2K_2^2\overline{p}^2_{\mathcal{A}_2^c}}\Bigg) \\
& + \Gamma(Q_2\lambda/2, n, s_{\mathcal{A}_1}, K_1+K_2, M_0, M_1, M_1^2\rho_{1,max}, \nu_1) \\
& + \Gamma(Q_2\lambda/2, n, s_{\mathcal{A}_2}, K_2, M_0M_1, M_1^2\rho_{2,max}, \nu_2).
\end{aligned}
\tag{3.60}
$$

To derive the upper bound for $P(\mathcal{E}_3^c)$, we use assumption (A2) to get

$$
\min_{k\in\mathcal{A}_1}\|\hat{\boldsymbol{\beta}}^{oracle}\| \ge \min_{k\in\mathcal{A}_1}\|\boldsymbol{\beta}^*\| - \|\hat{\boldsymbol{\beta}}^{oracle} - \boldsymbol{\beta}^*\|_{2,\infty}
$$

and

$$
\min_{k\in\mathcal{A}_2}\|\hat{\boldsymbol{\phi}}^{oracle}\| \ge \min_{k\in\mathcal{A}_2}\|\boldsymbol{\phi}^*\| - \|\hat{\boldsymbol{\phi}}^{oracle} - \boldsymbol{\phi}^*\|_{2,\infty}.
$$

It follows that

$$
\begin{aligned}
P(\mathcal{E}_3^c) &\leq P(\|\hat{\boldsymbol{\theta}}^{oracle} - \boldsymbol{\theta}^*\|_{2,\infty} > \overline{R}) \leq P(\|\hat{\boldsymbol{\theta}}^{oracle} - \boldsymbol{\theta}^*\|_{\infty} > \frac{\overline{R}}{\overline{p}_k}) \\
&\leq P\left(\left\| \frac{1}{n} \begin{pmatrix} \mathbf{X}_{\mathcal{A}_1}^\top W(\boldsymbol{\epsilon} + \boldsymbol{\eta}) \\ \mathbf{X}_{\mathcal{A}_2}^\top W\boldsymbol{\eta} \end{pmatrix} \right\|_2 \geq c_0 \phi_{\min} \frac{\overline{R}}{\overline{p}_k} \right) \\
&\leq P\left(\left\| \frac{1}{n} \mathbf{X}_{\mathcal{A}_1}^\top W(\boldsymbol{\epsilon} + \boldsymbol{\eta}) \right\|_2 \geq c_0 \phi_{\min} \frac{\overline{R}}{2\overline{p}_k} \right) + P\left(\left\| \frac{1}{n} \mathbf{X}_{\mathcal{A}_2}^\top W\boldsymbol{\eta} \right\|_2 \geq c_0 \phi_{\min} \frac{\overline{R}}{2\overline{p}_k} \right) \\
&\leq \Gamma(c_0 \phi_{\min} \frac{\overline{R}}{2\overline{p}_k}, n, s_{\mathcal{A}_1}, K_1 + K_2, M_0, M_1, M_1^2 \rho_{1,max}, \nu_1) + \\
&\qquad\qquad \Gamma(c_0 \phi_{\min} \frac{\overline{R}}{2\overline{p}_k}, n, s_{\mathcal{A}_2}, K_2, M_0 M_1, M_1^2 \rho_{2,max}, \nu_2).
\end{aligned}
\tag{3.61}
$$

This ends the proof of Theorem 3.7. □

3.8.5 Appendix **E** : Checking KKT condition

We have been proved that the GPER and COGPER algorithms hold in the descent property. In the following section, we show that those algorithms converge to a stationary point by checking KKT conditions. Theorically, the solutions in (3.6) and (3.20) are established based on KKT conditions, then, they must always verify exactly KKT conditions. But, the numerical solution may fail the KKT conditions. more details are given [105, 77]. We define numerical KKT conditions for the GPER approach with the penalties GLasso, GMCP, GSCAD and GLLA, respectively, as follow

$$
\begin{cases}
\|\nabla_k \varPsi_\tau(\boldsymbol{\beta}) + \lambda\omega_k \cdot \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2}\|_2 \leqslant \epsilon, & if \;\; \boldsymbol{\beta}_k \neq 0 \\
\|\nabla_k \varPsi_\tau(\boldsymbol{\beta})\|_2 \leqslant \lambda\omega_k + \epsilon, & if \;\; \boldsymbol{\beta}_k = 0,
\end{cases}
$$

$$
\begin{cases}
\|\nabla_k \varPsi_\tau(\boldsymbol{\beta}) + \lambda\omega_k \cdot \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} - \frac{\boldsymbol{\beta}_k}{\theta}\|_2 \leqslant \epsilon, & if \;\; \boldsymbol{\beta}_k \neq 0 \text{ and } \|\boldsymbol{\beta}_k\|_2 \leqslant \theta\lambda \\
\|\nabla_k \varPsi_\tau(\boldsymbol{\beta})\|_2 \leqslant \lambda\omega_k + \epsilon, & if \;\; \boldsymbol{\beta}_k = 0 \text{ and } \|\boldsymbol{\beta}_k\|_2 \leqslant \theta\lambda \\
\|\nabla_k \varPsi_\tau(\boldsymbol{\beta})\|_2 \leqslant \epsilon = 0, & if \;\; \|\boldsymbol{\beta}_k\|_2 > \theta\lambda,
\end{cases}
$$

$$\begin{cases} \|\nabla_k \Psi_\tau(\boldsymbol{\beta}) + \lambda\omega_k \cdot \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2}\|_2 \leqslant \epsilon, & if \;\; \boldsymbol{\beta}_k \neq 0 \text{ and } \|\boldsymbol{\beta}_k\|_2 \leqslant \lambda \\ \|\nabla_k \Psi_\tau(\boldsymbol{\beta})\|_2 \leqslant \lambda\omega_k + \epsilon, & if \;\; \boldsymbol{\beta}_k = 0 \text{ and } \|\boldsymbol{\beta}_k\|_2 \leqslant \lambda \\ \|\nabla_k \Psi_\tau(\boldsymbol{\beta}) + \frac{\theta}{\theta-1}\lambda\omega_k \cdot \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2} - \frac{\boldsymbol{\beta}_k}{(\theta-1)}\|_2 \leqslant \epsilon, & if \;\; \lambda < \|\boldsymbol{\beta}_k\|_2 \leqslant \theta\lambda \\ \|\nabla_k \Psi_\tau(\boldsymbol{\beta})\|_2 \leqslant \epsilon, & if \;\; \|\boldsymbol{\beta}_k\|_2 > \theta\lambda, \end{cases}$$

$$\begin{cases} \|\nabla_k \Psi_\tau(\boldsymbol{\beta}) + \lambda\omega'_k \cdot \frac{\boldsymbol{\beta}_k}{\|\boldsymbol{\beta}_k\|_2}\|_2 \leqslant \epsilon, & if \;\; \boldsymbol{\beta}_k \neq 0 \\ \|\nabla_k \Psi_\tau(\boldsymbol{\beta})\|_2 \leqslant \lambda\omega'_k + \epsilon, & if \;\; \boldsymbol{\beta}_k = 0. \end{cases}$$

To obtain the KKT conditions for COGPER approach, we replace $\Psi_\tau(.)$, $\boldsymbol{\beta}_k$ and (w_k,w'_k) in the above KKT conditions by $S_\tau(.)$, $\boldsymbol{\beta}_k$ or $\boldsymbol{\phi}_k$ and (u_k,u'_k), respectively.

CHAPITRE 4

LES PAQUETS R `GPQR` ET `GPER` POUR LA RÉGRESSION QUANTILE ET EXPECTILE EN GRANDE DIMENSION

4.1 Introduction

Ce chapitre décrit deux paquets R, `GPQR` et `GPER`, qui implémentent une nouvelle famille de méthodes de régression asymétriques pénalisées en grande dimension (régression quantile, régression expectile et régression expectile couplée). Dans cette thèse, nous avons traité la situation ou la structure groupes est variables est connue à priori. Nous avons donc implémenté nos méthodes proposées pour les trois pénalités décrites aux chapitres 2 et 3 : Group Lasso, Group Scad et group MCP. Dans le cadre de GPQR, nous avons combiné l'idée de l'approximation de la fonction de perte de quantile, qui n'est pas dérivable en zéro, par une fonction de perte modifiée qui est dérivable partout. De plus, nous avons implémenté un algorithme qui combinent le principe MM et l'algorithme de descente par coordonnées pour mettre à jour chaque groupe de coefficients d'une façon simple et et efficace. Pour GPER et COGPER, nous avons utilisé le principe MM et l'algorithme de descente par coordonnées directement sans approximation de la fonction de perte expectile, car cette dernière est dérivable partout.

On note que les deux paquets R disposent de documentations complètes sous forme de vignettes qui fournissent plus de détails sur la façon dont chaque méthode peut être ajustée avec quelques exemples d'exécution. Elles sont disponibles publiquement via Github (`https://github.com/ouhourane/These_vignette`). Dans ce chapitre, nous présentons un aperçu bref de ses deux paquets R.

GPQR et GPER résous le problème suivant

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \left(\frac{1}{n} \sum_{i=1}^{n} \rho_\tau(y_i - \mathbf{x}_i^\top \boldsymbol{\beta}) + \sum_{k=1}^{K} P_\lambda(\|\boldsymbol{\beta}_k\|_2) \right)$$

sur une grille de valeurs λ qui couvre le chemin de solution, tel que $\rho_\tau(u) = |\tau - \mathbb{1}_{(u<0)}||u|$ pour la régression quantile (GPQR) et $\rho_\tau(u) = |\tau - \mathbb{1}_{(u<0)}|u^2$ pour la régression expectile (GPER).

Le paquet R `GPER` permet aussi de résoudre le problème de la régression expectile couplée. Celle-ci est décrit en section 3.3 de cette thèse, et qui est défini par

$$(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\phi}}) = \arg\min_{\boldsymbol{\beta}, \boldsymbol{\phi}} \frac{\nu}{2n} \sum_{i=1}^{n} \rho_{0.5}(y_i - \mathbf{x}_i^\top \boldsymbol{\beta}) + \frac{1}{n} \sum_{i=1}^{n} \rho_\tau(y_i - \mathbf{x}_i^\top \boldsymbol{\beta} - \mathbf{x}_i^\top \boldsymbol{\phi}) + P_\lambda(\boldsymbol{\beta}) + P_\lambda(\boldsymbol{\phi}), \tag{4.1}$$

où $\rho_\tau(u) := |\tau - \mathbb{1}_{(u<0)}|u^2$. Dans ces deux paquets, on considère trois type de pénalités : group Lasso (GLasso), group MCP (GMCP) et group SCAD (GSCAD), qui sont définies dans (2.2), (2.3) et (2.4) de cette thèse.

R est un langage de programmation interprété, c'est-à-dire que le code R n'est pas directement exécuté par la machine. En conséquence, R est connu pour être plus lent qu'un autre langage compilé tel que Fortran, C, etc. En particulier, pour un code itératif, les boucles sont plus lentes en R qu'en langages compilés. Une façon de bénéficier de la rapidité de Fortran avec les avantages de R consiste à coder les boucles internes en Fortran et à les appeler depuis R. Pour accélérer nos algorithmes dans les deux paquets, les fonctions codifiant les algoithmes de GPER et GPQR sont écrit en Fortran, ce qui permet une économie considérable du temps d'exécution. Plusieurs fonctions auxiliaires dans les deux paquets (cv, predict, print, coef, etc.) sont prises des paquets `gglasso` [105] et `sales` [33].

Les approches GPQR et GPER utilisent l'algorithme de descente cyclique par bloc de coordonnées, qui mettent à jour de manière itérative un bloc de variables en fixant les autres blocs. Nos paquets calculent rapidement le chemin de solutions, car ils utilisent certaines techniques telles que la règle forte pour écarter les blocs de variables non significatives [91] et la technique de démarrage rapide, plus de détails sont donnés dans la section $2.3.4$.

4.2 Installation

Les deux paquets `GPQR` et `GPER` ne sont pas encore publiés sur le "Comprehensive R Archive Network (CRAN)", mais ils sont disponibles via GitHub comme de nombreux autres paquets R. Pour installer GPQR et GPER à partir de GitHub, nous exécutons les lignes de code suivantes dans la console R :

```
library(devtools)
devtools::install_github("https://github.com/
                  ouhourane/GPQR.git"
devtools::install_github("https://github.com/
                  ouhourane/GPER.git"
```

Dans cette vignette, nous démontrons comment utiliser GPQR(.) et GPER(.), les principales fonctions des paquets `GPQR` et `GPER` pour produire le chemin de solutions de la régression quantile/expectile avec pénalités de sélection par group de varoables. D'autres fonctions telles que predict(), coef(), cv.predict, cv.coef, etc., sont dérivées des paquets `gglasso` et `sales` avec quelques modifications.

4.3 Exemple 1 de notre article [77]

Dans cet exemple, notre objectif est d'illustré aux utilisateurs finaux comment exécuté nos méthodes dans R.

Pour l'illustration, nous utilisons les données du scénario 1 du chapître 2. Plus précisément, nous fixons la taille de l'échantillon à $n = 100$ observations et $p = 20$ variables. Les variables $X_j, j = 1 \ldots 20$, ont été générés comme suit :

- On génère $Z_j, j = 1, \ldots, 11$, selon la distribution gaussienne standard ;
- On prends $X_j = Z_1 + \epsilon_j, \epsilon_j \sim N(0, 0.1),\ j = 1, \ldots, 4$;
- $X_j = Z_2 + \epsilon_j, \epsilon_j \sim N(0, 0.1),\ j = 5, \ldots, 8$;
- $X_j = Z_3 + \epsilon_j, \epsilon_j \sim N(0, 0.1),\ j = 9, \ldots, 12$;
- $X_j = Z_{j-9}, j = 13, \ldots, 20$.

Le code suivant décrit comment générer la matrice des variables $\mathbf{X}$ (de dimension $n \times p$) à partir de la matrice de distribution normale multivariée $\mathbf{Z}$ (de dimension $n \times K$) avec le vecteur moyen des variables est $\mu_{\mathbf{Z}} = \mathbf{0}_{11}$ et la matrice de covariance des variables est $\mathbf{\Sigma}$, avec $\Sigma_{jk} = 0.5^{|j-k|}$.

```
library("MASS")
n = 100
K = 11
p = 20
MuVec<-rep(0,K)
v<-rep(1,K); SigmaMat<-diag(v)
```

```
for(j in 1:K)
    for(k in 1:K) SigmaMat[j,k] <- 0.5^abs(j-k)
Z <- mvrnorm(n,MuVec,SigmaMat,tol = 1e-6, empirical = FALSE)
X = NULL
for (h in 1:3) for (k in 1:4) X=cbind(X,Z[,h]+rnorm(n,0,0.1))
X = cbind(X,Z[,4:K])
```

Ainsi, on fixe les effets des prédicteurs à

$$\boldsymbol{\beta} = (\underbrace{3,3,3,3}_{G_1}, \underbrace{2,2,2,2}_{G_2}, \underbrace{-1,-1,-1,-1}_{G_3}, \underbrace{0,\ldots,0}_{G_4 - G_{11}})^\top$$

Au total, nous avons 11 groupes : G_1,G_2,...,G_{11}.

```
# Un vecteur d'entiers consécutifs décrivant # le regroupement des
prédicteurs
group=c(1,1,1,1,2,2,2,2,3,3,3,3,4:K)
```

La variable réponse Y est générée à partir du modèle de régression linéaire d'échelle comme suit

$$Y = \sum_{j=1}^{20} \beta_j X_j + \Phi(X_{20})\epsilon, \quad \epsilon \sim N(0,3),$$

où $\Phi(.)$ est la fonction de distribution cumulative d'une loi normale standard.

```
beta = c(rep(3,4),rep(2,4),rep(-1,4),rep(0,p-12))
Y<-X%*%beta+pnorm(X[,p])*rnorm(n,0,3)
```

4.4 Introduction au paquet `GPQR`

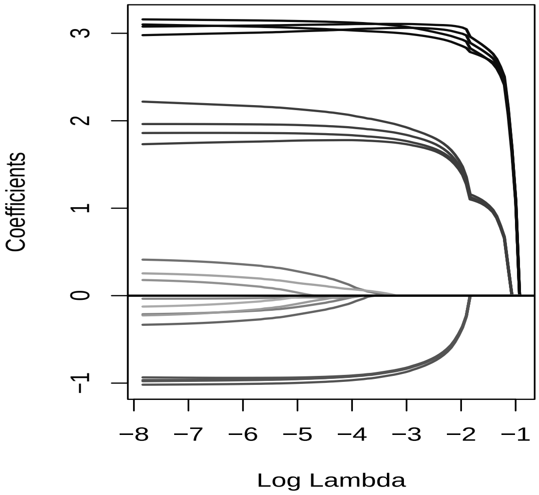
On ajuste le modèle de quantile en utilisant la fonction de base GPQR() avec de nombreux arguments d'entrée optionnels. Dans la ligne de code suivante, on exécute GPQR() pour la pénalité group Lasso (method = "GLasso"), le paramètre $\tau = 0,5$ (taux = 0,5), et la fonction de perte pseudo quantile $\Psi^{(1)}_{\tau,\delta}(.)$ (check = "f1") donnée par (2.5).

```
library("GPQR")
fit <- GPQR(x=X,y=Y,group=group, method="GLasso", check="f1",

taux=0.5)
```

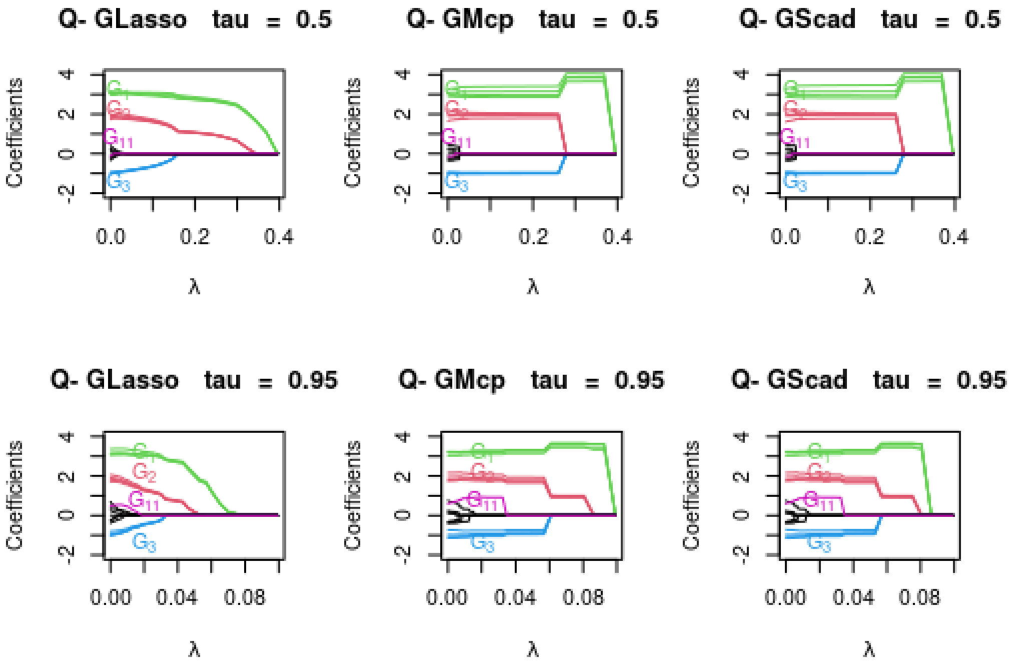
On utilise la fonction `plot` pour tracer le chemin de solutions extraite de objet GPQR ajusté par la fonction GPQR(.).

```
plot(fit)
```



Dans la figure ci-dessus, on reproduit une partie de la figure 2.2 présentée au chapitre 2, qui illustre comment l'approche GPQR peut être utile pour détecter les groupes de variables hétéroscédastiques. On utilise une fonction personnalisée `GPQR_illustration` pour tracer les chemins de chaque groupe avec une couleur différente pour une meilleur visualisation. Cela conduit à la figure ci-dessous. Le code de cette fonction est donné en annexe dans la section 4.7 .

```
GPQR_illustration("GLasso", 0.5)
GPQR_illustration("GMcp", 0.5)
GPQR_illustration("GScad", 0.5)
GPQR_illustration("GLasso", 0.95)
GPQR_illustration("GMcp", 0.95)
GPQR_illustration("GScad", 0.95)
```



4.5 Introduction au paquet `GPER` : l'approche GPER

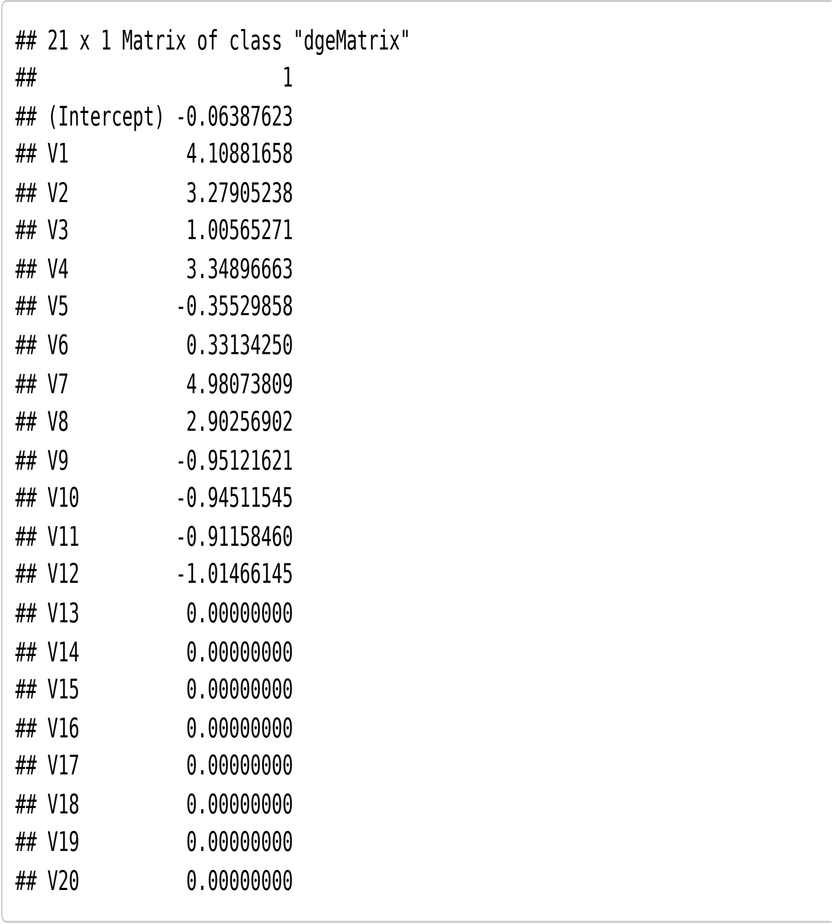
On ajuste le modèle GPER avec la pénalité GMcp en utilisant la fonction GPER() avec $\tau = 0.85$. Cela est donné par

```
library("GPER")
fit <- gper(x=X,y=Y,group=group, method="GMcp",
                        tau = 0.85)
```

L'objet `fit` est une liste contenant touts les informations pertinentes du modèle ajusté. Les utilisateurs peuvent explorer cet objet en regardant directement ses éléments, qui sont résumés sous forme de liste. Diverses fonctions sont fournies pour extraire des informations de l'objet GPER, tel que les fonctions plot, print, coef et predict, qui nous permettent d'exécuter facilement plusieurs tâches.

On peux obtenir les coefficients estimés pour une valeur spécifique de λ (ou de plusieurs valeurs) dans l'intervalle ($\lambda_{\min}$,$\lambda_{\max}$) :

```
coef(fit, s = 1)
```



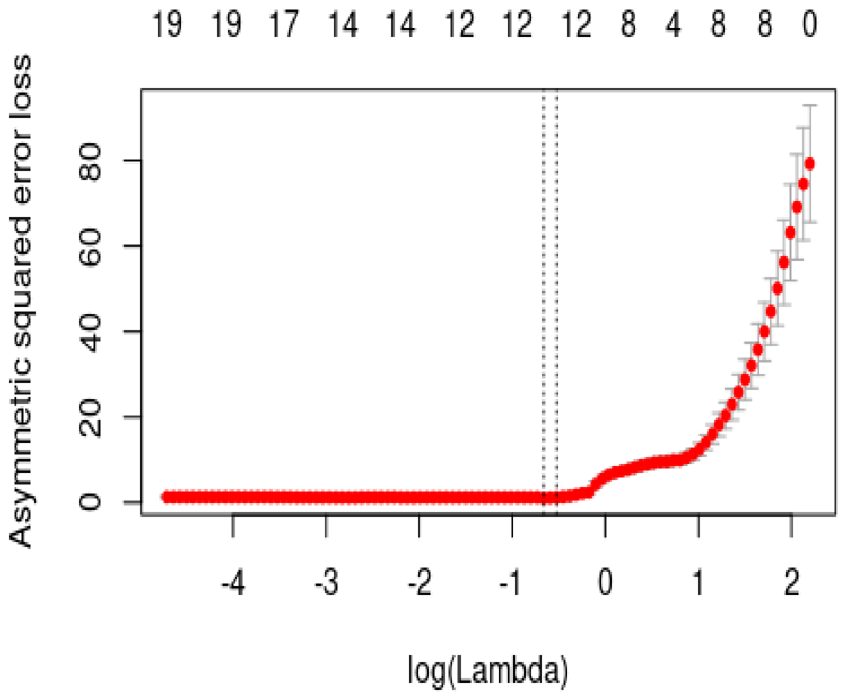
```
## 21 x 1 Matrix of class "dgeMatrix"
##                       1
## (Intercept) -0.06387623
## V1           4.10881658
## V2           3.27905238
## V3           1.00565271
## V4           3.34896663
## V5          -0.35529858
## V6           0.33134250
## V7           4.98073809
## V8           2.90256902
## V9          -0.95121621
## V10         -0.94511545
## V11         -0.91158460
## V12         -1.01466145
## V13          0.00000000
## V14          0.00000000
## V15          0.00000000
## V16          0.00000000
## V17          0.00000000
## V18          0.00000000
## V19          0.00000000
## V20          0.00000000
```

La fonction `cv.gper` peut être utilisée pour calculer la validation croisée k-fold pour le modèle GPER. Cette fonction renvoie une liste de sorties qui contient un objet de type cv.gper.

```
cvfit <- cv.gper(x=X, y=Y, group=group,
                    method="GScad", tau=0.5)
```

On peux visualiser l'erreur de la validation croisée en traçant l'objet cv.gper comme suit :

```
plot(cvfit)
```



La valeur optimale de λ peut être obtenue par les deux lignes pointillées verticales correspondantes à `lambda.min` et `lambda.1se`, où `lambda.min` est la valeur de λ correspondant au minimum d'erreur de la validation croisée, et `lambda.1 se` est la plus grande valeur de λ telle que l'erreur se situe dans une erreur standard des erreurs de la validation croisée pour `lambda.min`. Par exemple, les valeurs des coefficients $\hat{\beta}$, correspondant à `lambda.1 se`, est donné par

```
coef(cvfit, s = "lambda.1se")
```

```
## 21 x 1 Matrix of class "dgeMatrix"
##                      1
## (Intercept) -0.0570172
## V1           4.1587814
## V2           3.3942172
## V3           0.9708740
## V4           3.2325536
## V5          -0.3861691
## V6           0.3192177
## V7           5.0042467
## V8           2.8718166
## V9          -0.9276096
## V10         -0.9235594
## V11         -0.8954858
## V12         -0.9808504
## V13          0.0000000
## V14          0.0000000
## V15          0.0000000
## V16          0.0000000
## V17          0.0000000
## V18          0.0000000
## V19          0.0000000
## V20          0.0000000
```

`cv.cogper` est la fonction principale pour réaliser une validation croisée pour le modèle COGPER. Nous exécutons cette fonction avec $\tau = 0.9$ et la pénalité GLasso.

```
cvfit <- cv.cogper(x=X,y=Y,group=group,
                  method="GLasso",tau=0.9)
```

L'objet retourné est de la classe cogper qui contient tout les informations pertinentes du modèle ajusté qui sera utilisé par d'autres fonctions. Les fonctions plot, coef et predict peuvent être appliqués à l'objet ajusté pour obtenir facilement des résultats plus détaillés d'une manière similaire à celle illustrée précédemment.

On peux faire des prédictions en utilisant la fonction predict. Pour cela, les utilisateurs doivent fournir la matrice des variables et la ou les valeurs de λ auxquelles les prédictions doivent être faites.

```
predict(cvfit, newx = X[1:3,], s = "lambda.min")
```

```
## [1,] 13.404298
## [2,] -1.148949
## [3,] 12.756209
```

Par exemple, la valeur 13.404298 est la prédiction, $\mathbf{x}_1^\top \hat{\boldsymbol{\beta}}$, pour la première observation de $\mathbf{X}$.

La fonction suivante estime la solution du modèle COGPER par validation croisée en utilisant l'objet cv.cogper ajusté, et la valeur optimale choisie pour λ.

```
coef(cvfit, s = "lambda.min")
```

```
## $beta
## 21 x 1 Matrix of class "dgeMatrix"
##                        1
## (Intercept) -0.123161602
## V1           3.064646666
## V2           3.008475419
## V3           2.607082071
## V4           3.005998580
## V5           1.590238686
## V6           1.781564962
## V7           2.485536951
## V8           2.073337852
## V9          -1.176758860
## V10         -0.882879629
## V11         -0.832638781
## V12         -1.324573039
## V13          0.000000000
## V14         -0.051008187
## V15          0.000000000
## V16          0.000000000
## V17          0.000000000
## V18         -0.322078359
## V19          0.002044579
## V20          0.000000000
##
## $phi
## 21 x 1 sparse Matrix of class "dgCMatrix"
##                      1
## (Intercept) -1.2521914
## V1           .
## V2           .
## V3           .
## V4           .
## V5           .
## V6           .
## V7           .
## V8           .
## V9           .
## V10          .
## V11          .
## V12          .
## V13          .
## V14          .
## V15          .
## V16          .
## V17          .
## V18          .
## V19          .
## V20         -0.2205322
```

4.7 Annexe

La fonction suivante offre une version personnalisée de la fonction plot pour tracer l’objet GPQR ajusté.

```
GPQR_illustration <-function(penalty, taux){
   fit <- GPQR(x=X, y=Y, group=group,       method=penalty,
check="f1", taux=taux)
   main_lab = paste("Q-",penalty," ",       expression(tau), " = ",
taux)
   matplot(fit$lambda, t(fit$beta[1:4,]),       type = "l", col
= 3,lty = 1, ylim=c(-2,4),       lwd=1, ylab="Coefficients",
main=main_lab,       xlab = expression(lambda))
   matlines(fit$lambda,t(fit$beta[5:8,]),       type="l", col=2,
lty=1,lwd=1)
   matlines(fit$lambda,t(fit$beta[9:12,]),       type="l", col=4,
lty=1,lwd=1)
   matlines(fit$lambda,t(fit$beta[13:19,]),       type="l", col=2,
lty=1,lwd=1)
   matlines(fit$lambda,fit$beta[20,],       type="l", col=6,
lty=1,lwd=1)
   text(0.02,0.75, expression("G"[11]), col=6)
   text(0.02,3.2, expression("G"[1]), col=3)
   text(0.02,2.2, expression("G"[2]), col=2)
   text(0.02,-1.4, expression("G"[3]), col=4)
}
```

CONCLUSION

Cette thèse s'inscrit dans le cadre des travaux récents portant sur les modèles de régression asymétrique en grande dimension, elle traite en particulier les modèles de régression quantile, expectile et expectile couplée. Nous avons proposé de nouvelles méthodes de sélection et d'estimation des groupes de variables hétérogènes dans le cadre de la régression linéaire asymétrique. L'idée principale de nos travaux de recherche est de généraliser les pénalités qui garantissent la sélection des variables par groupe connus à priori aux modèles de régression asymétrique.

Dans le chapitre 2, nous avons proposé un algorithme de descente par blocs de coordonnées qui permet de trouver le chemin des solutions de la régression quantile régularisée avec les pénalités de groupe, à savoir la pénalité de group Lasso, les pénalités de groupe non convexes (group SCAD et group MCP) et leurs approximations locales. L'approche GPQR permet de sélectionner les groupes importants de variables (hétérogènes) et fournit une estimation de leurs effets sur la variable dépendante, simultanément.

Nous avons prouvé que le vitesse de convergence de l'approche GPQR avec la pénalité group Lasso est linéaire. De plus, la comparaison empirique a confirmé que notre approche GPQR est performante par rapport aux méthodes de régression quantile qui sélectionnent les variables individuellement (fréquentistes et bayésiennes) et les approches MCO pénalisées qui sélectionne les variables par groupe.

Bien que l'approche GPQR a démontré son utilité dans la sélection des gènes pertinents en analysant le jeu de données ADNI, les résultats ont également montré une certaine incohérence entre les trois pénalités. Comme nous l'avons indiqué dans la section 2.5, cela pourrait être le résultat de la sensibilité de la méthode à l'assignation des données utilisées dans la procédure de validation croisée [81]. En revanche, le choix d'un bon paramètre de réglage dépend du paramètre inconnu σ^2 qui est la variance du bruit [7]. Pour étudier les données réelles, la connaissance de l'écart-type nécessite une étude plus approfondie des données. Cependant, l'homogénéité de la variance des erreurs pourrait être une hypothèse forte à supposer pour les données réelles en grande dimension. Dans la littérature, les méthodes de régularisation pivotantes ont été introduites pour corriger ce problème [6] (c'est-à-dire pivotantes dans le sens où la méthode ne repose pas sur la connaissance de l'écart-type σ et n'a pas besoin d'estimer σ a priori). Combiner notre approche GPQR avec l'idée de méthodes pivotantes pourrait être une voie de recherche intéressante.

Comme le modèle de la régression quantile standard, la méthode GPQR peut être sensible au problème bien connu dans la littérature, à savoir le croisement des quantiles. Dans le cas des données de grande dimension, ce problème pourrait être plus persistant en raison de l'ajout de la pénalité. En effet, comme les courbes GPQR sont estimées individuellement, la monotonicité des courbes des quantiles peut être violée. De plus, les niveaux de rétrécissement de la pénalisation peuvent être différents selon les endroits, ce qui rend le croisement encore plus probable quand on estime les paramètres du modèle de la régression quantile en grande dimension. Pour contourner le problème de croisement des quantiles en petite dimension, plusieurs auteurs ont proposé une procédure d'estimation simultanée en imposant des contraintes sur les courbes de quantiles. Par exemple, [50] a supposé la condition d'égalité des pentes (c'est-à-dire que les plans des quantiles sont parallèles). [63] ont proposé une procédure d'estimation séquentielle des quantiles qui garantit le non-croisement, en imposant une contrainte sur la courbe de quantile que nous cherchons à estimer de ne pas croiser la courbe estimée avant. [9] ont ajouté des contraintes au problème d'optimisation de la régression quantile afin de garantir que les hyperplans des quantiles ne se croisent pas. La question de croisement des quantiles pour la régression quantile pénalisée en dimension élevée a été peu étudiée dans la littérature. Cette avenue pourrait être une voie de recherche intéressante dans le domaine de la régression quantile.

Dans le chapitre 3, nous avons introduit les pénalités de sélection des variables par bloc aux modèles de la régression des moindres carrés asymétriques, à savoir la régression expectile et la régression expectile couplée. Nous avons fourni ensuite les techniques de résolution des problèmes d'optimisation sous-jacents via l'algorithme CDA combiné avec le principe MM. D'un point de vue théorique, nous avons généralisé les résultats de [33] au cadre de la sélection des variables par groupe ; nous avons montré que nos modèles possèdent de propriété oracle. L'étude empirique sur des données simulées et réelles ont confirmé la bonne performance de la régression expectile par rapport à la régression symétrique en termes de la détection et la séparation de l'effet des groupes de variables hétérogènes sur la moyenne et la variance.

À partir de l'ensemble des travaux présentés dans cette thèse, nous pouvons dégager les conclusions suivantes :

— La performance des méthodes de la régression linéaire asymétrique pénalisée a été clairement démontré. En effet, les méthodes proposées produisent des modèles parcimonieux par bloc de variables d'une part. D'autre part, elles traitent le problème d'hétérogénéité en grande dimension en sélectionnant les groupes de variables hétérogènes importantes.

— Afin de maintenir l'homogénéité de cette thèse, nous nous sommes concentrés sur la généralisation des pénalités de sélection des variables par bloc sur la régression asymétrique en grande dimension. De point de vue algorithmique, l'implémentation de nos méthodes est basée sur la combinaison du principe MM et l'algorithme CDA. De point de vue théorique, notre contribution consiste à généraliser les propriétés oracles, établies par [33], sur la régression expectile et la régression expectile couplée d'une part ; d'autre part, nous avons démontré que l'algorithme proposé pour résoudre le problème de la régression quantile avec la pénalité group Lasso converge linéairement.

La régression robuste est un domaine active de la recherche en statistique en grande dimension. Elle s'intéresse à modéliser les données contenant des observations contaminées par des valeurs aberrantes. La présence de celles-ci peut biaiser significativement l'estimation des paramètres du modèle. En grand dimension, le risque de contamination des données est plus important. Comme nous l'avons vu dans le chapitre 2, l'approche GPQR assure la robustesse et la parcimonie du modèle simultanément. Chang *et al.* [16] ont proposé une nouvelle fonction de perte, appelée 'Tukey's Biweight'. Elle est définie par

$$\rho_d(u) = \begin{cases} \dfrac{d^2}{6}\left(1 - \left[1 - \dfrac{u^2}{d^2}\right]^3\right)\nVdash_{(|u| \leq d)}, \\ \dfrac{d^2}{6}\nVdash_{(|u| \geq d)}, \end{cases} \tag{4.2}$$

où d est un paramètre de réglage. La fonction de perte $\rho_d(.)$ est continûment différentiable partout. Si $|u| \leq d$, $\rho_d(.)$ est un polynôme de degré six ; sinon, elle est constante. Contrairement à la fonction de perte quantile, qui est linéaire par morceaux, et qui permet d'atténuer l'effet d'une valeur aberrante, sans toutefois annuler cet effet, la fonction $\rho_d(.)$ annule cet effet. En combinant les avantages de ces deux foncions de pertes, nous proposons la fonction de perte suivante

$$\rho_{\tau,d}(u) = \begin{cases} h(-d, 1-\tau, d) \text{ If } u \leq -d, \\ h(u, 1-\tau, d) \text{ If } -d \leq u \leq -d(1-\tau), \\ (\tau - 1)u \text{ If } -d(1-\tau) \leq u \leq 0, \\ \tau u \text{ If } 0 \leq u \leq d\tau, \\ h(u, \tau, d) \text{ If } \tau d \leq u \leq d, \\ h(d, \tau, d) \text{ If } u \geq d, \end{cases} \tag{4.3}$$

où

$$h(u, \tau, d) = f(u, \tau, d) - f(\tau d, \tau, d) - \tau^2 d$$

et

$$f(u, \tau, d) = \frac{d^2}{6(1-\tau^2)^2}\left(1 - \left[1 - \frac{u^2}{d^2}\right]^3\right).$$

Ainsi, si $|u| \leq d$, $\rho_{\tau,d}(u)$ est égale à la fonction de perte quantile, étant n'est pas dérivable en zéro. Sinon, $\rho_{\tau,d}(i)$ est une fonction de perte équivalente à $\rho_d(.)$ à une constante près.

De point de vue algorithmique, pour résoudre un problème d'optimisation de type Lasso avec la fonction de perte $\rho_{\tau,d}(u)$, nous pouvons utiliser des approximations similaires à (2.5) ou (2.6).

[62] ont montré que la régression quantile est plus robuste aux points aberrants que la régression expectile. [113] ont proposé une version robuste de la régression expectile en grande dimension avec les pénalités Lasso, SCAD et MCP. Dans des travaux futures, nous pouvons proposer une extension des pénalités group Lasso, group SCAD, group MCP à la régression expectile robuste en utilisant une fonction de perte identique à la fonction expectile pour les points non aberrants et une fonction pseudo-quantile pour les points aberrants.

BIBLIOGRAPHIE

[1] Aigner, D. J., Amemiya, T. et Poirier, D. J. (1976). On the estimation of production frontiers : maximum likelihood estimation of the parameters of a discontinuous density function. *International economic review*, 377–396.

[2] Alhamzawi, R., Yu, K. et Benoit, D. F. (2012). Bayesian adaptive lasso quantile regression. *Statistical Modelling*, *12*(3), 279–297.

[3] Aravkin, A. Y., Kambadur, A., Lozano, A. C. et Luss, R. (2014). Sparse quantile huber regression for efficient and robust estimation. *arXiv preprint arXiv :1402.4624*.

[4] Barry, A. D. (2019). La régression expectile pour l'analyse des données longitudinales.

[5] Belloni, A., Chernozhukov, V. *et al.* (2011a). l1-penalized quantile regression in high-dimensional sparse models. *The Annals of Statistics*, *39*(1), 82–130.

[6] Belloni, A., Chernozhukov, V. et Wang, L. (2011b). Square-root lasso : pivotal recovery of sparse signals via conic programming. *Biometrika*, *98*(4), 791–806.

[7] Bickel, P. J., Ritov, Y., Tsybakov, A. B. *et al.* (2009). Simultaneous analysis of lasso and dantzig selector. *The Annals of statistics*, *37*(4), 1705–1732.

[8] Bloomfield, P. et Steiger, W. L. (1983). *Least absolute deviations : theory, applications, and algorithms*. Springer.

[9] Bondell, H. D., Reich, B. J. et Wang, H. (2010). Noncrossing quantile regression curve estimation. *Biometrika*, *97*(4), 825–838.

[10] Breheny, P. (2015). grpreg : Regularization paths for regression models with grouped covariates. *R package version*, *2*, 8–1.

[11] Breheny, P. et Huang, J. (2011). Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. *The annals of applied statistics*, *5*(1), 232.

[12] Breheny, P. et Huang, J. (2015). Group descent algorithms for nonconvex penalized linear and logistic regression models with grouped predictors. *Statistics and computing*, *25*(2), 173–187.

[13] Briollais, L. et Durrieu, G. (2014). Application of quantile regression to recent genetic and-omic studies. *Human genetics*, *133*(8), 951–966.

[14] Bühlmann, P. et Van De Geer, S. (2011). *Statistics for high-dimensional data : methods, theory and applications*. Springer Science & Business Media.

[15] Candes, E. et Tao, T. (2007). The dantzig selector : Statistical estimation when p is much larger than n. *The annals of Statistics*, *35*(6), 2313–2351.

[16] Chang, L., Roberts, S. et Welsh, A. (2018). Robust lasso regression using tukey's biweight criterion. *Technometrics*, *60*(1), 36–47.

[17] Chiolero, A., Bovet, P. et Paccaud, F. (2005). Association between maternal smoking and low birth weight in switzerland : the eden study. *Swiss medical weekly*, *135*(35-36), 525–530.

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

[19] Daouia, A., Gijbels, I. et Stupfler, G. (2019). Extremiles : A new perspective on asymmetric least squares. *Journal of the American Statistical Association*, *114*(527), 1366–1381.

[20] Daouia, A., Gijbels, I. et Stupfler, G. (2021). Extremile regression. *Journal of the American Statistical Association*, 1–8.

[21] Daye, Z. J., Chen, J. et Li, H. (2012). High-dimensional heteroscedastic regression with an application to eqtl data analysis. *Biometrics*, *68*(1), 316–326.

[22] Durinck, S., Spellman, P. T., Birney, E. et Huber, W. (2009). Mapping identifiers for the integration of genomic datasets with the r/bioconductor package biomart. *Nature protocols*, *4*(8), 1184.

[23] Efron, B. (1991). Regression percentiles using asymmetric squared error loss. *Statistica Sinica*, 93–125.

[24] Efron, B., Hastie, T., Johnstone, I., Tibshirani, R. *et al.* (2004). Least angle regression. *The Annals of statistics*, *32*(2), 407–499.

[25] Efron, B., Hastie, T. et Tibshirani, R. (2007). Discussion : The dantzig selector : Statistical estimation when p is much larger than n. *The Annals of Statistics*, *35*(6), 2358–2364.

[26] Fan, J., Fan, Y. et Barut, E. (2014a). Adaptive robust variable selection. *Annals of statistics*, *42*(1), 324.

[27] Fan, J. et Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, *96*(456), 1348–1360.

[28] Fan, J. et Peng, H. (2004). Nonconcave penalized likelihood with a diverging number of parameters. *The annals of statistics*, *32*(3), 928–961.

[29] Fan, J., Xue, L. et Zou, H. (2014b). Strong oracle optimality of folded concave penalized estimation. *Annals of statistics*, *42*(3), 819.

[30] Fenske, N., Kneib, T. et Hothorn, T. (2011). Identifying risk factors for severe childhood malnutrition by boosting additive quantile regression. *Journal of the American Statistical Association*, *106*(494), 494–510.

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

[32] Friedman, J., Hastie, T. et Tibshirani, R. (2010b). Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, *33*(1), 1.

[33] Gu, Y., Zou, H. *et al.* (2016). High-dimensional generalizations of asymmetric least squares regression and their applications. *The Annals of Statistics*, *44*(6), 2661–2694.

[34] Hashem, H., Vinciotti, V., Alhamzawi, R. et Yu, K. (2016). Quantile regression with group lasso for classification. *Advances in Data Analysis and Classification*, *10*(3), 375–390.

[35] Hertz, J. M., Schell, G. et Doerfler, W. (1999). Factors affecting de novo methylation of foreign dna in mouse embryonic stem cells. *Journal of Biological Chemistry*, *274*(34), 24232–24240.

[36] Hofner, B., Mayr, A., Robinzonov, N. et Schmid, M. (2014). Model-based boosting in r : a hands-on tutorial using the r package mboost. *Computational statistics*, *29*(1-2), 3–35.

[37] Hohman, T. J., Koran, M. E. I. et Thornton-Wells, T. A. (2014). Genetic modification of the relationship between phosphorylated tau and neurodegeneration. *Alzheimer's & dementia : the journal of the Alzheimer's Association*, *10*(6), 637–645.

[38] Hong, M., Wang, X., Razaviyayn, M. et Luo, Z.-Q. (2017). Iteration complexity analysis of block coordinate descent methods. *Mathematical Programming*, *163*(1-2), 85–114.

[39] Hosmer Jr, D. W., Lemeshow, S. et Sturdivant, R. X. (2013). *Applied logistic regression*, volume 398. John Wiley & Sons.

[40] Huang, J. et Zhang, C.-H. (2012). Estimation and selection via absolute penalized convex minimization and its multistage adaptive applications. *Journal of Machine Learning Research*, *13*(Jun), 1839–1864.

[41] Hunter, D. R. et Lange, K. (2000). Quantile regression via an mm algorithm. *Journal of Computational and Graphical Statistics*, *9*(1), 60–77.

[42] Hunter, D. R. et Lange, K. (2004). A tutorial on mm algorithms. *The American Statistician*, *58*(1), 30–37.

[43] Jennings, L., Wong, K. et Teo, K. (1996). Optimal control computation to account for eccentric movement. *The ANZIAM Journal*, *38*(2), 182–193.

[44] Ji, Y., Lin, N. et Zhang, B. (2012). Model selection in binary and tobit quantile regression using the gibbs sampler. *Computational Statistics & Data Analysis*, *56*(4), 827–839.

[45] Jiang, C., Jiang, M., Xu, Q. et Huang, X. (2017). Expectile regression neural network model with applications. *Neurocomputing*, *247*, 73–86.

[46] Juban, R., Ohlsson, H., Maasoumy, M., Poirier, L. et Kolter, J. Z. (2016). A multiple quantile regression approach to the wind, solar, and price tracks of gefcom2014. *International Journal of Forecasting*, *32*(3), 1094–1102.

[47] Kadkhodaie, M., Sanjabi, M. et Luo, Z.-Q. (2014). On the linear convergence of the approximate proximal splitting method for non-smooth convex optimization. *Journal of the Operations Research Society of China*, *2*(2), 123–141.

[48] Kato, K. (2011). Group lasso for high dimensional sparse quantile regression models. *arXiv preprint arXiv :1103.1458*.

[49] Kim, S., Swaminathan, S., Shen, L., Risacher, S., Nho, K., Foroud, T., Shaw, L., Trojanowski, J., Potkin, S., Huentelman, M. *et al.* (2011). Genome-wide association study of csf biomarkers aβ1-42, t-tau, and p-tau181p in the adni cohort. *Neurology*, *76*(1), 69–79.

[50] Koenker, R. (1984). A note on l-estimates for linear models. *Statistics & probability letters*, *2*(6), 323–325.

[51] Koenker, R. (2005). *Quantile regression*. Numéro 38. Cambridge university press.

[52] Koenker, R. et Bassett Jr, G. (1978). Regression quantiles. *Econometrica : journal of the Econometric Society*, 33–50.

[53] Koenker, R. et D'orey, V. (1993). Computing dual regression quantiles and regression rank score. *J. Roy. Stat. Soc. C*, *43*, 410–414.

[54] Koenker, R. et Hallock, K. F. (2001). Quantile regression. *Journal of economic perspectives*, *15*(4), 143–156.

[55] Koenker, R. et Zhao, Q. (1994). L-estimatton for linear heteroscedastic models. *Journaltitle of Nonparametric Statistics*, *3*(3-4), 223–235.

[56] Kozumi, H. et Kobayashi, G. (2011). Gibbs sampling methods for bayesian quantile regression. *Journal of statistical computation and simulation*, *81*(11), 1565–1578.

[57] Lakhal-Chaieb, L., Greenwood, C. M., Ouhourane, M., Zhao, K., Abdous, B. et Oualkacha, K. (2017). A smoothed em-algorithm for dna methylation profiles from sequencing-based methods in cell lines or for a single cell type. *Statistical applications in genetics and molecular biology*, *16*(5-6), 333–347.

[58] Lange, K., Papp, J. C., Sinsheimer, J. S. et Sobel, E. M. (2014). Next-generation statistical genetics : Modeling, penalization, and optimization in high-dimensional data. *Annual Review of Statistics and Its Application*, *1*(1), 279–300.

[59] Li, J., Zhang, Q., Chen, F., Meng, X., Liu, W., Chen, D., Yan, J., Kim, S., Wang, L., Feng, W. *et al.* (2017). Genome-wide association and interaction studies of csf t-tau/aβ42 ratio in adni cohort. *Neurobiology of aging*, *57*, 247–e1.

[60] Li, Y. et Arce, G. R. (2004). A maximum likelihood approach to least absolute deviation regression. *EURASIP Journal on Advances in Signal Processing*, *2004*(12), 1–8.

[61] Li, Y. et Zhu, J. (2008). L 1-norm quantile regression. *Journal of Computational and Graphical Statistics*, *17*(1), 163–185.

[62] Liao, L., Park, C. et Choi, H. (2019). Penalized expectile regression : an alternative to penalized quantile regression. *Annals of the Institute of Statistical Mathematics*, *71*(2), 409–438.

[63] Liu, Y. et Wu, Y. (2009). Stepwise multiple quantile regression estimation using non-crossing constraints. *Statistics and its Interface*, *2*(3), 299–310.

[64] Luo, Z.-Q. et Tseng, P. (1992). On the linear convergence of descent methods for convex essentially smooth minimization. *SIAM Journal on Control and Optimization*, *30*(2), 408–425.

[65] Luo, Z.-Q. et Tseng, P. (1993). Error bounds and convergence analysis of feasible descent methods : a general approach. *Annals of Operations Research*, *46*(1), 157–178.

[66] Mayr, A., Binder, H., Gefeller, O. et Schmid, M. (2014). The evolution of boosting algorithms-from machine learning to statistical modelling. *arXiv preprint arXiv :1403.1452*.

[67] Meier, L., Van De Geer, S. et Bühlmann, P. (2008). The group lasso for logistic regression. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, *70*(1), 53–71.

[68] Meier, L., Van de Geer, S., Bühlmann, P. *et al.* (2009). High-dimensional additive modeling. *The Annals of Statistics*, *37*(6B), 3779–3821.

[69] Meinshausen, N. (2007). Relaxed lasso. *Computational Statistics & Data Analysis*, *52*(1), 374–393.

[70] Mkhadri, A. et Ouhourane, M. (2013). An extended variable inclusion and shrinkage algorithm for correlated variables. *Computational Statistics & Data Analysis*, *57*(1), 631–644.

[71] Mkhadri, A. et Ouhourane, M. (2015). A group visa algorithm for variable selection. *Statistical Methods & Applications*, *24*(1), 41–60.

[72] Mkhadri, A., Ouhourane, M. et Oualkacha, K. (2017). A coordinate descent algorithm for computing penalized smooth quantile regression. *Statistics and Computing*, *27*(4), 865–883.

[73] Newey, W. K. et Powell, J. L. (1987a). Asymmetric least squares estimation and testing. *Econometrica*, *55*(4), 819–47. Récupéré de `https://ideas.repec.org/a/ecm/emetrp/v55y1987i4p819-47.html`

[74] Newey, W. K. et Powell, J. L. (1987b). Asymmetric least squares estimation and testing. *Econometrica : Journal of the Econometric Society*, 819–847.

[75] Ogutu, J. O. et Piepho, H.-P. (2014). Regularized group regression methods for genomic prediction : Bridge, mcp, scad, group bridge, group lasso, sparse group lasso, group mcp and group scad. Dans *BMC proceedings*, p. S7. BioMed Central.

[76] Oh, H.-S., Lee, T. C. et Nychka, D. W. (2011). Fast nonparametric quantile regression with arbitrary smoothing methods. *Journal of Computational and Graphical Statistics*, *20*(2), 510–526.

[77] Ouhourane, M., Yang, Y., Benedet, A. L. et Oualkacha, K. (2021). Group penalized quantile regression. *Statistical Methods & Applications*, 1–35.

[78] Peng, B. et Wang, L. (2015). An iterative coordinate descent algorithm for high-dimensional nonconvex penalized quantile regression. *Journal of Computational and Graphical Statistics*, *24*(3), 676–694.

[79] Portnoy, S. et Koenker, R. (1997). The gaussian hare and the laplacian tortoise : computability of squared-error versus absolute-error estimators. *Statistical Science*, *12*(4), 279–300.

[80] Radchenko, P. et James, G. M. (2008). Variable inclusion and shrinkage algorithms. *Journal of the American Statistical Association*, *103*(483), 1304–1315.

[81] Roberts, S. et Nowak, G. (2014). Stabilizing the lasso against cross-validation variability. *Computational Statistics & Data Analysis*, *70*, 198–211.

[82] Rudelson, M., Vershynin, R. *et al.* (2013). Hanson-wright inequality and sub-gaussian concentration. *Electronic Communications in Probability*, *18*.

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

[84] Simon, N. et Tibshirani, R. (2012). Standardization and the group lasso penalty. *Statistica Sinica*, *22*(3), 983.

[85] Slawski, M. (2012). The structured elastic net for quantile regression and support vector classification. *Statistics and Computing*, *22*(1), 153–168.

[86] Sobotka, F., Kauermann, G., Waltrup, L. S. et Kneib, T. (2013). On confidence intervals for semiparametric expectile regression. *Statistics and Computing*, *23*(2), 135–148.

[87] Spady, D. W., Atrens, M. A. et Szymanski, W. A. (1986). Effects of mother's smoking on their infants' body composition as determined by total body potassium. *Pediatric research*, *20*(8), 716–719.

[88] Sun, R. et Hong, M. (2015). Improved iteration complexity bounds of cyclic block coordinate descent for convex problems. Dans *Advances in Neural Information Processing Systems*, 1306–1314.

[89] Tang, S., Cai, Z., Fang, Y. et Lin, M. (2021). A new quantile treatment effect model for studying smoking effect on birth weight during mother's pregnancy. *Journal of Management Science and Engineering*, *6*(3), 336–343.

[90] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 267–288.

[91] Tibshirani, R., Bien, J., Friedman, J., Hastie, T., Simon, N., Taylor, J. et Tibshirani, R. J. (2012). Strong rules for discarding predictors in lasso-type problems. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, *74*(2), 245–266.

[92] Turgeon, M., Oualkacha, K., Ciampi, A., Miftah, H., Dehghan, G., Zanke, B. W., Benedet, A. L., Rosa-Neto, P., Greenwood, C. M., Labbe, A. *et al.* (2016). Principal component of explained variance : an efficient and optimal data dimension reduction framework for association studies. *Statistical methods in medical research*, p. 0962280216660128.

[93] Venables, W. N. et Ripley, B. D. (2013). *Modern applied statistics with S-PLUS*. Springer Science & Business Media.

[94] Vershynin, R. (2010). Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv :1011.3027*.

[95] Waldmann, E., Kneib, T., Yue, Y. R., Lang, S. et Flexeder, C. (2013). Bayesian semiparametric additive quantile regression. *Statistical Modelling*, *13*(3), 223–252.

[96] Wang, H., Lengerich, B. J., Aragam, B. et Xing, E. P. (2019). Precision lasso : accounting for correlations and linear dependencies in high-dimensional genomic data. *Bioinformatics*, *35*(7), 1181–1187.

[97] Wang, L. (2013). The l1 penalized lad estimator for high dimensional linear regression. *Journal of Multivariate Analysis*, *120*, 135–151.

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

[99] Wei, F. et Zhu, H. (2012). Group coordinate descent algorithms for nonconvex penalized regression. *Computational Statistics & Data Analysis*, *56*(2), 316–326.

[100] Wilcox, A. J. (1993). Birth weight and perinatal mortality : the effect of maternal smoking. *American Journal of Epidemiology*, *137*(10), 1098–1104.

[101] Wu, T. T., Lange, K. *et al.* (2008). Coordinate descent algorithms for lasso penalized regression. *The Annals of Applied Statistics*, *2*(1), 224–244.

[102] Xu, Q., Ding, X., Jiang, C., Yu, K. et Shi, L. (2020). An elastic-net penalized expectile regression with applications. *Journal of Applied Statistics*, 1–26.

[103] Yang, Y., Zhang, T. et Zou, H. (2018). Flexible expectile regression in reproducing kernel hilbert spaces. *Technometrics*, *60*(1), 26–35.

[104] Yang, Y. et Zou, H. (2013). An efficient algorithm for computing the hhsvm and its generalizations. *Journal of Computational and Graphical Statistics*, *22*(2), 396–415.

[105] Yang, Y. et Zou, H. (2015a). A fast unified algorithm for solving group-lasso penalize learning problems. *Statistics and Computing*, *25*(6), 1129–1141.

[106] Yang, Y. et Zou, H. (2015b). Nonparametric multiple expectile regression via er-boost. *Journal of Statistical Computation and Simulation*, *85*(7), 1442–1458.

[107] Ye, F. et Zhang, C.-H. (2010). Rate minimaxity of the lasso and dantzig selector for the lq loss in lr balls. *Journal of Machine Learning Research*, *11*(Dec), 3519–3540.

[108] Yi, C. et Huang, J. (2017). Semismooth newton coordinate descent algorithm for elastic-net penalized huber loss regression and quantile regression. *Journal of Computational and Graphical Statistics*, *26*(3), 547–557.

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

[110] Zhang, C.-H. *et al.* (2010). Nearly unbiased variable selection under minimax concave penalty. *The Annals of statistics*, *38*(2), 894–942.

[111] Zhang, H., Jiang, J. et Luo, Z.-Q. (2013). On the linear convergence of a proximal gradient method for a class of nonsmooth convex minimization problems. *Journal of the Operations Research Society of China*, *1*(2), 163–186.

[112] Zhao, G., Teo, K. L. et Chan, K. (2005). Estimation of conditional quantiles by a new smoothing approximation of asymmetric loss functions. *Statistics and Computing*, *15*(1), 5–11.

[113] Zhao, J., Yan, G. et Zhang, Y. (2021). Robust estimation and shrinkage in ultrahigh dimensional expectile regression with heavy tails and variance heterogeneity. *Statistical Papers*, 1–28.

[114] Zhao, J., Yan, G. et Zhang, Y. (2022). Robust estimation and shrinkage in ultrahigh dimensional expectile regression with heavy tails and variance heterogeneity. *Statistical Papers*, *63*(1), 1–28.

[115] Zhao, J. et Zhang, Y. (2018). Variable selection in expectile regression. *Communications in Statistics-Theory and Methods*, *47*(7), 1731–1746.

[116] Zhou, H., Alexander, D. H., Sehl, M. E., Sinsheimer, J. S., Sobel, E. et Lange, K. (2011). Penalized regression for genome-wide association screening of sequence data. Dans *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, p. 106. NIH Public Access.

[117] Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American statistical association*, *101*(476), 1418–1429.

[118] Zou, H. et Hastie, T. (2003). Regression shrinkage and selection via the elastic net, with applications to microarrays. *JR Stat Soc Ser B*, *67*, 301–20.

[119] Zou, H. et Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the royal statistical society : series B (statistical methodology)*, *67*(2), 301–320.

[120] Zou, H. et Li, R. (2008). One-step sparse estimates in nonconcave penalized likelihood models. *Annals of statistics*, *36*(4), 1509.