GPER

Installation

Like many other R packages, the simplest way to obtain GPER is to install it from github. Type the following command in R console:

```r
library(devtools)
#> Le chargement a nécessité le package : usethis
devtools::install_github("https://github.com/ouhourane/GPER.git")
#> Skipping install of 'GPER' from a github remote, the SHA1 (fe585495) has not changed
#>       since last install.
#>   Use `force = TRUE` to force installation
library(GPER)
#> Le chargement a nécessité le package : Matrix
```

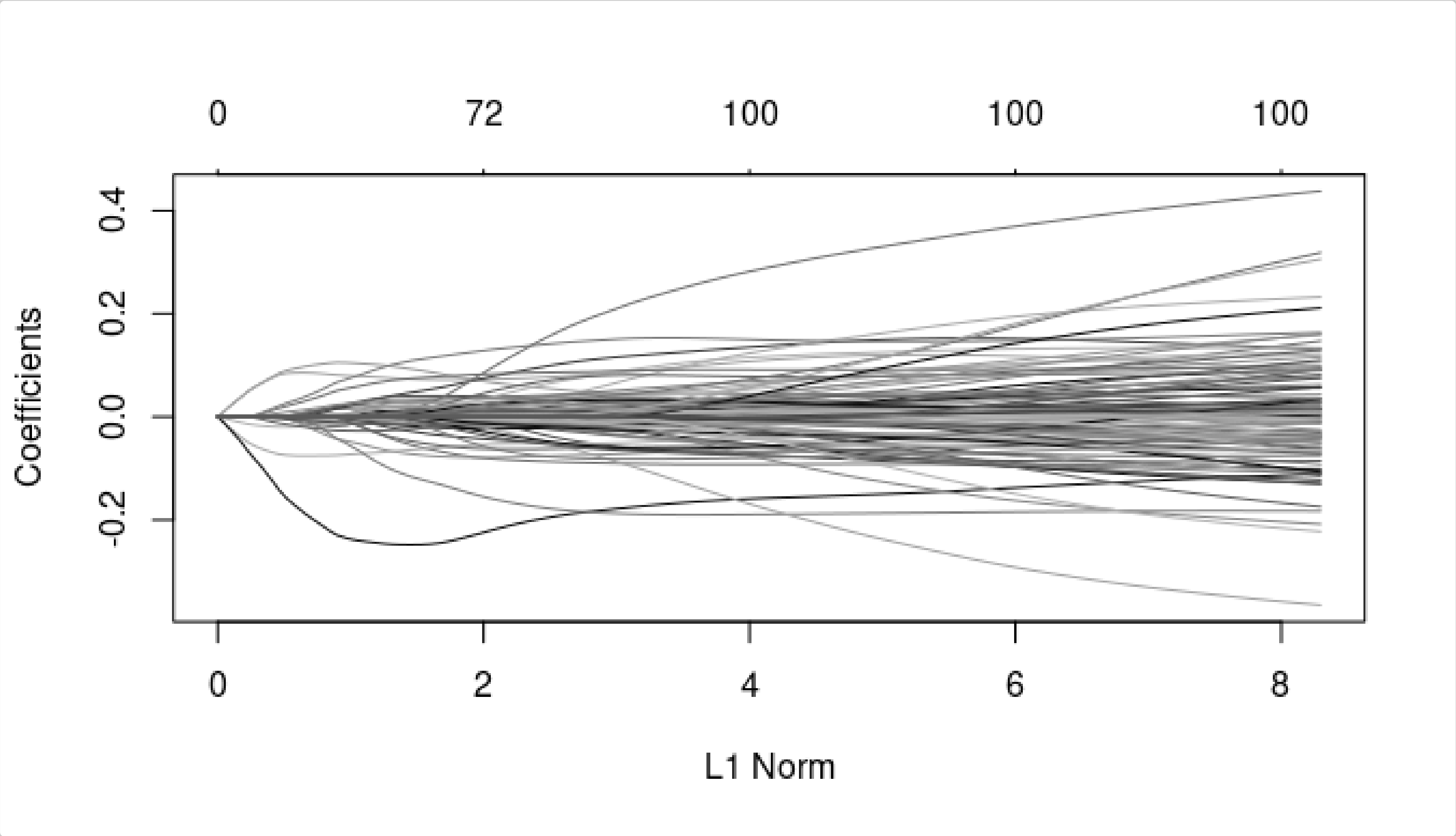
All graphs in this vignette are created using others functions. The code for these functions can be found in the GPER.Rmd file.

In this vignette, we demonstrate how to use the GPER() and cv.GPER() functions in the GPER package to fit the regularization path of parametric expectile regression with grouped penalties. This includes group selection methods such as group Lasso, group Mcp, and group Scad. The others function: predict(), coef(), cv.predict, cv.coef, ... are minor modifications or directly copied from the glmnet package.

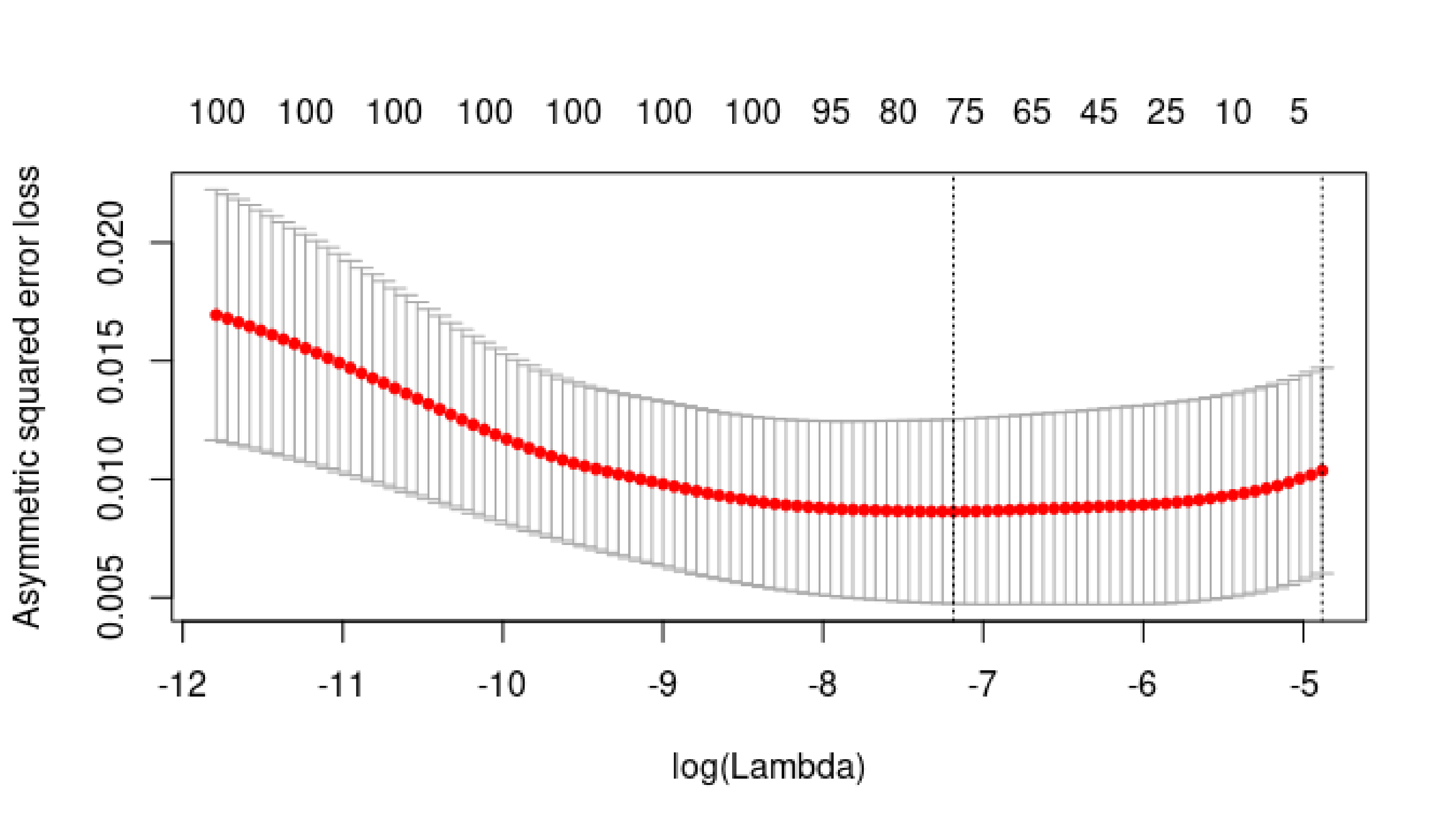
Bardet dataset

Gene expression data (20 genes for 120 samples) from the microarray experiments of mammalian eye tissue samples of Scheetz et al. (2006). This data set contains 120 samples with 100 predictors (expanded from 20 genes using 5 basis B-splines, as described in Yang, Y. and Zou, H. (2012)).

```r
library(GPER)
#load bardet dataset fron GSQR package
library(GPQR)
data(bardet)
group <- rep(1:20,each=5)
#run GPER for group Lasso penaly, taux = 0.5 and  with the penalty group Lasso
#and the first pseudo-quantile approximation loss function
fit <- gper(x=bardet$x,y=bardet$y,group=group,method="GLasso",tau=0.5)
# To produce a coefficient profile plot of the coefficient paths for a fitted GPER object.
plot(fit)
```



```
# cv.gper does k-fold cross-validation for GPER, produces a plot,
#and returns a value for lambda.
#function is based on the cv function from the glmnet package.
cv <- cv.gper(x=bardet$x,y=bardet$y,group=group,method="GLasso",tau=0.5)
plot(cv)
```



Birthwt data

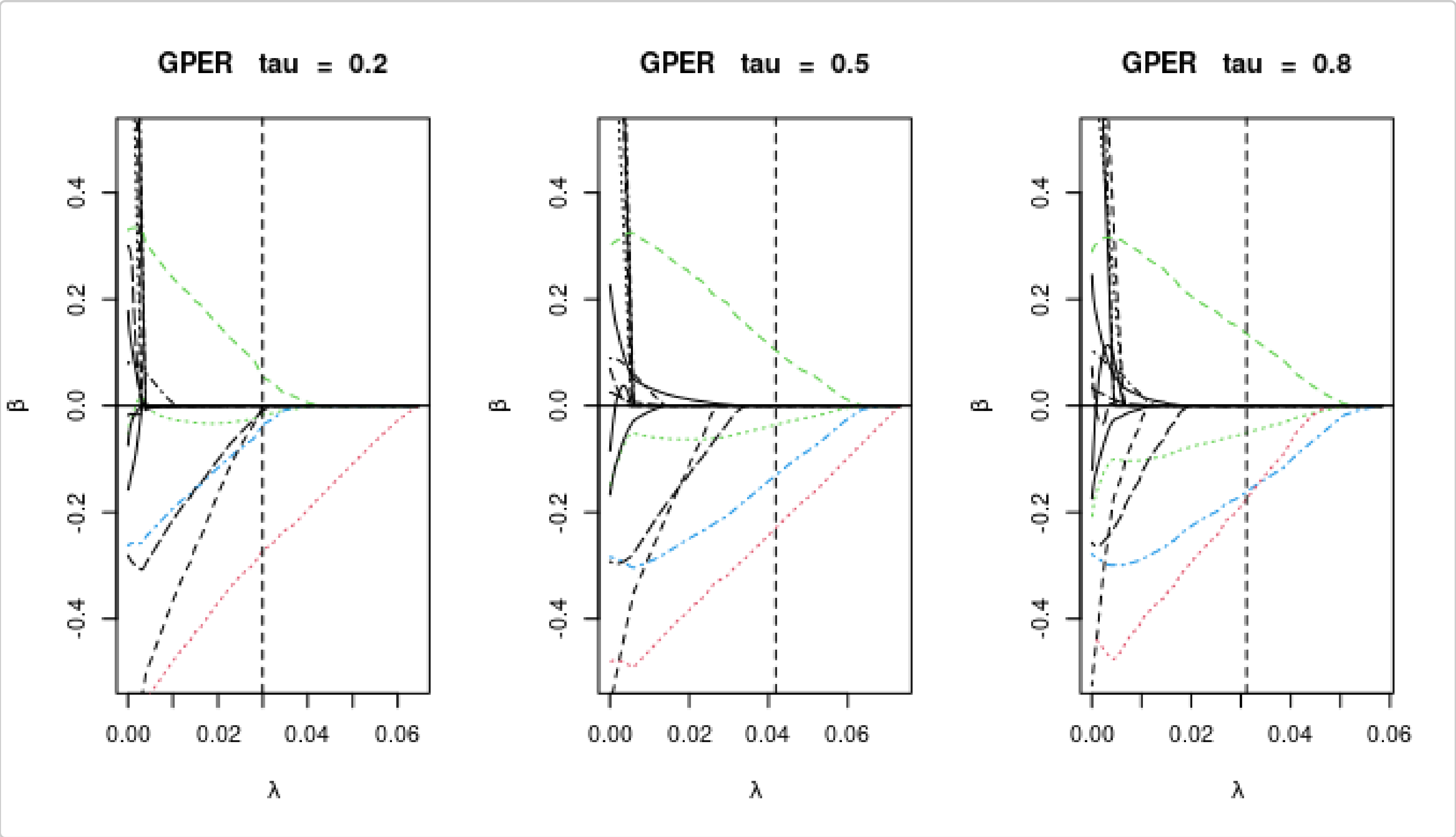
The birth weight data set records the birth weights of 189 babies and eight predictors concerning the mother. Among the eight predictors, two are continuous (mother's age in years and mother's weight in pounds at the last menstrual period) and six are categorical (mother's race (white, black or other), smoking status during pregnancy (yes or no), number of previous premature labours (0, 1 or 2 or more), history of hypertension (yes or no), presence of uterine irritability (yes or no), number of physician visits during the first trimester (0, 1, 2 or 3 or more)). The data were collected at Baystate Medical Center, Springfield, Massachusetts, during 1986.

we fitted the GPER and COGPER with the group Lasso penalty for all 189 babies with 5-fold cross-validation to obtain a better model estimation.

The function "plot_GPER" below produces the coefficient paths of GPER with fixed τ, are shown as a function of the tuning parameter

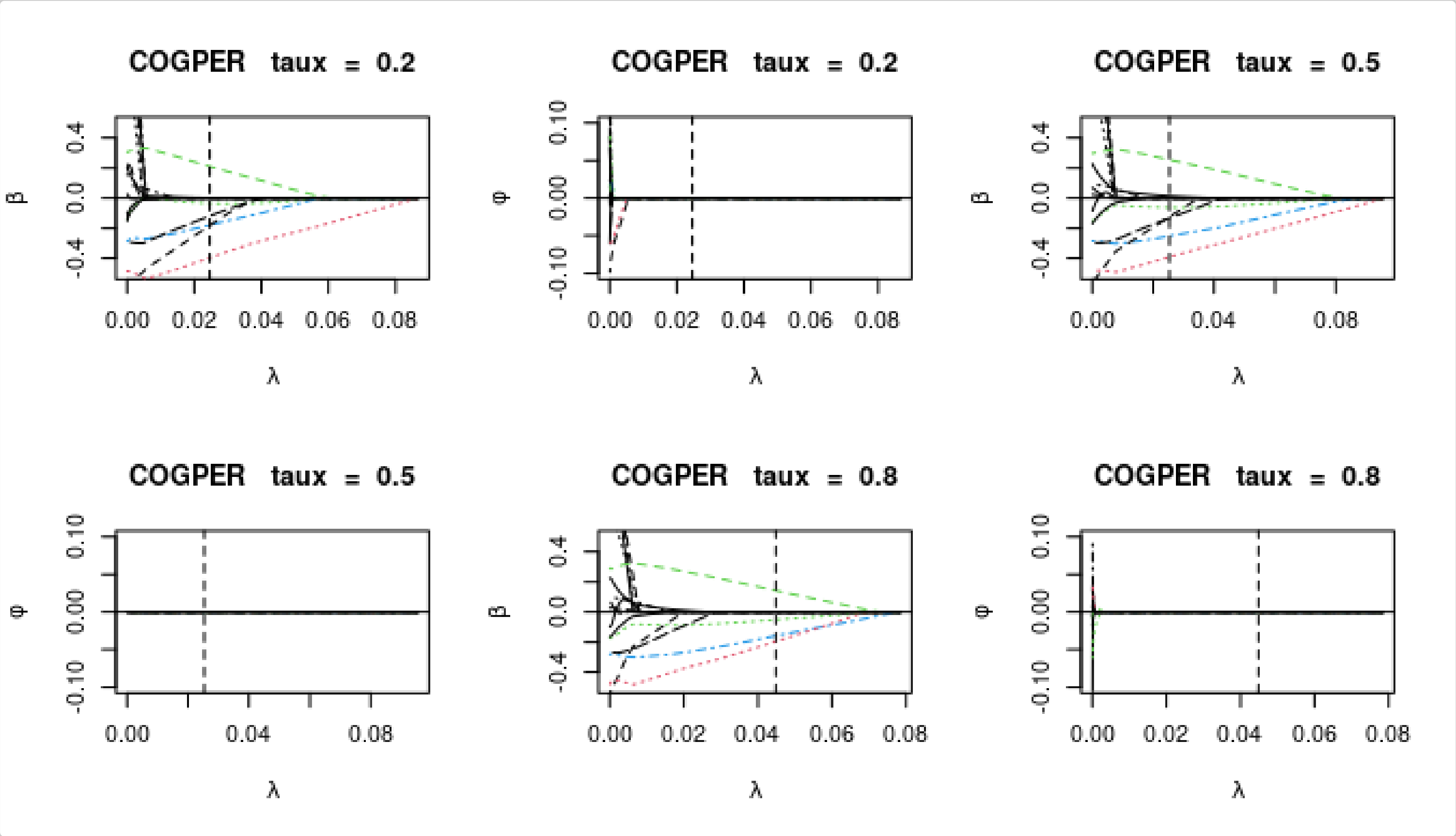
```
library(grpreg)
# load Birthwt data from grpreg package
data(Birthwt)
x <- Birthwt$X
group <- Birthwt$group
y <- Birthwt$bwt
group <- c(1,1,1,2,2,2,3,3,4,5,5,6,7,8,8,8)
bs <- sqrt(as.integer(as.numeric(table(group))))
# The function "plot_GPER" code
```

```r
plot_GPER <- function(taux){
  cv <- cv.gper(x=x,y=y,group=group,method="GLasso",tau=taux,eps=0.0001)
  matplot(cv$lambda,t(cv$gper.fit$beta),ylim=c(-0.5,0.5),ylab = expression(beta),
          xlab = expression(lambda),type="l",main=paste("GPER  ",expression(tau) ,
                                                                    " =
        ",taux),col=c(1,1,1,1,1,1,3,3,4,1,1,1,2,1,1,1))
  abline(h=0,col=1); abline(v=cv$lambda.1se,col=1,lty=2)
}
# running "plot_GPER" for three values of tau = 0.2, 0.5, 0.8
par(mfrow = c(1, 3))
plot_GPER(0.2)
plot_GPER(0.5)
plot_GPER(0.8)
```



The function “plot_COGPER” below produces the coefficients paths β and \bphi of COGPER respectivey with $\tau \in \{0.2, 0.5, 0.8\}$

```r
# running "plot_GPER" for three values of tau = 0.2, 0.5, 0.8
par(mfrow = c(2, 3))
plot_COGPER(0.2)
plot_COGPER(0.5)
plot_COGPER(0.8)
```



The impact of the smoking status is more important for GPER and COGPER with $\tau = 0.5$ or 0.8. The coefficient value of smoking status gives the total
effects on both mean and scale function, it is can not separate by GPER with $\tau = 0.8$. However, the heteroscedastic effect of smoking status, and estimates the amount of the the heteroscedastic effect and separate it's to the mean function.

Illustration example

We generate one data set of 50 observations and five predictors X_i from a normal standard distribution and the correlations among the columns in the design matrix was set equal to 0.5. We compute a cubic B-spline basis (W^1, W_i^2, W_i^3) from each predictor $X_i, \ i = 1, \ldots, 5$, and we set $X_1 = \Phi(W_1^1 + W_1^2 + W_1^3)$ and $X_i^j = W_i^j$ for $i = 2, \ldots, 5$, where $\Phi(.)$ is the standard normal CDF. In this heteroscedasticity model, we consider that G_2 and G_3 have an effect on the mean and G_1 has the effect only on the scale, we definite β as

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

For the second model, it is similar to the first one, except that G_1 has the effect on both mean and scale, β is given by

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

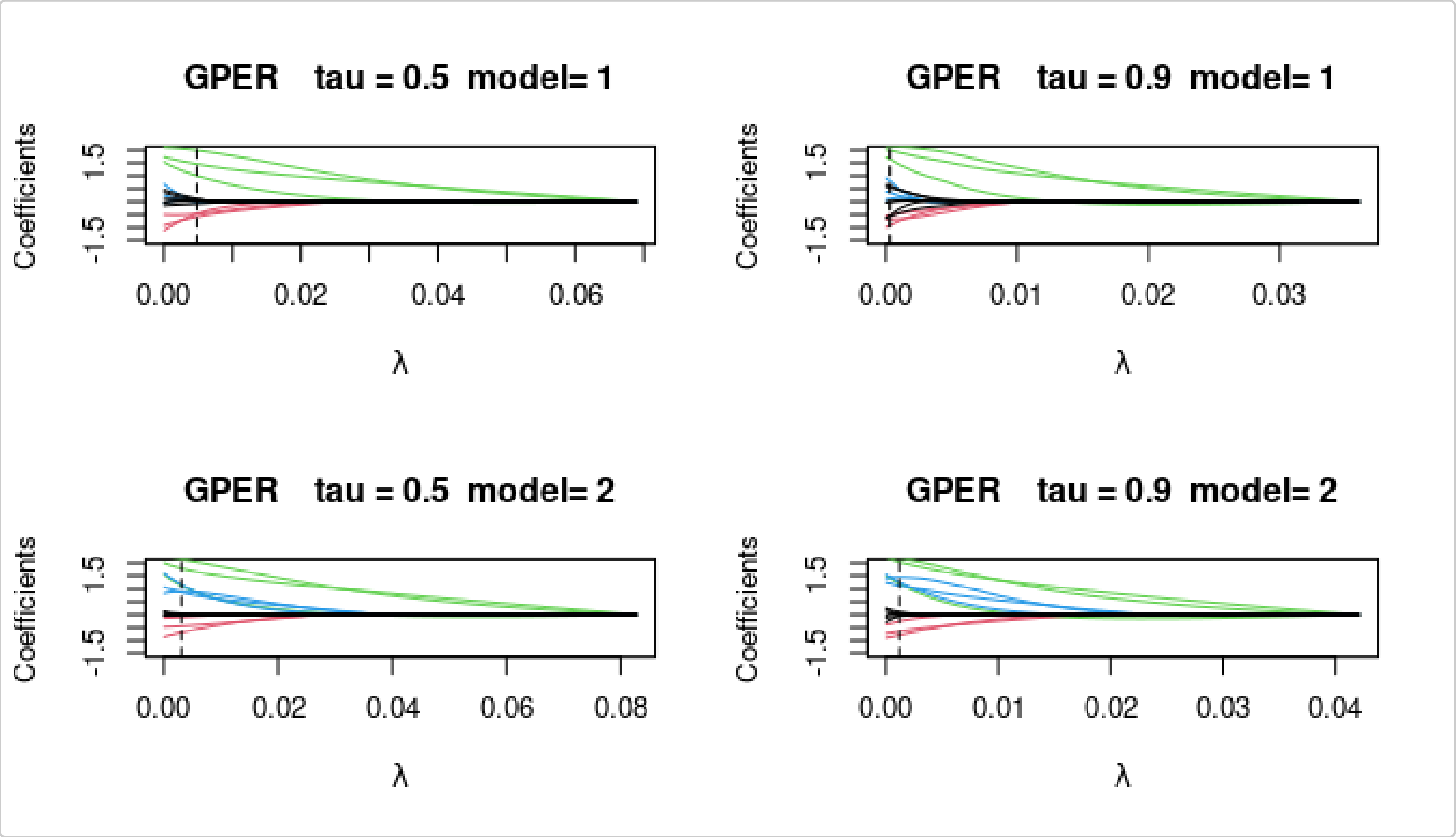
The design matrices were generated from a normal standard distribution. The response \by is generated as:

$$y = x^{\top}\beta + \Phi(G_1)\epsilon, \quad \epsilon \sim N(0, 1),$$

where $\Phi(\cdot)$ is the cumulative distribution function of the standard normal distribution.

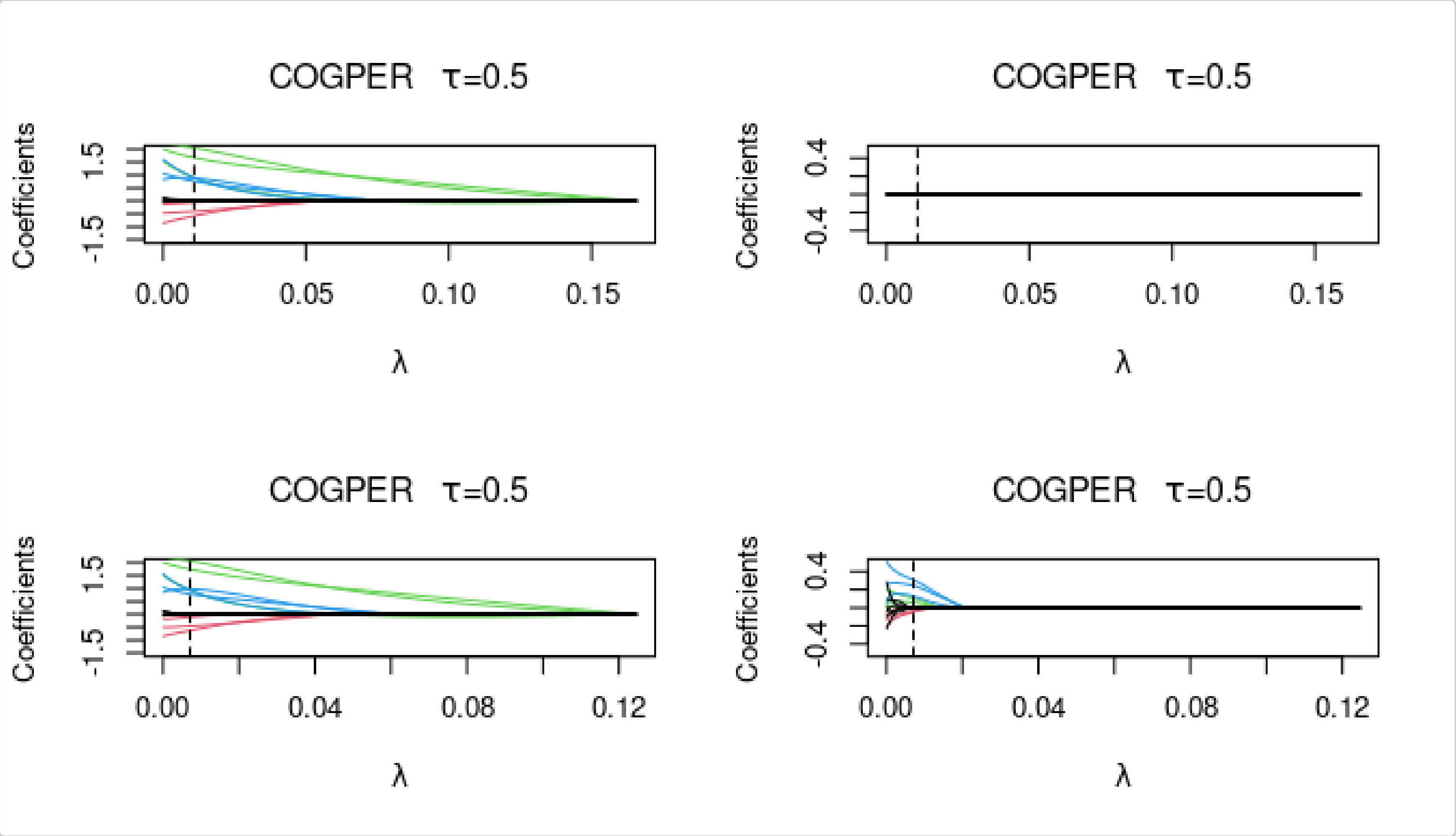
```
library("MASS")
#> 
#> Attachement du package : 'MASS'
#> L'objet suivant est masqué depuis 'package:grpreg':
#> 
#>     select
library(GPER)
library(splines)
par(mfrow = c(2, 2))
xlm=c(-1.5,2)
xlm1=c(-0.5,0.5)
ng=5

group <- rep(1:ng,each=3)
n=300;p=length(group);
betac1=c(rep(2,3),rep(-1,3),rep(0,3),rep(0,p-9))
betac2=c(rep(2,3),rep(-1,3),rep(1,3),rep(0,p-9))
cc1=c(1,2,3);cc2=c(4,5,6);cc3=c(7,8,9);cc4=c(10:p);
sig=1
Xtrain=NULL;
set.seed(1)
for(i in 1:ng) Xtrain=cbind(Xtrain,bs(runif(n),df=3))
## model 1
Ytrain<-Xtrain%*%betac1 + (pnorm(Xtrain[,7]+Xtrain[,8]+Xtrain[,9]))*rnorm(n,0,sig)
## running "plot_GPER_illustr" function for two value of tau = 0.5, 0.9 and model 1.
plot_GPER_illustr(0.5,1)
plot_GPER_illustr(0.9,1)
## model 2
Ytrain<-Xtrain%*%betac2 + (pnorm(Xtrain[,7]+Xtrain[,8]+Xtrain[,9]))*rnorm(n,0,sig)
## running "plot_GPER_illustr" function for two value of tau = 0.5, 0.9 and model 2.
plot_GPER_illustr(0.5,2)
plot_GPER_illustr(0.9,2)
```



At the top from figure above, 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, the signature of the Group G_1 appearing in the scale function, is detected by GPER-GLasso for $\tau = 0.9$; but it is not detected by least-square GLasso (GPER with $\tau = 0.5$). However, at the batton, when GPER-GLasso picks the tree groups G_1,G_2 and G_3, the coefficients values of each variables in the blue group G_1, estimated by the GPER-GLasso ($\tau = 0.95$), are superior to 1. Those coefficients values are greater than the true values 1. Thus, the value of $\widehat{\backslash\mathrm{bbeta}}_{G_1}$ gives the total effects of G_1 on both mean and scale function, it is can not separate by GPER. This lead us to propose the Coupled Group Expectile Regression (COGPER) for analyzing the heteroscedasticity in high-dimensional data, and separating these two effects.

```
## running "plot_GPER_illustr" function for two value of tau = 0.5, 0.9 and model 2.
par(mfrow = c(2, 2))
plot_CoGPER_illustr(0.5,2)
plot_CoGPER_illustr(0.9,2)
```



The figure above shows that COGPER-GLasso has a tendency to select the groups of variables that have effect on the conditional $\tau-$ mean for any $\tau \in (0.5, 0.95)$. Furthermore, the heteroscedastic effect of group G_1 in the scale function is often recovered (Blue-color group) when fitting model for the 0.95th conditional mean, but it is not the case for the COGPER-GLasso model with $\tau = 0.5$. This shows that the COGPER-GLasso can not only be used to detect the heteroscedastic effect of groups, but also estimates the amount of the heteroscedastic effect and separate it's to the mean function. Indeed, the coefficients values of G_1 are similar to the true values $(1, 1, 1)$ for both value of $\tau \in (0.5, 0.95)$, which is not the case in the right-end graph of the figure 1. The estimated value of $\hat{\beta}_{G_1}$ gives only effect of G_1 on mean function. The estimated value of the scale effect ϕ is null for all the groups with $\tau = 0.5$. On contrary, the group G_1 is the most important group which has effect on the scale function.