QICD: Iterative Coordinate Descent Algorithm for High-dimensional Nonconvex Penalized Quantile Regression

Bo Peng

8 November 2014

The QICD algorithm combines the idea of the Majorization Minimization (MM) algorithm with that of the coordinate descent algorithm. More specifically, we first replace the non-convex penalty function by its majorization function to create a surrogate objective function. Then we minimize the surrogate objective function with respect to a single parameter at each time and cycle through all parameters until convergence. For each univariate minimization problem, we only need to compute a one-dimensional weighted median, which ensures fast computation. See Peng and Wang (2014), for more details. We introduce a new `R` package `QICD` which implements this iterative coordinate descent algorithm on non-convex penalized quantile regression model. The `QICD` package implements High dimensional BIC (HBIC, see Lee,Noh and Park (2014)) and k fold cross validation as tuning parameter selection criterion.

This vignette contains only a brief introduction to utilize `QICD` to solve non-convex penalized quantile regression under high-dimensional settings. We consider a random sample $\{Y_i, \mathbf{x}_i\}, i = 1, 2, \ldots, n$ and assume $Y_i = \mathbf{x}_i^T \boldsymbol{\beta} + \epsilon_i$, where $\mathbf{x}_i = (x_{i0}, x_{i1}, \ldots, x_{ip})^T$ is a (p+1)-dimensional vector of covariates with $x_{i0} = 1$, $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_p)^T$ is the vector of parameters, and ϵ_i is the random error.The true value $\boldsymbol{\beta}$ is assumed to be sparse in the sense most of its components are equal to zero. We are interested in identifying and estimating the nonzero component of $\boldsymbol{\beta}$ when $p >> n$.

A popular approach of solving this problem is to use penalized quantile regression for large-scale data analysis. The penalized quantile regression estimator for $\boldsymbol{\beta}$ is obtained by minimizing

$$Q(\boldsymbol{\beta}) = n^{-1} \sum_{i=1}^{n} \rho_\tau(Y_i - \mathbf{x}_i^T \beta) + \sum_{j=1}^{p} p_\lambda(|\beta_j|)$$

where $\rho_\tau(u) = u\{\tau - I(u < 0)\}$ is the check loss funtion. The tuning parameter λ in the penalty function $p_\lambda(\cdot)$ controls the model complexity and goes to zero at an approriate rate. In this vignette, we only consider a general class of nonconvex penalty function, which in particular includes the two popular nonconvex penaltyies: SCAD and MCP. The SCAD penalty function Fan and Li (2001) is defined by

$$p_\lambda(|\beta|) = \lambda|\beta| I(0 \leq |\beta| < \lambda) + \frac{a\lambda|\beta| - (\beta^2 + \lambda^2)/2}{a-1} I(\lambda \leq |\beta| \leq a\lambda) + \frac{(a+1)\lambda^2}{2} I(|\beta > a\lambda|)$$

for some $a > 2$; while the MCP penalty function Zhang (2010) has the form

$$p_\lambda(|\beta|) = \lambda(|\beta| - \frac{\beta^2}{2a\lambda}) I(0 \leq |\beta| < a\lambda) + \frac{a\lambda^2}{2} I(|\beta| \geq a\lambda)$$

for some $a > 1$. Both penalty functions are singular at the origin to achieve sparsity of estimation. They also both remain constant when $|\beta|$ exceeds $a\lambda$, which avoids over-penalizing large coefficients and alleviates the bias problem associated with Lasso.

To implement our package, we use the same setting in Peng and Wang (2014). To generate the covariates $X_1, X_2, \ldots, X_p$, we first generate $(\tilde{X}_1, \tilde{X}_2, \ldots, \tilde{X}_p)^T$ from the multivariate normal distribution $N_p(0, \mathbf{\Sigma})$ with $\mathbf{\Sigma} = (\sigma_{jk})_{p\times p}$ and $\sigma_{jk} = 0.5^{|j-k|}$. Then we set $X_1 = \phi(\tilde{X}_1)$ and $X_j = \tilde{X}_j$ for $j = 2, 3, \ldots, p$, where $\phi(\cdot)$ is the cumulative distribution function of the standard normal distribution. Then we can generate the response variable from the following location-scale regression model:

$$Y = X_6 + X_{12} + X_{15} + X_{20} + 0.7X_1\epsilon$$

where the random error $\epsilon \sim N(0, 1)$ is independent of the covariates. It is noteworthy that in this model, the τth quantile function is $X_6 + X_{12} + X_{15} + X_{20} + 0.7X_1\phi^{-1}(\tau)$, where $\phi^{-1}(\tau)$ denotes the τth conditional quantile of the standard normal distribution. Hence, X_1 does not influence the center of the conditional distribution, but plays an important role when considering other conditional quantiles.

In this example, we consider sample size $n = 300$, covariates dimension $p = 1000$ and three different quantiles $\tau = 0.3, 0.5, 0.7$. We use different tuning paramerter λ for different quantiles as follows.

```
> library(QICD)
> library(mvtnorm)
> set.seed(123)
> n <- 300
> p <- 1000
> Sigma=0.5^abs(outer(1:p,1:p,'-'))
> X=rmvnorm(n,mean=rep(0,p),sigma=Sigma)
> epsilon=rnorm(n)
> Y=X[,6]+X[,12]+X[,15]+X[,20]+0.7*pnorm(X[,1])*epsilon
> intercept<-1
> #include intercept
> beta1=rep(0,p+1)
> #initial value to be zero
> obj_tau3=QICD(Y,X,beta1,tau=0.3,lambda=9,funname="scad")
> obj_tau5=QICD(Y,X,beta1,tau=0.5,lambda=15,funname="scad")
> obj_tau7=QICD(Y,X,beta1,tau=0.7,lambda=8.5,funname="scad")
```

Then we can compare the coefficient estimates for different quantiles $\tau = 0.3, 0.5, 0.7$. The results, actually, are very close to the true parameter. Also,since X_1 does not influence the center of the conditional distribution, but plays an important role when considering other conditional quantiles. The coefficient for X_1 is zero for quantile $\tau = 0.5$ but none zero for other quantiles.

```
> res=data.frame(
+   V1=obj_tau3$beta_final[c(1,6,12,15,20)]
+   ,V2=obj_tau5$beta_final[c(1,6,12,15,20)]
+   ,V3=obj_tau7$beta_final[c(1,6,12,15,20)]
+ )
> colnames(res)=c("tau=0.3","tau=0.5","tau=0.7")
> rownames(res)=c(1,6,12,15,20)
> print(res,digits=6)

        tau=0.3  tau=0.5     tau=0.7
1  -9.76954e-05 0.000000 0.000114096
6   9.42517e-01 0.973327 0.832316420
12  8.96578e-01 0.987515 0.881342040
15  1.00279e+00 1.014146 1.044361246
20  1.00318e+00 1.029070 1.013159680
```

However, the tuning parameter λ is always unknow in reality. Cross-validation and High-dimensional BIC (HBIC) Lee,Noh and Park (2014) are used for tuning parameter selection. In practice, we prefer the HBIC since Cross-validation is time-consuming when p is notably large and may result in overfitting (see Wang (Li and Tsai)). For HBIC, let $\boldsymbol{\beta}_\lambda = (\beta_{\lambda,1}, \ldots, \beta_{\lambda,p})$ be the penalized estimator obtained with the tuning parameter λ; and let $\mathcal{S} \equiv \{j : \beta_{\lambda,j} \neq 0, 1 \leq j \leq p\}$ be the index set of covariates with nonzero coefficients. Define

$$\mathrm{HBIC}(\lambda) = \log\big(\sum_{i=1}^{n} \rho_\tau(Y_i - \mathbf{x}_i^T \boldsymbol{\beta}_\lambda)\big) + |\mathcal{S}_\lambda| \frac{\log(\log n)}{n} C_n,$$

where $|\mathcal{S}_\lambda|$ is the cardinality of the set $\mathcal{S}_\lambda$, and C_n is a sequence of positive constants diverging to infinity as n increases. We select the value of λ that minimizes $\mathrm{HBIC}(\lambda)$. In practice, we recommend to take $C_n = O(\log(p))$, which we find to work well in a variety of settings. However, the adjustment for C_n is still not easy in real application cases. A HBIC curve is displayed in

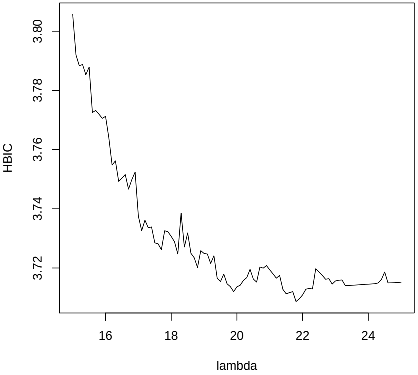

Figure 1: HBIC trends for $\tau = 0.5$

Figure 1. The best λ is around 22. Figure 2 presents the cross-validation results. This process is time-consuming, but the optimal λ seems close to the one selected by HBIC.

References

Fan, J. and Li, R. (2001). Variable Selection via Nonconcave Penalized Likelihood and its Oracle Propertie. *Journal of the American Statistical Association*, **96 (456)**, http://orfe.princeton.edu/~jqfan/papers/01/penlike.pdf

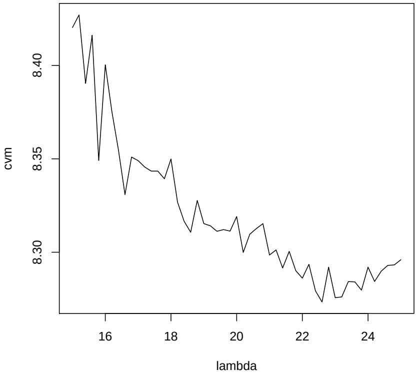

Figure 2: cross validation trends for $\tau = 0.5$

Lee,Noh and Park (2014). Model Selection via Bayesian Information Criterion for Quantile Regression Models. *Journal of the American Statistical Association*, **109 (505)**, `http://www.tandfonline.com/doi/abs/10.1080/01621459.2013.836975#.VF04ePldWeA`

Peng,B. and Wang, L. (2014). An Iterative Coordinate Descent Algorithm for High-dimensional Nonconvex Penalized Quantile Regression. *Journal of Computational and Graphical Statistics*, `http://users.stat.umn.edu/~wangx346/research/QICD.pdf`.

Wang, H., Li, R., and Tsai, C. L (2007). Tuning parameter selectors for the smoothly clipped absolute deviation method. *Biometrika*, **94 (3)**, `http://biomet.oxfordjournals.org/content/94/3/553.short`.

Zhang,C.H.(2010). Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, **38 (2)**, `http://projecteuclid.org/download/pdfview_1/euclid.aos/1266586618`.