Regularization and Variable Selection for Parametric Models (6)

February 8, 2012

```
> library(lqa)

> library(catdata)
> data(heart, package="catdata")
> X<-heart[,-1]
> y<-heart[,1]
> X.std<-scale(X)
> p<-ncol(X)
> n<-length(y)
> family <- binomial()
> n.fold<-10
> ylab.text<-""
> xlab.text<-""
> Width = 6
> Height = 6
> oma.vec<-c(1,1,1,3)
> size.axis=1.4
> size.lab=1.4
> size.main=1.4
> size.right=1.2
> size.width=2.0
> colour=1
> library(penalized)
```

Welcome to penalized. For extended examples, see vignette("penalized").

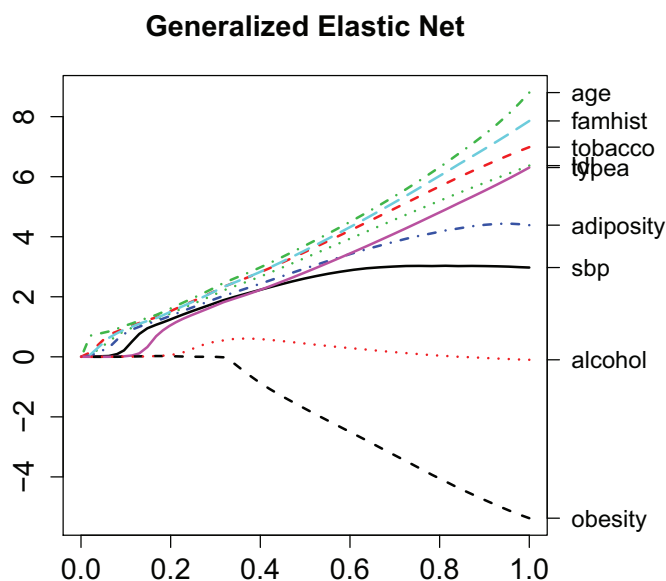GENERALIZED ELASTIC NET

```
> alpha <- 0.7
> g<-5
> ### COEF BUILD-UPS
>
> main.text<-"Generalized Elastic Net"
> penalty.family<-genet
> Plot.mat<-plot.lqa (y = y, x = X, family=family, penalty.family=penalty.family,
+ offset.values = c (NA, alpha,g),add.MLE = FALSE, ret.true=TRUE,really.plot = FALSE,
+ show.standardized=TRUE,gamma=0.01)
```

```
> par(oma=oma.vec,cex.axis=size.axis,cex.lab=size.lab,cex.main=size.main)
> matplot(Plot.mat$s1,Plot.mat$beta.mat,type="l",ylab=ylab.text,xlab=xlab.text,
+ main=main.text,lwd=size.width)
> axis(4, at = Plot.mat$beta.mat[1, ], labels = colnames(X), adj = 0, las = 1,
+ cex.axis=size.right)
```

Generalized Elastic Net

IMPROVED CORRELATION BASED
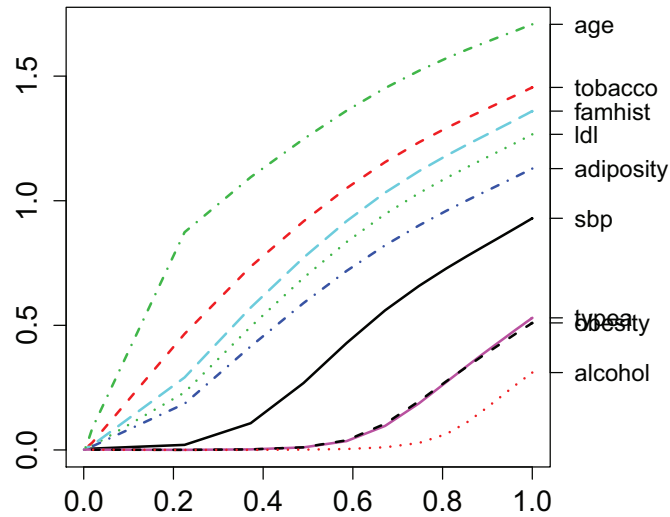
```
> lambda2<-0.1
> main.text<-"Improved Correlation Based"
> penalty.family<-icb
> Plot.mat<-plot.lqa (y = y, x = X, family=family, penalty.family=penalty.family,
+ offset.values = c (NA, lambda2),add.MLE = FALSE, ret.true=TRUE,really.plot = FALSE,
+ show.standardized=TRUE,gamma=0.01)

> par(oma=oma.vec,cex.axis=size.axis,cex.lab=size.lab,cex.main=size.main)
> matplot(Plot.mat$s1,Plot.mat$beta.mat,type="l",ylab=ylab.text,xlab=xlab.text,
+ main=main.text,lwd=size.width)
> axis(4, at = Plot.mat$beta.mat[1, ], labels = colnames(X), adj = 0, las = 1,
+ cex.axis=size.right)
```

Improved Correlation Based
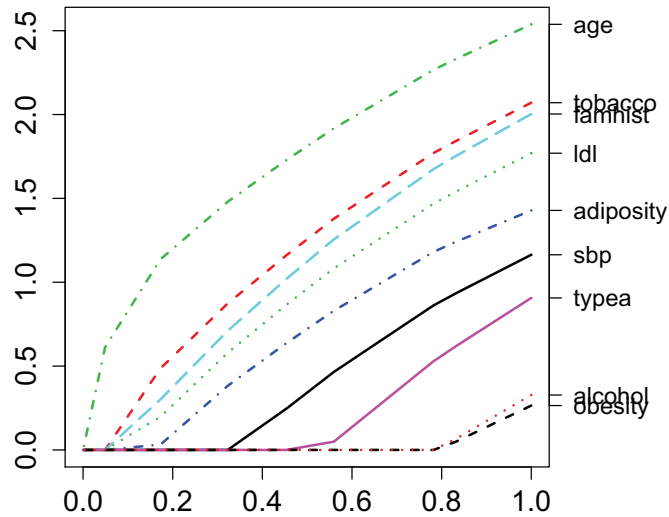


```
> Path<-matrix(0,60,p)
> lambda1<-exp(seq(-10, 6, length = 60))
> lambda2 = 1
> for(i in 60:1)
+ {
+ Path[i,]<-coef(penalized (y,penalized=X.std/sqrt(n), lambda1 = lambda1[i],
+ lambda2 = lambda2,model="logistic",standardize=FALSE),"penalized")
+ }

> par(oma=oma.vec,cex.axis=size.axis,cex.lab=size.axis,cex.main=size.main)
> matplot(rowSums(abs(Path))/max(rowSums(abs(Path))),Path,type="l",ylab=ylab.text,
+ xlab=xlab.text,main="Enet with penalized",lwd=size.width)
> axis(4, at = Path[1, ], labels = colnames(X), adj = 0, las = 1,cex.axis=size.right)
```

Enet with penalized

LASSO mit penalized

```
> Path<-matrix(0,60,p)
> lambda1<-exp(seq(-10, 6, length = 60))
> lambda2 = 0
> for(i in 60:1)
+ {
+ Path[i,]<-coef(penalized (y,penalized=X.std/sqrt(n), lambda1 = lambda1[i],
+ lambda2 = lambda2,model="logistic",standardize=FALSE),"penalized")
+ }

> par(oma=oma.vec,cex.axis=size.axis,cex.lab=size.axis,cex.main=size.main)
> matplot(rowSums(abs(Path))/max(rowSums(abs(Path))),Path,type="l",ylab=ylab.text,
+ xlab=xlab.text,main="Lasso with penalized",lwd=size.width)
> axis(4, at = Path[1, ], labels = colnames(X), adj = 0, las = 1,cex.axis=size.right)
```

Lasso with penalized