Comparison on Lasso and Relaxed Lasso Model

Mingyang Yan Jingyan Jiang

December 15, 2019

No safe wall between variable selection and model inference.

1 Prologue

It deserves to know that how variable selection process affects our subsequent statistical inference and why statistical inference in high-dimensional situations is very complicated. In statistical and econometric analysis, many times we go through a process of model selection: that is, we use data to decide whether some variables should be added to our model.

In the classic analysis, we believe that as long as our Variable Selection process is consistent, that is, when the sample is large enough, our method can select the correct variable, then the method is considered valid. We can regard the model we selected using the data we have and some specific variable selection methods (such as: AIC, BIC, LASSO, SCAD, etc.) as a correct model. By inferring the model at this new starting point, we can get our normal confidence interval in general. But things are often not so simple. We only have limited data, and no theory can guarantee that our model selection method can choose a real model on the limited data.

In this context, we compare the performance of Lasso and Relaxed Lasso, and we find that by running a Multi-stage Lasso, our results can be improved.

2 Lasso Model

2.1 Data Simulation

We design a "true model" in this section and use it to generate our data.

We have 110 variables, and normal error terms in a simple linear model. Ten of them are non-zero coefficients. For all the other β, the underlying

true values are 0.

$$x_1, \ldots, x_{110}, e \sim^{i.i.d.} \mathcal{N}(0,1)$$
$$y = \beta_1 x_1 + \cdots + \beta_{110} x_{110} + e$$

. where

$$\beta_1 = \cdots = \beta_{10} = 10$$
$$\beta_{11} = \cdots = \beta_{110} = 0$$

```
# True model
p = 110 # number of variables
s = 10 # number of non-zero coefficients
beta = c(rep(10,s),rep(0,p-s))

# Training Set
n = 100 # sample size
Xtr = list()
Ytr = list()
# Generate 100 train data sets
for (i in 1:100) {
  Xtr[[i]]<- matrix(rnorm(n*p),ncol=p);
  Ytr[[i]] = Xtr[[i]]%*%beta + rnorm(n)
}

# Test set
n = 1e5 # sample size
Xte = matrix(rnorm(n*p),ncol=p)
Yte = Xte%*%beta + rnorm(n)
```

2.2 Training Process

```
#########
# Lasso #
#########
require(glmnet)
betahat.lasso = matrix(0,nrow = 110)
sumerr.lasso = 0
for (j in 1:100) {
  cv.lasso = cv.glmnet(Xtr[[i]],Ytr[[i]],intercept=FALSE,alpha=1)
  lambda.star = cv.lasso$lambda.min
  fit.lasso = glmnet(Xtr[[i]],Ytr[[i]],intercept=FALSE,alpha=1,lambda=lambda.star)
  if (j == 1) {
    betahat.lasso = betahat.lasso + fit.lasso$beta
  } else {
    betahat.lasso = rbind(betahat.lasso, fit.lasso$beta)
  }
  pred.lasso = predict(fit.lasso,Xte)
  sumerr.lasso = sumerr.lasso + mean((Yte-pred.lasso)^2)
}

# test err
err.lasso = sumerr.lasso/100
```

We use "glmnet" package in R to train Lasso method on our dataset, and the test error we get is **1.7584**.

```
# number of non-zero estimated coefficients:
nncl = sum(betahat.lasso[,1]!=0)
```

The result of nncl (i.e. number of non-zero estimated coefficients form Lasso) is **1300**, and the true value is **1000**.

3 Relaxed Lasso

Then we try Relaxed Lasso and repeat the process in previous section.

```
#################
# Relaxed Lasso #
#################
require(relaxo)

betahat.relaxo = matrix(0,nrow = 110)
sumerr.relaxo = 0

for (k in 1:100) {
  fit.relaxo = cvrelaxo(Xtr[[i]],Ytr[[i]])
  if (k == 1) {
    betahat.relaxo = betahat.relaxo + t(fit.relaxo$beta)
  } else {
    betahat.relaxo = rbind(betahat.relaxo, t(fit.relaxo$beta))
  }
  pred.relaxo = predict(fit.relaxo,Xte)
  sumerr.relaxo = sumerr.relaxo + mean((Yte-pred.relaxo)^2)
}

# test err
err.relaxo = sumerr.relaxo/100
```

The error rate we get from relaxed lasso is **1.1961**, which is smaller than **1.7584** from Lasso.

```
# number of non-zero estimated coefficients:
nncr = sum(betahat.relaxo[,1]!=0)
```

The result of nncr (i.e. number of non-zero estimated coefficients form Relaxed Lasso) is **1153**, and the true value is **1000**.

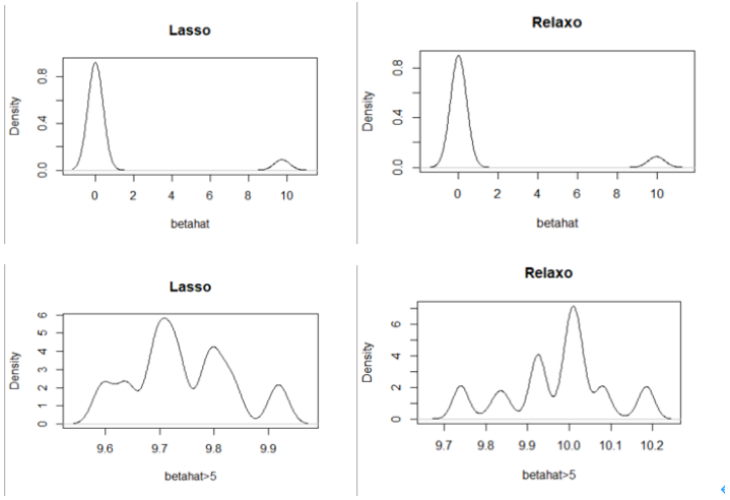
4 Results Comparison

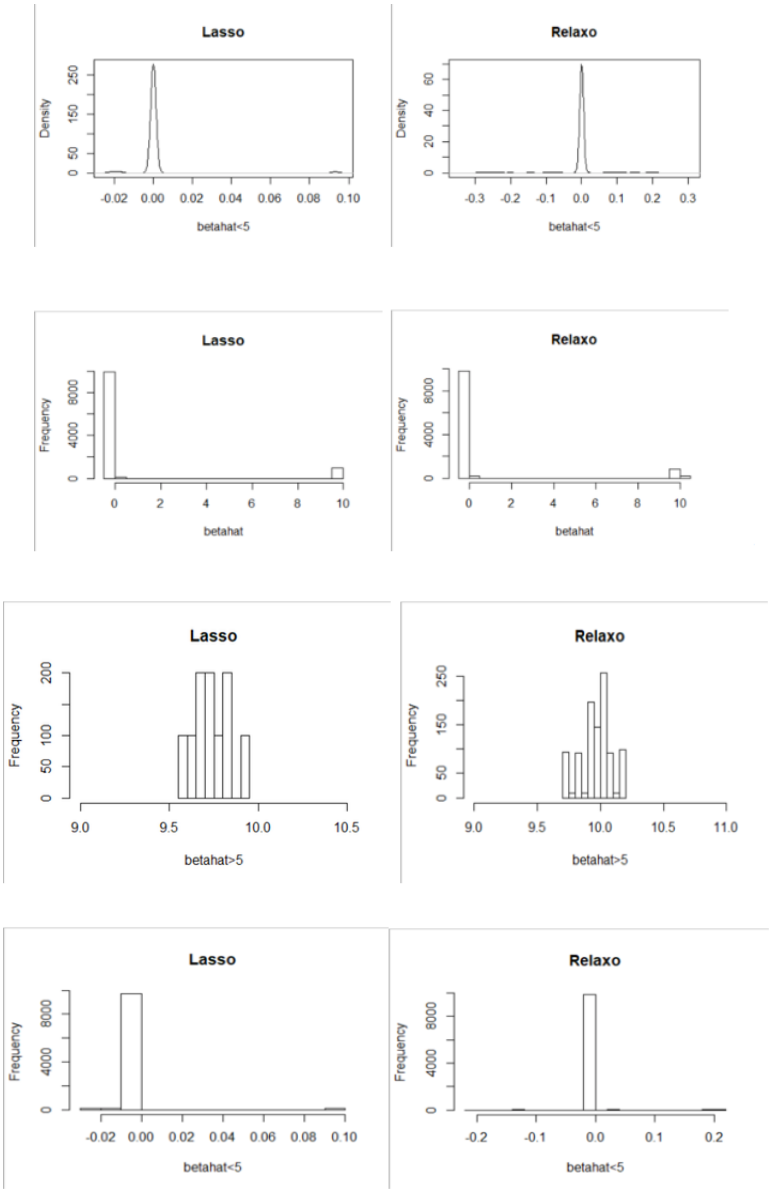
```
# distribution visualization
# density
plot(density(betahat.lasso[,1]),main = "Lasso", xlab="betahat")
plot(density(betahat.lasso[betahat.lasso[,1]>5, xlim = c(9,12)]),main = "Lasso", xlab="betahat>5")
plot(density(betahat.lasso[betahat.lasso[,1]<5]),main = "Lasso", xlab="betahat<5")
# histogram
hist(betahat.lasso[,1], main = "Lasso", xlab="betahat")
hist(betahat.lasso[betahat.lasso[,1]>5], xlim = c(9,10.5),main = "Lasso", xlab="betahat>5")
hist(betahat.lasso[betahat.lasso[,1]<5],main = "Lasso", xlab="betahat<5")
```

Plot distribution of Lasso

```
# distribution visualization
# density
plot(density(betahat.relaxo[,1]),main = "Relaxo", xlab="betahat")
plot(density(betahat.relaxo[betahat.relaxo[,1]>5]),main = "Relaxo", xlab="betahat>5")
plot(density(betahat.relaxo[betahat.relaxo[,1]<5]),main = "Relaxo", xlab="betahat<5")
# histogram
hist(betahat.relaxo[,1], main = "", xlab="betahat")
hist(betahat.relaxo[betahat.relaxo[,1]>5],xlim = c(9,11),main = "Relaxo", xlab="betahat>5")
hist(betahat.relaxo[betahat.relaxo[,1]<5],main = "Relaxo", xlab="betahat<5")
```

Plot distribution of Relaxed Lasso





4.1 Performance Evaluation

Since the estimation using Lasso would be affected more significantly by the noise data than that using Relaxed Lasso, the performance of Lasso is considered to be poorer than Relaxed Lasso theoretically, which is also confirmed by our simulation results. The averaged test error of Lasso is 1.7584, which is smaller than the averaged test error of Relaxed Lasso, 1.1961. Besides, fitting with the 100 training data sets, the sum of the number of non-zero estimated coefficients in the 100 models using Relaxed Lasso is 1153, which is more closed to the true value of 1000 than that using Lasso,

1300.

In addition, from the density diagram and histogram of Lasso and Relaxed Lasso when the estimated beta is larger than 5, which approximately represent the distribution of the non-zero estimated beta, we can see that the mode of $\hat{\beta}$ in Lasso model is around 9.7, while the mode of $\hat{\beta}$ in Relaxed Lasso is around the true value of 10.

To sum up, from the simulation results we obtained, we can conclude that the performance of Relaxed Lasso is better than Lasso in our settings.

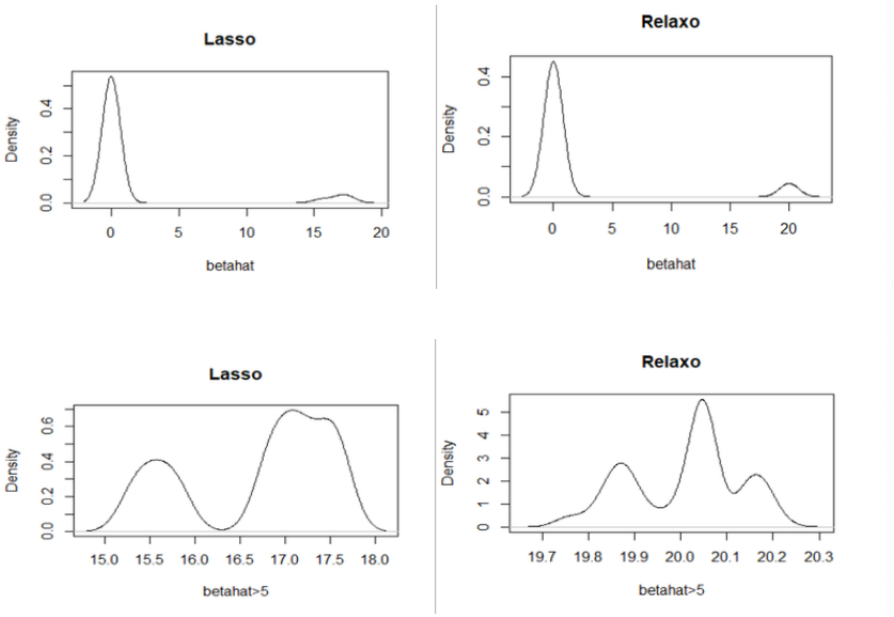
5 Change settings

5.1 Change the size of β

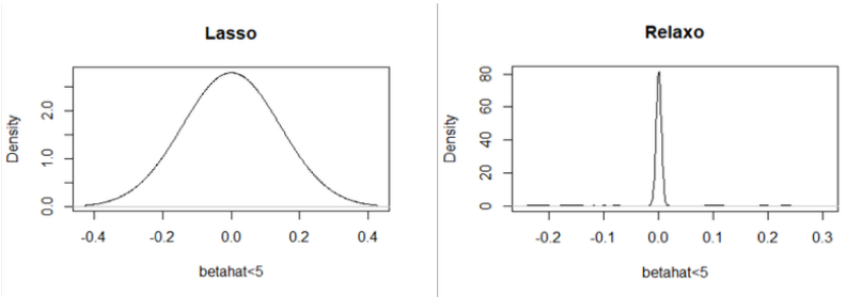
Now we let$\beta_1 = ... = \beta_{10} = 20, \beta_{11} = ... = \beta_{110} = 0$

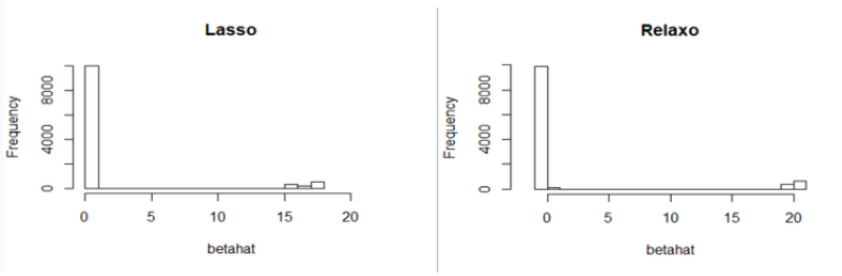
```
# True model
p = 110 # number of variables
s = 10 # number of non-zero coefficients, can be changed for different sparsities
beta = c(rep(20,s),rep(0,p-s)) # can be changed for different beta sizes

# Training Set
n = 100 # sample size
Xtr = list()
Ytr = list()
# Generate 100 training data sets
for (i in 1:100) {
  Xtr[[i]]<- matrix(rnorm(n*p),ncol=p);
  Ytr[[i]] = Xtr[[i]]%*%beta + rnorm(n)
}

# Test set
n = 1e5 # sample size
Xte = matrix(rnorm(n*p),ncol=p)
Yte = Xte%*%beta + rnorm(n)
```

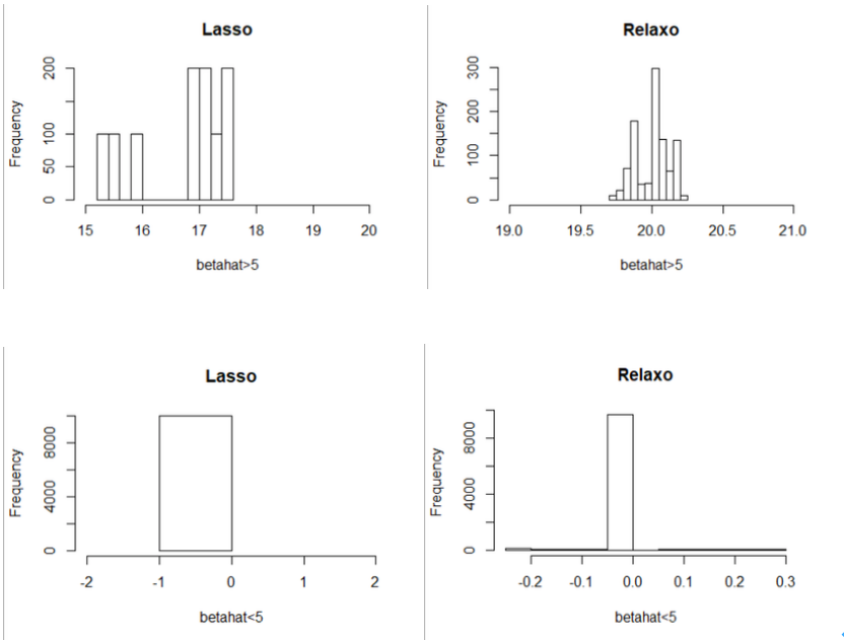
Other codes are similar with those in Section 1.
Now we compare the distribution.


Lasso
Density
0.0
0.2
0.4
0
5
10
15
20
betahat
Relaxo
Density
0.0
0.2
0.4
0
5
10
15
20
betahat

Lasso
Density
0.0
0.2
0.4
0.6
15.0
15.5
16.0
16.5
17.0
17.5
18.0
betahat>5
Relaxo
Density
0
1
2
3
4
5
19.7
19.8
19.9
20.0
20.1
20.2
20.3
betahat>5


Lasso
Density
0.0
1.0
2.0
-0.4
-0.2
0.0
0.2
0.4
betahat<5
Relaxo
Density
0
20
40
60
80
-0.2
-0.1
0.0
0.1
0.2
0.3
betahat<5


Lasso
Frequency
0
4000
8000
0
5
10
15
20
betahat
Relaxo
Frequency
0
4000
8000
0
5
10
15
20
betahat



5.1.1 Performance Evaluation

From the formula of Lasso regression, we can know that the estimated coefficients of Lasso tend to be shrunk too much due to the noise data thus affecting its accuracy. Intuitively, this over shrinkage effect would be more significant in terms of magnitude when the beta size increase. That is consistent with our simulation results increasing the beta size from 10 to 20. Although the sum of the number of non-zero estimated coefficients in the 100 models using Lasso, 1000, is more closed to the true value of 1000 than that of Relaxed Lasso, 1310, the averaged test error with the Lasso regression increases to be 115.0959, which is much larger than the 1.2404 averaged test error with Relaxed Lasso.

Besides, from the density diagram and histogram of Lasso and Relaxed Lasso when the estimated beta is larger than 5, which approximately represent the distribution of the non-zero estimated beta, we can see that the mode of $\hat{\beta}$ in Lasso model is around 17, while the mode of$\hat{\beta}$ in Relaxed Lasso is around the true value of 20.

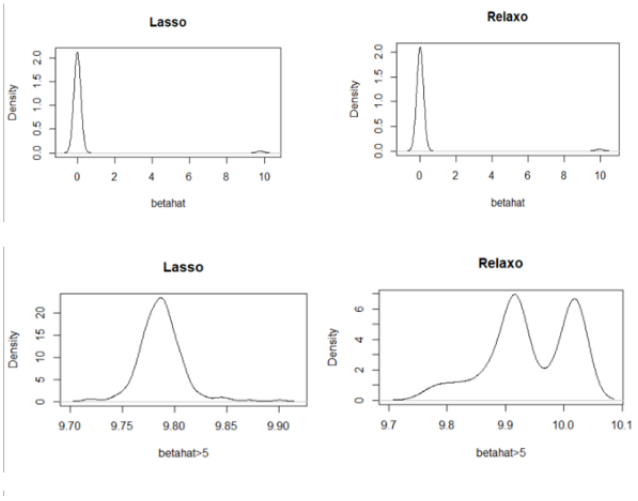
In this case, we can figure out that the performance of Relaxed Lasso is better than Lasso when the true beta size is relatively larger.

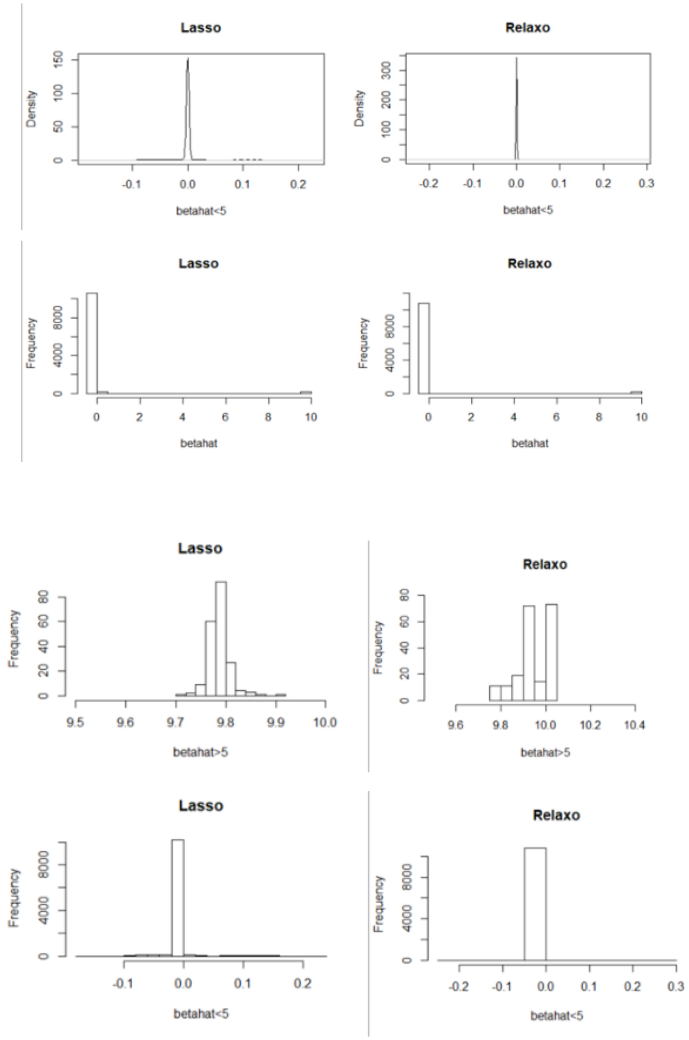
5.2 Change the sparsity of β

Now we let $\beta_1 = \beta_2 = 10, \beta_3 = ... = \beta_{110} = 0$

```
# True model
p = 110 # number of variables
s = 2 # number of non-zero coefficients, can be changed for different sparsities
beta = c(rep(10,s),rep(0,p-s)) # can be changed for different beta sizes

# Training Set
n = 100 # sample size
Xtr = list()
Ytr = list()
# Generate 100 training data sets
for (i in 1:100) {
  Xtr[[i]]<- matrix(rnorm(n*p),ncol=p);
  Ytr[[i]] = Xtr[[i]]%*%beta + rnorm(n)
}

# Test set
n = 1e5 # sample size
Xte = matrix(rnorm(n*p),ncol=p)
Yte = Xte%*%beta + rnorm(n)
```

Again, we compare the distribution.





5.2.1 Performance Evaluation

The over shrinkage effect we mentioned above might also be minor in terms of magnitude when the sparsity of beta increase based on the formula of Lasso regression. In this case, we decrease the number of non-zero true beta from 10 to 2 and have proved this hypothesis. The performance of Lasso does increase as its averaged test error is 1.1316, which is still larger but more closed to the averaged test error of Relaxed Lasso, 1.0162. And the sum of the number of non-zero estimated coefficients in the 100 models using Lasso, 210, is also more closed to the true value of 200 than that of Relaxed Lasso, 110.

Additionally, from the density diagram and histogram of Lasso and Relaxed Lasso when the estimated beta is larger than 5, which approximately

represent the distribution of the non-zero estimated beta, we can see that the mode of $\hat{\beta}$ in Lasso model is around 9.8. That is more closed to the true value 10 compared with the 9.7 in our first simulation, while the mode of $\hat{\beta}$ in Relaxed Lasso is still around 10.

In this case, we can find that the difference in the performance of Lasso and Relaxed Lasso is relatively smaller when the sparsity of true beta increase, but the Relaxed Lasso is still better than Lasso.

6 Acknowlegement

[1] Lee, J. D., D. L. Sun, Y. Sun, and J. E. Taylor. 2016. "Exact Post-selection Inference with Application to the Lasso," The Annals of Statistics, 44(3).
[2] Meinshausen, N. 2007. "Relaxed Lasso," Computational Statistics&Data Analysis,52(1).
[3] Taylor, J. and R. Tibshirani. 2017. "Post-selection inference for L1-penalized likelihood models," The Canadian Journal of Statistics, 46(1).
[4] Github, Jiaming Mao, 2019, https://jiamingmao.github.io/data-analysis/
[5] Wikipedia, https://en.wikipedia.org/wiki/Lasso