Title
Using simulation to evaluate confidence interval construction by thresholding the group lasso

Permalink
https://escholarship.org/uc/item/18b60696

Author
He, Jia

Publication Date
2015

Peer reviewed|Thesis/dissertation
Using simulation to evaluate confidence interval construction by thresholding the group lasso

A dissertation submitted in partial satisfaction
of the requirements for the degree
Master of Science in Statistics

by

Jia He

2015
Abstract of the Dissertation

Using simulation to evaluate confidence interval construction by thresholding the group lasso

by

Jia He

Master of Science in Statistics

University of California, Los Angeles, 2015

Professor Qing Zhou, Chair

In terms of constructing confidence intervals for coefficients in high-dimensional linear models under group norm regularization, the standard residual bootstrap method has been proven to be inconsistent and its performance is unstable especially for coefficients in active groups. In this thesis, we consider a thresholding method which forces the close-to-zero estimated coefficients to be exact zero. Through simulations, this method outperforms the standard method in terms of the confidence interval coverage rate. Instead of setting a hard threshold on \( \hat{\beta} \), we also consider a method which selects groups with high chance of being active in multiple simulated datasets. After thresholding, two other confidence interval building methods are proposed, and in this thesis, these three methods are compared in terms of coverage rate for both active groups and non-active groups under different conditions. Modified thresholding confidence interval construction method is most consistent and accurate method based on the simulation results.
The dissertation of Jia He is approved.

Nicolas Christou

Yingnian Wu

Qing Zhou, Committee Chair

University of California, Los Angeles
2015
To my mother and father...

who—among so many other things—
saw to it that I learned to touch-type
while I was still in elementary school
# Table of Contents

1 Introduction ................................................. 1

2 Residual bootstrap method ................................. 4

3 Threshold set up .............................................. 8

4 Thresholding confidence interval construction method .... 16

5 Modified thresholding confidence interval construction method 19
   5.1 Introduction to the method ............................. 19
   5.2 Between-group correlation coefficient = 0 ............... 21

6 Conclusion and Future Work ................................. 23

References ....................................................... 25
List of Figures

3.1 \(n=200, p=400, \rho=0.1\) .................................................. 9
3.2 \(n=500, p=1000, \rho=0.1\) .................................................. 10
3.3 \(n=1000, p=2000, \rho=0.1\) ............................................. 11
3.4 method2: \(n=200, p=400, \rho=0.1\) ................................. 12
3.5 method2: \(n=500, p=1000, \rho=0.1\) ................................. 13
3.6 method2: \(n=1000, p=2000, \rho=0.1\) ................................. 13
## List of Tables

2.1 residual bootstrap: n=200, p=400 .................................. 6
2.2 residual bootstrap: n=500, p=1000 ................................. 6
2.3 residual bootstrap: n=1000, p=2000 .............................. 7

4.1 thresholding method: n=200, p=400 ............................... 17
4.2 thresholding method: n=500, p=1000 ............................. 17
4.3 thresholding method: n=1000, p=2000 ........................... 18

5.1 thresholding method: n=200, p=400 ............................... 20
5.2 thresholding method: n=500, p=1000 ............................. 20
5.3 thresholding method: n=1000, p=2000 ........................... 20
5.4 thresholding method: ρ=0, n=200, p=400 ........................ 21
5.5 thresholding method: ρ=0, n=500, p=1000 ....................... 21
5.6 thresholding method: ρ=0, n=1000, p=2000 ...................... 22
CHAPTER 1

Introduction

Consider the group lasso model proposed by Yuan&Lin[7], it solves the optimization problem:

\[
\min_{\beta \in \mathbb{R}^p} (||y - \sum_{l=1}^{L} X_l \beta_l||_2^2 + \lambda \sum_{l=1}^{L} \sqrt{p_l}||\beta_l||_2)
\]

where \(p_l\) represents the size of the group, and different from the ordinary lasso, all of the variables within a certain group would either all be in the final model or none of them would be in the final model. Algorithm is proposed to solve group lasso combining with lasso (\(L_1\)) by Friedman,Hastie&Tibshirani[3]. In this thesis, I mainly use R package grpreg to solve group lasso problems.

In terms of group lasso model, we are not only interested in variable estimated coefficients, sometimes we are more interested in building confidence interval for each individual variable coefficient. However, it’s an non-linear parameter estimation procedure, it’s difficult to measure the variance for parameter estimates. Chatterjee & Lahiri[1] applied a residual bootstrap method to a lasso problem in order to construct confidence interval for each coefficient and they points out that regular residual bootstrap is consistent when certain mild regularity conditions are satisfied, and the approximation will converge weakly. Besides, it’s worthwhile to note that bootstrap might fail if one or more than one parameters for regression equal to one. To obtain more consistent result, they propose a modified bootstrap method which offers valid approximation to the distribution of lasso estimator even in cases where the standard bootstrap method fails. Inconsistency occurs when some of the parameters are zero, so the trick they use is to force estima-
tors that are close to zero to be exact zero. After choosing appropriate tuning parameters, modified bootstrap method provide a more robust and consistent approximation than the standard bootstrap method. Bootstrap replications could also help with variable selection, as demonstrated by Bach[2].

A perturbation resampling method is proposed by Minnier, Tian& Cai [5] to build confidence interval especially for finite samples, it provides a way to estimate both the covariance matrix and confidence regions. Method is proven to be more consistent through simulation experiments compared with standard bootstrap method.

For high-dimensional linear models, Zhang & Zhang [8] proposed another method to construct confidence interval for low-dimension parameters, and use simulation results to support their result. Javanmard & Montanari [4] proposed an algorithm for confidence interval construction and p values calculation mainly based on a ’de-biased’ version of regularized M-estimators, if certain assumptions are met, an asymptotically valid confidence intervals can be easily constructed. van de Geer et al [6] extended Zhang & Zhang’s [8] work from linear models to generalized linear models with convex loss functions, and their method is a more general one constructing confidence interval and p values for low-dimension parameters in a high-dimensional models.

In this thesis I apply the method proposed by Zhou [9] to threshold the Lasso-type estimator instead of applying the bootstrap directly, and this thesis is mainly focused on high-dimensional models, with a small number of observations n and a large number of parameters p. Besides, all the methods are evaluated based on simulations for different combinations of (n,p), number of active groups and so on.

The thesis is organized as follows: after defining the notations, we utilize simulations to evaluate the standard residual bootstrap for different combinations of (n,p), number of active groups and range of parameters, and the evaluation is based on the percentage that the true parameters fall into the constructed confi-
dence interval. In next section, we develop a simulation method to select active groups and force the coefficients of the rest groups to be exact zero. After the thresholding, we compare two methods of confidence interval construction: the first one uses group lasso regression estimator vector as the initial estimator vector while the second one uses linear regression estimator vector for the active groups. The thesis concludes a conclusion and future works in the end.
CHAPTER 2

Residual bootstrap method

Given Y, X and \( \lambda \), through group lasso algorithm, we can obtain \( \hat{\beta} \) as our coefficient estimator vector. Usually, using a bootstrap method, we can get a 95% confidence interval for each variable. Here we apply a residual bootstrap [10] &[11]:

\( \hat{\beta} \) is our lasso estimator, \( \hat{\beta}_j \) for jth variable. The residuals are defined as the following:

\[
e_i = y_i - \tilde{x}_i' \hat{\beta}, \quad i = 1, \ldots, n
\]

Then we center the residuals and obtain a set of centered residuals \( \{e_i - \bar{e} : i = 1, \ldots, n\} \), \( \bar{e} = \frac{\sum_{i=1}^{n} e_i}{n} \). Then we draw bootstrap sample from this set of centered residuals with replacement and get a set of new residuals \( \{e^*_i : i = 1, \ldots, n\} \). Based on the new residuals, we can form a \( y^* \) vector:

\[
y^*_i = \tilde{x}_i' \hat{\beta} + e^*_i, \quad i = 1, \ldots, n
\]

In each simulation, we repeat the same procedure and have a \( y^* \) response variable vector and X matrix, through group lasso algorithms, we can obtain lasso estimators. In one experiment, when we have enough simulations, based on the 2.5% and 97.5% quantiles, we can get a 95% confidence interval for each variable’s coefficient. If we do this \( N \) times, we would expect that 0.95*\( N \) times the true coefficients would fall into the constructed confidence interval. However, it’s time consuming and inefficient to apply those algorithms hundreds of times[9], what’s more, a residual bootstrap could be inconsistent[1], especially the inconsistency of the standard residual bootstrap arises when some components of \( \beta \) are zero.
In this chapter, we build confidence intervals using the regular residual bootstrap. Before going into details of this experiment, we need to specify the meaning of some letters which will be extensively used in this thesis:

- **n**: number of observations
- **p**: number of variables
- **m**: number of simulations in one experiment
- **p_group**: number of variables in one group
- **q**: number of active groups
- **ρ**: the correlation coefficients between each group
- **p_range**: the range of active variable coefficients

In this thesis, we always set **p_group** to be 10 which means each group consists of 10 variables.

For standard bootstrap method, we are going to have three combinations of (n,p): (n=200,p=400), (n=500,p=1000), (n=1000,p=2000). And for each combination, we set ρ=0.1 which means groups are weakly correlated. **p_range** to be either (0.5,1) or (1,5), with q to be either 5 or 10. So far, for each combination of (n,p), we have four segments, and we are going to run m=400 simulations for each segment, so we can build one confidence interval for each coefficient based on m times simulations i.e. one experiment.

In this thesis, we are going to use the following metrics to measure the confidence interval construction methods:

- **p_active**: the percentage of the true active variables’ coefficients that fall into the constructed confidence interval.
- **sd_active**: standard deviation of **p_active** in all experiments.
- **p_non_active**: the percentage of the true non-active variables’ coefficients that fall...
into the constructed confidence interval.

sd_non_active: standard deviation of p_non_active in all experiments.
p_overall: the percentage of the true variables’ coefficients that fall into the constructed confidence interval.

20 experiments are set for each segment and this applies to other methods. Since each simulation is the same process, parallelism is used for speed purpose. We have the following 3 tables for each combination of (n,p).

<table>
<thead>
<tr>
<th>p_range,q</th>
<th>p_active</th>
<th>sd_active</th>
<th>p_non_active</th>
<th>sd_non_active</th>
<th>p_overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>p_range=(0.5,1), q=5</td>
<td>0.606</td>
<td>0.071</td>
<td>0.999</td>
<td>0.004</td>
<td>0.950</td>
</tr>
<tr>
<td>p_range=(1,5), q=5</td>
<td>0.494</td>
<td>0.045</td>
<td>1.000</td>
<td>0.000</td>
<td>0.937</td>
</tr>
<tr>
<td>p_range=(0.5,1), q=10</td>
<td>0.613</td>
<td>0.053</td>
<td>0.990</td>
<td>0.019</td>
<td>0.895</td>
</tr>
<tr>
<td>p_range=(1,5), q=10</td>
<td>0.522</td>
<td>0.054</td>
<td>0.990</td>
<td>0.021</td>
<td>0.873</td>
</tr>
</tbody>
</table>

Table 2.1: residual bootstrap: n=200, p=400

<table>
<thead>
<tr>
<th>p_range,q</th>
<th>p_active</th>
<th>sd_active</th>
<th>p_non_active</th>
<th>sd_non_active</th>
<th>p_overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>p_range=(0.5,1), q=5</td>
<td>0.544</td>
<td>0.051</td>
<td>1.000</td>
<td>0.000</td>
<td>0.977</td>
</tr>
<tr>
<td>p_range=(1,5), q=5</td>
<td>0.366</td>
<td>0.059</td>
<td>1.000</td>
<td>0.000</td>
<td>0.968</td>
</tr>
<tr>
<td>p_range=(0.5,1), q=10</td>
<td>0.583</td>
<td>0.043</td>
<td>1.000</td>
<td>0.000</td>
<td>0.958</td>
</tr>
<tr>
<td>p_range=(1,5), q=10</td>
<td>0.446</td>
<td>0.035</td>
<td>1.000</td>
<td>0.000</td>
<td>0.945</td>
</tr>
</tbody>
</table>

Table 2.2: residual bootstrap: n=500, p=1000

As we can see from the table, for those inactive groups, their true coefficients are all 0, and almost all the confidence interval covers 0. So the bootstrap method works good for inactive groups. Nonetheless, for active groups, the percentage
<table>
<thead>
<tr>
<th>p_range, q</th>
<th>p_active</th>
<th>sd_active</th>
<th>p_non_active</th>
<th>sd_non_active</th>
<th>p_overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>range=(0.5,1), q=5</td>
<td>0.400</td>
<td>0.045</td>
<td>1.000</td>
<td>0.000</td>
<td>0.985</td>
</tr>
<tr>
<td>range=(1,5), q=5</td>
<td>0.312</td>
<td>0.049</td>
<td>1.000</td>
<td>0.000</td>
<td>0.983</td>
</tr>
<tr>
<td>range=(0.5,1), q=10</td>
<td>0.494</td>
<td>0.032</td>
<td>1.000</td>
<td>0.000</td>
<td>0.975</td>
</tr>
<tr>
<td>range=(1,5), q=10</td>
<td>0.339</td>
<td>0.041</td>
<td>1.000</td>
<td>0.000</td>
<td>0.967</td>
</tr>
</tbody>
</table>

Table 2.3: residual bootstrap: n=1000, p=2000

is far below 0.95, and as n, p increases, the percentage decreases which indicates bootstrap is not consistent in confidence interval construction. Besides, we find out the number of active groups has no significant influence on percentage, and the range of coefficients matters, the larger the coefficients, the lower the percentage. Large coefficient estimation might lead to large variation.
A modified bootstrap method was proposed by A. Chatterjee and S. N. Lahiri [1], and was showed that it can be used to consistently estimate the distribution and variance of the Adaptive Lasso estimator. Its main idea is to set a threshold to force the coefficients near 0 to be exact 0, the idea of confidence interval construction method we applied in this thesis is similar.
CHAPTER 3

Threshold set up

3.1 Threshold set up

The key is to set a threshold that can separate the zero coefficients and non-zero coefficients, since the estimators of the zero variables’ coefficients are usually closer to 0 than the estimators of the active variables’ coefficients. However, in reality, coefficients of the active variables could be small, leading to small estimators, so it’s not easy to set a hard threshold and select variables whose estimators are larger than that.

Here we apply a simulation method to help select active variables. Given X matrix and Y, first, we use cross validation to obtain a $\lambda$, and through group lasso algorithm, we can obtain the lasso coefficient estimators $\hat{\beta}$.

$$\hat{\beta} = \min_{\beta} \left( \frac{1}{2} \| y - \sum_{l=1}^{p_{\text{group}}} X^{(l)} \beta^{(l)} \|_2^2 + \lambda \sum_{l=1}^{p_{\text{group}}} \sqrt{p_l} \| \beta^{(l)} \|_2 \right)$$

where $X^{(l)}$ is the submatrix of X columns to components in group l, $p_l$ is the number of components in group l and $\beta^{(l)}$ is the component vector of group l[3].

For jth simulation, we simulated an $\epsilon_j$ as new error vector, in this case, each item variable in $\epsilon_j$ follows a normal distribution N(0,1), so we simulate from N(0,1) n times and form an $\tilde{\epsilon}_j$. Then we obtain our new response variable $Y_j$:

$$Y_j = X\hat{\beta} + \tilde{\epsilon}_j$$
Then, for the same $\lambda$, X and $Y_j$, through lasso regression, we obtain a new estimator of $\beta$: $\hat{\beta}_{m,j}$(jth estimator in m simulations). For each $\hat{\beta}_i$ in $\hat{\beta}$ vector, we have a set consisting of m estimators $\{\hat{\beta}_{im,j} : j = 1, \ldots, m\}$ so that we can calculate the percentage of the estimators that are active (the percentage of the estimators not equal to 0). The higher the percentage, the more likely this variable is an non-zero variable.

We have the first q groups to be non-active with the between group correlation coefficients $\rho$ to be 0.1, and $\epsilon$ is from $N(0,1)$. We expect the percentage of the first m groups to be significantly higher than the rest of the groups, so that this method could be applied to active group selection. We test it for different combination of (n,p), different range of active coefficients and different q.

(a) n=200, p=400, q=5, range(0.5,1)  
(b) n=200, p=400, q=5, range(1,5)  
(c) n=200, p=400, q=10, range(0.5,1)  
(d) n=200, p=400, q=10, range(1,5) 

Figure 3.1: n=200, p=400, $\rho=0.1$
From the plot 3.1 to 3.3, we can find out that the performance of the active groups is very consistent: the percentage of the true active groups is always 1.00 regardless of the n, p, p group etc. While for inactive groups, the performance varies. When n is large enough, the percentage difference is very large, and that’s because we have more data, estimation is more accurate and variance decreases accordingly. If we set the cutoff percentage to be 0.90, then in most cases, we will have a very high accuracy predicting active groups.

When n is small, especially when the magnitude of true parameters is small (between 0.5 and 1 in this case), chances are that percentage of certain inactive group will be larger than 0.9 which will lead to misclassification of active groups, luckily, the error of predicting an inactive group to be active is acceptable, because the estimated coefficient is small, and won’t have a great influence on the predicted response.
There is another method for active group selection. The idea is similar to the previous one, we also apply a simulation method to select active groups. In each simulation, instead of randomly generating error term, we randomly pick half of the X observations from X matrix without replacement, along with the corresponding Y. We can train the group lasso model and obtain the estimated coefficients for each group. After m simulations, we obtain a percentage of true coefficient coverage for each group. This method proves to be more reliable than the previous one when dealing with small sparse dataset because this way of drawing samples will decrease the variance and simply by increasing the number of simulations m, we will get more stable result.

The simulation tests are shown below:

When n=200, p=400, this is the case with most variation, and percentages of
the all the non-active groups are all below 0.75, so we call well separate active
groups and non-active groups by choosing the thresholding percentage to be 0.9.
The most separable segment is when q=5 and range is (1,5). The estimation is
more precise when we have less non-zero components in the model. And larger
coefficients will make active groups more distinct.

When n=500, p=1000, percentages from all non-active groups are below 0.5,
and it’s more consistent and robust compared to the previous method.

When dataset is large enough, the variance of the first method is smaller,
onetheless, there is no big difference between those two thresholding methods,
they both well classify the active groups and inactive groups. It is recommended
to use the second method if the dataset is not large enough, and it’s safe to set
the threshold percentage to be 0.9. In the following parts of the thesis, we will
use 0.9 as our threshold percentage.
Figure 3.5: method2: n=500, p=1000, \( \rho = 0.1 \)

Figure 3.6: method2: n=1000, p=2000, \( \rho = 0.1 \)
3.2 Modified residual bootstrap method

We will apply the thresholding to residual bootstrap method, and we obtain a set of lasso estimator $\{\tilde{\beta}_i : i = 1, \ldots, p\}$

$$\tilde{\beta}_i = \hat{\beta}_i I(percentage(\hat{\beta}_i) > 0.9)$$

where $\hat{\beta}$ is the regular Lasso coefficient estimator, and as the size of the data $n$ approximates infinity, for a nonzero component $\beta_i$, its corresponding estimate $\tilde{\beta}_i$ will equal to $\beta_i$ with high probability, and for a zero component $\beta_i$, the estimate $\tilde{\beta}_i$ will equal to 0 with high probability. Our main goal to set the thresholding is to obtain the indices of zero components with probability tending to 1 as $n$ increases. Then, we can get a list of residuals $\{e_i : i = 1, \ldots, n\}$

$$e_i = y_i - \bar{x}_i' \tilde{\beta}, \quad i = 1, \ldots, n$$

Next we center the residuals and obtain a set of centered residuals $\{e_i - \bar{e} : i = 1, \ldots, n\}$, $\bar{e} = \frac{\sum_{i=1}^{n} e_i}{n}$. Then we draw bootstrap sample from this set of centered residuals with replacement and get a set of new residuals $\{e_i^{**} : i = 1, \ldots, n\}$. Based on the new residuals, we can form a $y^{**}$ vector:

$$y_i^{**} = \bar{x}_i' \tilde{\beta} + e_i^{**}, \quad i = 1, \ldots, n$$

In each simulation, through group lasso algorithm, we can obtain the lasso estimator for each component. After $m$ simulations, a confidence interval can be constructed for each component. We also test it for different combinations of $(n,p)$, different $q$ and different $p_{range}$. However, when compared the simulation result with that of the regular residual bootstrap method, there is no significant improvement on $p_{active}$, and $p_{nonactive}$ is high for both two methods. What’s more, when $n$, $p$ increases, $p_{active}$ decreases which means increasing the size of the dataset won’t lead to more accurate prediction result. So far, residual bootstrap is good at variable selection but it doesn’t seem to be an ideal method to
construct confidence interval for non-zero components.
CHAPTER 4

Thresholding confidence interval construction
method

After thresholding, we obtain \{\tilde{\beta}_i : i = 1, \ldots, p\}:

\[ \tilde{\beta}_i = \hat{\beta}_i I(\text{percentage}(\hat{\beta}_i) > 0.9) \]

Which force the coefficients close to 0 to be exact 0. Next, we start resampling, in each simulation, we draw \( \epsilon^* \) from N(0, I), and then we construct \( y^* \):

\[ y^*_i = x_i^T \tilde{\beta} + \epsilon^*_i : i = 1, \ldots, n \]

Set \( y^* \) as response variable, \( x \) as explanatory variable matrix, and \( \lambda \) (we use the \( \lambda \) chosen by cross validation at the beginning), by using group lasso algorithm, we can obtain a coefficient estimator vector: \( \hat{\beta}^* \).

In one experiment, we have \( m \) simulations, so we have set of estimated \( \{\tilde{\beta}^*_j : j = 1, \ldots, m\} \). Our goal is to construct a confidence interval for \( \beta_0 \) based on the sample mean and sample standard deviation. Sample mean is \( \hat{\beta} \), thus the 95% confidence interval is

\[ (\hat{\beta} - q_{0.025}, \hat{\beta} + q_{0.975}) \]

where \( q_{0.025} \) and \( q_{0.975} \) are the 0.025th and 0.975th quantiles. The assumption is that the standard deviation of \( \hat{\beta} \) would be close to the standard deviation of \( \tilde{\beta} \) when \( n \) is large enough. We take the difference vector set into consideration \( \{\tilde{\beta}^*_j - \hat{\beta} : j = 1, \ldots, m\} \), so the quantiles constructed by the difference vector set would be a good estimation of the quantiles of \( \{\hat{\beta}_j - \beta_0 : j = 1, \ldots, m\} \).
For each $\beta_i, i = 1, \ldots, p$, find the corresponding quantiles of the difference vector set, and then build confidence interval for $\beta_0$.

We are going to have the same experiments as before, and 3 tables for each combination of $(n,p)$ are listed below:

<table>
<thead>
<tr>
<th>$p_{\text{range}}$, $q$</th>
<th>$p_{\text{active}}$</th>
<th>sd_active</th>
<th>$p_{\text{non active}}$</th>
<th>sd_non_active</th>
<th>$p_{\text{overall}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_{\text{range}}=(0.5,1)$, $q=5$</td>
<td>0.861</td>
<td>0.065</td>
<td>0.965</td>
<td>0.018</td>
<td>0.952</td>
</tr>
<tr>
<td>$p_{\text{range}}=(1,5)$, $q=5$</td>
<td>0.783</td>
<td>0.077</td>
<td>0.992</td>
<td>0.014</td>
<td>0.966</td>
</tr>
<tr>
<td>$p_{\text{range}}=(0.5,1)$, $q=10$</td>
<td>0.739</td>
<td>0.041</td>
<td>0.949</td>
<td>0.037</td>
<td>0.896</td>
</tr>
<tr>
<td>$p_{\text{range}}=(1,5)$, $q=10$</td>
<td>0.300</td>
<td>0.040</td>
<td>0.911</td>
<td>0.056</td>
<td>0.758</td>
</tr>
</tbody>
</table>

Table 4.1: thresholding method: $n=200$, $p=400$

<table>
<thead>
<tr>
<th>$p_{\text{range}}$, $q$</th>
<th>$p_{\text{active}}$</th>
<th>sd_active</th>
<th>$p_{\text{non active}}$</th>
<th>sd_non_active</th>
<th>$p_{\text{overall}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_{\text{range}}=(0.5,1)$, $q=5$</td>
<td>0.914</td>
<td>0.031</td>
<td>0.980</td>
<td>0.020</td>
<td>0.977</td>
</tr>
<tr>
<td>$p_{\text{range}}=(1,5)$, $q=5$</td>
<td>0.862</td>
<td>0.051</td>
<td>1.000</td>
<td>0.000</td>
<td>0.993</td>
</tr>
<tr>
<td>$p_{\text{range}}=(0.5,1)$, $q=10$</td>
<td>0.845</td>
<td>0.042</td>
<td>0.989</td>
<td>0.011</td>
<td>0.975</td>
</tr>
<tr>
<td>$p_{\text{range}}=(1,5)$, $q=10$</td>
<td>0.460</td>
<td>0.067</td>
<td>0.997</td>
<td>0.005</td>
<td>0.944</td>
</tr>
</tbody>
</table>

Table 4.2: thresholding method: $n=500$, $p=1000$

From the tables 4.1, 4.2 and 4.3, we observe that the number of active groups is a factor that has a negative effect on active group’s prediction accuracy, which means when we have more non-zero coefficients to estimate, $p_{\text{active}}$ is lower and further from 95%.

After the thresholding, nearly all the coefficient estimators from inactive groups are forced to be 0s, so the $p_{\text{non active}}$ is consistently close to 1 when $n$ is large enough.
<table>
<thead>
<tr>
<th>p_{\text{range},q}</th>
<th>p_{\text{active}}</th>
<th>sd_{\text{active}}</th>
<th>p_{\text{non-active}}</th>
<th>sd_{\text{non-active}}</th>
<th>p_{\text{overall}}</th>
</tr>
</thead>
<tbody>
<tr>
<td>p_{\text{range}}=(0.5,1), q=5</td>
<td>0.866</td>
<td>0.040</td>
<td>0.998</td>
<td>0.003</td>
<td>0.995</td>
</tr>
<tr>
<td>p_{\text{range}}=(1,5), q=5</td>
<td>0.862</td>
<td>0.047</td>
<td>1.000</td>
<td>0.000</td>
<td>0.997</td>
</tr>
<tr>
<td>p_{\text{range}}=(0.5,1), q=10</td>
<td>0.859</td>
<td>0.041</td>
<td>0.998</td>
<td>0.003</td>
<td>0.991</td>
</tr>
<tr>
<td>p_{\text{range}}=(1,5), q=10</td>
<td>0.845</td>
<td>0.042</td>
<td>0.989</td>
<td>0.011</td>
<td>0.975</td>
</tr>
</tbody>
</table>

Table 4.3: thresholding method: n=1000, p=2000

When n, p increase, the p_{\text{active}} approaches 95%, however, it’s still not very close, and in some cases, the performance is unstable. Besides, if the absolute values of non-zero group parameters are larger (range from (1,5) in this thesis), then their corresponding p_{\text{active}} is lower. This is because the \tilde{\beta} is a biased estimator of \beta_0, and generally, the bias is larger when the coefficients are larger, which leads to inaccurate p_{\text{active}}. We are going to modify the bias in the following method.
CHAPTER 5

Modified thresholding confidence interval construction method

5.1 Introduction to the method

To adjust the bias of the previous method, we apply another thresholding method: after thresholding, we obtain a \( \{ \tilde{\beta}_i : i = 1, \ldots, p \} \). From \( \tilde{\beta}_i \), we get a variable set for active groups:

\[
A = \{ \tilde{\beta}_i : \tilde{\beta}_i \neq 0, i = 1, \ldots, p \}
\]

In order to get unbiased estimator, fit a ordinary least square for variables in \( A \):

\[
Y \sim X_{[A]} \tilde{\beta}^{OLS}_{[A]} + \epsilon, \quad \epsilon \sim N(0, I)
\]

And for variables not in \( A \),

\[
\tilde{\beta}^{OLS}_{[-A]} = \bar{0}
\]

In each simulation, we follow the same procedure in 2.2, however, we replace \( \tilde{\beta} \) with \( \tilde{\beta}^{OLS} \). The estimator is expected be robust and provide more accurate result. Do the same experiments and the tables are displayed below:
<table>
<thead>
<tr>
<th>p_range,q</th>
<th>p_active</th>
<th>sd_active</th>
<th>p_non_active</th>
<th>sd_non_active</th>
<th>p_overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>p_range=(0.5,1), q=5</td>
<td>0.871</td>
<td>0.063</td>
<td>0.994</td>
<td>0.011</td>
<td>0.979</td>
</tr>
<tr>
<td>p_range=(1,5), q=5</td>
<td>0.856</td>
<td>0.067</td>
<td>0.966</td>
<td>0.019</td>
<td>0.953</td>
</tr>
<tr>
<td>p_range=(0.5,1), q=10</td>
<td>0.708</td>
<td>0.058</td>
<td>0.984</td>
<td>0.017</td>
<td>0.915</td>
</tr>
<tr>
<td>p_range=(1,5), q=10</td>
<td>0.697</td>
<td>0.055</td>
<td>0.997</td>
<td>0.008</td>
<td>0.922</td>
</tr>
</tbody>
</table>

Table 5.1: thresholding method: n=200, p=400

<table>
<thead>
<tr>
<th>p_range,q</th>
<th>p_active</th>
<th>sd_active</th>
<th>p_non_active</th>
<th>sd_non_active</th>
<th>p_overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>p_range=(0.5,1), q=5</td>
<td>0.915</td>
<td>0.033</td>
<td>0.980</td>
<td>0.020</td>
<td>0.977</td>
</tr>
<tr>
<td>p_range=(1,5), q=5</td>
<td>0.913</td>
<td>0.033</td>
<td>1.000</td>
<td>0.000</td>
<td>0.996</td>
</tr>
<tr>
<td>p_range=(0.5,1), q=10</td>
<td>0.838</td>
<td>0.035</td>
<td>0.990</td>
<td>0.0113</td>
<td>0.975</td>
</tr>
<tr>
<td>p_range=(1,5), q=10</td>
<td>0.841</td>
<td>0.034</td>
<td>1.000</td>
<td>0.001</td>
<td>0.984</td>
</tr>
</tbody>
</table>

Table 5.2: thresholding method: n=500, p=1000

<table>
<thead>
<tr>
<th>p_range,q</th>
<th>p_active</th>
<th>sd_active</th>
<th>p_non_active</th>
<th>sd_non_active</th>
<th>p_overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>p_range=(0.5,1), q=5</td>
<td>0.912</td>
<td>0.032</td>
<td>0.999</td>
<td>0.002</td>
<td>0.997</td>
</tr>
<tr>
<td>p_range=(1,5), q=5</td>
<td>0.906</td>
<td>0.031</td>
<td>1.000</td>
<td>0.000</td>
<td>0.998</td>
</tr>
<tr>
<td>p_range=(0.5,1), q=10</td>
<td>0.871</td>
<td>0.041</td>
<td>0.998</td>
<td>0.003</td>
<td>0.992</td>
</tr>
<tr>
<td>p_range=(1,5), q=10</td>
<td>0.869</td>
<td>0.035</td>
<td>1.000</td>
<td>0.000</td>
<td>0.995</td>
</tr>
</tbody>
</table>

Table 5.3: thresholding method: n=1000, p=2000
5.2 Between-group correlation coefficient = 0

In previous experiments, we set the between group covariance coefficient to be 0.1, here we study how correlation coefficient will influence the $p_{\text{active}}$ in our modified thresholding method.

If the groups are uncorrelated, the experiment result is shown below. the tables show that $p_{\text{active}}$ is much closer to 0.95 when the between-group covariance coefficient equals to 0. When we have smaller between-group covariance coefficient, $p_{\text{active}}$ will converge faster to 0.95.

<table>
<thead>
<tr>
<th>p_range,q</th>
<th>p_active</th>
<th>sd_active</th>
<th>p_non_active</th>
<th>sd_non_active</th>
<th>p_overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>p_range=(0.5,1), q=5</td>
<td>0.914</td>
<td>0.036</td>
<td>0.936</td>
<td>0.019</td>
<td>0.933</td>
</tr>
<tr>
<td>p_range=(1,5), q=5</td>
<td>0.924</td>
<td>0.035</td>
<td>0.997</td>
<td>0.008</td>
<td>0.988</td>
</tr>
<tr>
<td>p_range=(0.5,1), q=10</td>
<td>0.839</td>
<td>0.036</td>
<td>0.936</td>
<td>0.019</td>
<td>0.933</td>
</tr>
<tr>
<td>p_range=(1,5), q=10</td>
<td>0.841</td>
<td>0.049</td>
<td>0.982</td>
<td>0.019</td>
<td>0.947</td>
</tr>
</tbody>
</table>

Table 5.4: thresholding method: $\rho=0$, n=200, p=400

<table>
<thead>
<tr>
<th>p_range,q</th>
<th>p_active</th>
<th>sd_active</th>
<th>p_non_active</th>
<th>sd_non_active</th>
<th>p_overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>p_range=(0.5,1), q=5</td>
<td>0.926</td>
<td>0.040</td>
<td>0.945</td>
<td>0.022</td>
<td>0.944</td>
</tr>
<tr>
<td>p_range=(1,5), q=5</td>
<td>0.931</td>
<td>0.038</td>
<td>1.000</td>
<td>0.000</td>
<td>0.997</td>
</tr>
<tr>
<td>p_range=(0.5,1), q=10</td>
<td>0.921</td>
<td>0.031</td>
<td>0.954</td>
<td>0.015</td>
<td>0.951</td>
</tr>
<tr>
<td>p_range=(1,5), q=10</td>
<td>0.905</td>
<td>0.037</td>
<td>1.000</td>
<td>0.000</td>
<td>0.990</td>
</tr>
</tbody>
</table>

Table 5.5: thresholding method: $\rho=0$, n=500, p=1000
<table>
<thead>
<tr>
<th>p_range,q</th>
<th>p_active</th>
<th>sd_active</th>
<th>p_non_active</th>
<th>sd_non_active</th>
<th>p_overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>p_range=(0.5,1), q=5</td>
<td>0.935</td>
<td>0.028</td>
<td>0.967</td>
<td>0.017</td>
<td>0.966</td>
</tr>
<tr>
<td>p_range=(1,5), q=5</td>
<td>0.930</td>
<td>0.027</td>
<td>1.000</td>
<td>0.000</td>
<td>0.998</td>
</tr>
<tr>
<td>p_range=(0.5,1), q=10</td>
<td>0.925</td>
<td>0.026</td>
<td>0.982</td>
<td>0.013</td>
<td>0.979</td>
</tr>
<tr>
<td>p_range=(1,5), q=10</td>
<td>0.913</td>
<td>0.027</td>
<td>1.000</td>
<td>0.000</td>
<td>0.996</td>
</tr>
</tbody>
</table>

Table 5.6: thresholding method; $\rho=0$, $n=1000$, $p=2000$
CHAPTER 6

Conclusion and Future Work

In this thesis, we first presented the traditional residual bootstrap method to build confidence interval for lasso coefficient estimators, we apply it to the group lasso problem. However, shown by simulation experiments, constructed confidence intervals have low coverage for true coefficients and this method has been proven to be inconsistent. Then a percentage thresholding method has been proposed, force close-to-zero estimate to be zero, and a corresponding modified residual bootstrap is applied yet only to boost insignificant improvement. To obtain more accurate confidence interval especially for active groups, a thresholding confidence construction method is proposed, the key is to threshold the group lasso estimator \( \tilde{\beta} \) and then simulated estimators based on \( \tilde{\beta} \) to estimate the variance. In terms of the coverage rate, this method shows an considerable improvement, but for most cases, the coverage rate for active groups is still far from 0.95 which is the ideal coverage rate for 95% confidence interval. A linear thresholding confidence construction method is put in, instead of using \( \tilde{\beta} \) in the procedure of simulations, we utilize linear regression to estimate the active components which yields less biased estimators of true coefficients. This method outperforms previous confidence construction method as its coverage ratio approximates 0.95 faster when data size grows.

One key factor in influencing the confidence interval building is the between-group correlations, if the groups are well picked and the between-group correlation coefficients are small, the constructed confidence intervals of non-zero parameters are
more likely to have a coverage rate closer to 0.95, in this thesis, we only consider the cases when $\rho = 0$ and $\rho=0.1$, in the future works, we can extend the range of the $\rho$ to further explore the relationship of the $p_{\text{active}}$ and $\rho$.

So far, all the discussion in this thesis is based on constructing confidence interval for individual component. In fact, we can build a joint confidence interval for all the components in one group for group lasso problem. Experiments can be set accordingly to evaluate the coverage rate for different joint confidence interval building methods.


