Bootstrapping in Lasso with a Case Study

Biswajit Das
Anubhab Bose
Soumya Ghatak
Arkoprobho Paul

Project Guide: Arunabha Majumdar (IIT Hyderabad)

August 29, 2022

Introduction

Introduction:

LASSO is popularly used to achieve better prediction accuracy, better interpretation, model selection. As the lasso estimates don't have any closed form solution, the Bootstrap technique is used to find out the underlying distribution of the estimated coefficients. But there are some restrictions in using the Bootstrap, like: if we just simply use the bootstrap method, we may not get a good approximation of the estimated distribution.

So, here we will discuss some alternate method to get a better approximation of the estimated distribution.

Sparse Statistical Model

- A Sparse Statistical Model is one in which only a relatively small number of parameters (or predictors) play an important role.
- For example, consider a linear regression model:

$$
\begin{aligned}
y_i &= \beta_0 + \sum_{j=1}^{p} x_{ij}\beta_j + e_i \qquad , i = 1(1)N \\
&= \beta_0 + \boldsymbol{x_i}^{\boldsymbol{T}}\boldsymbol{\beta} + e_i \qquad , i = 1(1)N
\end{aligned}
$$

  where, $y_i, i = 1(1)N$ is our outcome variable,
  $\boldsymbol{x_i} = (x_{i1}, x_{i2}, ..., x_{ip})^T$ are our predictor variables,
  β_0 and $\boldsymbol{\beta} = (\beta_1, \beta_2, ..., \beta_p)$ are unknown parameters,
  and e_i is the error term.
- We say that the above model is **Sparse** or the model has **Sparsity**, if most of the elements of $\boldsymbol{\beta}$ are zero, *i.e.* there are only few non-zero elements in $\boldsymbol{\beta}$.

Sparse Statistical Model

- The method of least squares provides estimates of the parameters by minimizing the least-squares objective function:

$$\underset{\boldsymbol{\beta_0},\boldsymbol{\beta}}{minimize} \sum_{i=1}^{N} (y_i - \beta_0 - x_i^T \boldsymbol{\beta})^2$$

– This is our usual "least-squares" estimator for the pair $(\beta_0, \boldsymbol{\beta})$, which is based on minimizing the squared-error loss.

- **But why do we need an alternative method?**

Sparse Statistical Model

Well, there are three reasons:

- **1. Prediction Accuracy:** The least-squares estimate often is unbiased but has large variance, and prediction accuracy can sometimes be improved by shrinking the values of the regression coefficients, or setting some coefficients to zero. By doing so, we introduce some bias but reduce the variance of the predicted values, and hence may improve the overall prediction accuracy (as measured in terms of the mean-squared error).
- **2. Better Interpretation:** The second reason is for the purposes of interpretation. With a large number of predictors, we often would like to identify a smaller subset of these predictors that exhibit the strongest effects.
- **3. Avoiding Non-estimability:** In our model, if $p > N$, we can not estimate the parameters by the least-squares method. So, for this reason also we need an alternative method.

Lasso Regression

- The acronym "LASSO" stands for Least Absolute Shrinkage and Selection Operator.
- Lasso regression performs l_1 regularization.
- Lasso regression is a type of linear regression that uses shrinkage. Shrinkage is where data values are shrunk towards a central point, like the mean. The lasso procedure encourages simple, sparse models (*i.e.* models with fewer parameters). This particular type of regression is well-suited for models showing high levels of muticollinearity or when we want to automate certain parts of model selection, like variable selection/parameter elimination.
- In the Lasso Regression, our least-squares objective function becomes:

$$\underset{\boldsymbol{\beta_0},\boldsymbol{\beta}}{minimize}\{\frac{1}{2N}\sum_{i=1}^{N}(y_i-\beta_0-\boldsymbol{x_i}^{\boldsymbol{T}}\boldsymbol{\beta})^2 \text{ subject to } \sum_{j=1}^{p}|\beta_j|\leq t$$

Oracle Procedure

- Let us consider a linear regression model:

$$\boldsymbol{y} = \boldsymbol{X}\beta^* + \boldsymbol{e}$$

  , where $\boldsymbol{y}$ is the response vector, $\boldsymbol{X} = (\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_p})$ is the predictor matrix and $\boldsymbol{e}$ is the error vector
- Let, $\mathcal{A} = \{j : \beta_j^* \neq 0\}$ and further assume that $|\mathcal{A}| = p_0 < p$. Thus the true model depends only on a subset of the predictors. Let's denote the coefficient estimator produced by a fitting procedure δ, by $\hat{\boldsymbol{\beta}}(\delta)$.

Using the language of Fan and Li (2001), we call δ an oracle procedure if $\hat{\boldsymbol{\beta}}(\delta)$ (asymptotically) has the following oracle properties:

- Identifies the right subset model, $\mathcal{A} = \{j : \hat{\beta}_j \neq 0\}$
- Has the optimal estimation rate, $\sqrt{n}(\hat{\boldsymbol{\beta}}(\delta)_{\mathcal{A}} - \boldsymbol{\beta}^*_{\mathcal{A}}) \xrightarrow{d} N(\mathbf{0}, \boldsymbol{\Sigma}^*)$, where Σ^* is the covariance matrix knowing the true subset model.

Lasso Variable Selection

- We adopt the setup of Knight and Fu (2000) for the asymptotic analysis. We assume two conditions:
  - $y_i = \boldsymbol{x_i}^{\boldsymbol{T}}\beta + e_i, i = 1(1)n$, where $e_i's$ are *i.i.d.* random variables with mean 0 and variance σ^2.
  - $\frac{1}{n}X^TX \to C$, where C is a positive definite matrix.
- Now, without loss of generality, let $\mathcal{A} = \{1, 2, ..., p_0\}$ and $\mathbf{C} = \begin{pmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} \\ \mathbf{C}_{21} & \mathbf{C}_{22} \end{pmatrix}$, where $\mathbf{C}_{11}$ is a $p_0 \times p_0$ matrix.
- We consider the lasso estimates, $\hat{\boldsymbol{\beta}}^{(n)}$, as

$$\hat{\boldsymbol{\beta}}^{(n)} = \underset{\boldsymbol{\beta}}{arg\,min} \|\boldsymbol{y} - \sum_{j=1}^{p} \boldsymbol{x_j}^{\boldsymbol{T}}\beta_j\|^2 + \lambda_n \sum_{j=1}^{p} |\beta_j|$$

  where λ_n varies with n. Let $\mathcal{A}_n = \{j : \hat{\beta}_{j_{(n)}} \neq 0\}$.
- The lasso variable selection is consistent if and only if $lim_n\ P(\mathcal{A}_n = \mathcal{A}) = 1$.

Adaptive Lasso

- In certain scenario, lasso cannot be an oracle procedure. In lasso, the coefficients are equally penalized in the l_1 penalty.
- So, we introduce **Adaptive Lasso**, that more penalizes those coefficients, which are truly insignificant than those coefficients, which are significant.
- So, we introduce weights to the penalty on each lasso produced coefficients.
- Let's consider the weighted lasso:

$$\hat{\beta}^{(n)} = \underset{\boldsymbol{\beta}}{arg\,min} \|\boldsymbol{y} - \sum_{j=1}^{p} \boldsymbol{x_j^T} \beta_j\|^2 + \lambda_n \sum_{j=1}^{p} w_j |\beta_j|$$

where **w** is a known weights vector.

Adaptive Lasso

Adaptive Lasso: Suppose that, $\hat{\boldsymbol{\beta}}$ is a root-n-consistent estimator to $\boldsymbol{\beta}^*$; for example, we can use $\hat{\boldsymbol{\beta}}$(ols). Select a $\gamma > 0$, and define the weight vector $\hat{\mathbf{w}} = 1/|\hat{\boldsymbol{\beta}}|^\gamma$. Then the adaptive lasso estimates $\hat{\boldsymbol{\beta}}^{*(n)}$ are given by:

$$\hat{\boldsymbol{\beta}}^{*(n)} = \underset{\boldsymbol{\beta}}{arg\,min} \|\boldsymbol{y} - \sum_{j=1}^{p} \boldsymbol{x_j^T} \beta_j\|^2 + \lambda_n \sum_{j=1}^{p} \hat{w}_j |\beta_j|$$

Theorem: Adaptive Lasso have Oracle Property

Suppose, $\lambda_n/\sqrt{n} \to 0$ and $\lambda_n n^{(\gamma-1)/2} \to \infty$. Then, the adaptive lasso estimates must satisfy the followings:

- **Consistency in Variable Selection:** $lim_n\ P(\mathcal{A}_n^* = \mathcal{A}) = 1$.
- **Asymptotic Normality:** $\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}^{*(n)} - \boldsymbol{\beta}_{\mathcal{A}}^*) \xrightarrow{d} N(\mathbf{0}, \sigma^2 \times \mathbf{C}_{11}^{-1})$

So, Adaptive Lasso have the Oracle Property.

Incosistent Lasso Path

- Let us consider a scenario, where $p_0 = 2m + 1 \geq 3$ and $p = p_0 + 1$ that is there is only one insignificant predictor.
- Let, $\mathbf{C_{11}} = (1 - \rho_1)\mathbf{I_{p_0}} + \rho_1\mathbf{J_1}$, where $\mathbf{J_1}$ is the matrix of 1's, $\mathbf{C_{12}} = \rho_2\vec{\mathbf{1}}$ and $\mathbf{C_{22}} = 1$.
- If ρ_1 and ρ_2 satisfy the condition, $|\frac{\rho_2}{1+(p_0-1)\rho_1}| > 1$,then the Lasso variable selection is inconsistent.

Incosistent Lasso Path:Simulation

- This scenario is demonstrated by a simulation study here. We let $y = x^T\beta + N(0, \sigma^2)$.The true regression coefficients are considered as $\beta = (5, 5, 5, 0)$.The predictors $x_i(i = 1, .., n)$ are iid N(0,C) where C is the same matrix taken above with $\rho_1 = -0.39 and \rho_2 = 0.23$.
- Note that, under this simulation setup, the Lasso variable selection is expected to be inconsistent.
- To demonstrate this numerically, we simulated 100 data sets from the above mentioned model for three different sample size(n). For each of the data set, we compute the entire solution to the lasso path and estimate the probability of the lasso solution path containing the true model. We repeat the same process for adaptive lasso for $\gamma = 1$ and $\gamma = 2$.

Incosistent Lasso Path:Simulation

- The simulation results are tabulated as below:

	$n = 60$	n=120	n=300
lasso($\gamma = 0$)	0.594	0.268	0.24
adalasso($\gamma = 1$)	0.71	0.75	0.8
adalasso($\gamma = 2$)	0.83	0.84	1

- From the table,we observe that Lasso variable selection is not consistent and has a low probability of capturing the true model regardless of the sample size, n.
- The adaptive lasso variable selection is consistent and captures the true model with high precision as n increases.

Consistency of Lasso Variable Selection

Remark:

- Note that, if $p = 2$, then the lasso selection is consistent with a proper choice of λ_n.
- The orthogonal design ensures the necessary condition and consistency of the lasso Solution.

Bootstrap

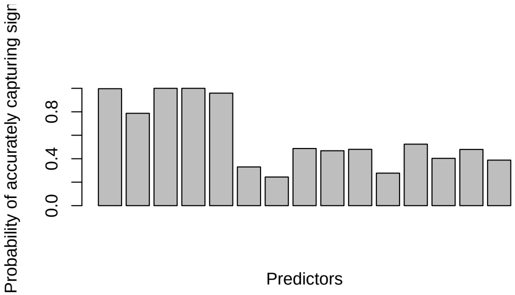
- The bootstrap method is a resampling technique to infer about the underlying distribution of the estimate by sampling a data set with replacement.
- Bootstrapping in regression is used to give us an insight on how variable the model parameters are that is how much random variation there is in the regression coefficients.
- The algorithm for bootstrap of residuals in regression is as below:
  - Fit regression model to the original data.
  - Extract residuals from the fit.
  - Create new y values using the residual samples.
  - Fit regression model with new y values selected with replacement and store the regression coefficients.
  - Obtain standard deviation of the regression coefficients.

Residual Bootstrap

- Knight and Fu (2000) considered the residual bootstrap method for Lasso estimator and sketched out its asymptotic behaviour. The asymptotic behaviour of the bootstrapped Lasso estimator is a random measure on $\mathbb{R}^p$ and that the bootstrap is inconsistent whenever one or more components of the regression parameter is 0.
- To appreciate; why the residual approximation have a random limit; first observe that the Lasso Estimators of the non zero components of β estimate their signs correctly with high probability but the estimators of the zero components take both positive and negative values with positive probabilities , thereby erring the capture of the target sign closely.
- The residual bootstrap mimics the main feature of the regression model closely but it fails to reproduce the sign of the zero components of β with sufficient accuracy leading to random limit.

Residual Bootstrap

- The fact that residual bootstrap erroneously captures the sign of the true coefficients is demonstrated by the simulation below where we have calculated the probabilities of bootstrap accurately capturing the true sign of the coefficients.
- Observe that the probabilities of capturing the signs of the significant coefficients is higher than the insignificant coefficients.



Modified Residual Bootstrap

- So we have devised a modified bootstrap method which acts as an improvement to the residual bootstrap method on the fitted Lasso model.
- The concept of modified bootstrap is to force components of the Lasso estimator, $\hat{\boldsymbol{\beta}}_{\boldsymbol{n}}$ to be exactly zero whenever they are close to zero.
- Since original Lasso estimator is root-n consistent; its fluctuations are of the order $n^{-0.5}$ around true value. So, a neighbourhood of order larger than $n^{-0.5}$ around the true value will contain the value of the Lasso estimator with higher probability.
- To that end, let a_n be a sequence of real numbers such that $a_n + (n^{-0.5} logn)a_n^{-1} \longrightarrow 0$ as $n \longrightarrow \infty$.For example, $a_n = \frac{c}{n^\delta}$ satisfies the condition of the sequence for all positive real number c and all δ lying in (0,0.5).

Modified Residual Bootstrap

- We threshold the components of the Lasso estimator $\hat{\beta}_n$ at a_n and define the modified Lasso estimator as:

$$\hat{\gamma}_n = (\hat{\gamma_{n,1}}, ..., \hat{\gamma_{n,p}})$$

with

$$\hat{\gamma_{n,j}} = \hat{\beta_{n,j}} \boldsymbol{I}(\{|\hat{\beta_{n,j}}| >= a_n)$$

- In particular, the shrinkage by a_n accomplishes our main objective to capture the signs of the zero coefficients precisely with probability tending to 1 for large n.
- Practically, the value of δ is chosen by 10 fold cross validation method.

Modified Residual Bootstrap

- The algorithm for residual bootstrapping on the modified estimate is described as below:
  - The modified residuals $r_i; i = 1, 2.., n$ based on the estimator $\hat{\gamma}_n$ are:
  $$r_i = y_i - x_i \hat{\gamma}_n$$
  - We select a random sample with replacement as $(e_1^{**}, ..., e_n^{**})$ from the centered residuals and set $y_i^{**} = x_i \hat{\gamma}_n + e_i^{**}$.
  - The modified bootstrap Lasso estimates are:
  $$\hat{\beta}_n^{**} = \underset{\mathbf{u} \in \mathbb{R}^p}{argmin} \{ \frac{1}{2N} \sum_{i=1}^{N} (y_i^{**} - x_i^T u)^2 + \lambda_n \sum_{j=1}^{p} |u_j| \}$$
- In this case; the asymptotic distribution of the bootstrapped lasso estimator is the the original distribution of the estimator and the modified bootstrap gives a valid and consistent approximation to the distribution of any statistic T_n.

Adaptive Lasso Residual Bootstrap

- The adaptive lasso residual bootstrap also acts as an improvement over the residual bootstrap on Lasso estimates.
- Let $\hat{\alpha}_n$ denote the adaptive lasso estimates.The adaptive lasso residuals are:

$$g_i = y_i - x_i\hat{\alpha}_n$$

- We select a random sample with replacement as $(e_1^{**}, ..., e_n^{**})$ from the centred residuals and set $y_i^{**} = x_i\hat{\alpha}_n + e_i^{**}$.
- The Adaptive Lasso bootstrap estimates are:

$$\hat{\beta_n^{**}} = \underset{\mathbf{u}\in\mathbb{R}^p}{argmin}\{\frac{1}{2N}\sum_{i=1}^{N}(y_i^{**} - x_i^T u)^2 + \lambda_n \sum_{j=1}^{p}\frac{|u_j|}{|\beta_{j,n}^{+}|}\}$$

where $\beta_n^{+} = (\beta_{1,n}^{+}, ..\beta_{p,n}^{+})$ are the OLS estimates over $(y_1^{**}, ..., y_n^{**})$.

Simulation 1 : Set Up

- Simulation setup:
  N=500,p=15
  $\beta_1 = 1, \beta_2 = 1.5, \beta_3 = 2, \beta_4 = -1,$
  $\beta_5 = 2.5, \beta_i = 0, \forall i \in \{6, 12, 13, ..., 15\}$
  The design matrix X is generated from standard normal distribution .
  And y variable is generated as $y_i = \mathbf{x}_i^T \beta + e_i$,where $e_i \sim N(0, 9)$

- Here we do bootstrap of residuals and generate $B = 500$ bootstrap samples and estimate bootstrap replicates of the coefficients .

Simulation 1 : Choice of δ in modified bootstrap and γ in adaptive lasso

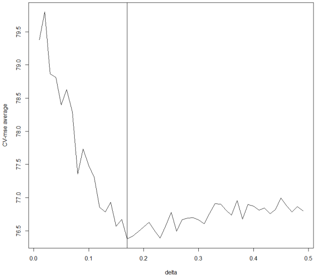


Figure: cv-mse average with varying δ

- From cross-validation we get that for $\delta = 0.17$ the average mse is lowest , so for modified bootstrap method we have taken $\delta = 0.17$.
- For adaptive lasso we have taken $\gamma = 2$ in this simulation .

Simulation 1: Estimates

i	β_i	$\hat{\beta}_i^{lasso}$(se)	$\hat{\beta}_i^{adalasso}$(se)	$\hat{\beta}_i^{mod.lasso}$(se)
1	1	0.642(0.35)	0.869(0.481)	0.642(0.425)
2	1.5	0.806(0.376)	1.035(0.485)	0.806(0.390)
3	2	1.334(0.428)	1.608(0.448)	1.334(0.403)
4	-1	-0.832(0.387)	-1.064(0.492)	-0.832(0.430)
5	2.5	1.861(0.419)	2.147(0.411)	1.861(0.415)
6	0	——	——	——
7	0	——	——	——
8	0	-0.0639(0.184)	——	——
9	0	——	——	——
10	0	——	——	——
11	0	-0.063(0.205)	——	——
12	0	——	——	——
13	0	——	——	——
14	0	——	——	——
15	0	-0.019(0.205)	——	——

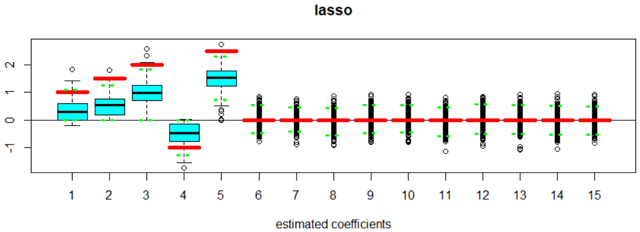
Simulation 1: Bias Estimation

i	β_i	Lasso	Ada.Lasso	Mod.Lasso
1	1	0.658	0.335	0.275
2	1.5	0.981	0.608	0.644
3	2	1.004	0.458	0.643
4	-1	0.489	0.108	0.089
5	2.5	0.987	0.382	0.594
6	0	——	——	——
7	0	——	——	——
8	0	0.010	——	——
9	0	——	——	——
10	0	——	——	——
11	0	0.033	——	——
12	0	——	——	——
13	0	——	——	——
14	0	——	——	——
15	0	0.0167	——	——

Simulation 1 : Confidence Interval

j	β_j	CI^{lasso}	$CI^{adalasso}$	$CI^{mod.lasso}$
1	1	(0.000,1.092)	(0.000,1.538)	(0.000,1.577)
2	1.5	(0.000,1.267)	(0.000,1.771)	(0.000,1.663)
3	2	(0.000,1.833)	(0.589,2.412)	(0.599,2.19)
4	-1	(-1.289,0.000)	(-1.704,0.000)	(-1.807,0.000)
5	2.5	(0.732,2.312)	(1.135,2.749)	(1.135,2.749)
6	0	——	——	——
7	0	——	——	——
8	0	(-0.574,0.422)	——	——
9	0	——	——	——
10	0	——	——	——
11	0	(-0.588,0.451)	——	——
12	0	——	——	——
13	0	——	——	——
14	0	——	——	——
15	0	(-0.544,0.485)	——	——

Simulation 1 : Inference Summary



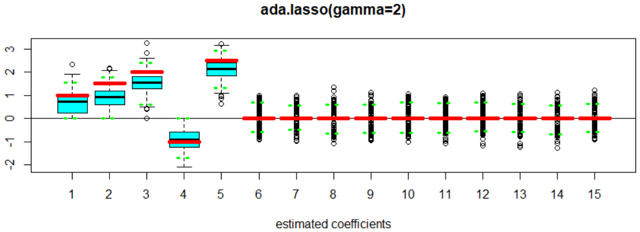


Figure: Estimated distribution of estimated coefficients along with confidence intervals and original coefficients using Lasso and Adaptive

Simulation 1 : Inference Summary

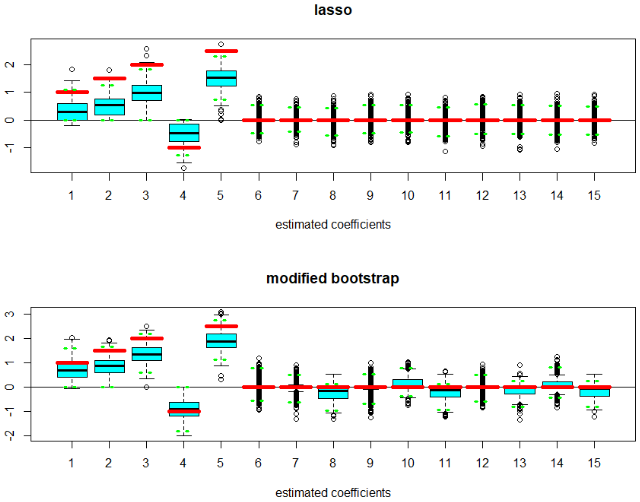


Figure: Estimated distribution of estimated coefficients along with confidence intervals and original coefficients using lasso and modified

Simulation 1 : Variable Selection

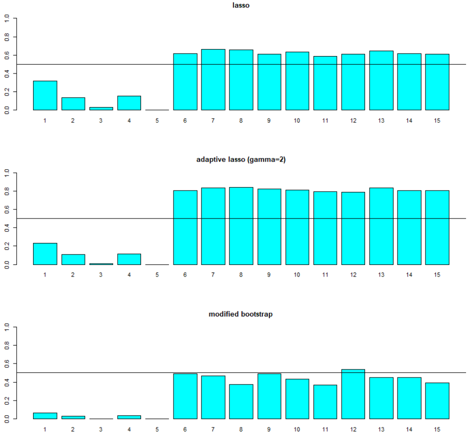


Figure: Proportion of times each coefficient is 0 in the bootstrap distribution or estimated $P[\hat{\beta}_i = 0]$

Simulation 2 : Set Up

- Simulation setup:
  N=300,p=400
  $\beta_1 = 1, \beta_2 = 0.5, \beta_3 = 0.64, \beta_4 = -0.4,$
  $\beta_5 = 0.6, \beta_6 = 0.5, \beta_7 = 0.7, \beta_8 = 1.2$
  $\beta_9 = 0.8, \beta_{10} = 0.24, \beta_i = 0, \forall i \in \{11, 12, 13, ..., 400\}$
  The design matrix X is generated from standard normal distribution .
  And y variable is generated as $y_i = \mathbf{x}_i^T \beta + e_i$,where $e_i \sim N(0, 1)$

- Here we do bootstrap of residuals and generate $B = 500$ bootstrap samples and estimate bootstrap replicates of the coefficients .

Simulation 1 : Choice of δ in modified bootstrap and γ in adaptive lasso

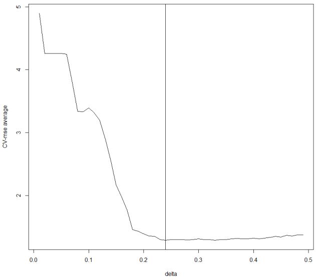


Figure: cv-mse average with varying δ

- From cross-validation we get that for $\delta = 0.24$ the average mse is lowest , so for modified bootstrap method we have taken $\delta = 0.24$.
- For adaptive lasso we have taken $\gamma = 2$ in this simulation .

Simulation 2: Estimates

i	β_i	$\hat{\beta}_i^{lasso}$(se)	$\hat{\beta}_i^{adalasso}$(se)	$\hat{\beta}_i^{mod.lasso}$(se)
1	1	0.962(0.072)	1.061(6.87e-02)	0.962(0.072)
2	0.5	0.386(0.06)	0.45(7.13e-02)	0.386(0.074)
3	0.64	0.411(0.062)	0.46(8.87e-02)	0.411(0.067)
4	-0.4	-0.362(0.069)	-0.411(8.06e-02)	-0.362(0.063)
5	0.6	0.482(0.067)	0.538(9.28e-02)	0.482(0.074)
6	0.5	0.474(0.066)	0.529(6.62e-02)	0.474(0.072)
7	0.7	0.452(0.071)	0.520(8.99e-02)	0.452(0.071)
8	1.2	1.094(0.06)	1.197(5.64e-02)	1.094(0.078)
9	0.61	0.539(0.072)	0.615(8.45e-02)	0.539(0.07)
10	0.8	0.698(0.06)	0.759(5.82e-02)	0.699(0.076)

Simulation 2: Estimates

	No. of non-zero estimated coefficients
Lasso	34
Ada.Lasso	12
Mod.Lasso	10

- Lasso estimated 24 (34-10) extra coefficients as significant (non-zero) which are insignificant(zero) in the simulation setup . Where as adaptive lasso and modified lasso estimated 2 and 0 respectively such coefficients .

i	β_i	Lasso	Ada.Lasso	Mod.Lasso
1	1	0.205	0.129	0.134
2	0.5	0.328	0.276	0.218
3	0.64	0.116	0.036	0.053
4	-0.4	0.245	0.178	0.115
5	0.6	0.113	0.006	0.013
6	0.5	0.36	0.273	0.268
7	0.7	0.205	0.015	0.112
8	1.2	0.196	0.07	0.034
9	0.61	0.184	0.059	0.089
10	0.8	0.002	0.000	0.007

j	β_j	CI^{lasso}	$CI^{adalasso}$	$CI^{mod.lasso}$
1	1	(0.707,0.977)	(0.905,1.172)	(0.792,1.09)
2	0.5	(0.171,0.409)	(0.216,0.501)	(0.229,0.5)
3	0.64	(0.182,0.433)	(0.157,0.517)	(0.23,0.497)
4	-0.4	(-0.407,-0.139)	(-0.511,-0.195)	(-0.49,-0.199)
5	0.6	(0.227,0.5)	(0.233,0.6)	(0.34,0.633)
6	0.5	(0.251,0.514)	(0.363,0.616)	(0.35,0.641)
7	0.7	(0.192,0.469)	(0.222,0.588)	(0.278,0.583)
8	1.2	(0.891,1.119)	(1.08,1.303)	(0.955,1.234)
9	0.61	(0.266,0.554)	(0.36,0.691)	(0.424,0.730)
10	0.8	(0.495,0.739)	(0.623,0.853)	(0.596,0.835)

Simulation 2 : Inference Summary

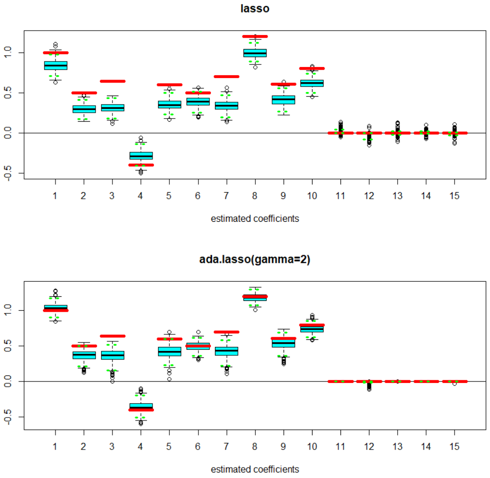


Figure: Estimated distribution of estimated coefficients along with confidence intervals and original coefficients using Lasso and Adaptive Lasso

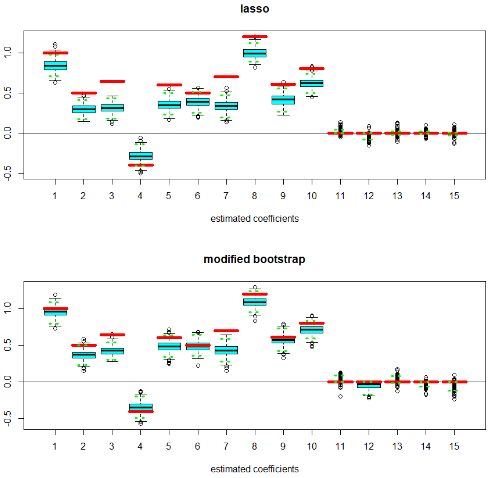


Figure: Estimated distribution of estimated coefficients along with confidence intervals and original coefficients using lasso and modified lasso(bootstrap)

Simulation 2 : Variable Selection

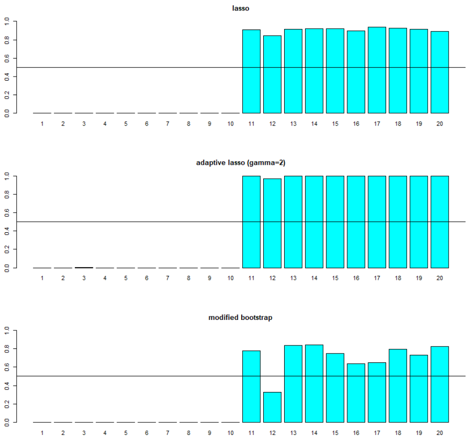


Figure: Proportion of times each coefficient is 0 in the bootstrap distribution estimated $P[\hat{\beta}_i = 0]$

Diabetes: Introduction

- The dataset consists observations on 442 patients.
- The response of interest here is a quantitative measure of disease progression one year after baseline.
- There are ten baseline variables here. They are as follows :
  - AGE : Age of the patient
  - SEX : Sex of the patient
  - BMI : Body Mass Index
  - BP : Blood Pressure
  - TC : Total Cholesterol
  - LDL : Low Density Lipoprotein
  - HDL : High Density Lipoprotein
  - TCH : Taxotere, Carboplatin and Herceptin
  - LTG : Serum Concentration of Lamorigine
  - GLU : Glucose Level
- The other variables that are being used are the interaction terms of the baseline variables. Hence, there are 55 variables in total.

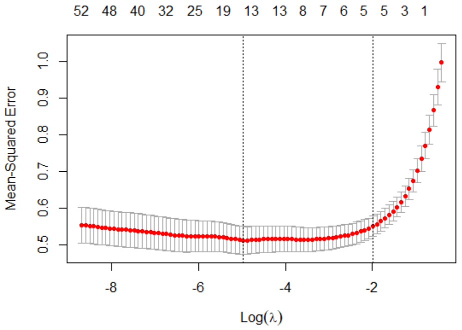
Diabetes: Summary

- A brief summary of the baseline variables is given below :

*	Min	Median	Mean	Max
AGE	19.00	50.00	48.52	79.00
BMI	18.00	25.70	26.38	42.20
BP	62.00	93.00	94.65	133.00
TC	97.0	186.0	189.1	301.0
LDL	41.60	113.00	115.44	242.40
HDL	22.00	48.00	49.79	99.00
TCH	2.00	4.00	4.07	9.09
LTG	3.258	4.620	4.641	6.107
GLU	58.00	91.00	91.26	124.00

- We divide the dataset into two parts - training part and testing part. We use the training part (which contains 320 observations of the dataset) to fit the various models and the testing part to assess how good our models are in comparison to each other.

Diabetes: LASSO



- The red dotted line here depicts the Mean Squared Error calculated by cross validation method for varying values of λ.
- From the above plot we obtain the following information.

	λ	Degrees of Freedom
Min	0.0069	11
1SE	0.1367	4

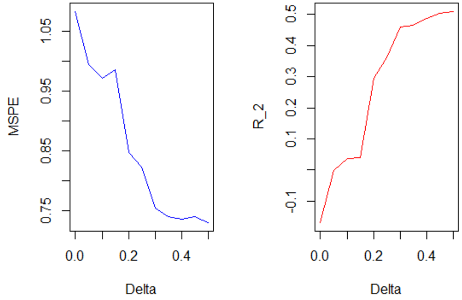
Diabetes: LASSO

- We obtain $\lambda_{min} = 0.0069$ and there are 11 non-zero coefficients in our fitted model at $\lambda = \lambda_{min}$ as shown in the table below.

i	$\hat{\beta}_i$	$se(\hat{\beta})$
SEX	-0.1251	0.0696
LTG	0.3048	0.1376
$AGE.SEX$	0.1596	0.1017
$AGE.HDL$	-0.1768	0.1070
$SEX.TCH$	-0.1581	0.0992
$BMI.BP$	0.3999	0.1568
$BMI.HDL$	-0.0032	0.0677
$BMI.TCH$	0.1450	0.0549
$BP.HDL$	-0.0102	0.0571
$LDL.HDL$	-0.0256	0.0497
$HDL.GLU$	-0.0076	0.0583

Diabetes: Determining the Value of δ

- We have a sequence of values of δ ranging between 0 and 0.5.
- For each value of δ we estimate the modified coefficients and calculate the Mean Squared Prediction Error and R^2 for each of the fitted models through 10-fold cross validation.



- From the plot we observe that for $\delta = 0.5$, the Mean Squared Prediction Error is the lowest as well as the R^2 is the highest.

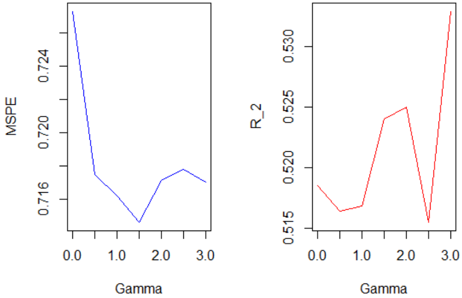
Diabetes: Modified LASSO

- Hence, we choose 0.5 to be the suitable value of δ and proceed to estimate the modified coefficients of our model.
- The table below shows us the modified coefficients of our model.

i	$\hat{\beta}_i$	$se(\hat{\hat{\beta}})$
SEX	-0.1251	0.0564
LTG	0.3048	0.1238
$AGE.SEX$	0.1596	0.0684
$AGE.HDL$	-0.1768	0.0856
$SEX.TCH$	-0.1581	0.1041
$BMI.BP$	0.3999	0.1559
$BMI.TCH$	0.1450	0.0764

Diabetes: Determining the Value of γ

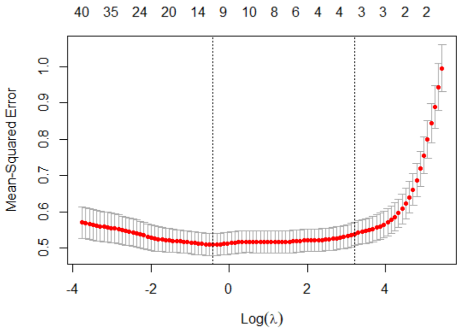
- We have a sequence of values of γ ranging between 0 and 3.
- For each value of γ, we fit the corresponding model and calculate the Mean Squared Prediction Error and R^2 for each of the fitted models through 10-fold cross validation.



- From the plot we observe that for $\gamma = 3.0$, the R^2 is the highest and the MSPE is relatively low.

Diabetes: Adaptive LASSO

- Hence, we choose 3.0 to be the suitable value of γ and proceed further



- The red dotted line here depicts the Mean Squared Error calculated by cross validation method for varying values of λ.

- From the above plot we obtain the following information.

	λ	Degrees of Freedom
Min	0.6580	9
1SE	24.762	4

- We obtain $\lambda_{min} = 0.6580$ and there are 9 non-zero coefficients in our fitted model at $\lambda = \lambda_{min}$ as shown in the table below.

i	$\hat{\beta}_i$	$se(\hat{\hat{\beta}})$
SEX	-0.1283	0.0917
LTG	0.3170	0.1665
AGE.SEX	0.2095	0.1312
AGE.HDL	-0.2219	0.1266
SEX.TCH	-0.2131	0.1204
BMI.BP	0.3921	0.1854
BMI.TCH	0.1608	0.0802
LDL.HDL	-0.0145	0.0765
HDL.GLU	-0.0056	0.0793

Remarks and Model Accuracy

- We observe that 7 of the variables are common in all the three models that we have fitted.
- Now we would like to assess how good are models are by comparing the Mean Squared Prediction Error and the R^2 for each of the models.

	MSPE	R^2
LASSO	374.8820	0.5204
Modified LASSO	517.5898	0.5181
Adaptive LASSO	380.1482	0.5221

Diabetes: Comparing CI

- We estimate the confidence intervals through residual bootstrap technique.

	LASSO	Modified LASSO	Adaptive LASSO
SEX	[-0.2134 , 0.0000]	[-0.1403 , 0.0000]	[-0.3374 , 0.0000]
LTG	[0.0000 , 0.3832]	[0.0000 , 0.3504]	[0.0000 , 0.5856]
AGE.SEX	[- 0.0373 , 0.3794]	[0.0000 , 0.2368]	[- 0.0538 , 0.4775]
AGE.HDL	[-0.4237 , 0.0000]	[-0.2793 , 0.0000]	[-0.4542 , 0.0000]
SEX.TCH	[-0.2845 , 0.0000]	[-0.3542 , 0.0000]	[-0.4128 , 0.0000]
BMI.BP	[0.0000 , 0.4663]	[0.0000 , 0.4841]	[-0.0078 , 0.5323]
BMI.HDL	[-0.2171 , 0.0652]	——	——
BMI.TCH	[0.0000 , 0.2082]	[0.0000 , 0.1762]	[0.0000 , 0.1512]
BP.HDL	[-0.1519 , 0.1889]	——	——
LDL.HDL	[-0.1742 , 0.0001]	——	[-0.2424 , 0.0000]
HDL.GLU	[-0.1219 , 0.0112]	——	[-0.2194 , 0.0010]