The Bayesian Lasso : A Survey

Kai Cui

Bayesian Lasso (Park and Casella, 2008)

Park, Trevor, and George Casella. "The Bayesian lasso." Journal of the American Statistical Association 103.482 (2008): 681-686.

Outline

1. Survey wide data and frequentist Lasso
2. Toy example
3. Bayesian duality
4. Bayesian Lasso :
   4.1 Implementation
   4.2 Example
5. Conclusion

Introduction

Wide Data

- Modern data sets are wide: $p \gg n$, more features than the sample size.
- Biological data: Microarrays, often tens of thousands of genes (features); while only tens of hundreds of samples.
- Many applications in computer vision deal with high-dimensional data.
- Different types of linear models are popular tools for such examples: e.g. linear regression, logistic regression. However, for these datasets, we cannot fit these models using standard approaches.

Approaches

Three Main Approaches for Improving Linear Model:

- Subset Selection: Identifying a subset of the p predictors that we believe to be related to the response, then fit a model on the reduced set of variables. (e.g. Best subset selection)
- Shrinkage: fitting a model involving all p predictors. However, the estimated coefficients are shrunken towards zero. Shrinkage has the effect of reducing variance. (e.g. Lasso and Ridge)
- Dimension Reduction: Projecting the p predictors into a M-dimensional subspace, where $M < p$. Then these M projections are used as predictors to fit a linear regression model by least squares. (e.g. PCA)

Issues

- Best subset selection: entails fitting 2^p models - infeasible beyond moderate values of p.
- Dimension reduction: Linear combination of covariates are harder to interpret, if not impossible.

Lasso

- The name is actually an acronym for Least Absolute Selection and Shrinkage Operator.
- The lasso is a method for simultaneous shrinkage and model selection in regression problems.
- It is most commonly applied to the linear regression model:

$$y = \mu 1_n + X\beta + \epsilon$$

Lasso(cont.)

- Lasso regression solves the following optimization problem:

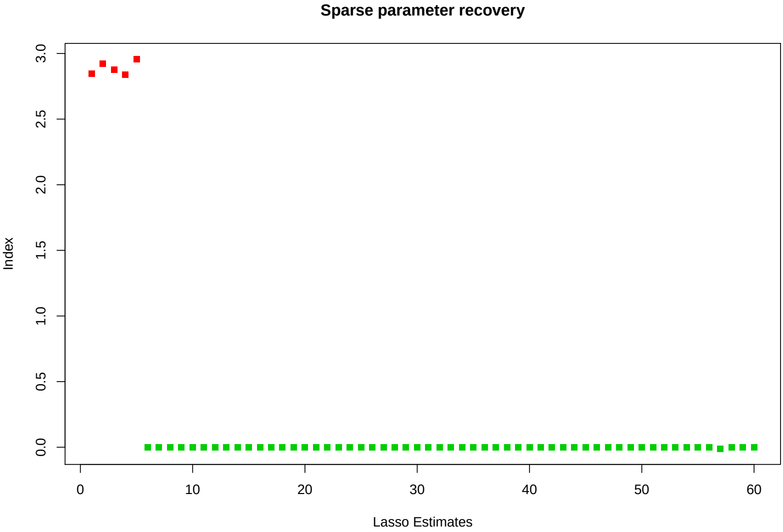
$$\min_{\beta} (\tilde{y} - X\beta)^T(\tilde{y} - X\beta) + \lambda \sum_{j=1}^{p} |\beta_j|$$

Where $\tilde{y} = y - \bar{y}1_n$ is the mean-centered response vector. The left term is RSS of the model; while the right term is shrinkage penalty ℓ_1.

A Toy Example of Lasso

- Generate X, y with $n = 30, p = 60$ with a sparse β vector with 5 non-zero elements.
- Our goal is to see if Lasso can recover the 5 non-zero β's correctly.

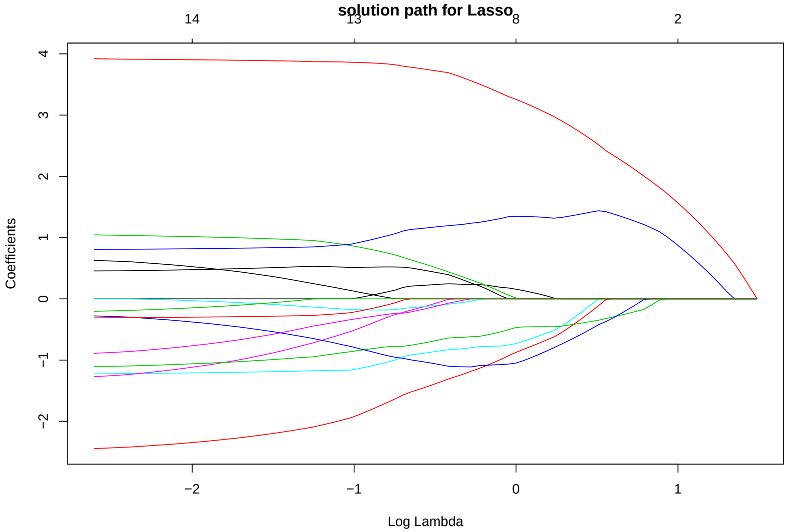
A Toy Example of Lasso
Variable Selection



- Lasso can select the truely non-zero coefficients.

A Toy Example of Lasso

Solution Path



Lasso vs Ridge

The difference between Lasso and Ridge

- Lasso uses an ℓ_1 penalty $||\beta||_1$.
- Ridge uses a (squared) ℓ_2 penalty $||\beta||_2^2$.
- Their solutions behave very differently. e.g. Ridge doesn't perform variable selection.

Lasso vs Ridge

Visualizing Ridge and Lasso with two predictors

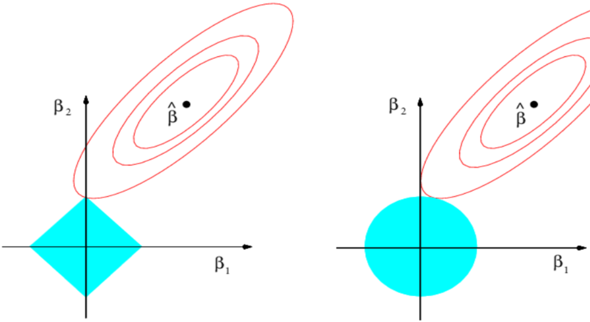


Figure: Contours of the error and constraint functions for the lasso (left) and ridge regression (right). The solid blue areas are the constraint regions, $|\beta_1| + |\beta_2| \leq s$ and $\beta_1^2 + \beta_2^2 \leq s$, while the red ellipses are the contours of the RSS.

Lasso vs Ridge

Why use Lasso instead of Ridge?

- Ridge regression shrinks all the coefficients to a non-zero value.
- The Lasso shrinks some of the coefficients all the way to zero.
- Lasso especially effective when most variables are not useful for prediction.

Bayesian Duality

Regularization leads to an optimization problem of the form

$$\min_{\beta \in \mathbb{R}^d} \{ l(y \mid \beta) + \text{pen}_\lambda(\beta) \}$$

Probabilistic approach leads to a Bayesian hierarchical model

$$p(y \mid \beta) \propto \exp\{-l(y \mid \beta)\} , \quad p_\lambda(\beta) \propto \exp\{-\text{pen}_\lambda(\beta)\}.$$

Regularized estimate = Posterior mode.

ℓ_1 penalty designed to perform selection, while ridge and ℓ_2 shrinkage.

Need for Bayesian Interpretation

- Bayesian hierarchical model leads to better uncertainty quantification for the parameters.
- Eliminates the need for ad-hoc tuning of parameters. Full Bayes method learns parameters from the data.

Bayesian Interpretation

- Ridge: β_j^R is the posterior mode, with a Normal prior on β.
- Lasso: β_j^L is the posterior mode, with a Laplace prior on β.

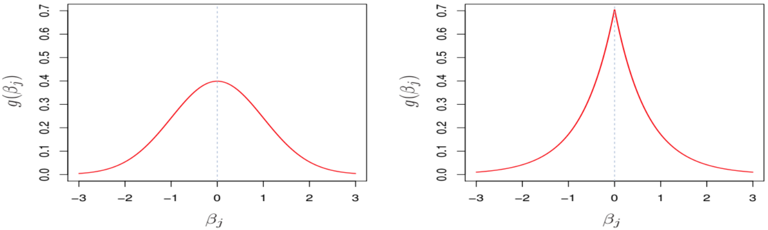


Figure: Left: Ridge regression is the posterior mode for $\boldsymbol{\beta}$ under a Gaussian prior. Right: The lasso is the posterior mode for $\boldsymbol{\beta}$ under a double-exponential prior.

Bayesian Lasso

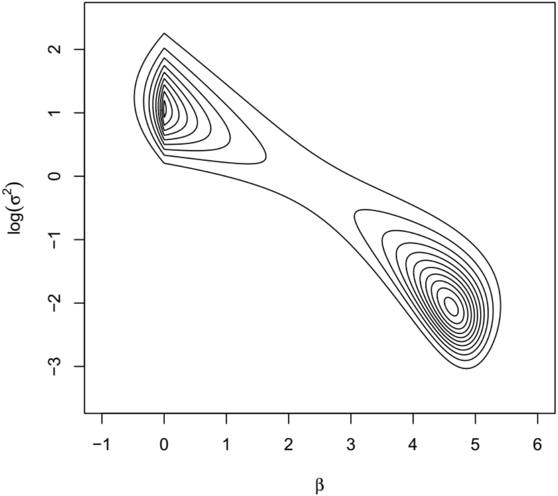
- The form of this expression suggests that the Lasso may be interpreted as a Bayesian posterior mode estimate when the regression parameters β_i have independent and identical double-exponential priors.
- The conditional prior is:

$$\pi(\beta|\sigma^2) = \frac{\lambda}{2\sqrt{\sigma^2}} e^{-\lambda|\beta_j|/\sqrt{\sigma^2}}$$

  - Conditioning on σ^2 is important as it ensures that the full posterior is unimodal.
  - Convergence of Gibbs sampler will be slower without unimodality.

Unconditional prior may lead to bimodal posteriors

- Consider the unconditional prior $\beta \sim \frac{\lambda}{2} e^{-\lambda|\beta_i|}$.
- Artificial example: p = 1, n = 10, $X^T X = 1$, $X^T y = 5$, $y^T y = 26$ and $\lambda = 3$.
- The posterior distributions of $(ln\sigma^2, \beta)$ are bimodal.



Bayesian Lasso

The lasso estimate can be considered as the mode of the posterior distribution of β.

$$\hat{\beta}_L = arg \max_{\beta} p(\beta|y, \sigma^2, \tau)$$

when

$$p(\beta|\tau) = (\tau/2)^p exp(-\tau||\beta||_1)$$

and the likelihood on

$$p(y|\beta, \sigma^2) = N(y|X\beta, \sigma^2 I_n)$$

For any fixed values $\sigma^2 > 0$, $\tau > 0$, the posterior mode β is the lasso estimate with penalty $\lambda = 2\tau\sigma^2$.

Gibbs sampling steps

We use the improper prior density $\pi(\sigma^2) = 1/\sigma^2$.

1. The full conditional for β is multivariate:

$$\beta \sim N(A^{-1}X^T y, \sigma^2 A^{-1})$$
$$A = X^T X + D_\tau^{-1}$$
$$D_\tau = diag(\tau_1^2, ..., \tau_p^2)$$

2. $\sigma^2 \sim$ Inv-gamma(a, b):

$$a = (n+p)/2$$
$$b = (y - X\beta)^T (y - X\beta)/2 + \beta^T D_\tau^{-1} \beta/2$$

3. 1 / $\tau_i^2 \sim$ Inv-gamma(a, b):

$$a = \sqrt{\lambda^2 \sigma^2 / \beta_j^2}$$
$$b = \lambda^2$$

R Code

```
Blas_gibbs = function(x, y, r, delta, iter) {
  n <- nrow(x); m <- ncol(x); XtX <- t(x) %*% x; xy <- t(x) %*% y
  coef <- matrix(0, iter, m); sample_sigma <- rep(0, iter)
  sample_iTau <- matrix(0, iter, m); sample_lambda <- rep(0, iter)
  beta <- drop(backsolve(XtX + diag(nrow=m), xy))
  residue <- drop(y - x %*% beta)
  sigma2 <- drop((t(residue) %*% residue) / n)
  iTau2 <- 1 / (beta * beta)
  lambda <- m * sqrt(sigma2) / sum(abs(beta))
  for (k in seq(iter)) {
    # sample beta
    invD <- diag(iTau2)
    invA <- solve(XtX + invD)
    mean <- invA %*% xy
    varcov <- sigma2 * invA
    beta <- drop(rmnorm(1, mean, varcov))
    coef[k,] <- beta
    # sample sigma^2
    shape <- (n+m-1)/2
    residue <- drop(y - x %*% beta)
    scale <- (t(residue) %*% residue + t(beta) %*% invD %*% beta)/2
    sigma2 <- rigamma(1, shape, 1/scale)
    sample_sigma[k] <- sigma2
    # sample tau^2
    muPrime <- sqrt(lambda^2 * sigma2 / beta^2)
    lambdaPrime <- lambda^2
    iTau2 <- rep(0, m)
    for (i in seq(m)) {iTau2[i] <- rgamma(1, muPrime[i], lambdaPrime)}
    sample_iTau[k, ] <- iTau2
    # update lambda
    shape = r + m/2
    scale = delta + sum(1/iTau2)/2
    lambda <- rgamma(1, shape, 1/scale)
    sample_lambda[k] <- lambda}
  coef}
```

Bayesian Lasso on a real data set

Diabetes study: 442 diabetes patients were measured on 10 baseline variables:age, sex, body mass index, average blood pressure and six blood serum measurements.

	AGE	SEX	BMI	BP	Serum measurements						Response
Patient	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	y
1	59	2	32.1	101	157	93.2	38	4	4.9	87	151
2	48	1	21.6	87	183	103.2	70	3	3.9	69	75
3	72	2	30.5	93	156	93.6	41	4	4.7	85	141
4	24	1	25.3	84	198	131.4	40	5	4.9	89	206
5	50	1	23.0	101	192	125.4	52	4	4.3	80	135
6	23	1	22.6	89	139	64.8	61	2	4.2	68	97
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Figure 1: Diabetes study

Bayesian Lasso on a real data set

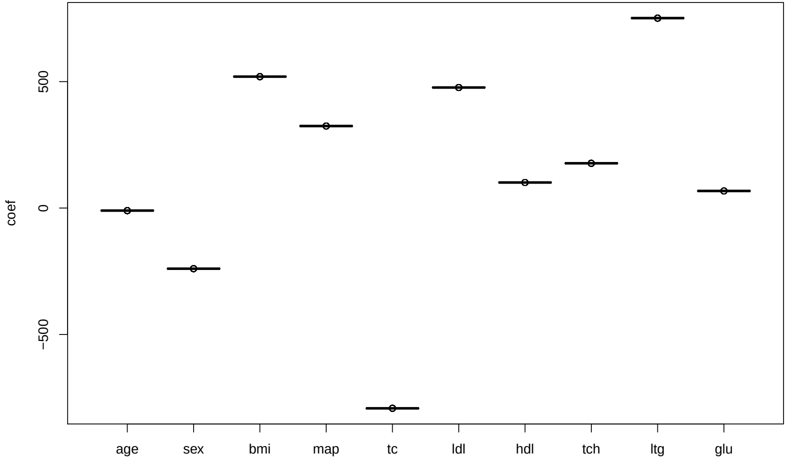
```r
library(monomvn) # blasso package
iter_number = 10000 #total number of MCMC samples to be collected
burn_in = 2000
## Calculate Bayesian Lasso by my implementation
blas_imp <- Blas_gibbs(x, y, delta = 0.1, r = 2,
                         iter = iter_number)
## Calculate Bayesian Lasso by  "monomvn"
blas_pkg <- blasso(x, y, T = iter_number, verb = 0)
## Get the mode for each column
blas_pkg_beta = apply(blas_pkg$beta[burn_in:iter_number, ], 2, median)
blas_imp_beta = apply(blas_imp[burn_in:iter_number, ], 2, median)
lasso_pkg_beta = apply(lasso_pkg$beta, 1, median)
## Get the column names
parameters = colnames(diabetes$x2)[1:10]
## plot
boxplot(blas_imp, ylim = range(-1000:1000), xlab="",
        ylab="coef", xaxt = "n")
axis(1, at=1:10, labels=parameters[1:10])
```

Bayesian Lasso on a real data set

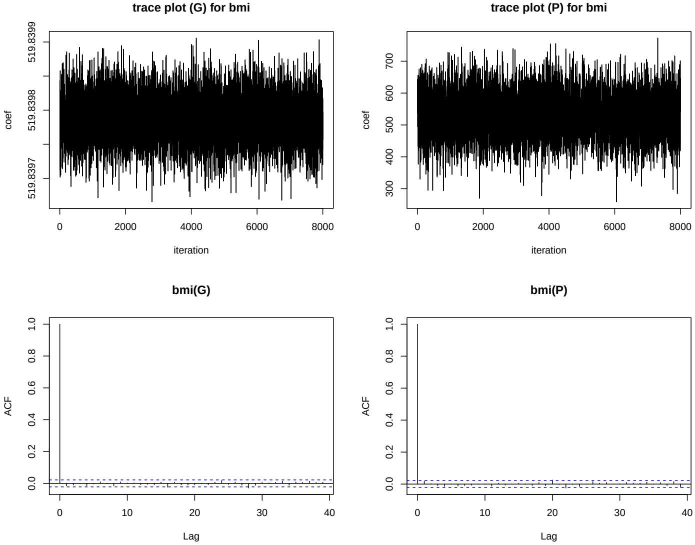
Parameter Estimates (Posterior Median)

	Freq. Lasso	Bayes Lasso (Gibbs)	Bayes Lasso (monomvn)
age	0	-10	0
sex	-204	-240	-201
bmi	521	520	531
map	301	324	307
tc	-123	-792	-68
ldl	0	477	0
hdl	-69	101	-210
tch	22	177	0
ltg	518	751	510
glu	57	68	0

Bayesian Lasso on a real data set



Trace plot for BMI



Conclusion

- The Bayesian lasso estimates seem to be a compromise between the Lasso and Ridge regression estimates: the paths are smooth, like Ridge regression, but are more similar in shape to the Lasso path.
- The Bayesian lasso is easy to implement and automatically provides interval estimates for all parameters, including the error variance.
- R package `monomvn` provides computing tool for Bayesian Lasso and other recent shrinkage priors, e.g. horseshoe.