Variable Selection Using Elastic Net

A Gentle Introduction to Penalized Regression

Mohamad Hindawi, PhD, FCAS

Antitrust Notice

- The Casualty Actuarial Society is committed to adhering strictly to the letter and spirit of the antitrust laws. Seminars conducted under the auspices of the CAS are designed solely to provide a forum for the expression of various points of view on topics described in the programs or agendas for such meetings.

- Under no circumstances shall CAS seminars be used as a means for competing companies or firms to reach any understanding – expressed or implied – that restricts competition or in any way impairs the ability of members to exercise independent business judgment regarding matters affecting competition.

- It is the responsibility of all seminar participants to be aware of antitrust regulations, to prevent any written or verbal discussions that appear to violate these laws, and to adhere in every respect to the CAS antitrust compliance policy.

Have you ever…

- …needed to build a realistic model with not enough data?
- …wanted to keep in your model highly correlated variables that capture different characteristics?
- …had highly correlated variables that made your model unstable? (Was it easy to find the source of the problem?)
- …had hundreds or thousands of highly redundant predictors to consider?
- …felt you had too little time to build a model?

You came to the right place!

Agenda

- The variable selection problem
  - Classic variable selection tools
  - Challenges
- Introduction to penalized regression
  - Ridge regression
  - LASSO
  - Elastic Net
- Extension to GLM
- Appendix
  - Close relatives to LASSO and Elastic Net
  - Bayesian interpretation of penalized regression

Goals of predictive modeling

- The goal is to build a model that ensures accurate prediction on future data
- How:
  - Choose the correct model structure
  - Choose variables that are predictive
  - Obtain the coefficients
- Many techniques:
  - Linear regression
  - GLM
  - Survival analysis – Cox's partial likelihood
  - …and many more!
- Variable selection:
  - Recover the true non-zero variables
  - Estimate coefficients close to their true value

Classic variable selection tools: Exhaustive methods

- Brute-force search
- For each $k \in \{1,2,\ldots,p\}$, find the subset of "best" variables of size k
  - For example: the smallest residual sum of squares (RSS)
- Choosing k can be done using:
  - AIC
  - Cross-validation
- Do not need to examine all possible subsets
  - "Leaps and bounds" techniques by Furnival and Wilson (1974)
- Never practical for even small number of variables or small datasets

Classic variable selection tools : Greedy algorithms

- More constrained than exhaustive methods
- ***Forward stepwise selection***
  - Starts with the intercept and then sequentially adds into the model the predictor that most improves the fit
- ***Backward stepwise selection***
  - Starts with the full model and sequentially deletes the predictor that has the least impact on the fit
- ***Hybrid stepwise selection***
  - Considers both forward and backward moves

Challenges

- Discrete process — ***variables are either retained or discarded but nothing in between***
- Issues:
  - **Unstable** — small changes in the data produce changes in the chosen variables
  - Models built this way usually exhibit **low prediction accuracy** on future data
  - **Computationally prohibitive** when the number of predictors is large

Challenges

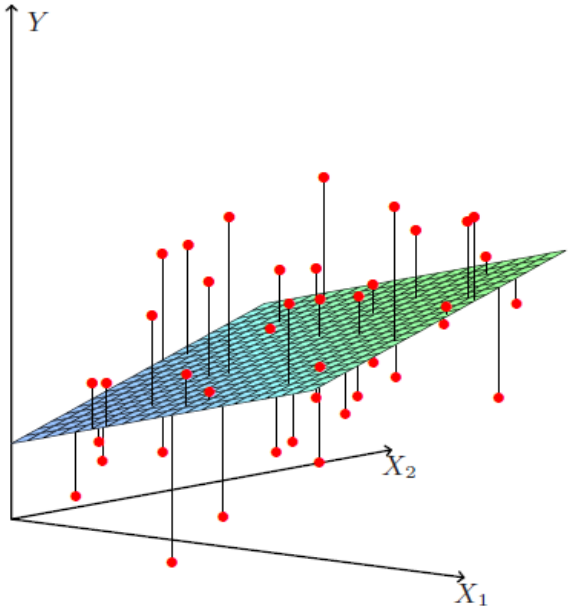
- Severely limits the number of variables to include in a model, especially for models built on small datasets
  - Certain lines of business
    - Boat, motorcycle, GL
  - Certain type of models
    - Fraud models, retention models
- Problems
  - Over-fitting
  - Under-fitting
  - …and don't forget multicollinearity
- Many regularization techniques provide a "more democratic" and smoother version of variable selection

Quick review of linear models

- Target variable (y)
  - Profitability (pure premium, loss ratio)
  - Retention
  - Fraudulent claims
- Predictive variables $\{x_1, x_2, \ldots, x_p\}$
  - "Covariates" – used to make predictions
  - Policy age, credit, vehicle type, etc.
- Model structure

$$y = \alpha + \beta_1 \cdot x_1 + \cdots + \beta_p \cdot x_p$$

- Solution is given by



$$\widehat{\boldsymbol{\beta}}_{\boldsymbol{OLS}} = \arg\min \left\{ \sum_{i=1}^{N} \left(y_i - \alpha - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\}$$

Penalization methods

- Generally, a penalized problem can be described as:

$$\widehat{\boldsymbol{\beta}}_{\mathbf{Penalized}} = \arg\min \left\{ \sum_{i=1}^{N} \left(y_i - \alpha - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \cdot J(\beta_1, \ldots, \beta_p) \right\}$$

  $J(\cdots)$ is a positive penalty for $\{\beta_1, \ldots, \beta_p\}$ not equal to zero

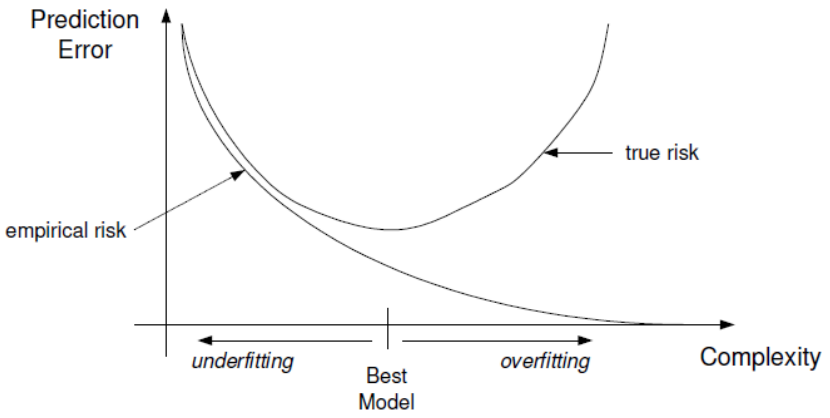
- Unlike subset selection methods, penalization methods are:
  - More continuous
  - Somewhat shielded from high variability
- All methods shrink coefficients toward zero
  - Some methods also do variable selection

The classic bias-variance trade-off

- Penalized regression produces estimates of coefficients that are biased
- The common dilemma: reduction in variance at the price of increased bias

$$MSE = Var(\hat{\beta}) + Bias(\hat{\beta})^2$$

- If bias is a concern, use penalized regression to choose variables and then fit unpenalized model
- Use cross validation to see which method works better



Penalization methods

$$\widehat{\boldsymbol{\beta}}_{\textbf{Penalized}} = \arg\min\left\{\sum_{i=1}^{N}\left(y_i - \alpha - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda \cdot J(\beta_1, \ldots, \beta_p)\right\}$$

- Different methods use different penalty functions:
  - Ridge Regression : L^2
  - LASSO : L^1
  - Elastic Net : combination of L^1 and L^2
- To use penalized regression, data needs to be normalized:
  - Center y around zero
  - Center each x_i around zero and standardized to have SD = 1

Ridge regression

- Ridge regression uses L^2 penalty function, i.e. "sum of squares"

$$\widehat{\boldsymbol{\beta}}_{\boldsymbol{Ridge}} = \arg\min \left\{ \sum_{i=1}^{N} \left(y_i - \alpha - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \cdot \sum_{j=1}^{p} {\beta_j}^2 \right\}$$

- Used to penalize large parameters
- λ is a tuning parameter; for every λ there is a solution

Ridge regression

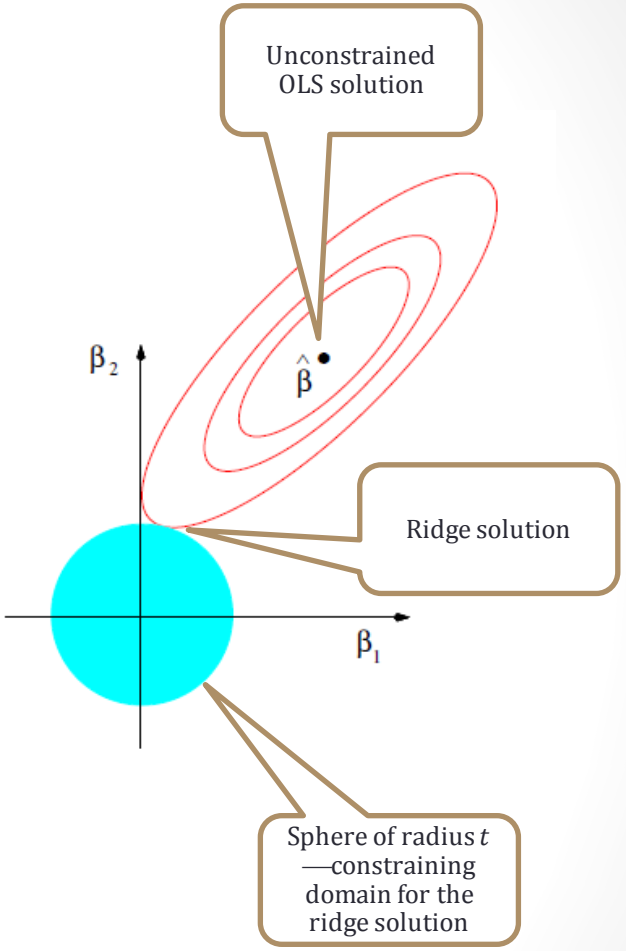
- Equivalent way to write the ridge problem:

$$\widehat{\boldsymbol{\beta}}_{\boldsymbol{Ridge}} = \arg\min \left\{ \sum_{i=1}^{N} \left(y_i - \alpha - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\}$$

subject to

$$\sum_{j=1}^{p} {\beta_j}^2 \leq t$$

- Ridge regression shrinks parameters, but never forces any to be zero

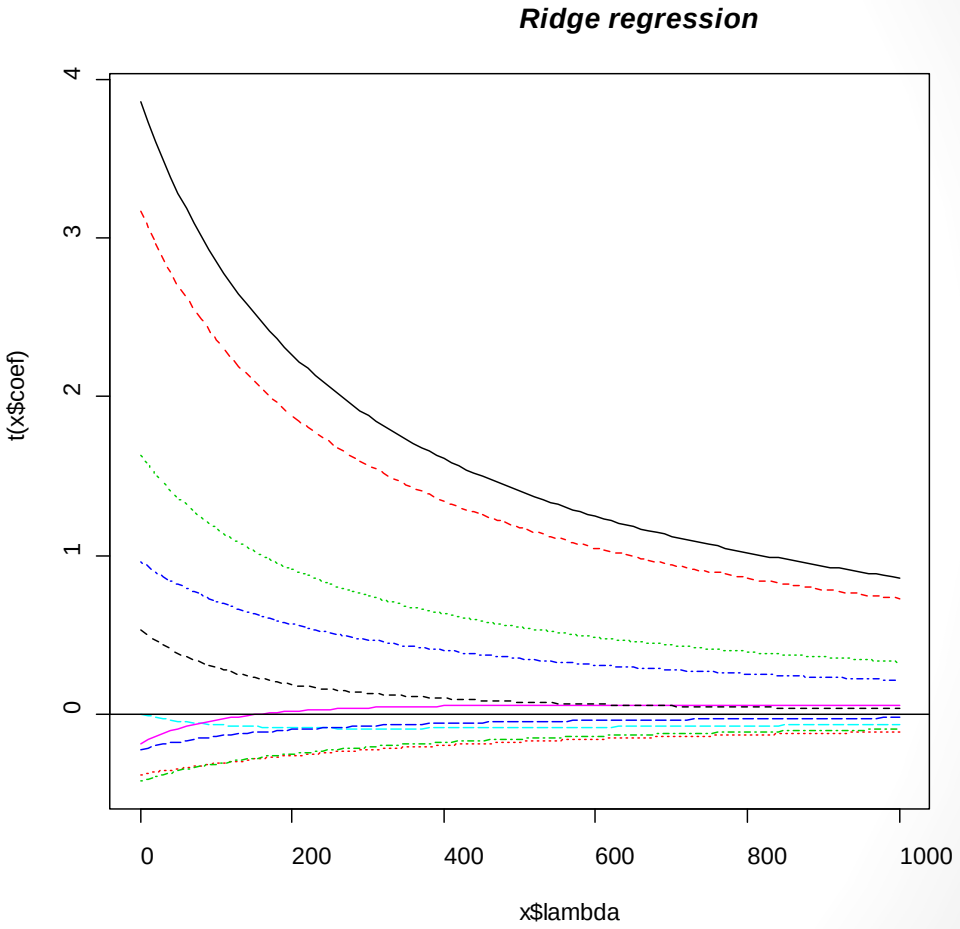


Ridge regression example using R

- Simulated data with 10 variables and 500 observations
- True model:

$$y = 4 \cdot x_1 + 3 \cdot x_2 + 2 \cdot x_3 + x_4$$

- Fit using package (MASS) in R
  - lm.ridge

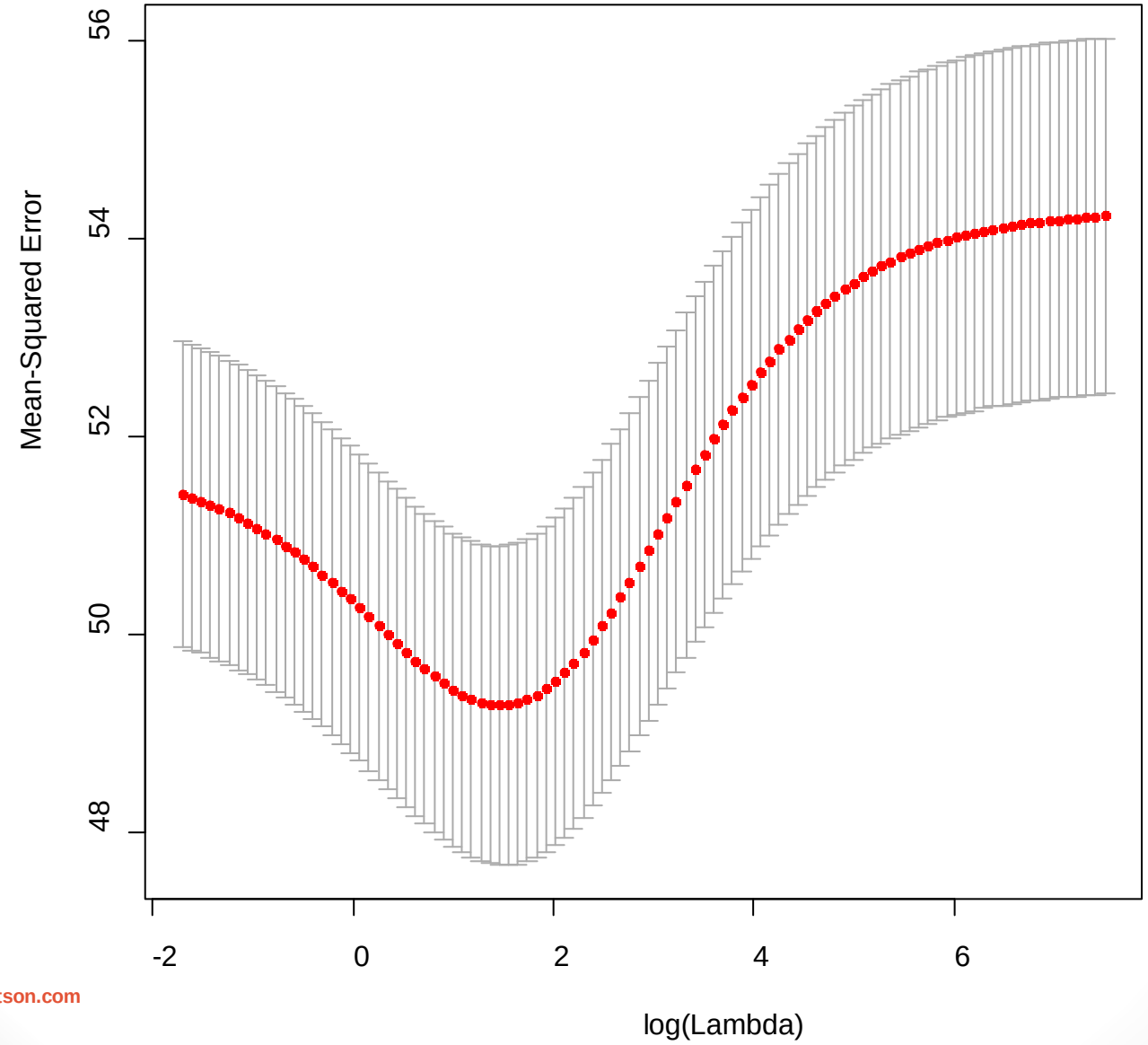


How to choose the tuning parameter λ?

- Use cross validation
- How it works:
  - Randomly divide data into *N* equal pieces

  - For each piece, estimate model from the other N-1 pieces
  - Test the model fit (e.g., sum of squared errors) on the remaining piece
  - Add up the N sum of square errors
  - Plot the sum vs. λ
- Recommendation: If possible, use separate years of data as the folds

How to choose the tuning parameter λ?



Simple example: Ridge regression – multicollinearity

- Ridge regression controls well for multicollinearity
  - Deals well with high correlations among predictors
- Simple example:
  - True model

$$y = 2 + x_1$$

  - Assume x_2 is another variable such that $x_2 = x_1$
  - Notice that $y = 2 + \beta_1 \cdot x_1 + (1 - \beta_1) \cdot x_2$ should be an equivalent linear model
  - Ridge regression tries to fit the data so that it will minimize ${\beta_1}^2 + {\beta_2}^2$
  - Ridge solution tries to split the coefficients as equally as possible between the two variables

$$y = 2 + ½\, x_1 + ½\, x_2$$

Ridge regression summary

- Uses L^2 penalty function
- Shrinks all coefficients, but does not force any to be zero
- Deals well with correlation between variables

LASSO

- **LASSO = L**east **A**bsolute **S**hrinkage and **S**electing **O**perator
- Introduced by Tibshirani in 1996
- Uses L^1 penalty function, i.e. sum of absolute values

$$\widehat{\boldsymbol{\beta}}_{\boldsymbol{LASSO}} = \arg\min \left\{ \sum_{i=1}^{N} \left(y_i - \alpha - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \cdot \sum_{j=1}^{p} |\beta_j| \right\}$$

- As usual, data needs to be normalized

LASSO

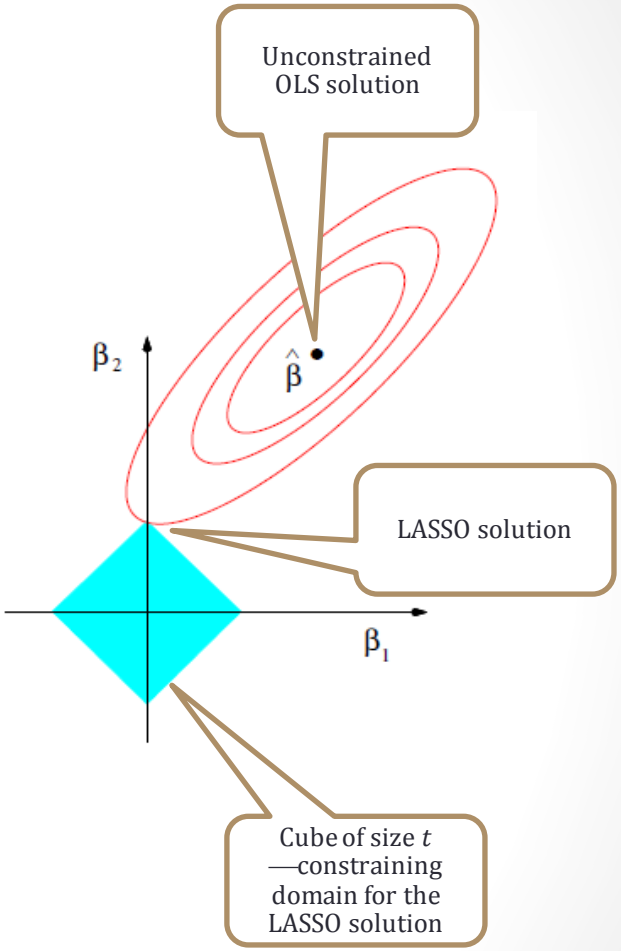
- Equivalent way to write the LASSO problem:

$$\widehat{\boldsymbol{\beta}}_{\boldsymbol{LASSO}} = \arg\min \left\{ \sum_{i=1}^{N} \left(y_i - \alpha - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\}$$

subject to

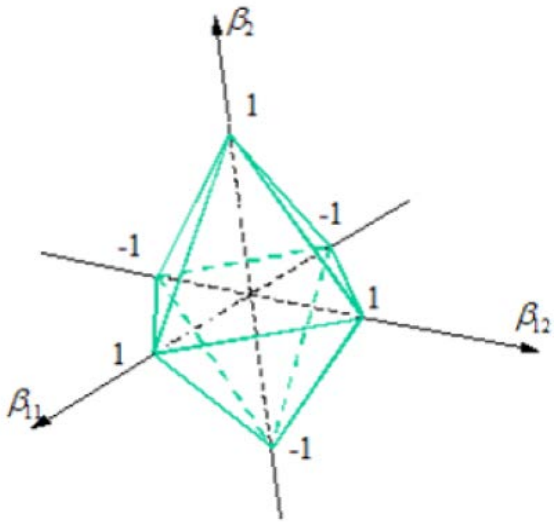
$$\sum_{j=1}^{p} |\beta_j| \leq t$$

- For every t, there is a unique solution
  - $t \to 0$: constant model
  - $t \to \infty$: OLS model



LASSO

- Example of LASSO domain in three dimensions

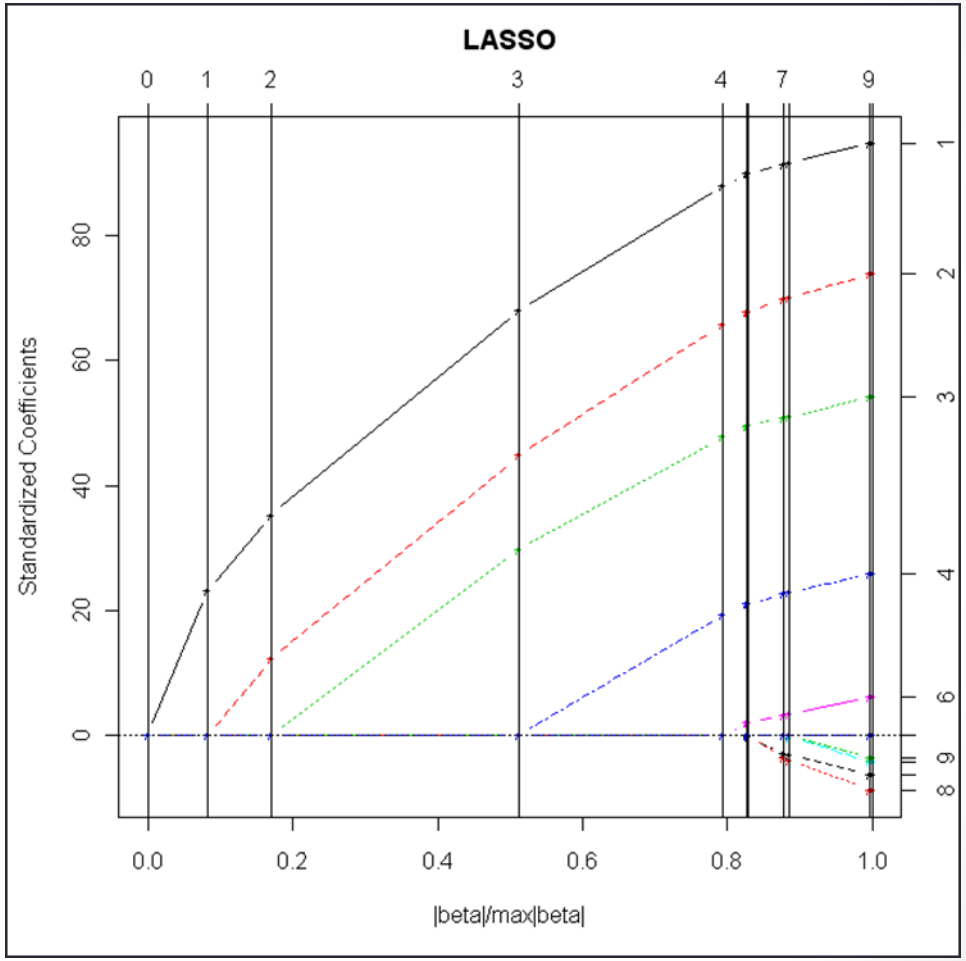


LASSO example using R

- Simulated dataset with 10 variables and 500 observations
- $Corr(x_i, x_j) = 0.5$
- True model:

$$y = 4 \cdot x_1 + 3 \cdot x_2 + 2 \cdot x_3 + x_4$$

- Fit using package "elasticnet" in R



LASSO example 2 using R

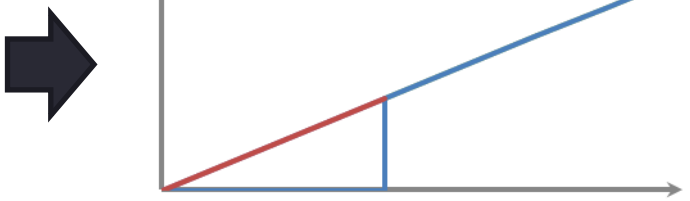
- Fitting LASSO curve for linear models is extremely fast
- This example used 100k of simulated data and 100 variables

```
LASSO sequence
Computing X'X .....
LARS Step 1 :    Variable 37    added
LARS Step 2 :    Variable 12    added
LARS Step 3 :    Variable 49    added
LARS Step 4 :    Variable 82    added
LARS Step 5 :    Variable 42    added
LARS Step 6 :    Variable 19    added
LARS Step 7 :    Variable 1     added
LARS Step 8 :    Variable 7     added
LARS Step 9 :    Variable 89    added
LARS Step 10 :   Variable 22    added
LARS Step 11 :   Variable 4     added
LARS Step 12 :   Variable 50    added
LARS Step 13 :   Variable 23    added
LARS Step 14 :   Variable 65    added
LARS Step 15 :   Variable 72    added
LARS Step 16 :   Variable 60    added
LARS Step 17 :   Variable 44    added
LARS Step 18 :   Variable 94    added
LARS Step 19 :   Variable 61    added
LARS Step 20 :   Variable 55    added
LARS Step 21 :   Variable 48    added
LARS Step 22 :   Variable 79    added
LARS Step 23 :   Variable 70    added
LARS Step 24 :   Variable 81    added
LARS Step 25 :   Variable 97    added
LARS Step 26 :   Variable 17    added
.......
```

Simple illustration : Orthonormal design matrix

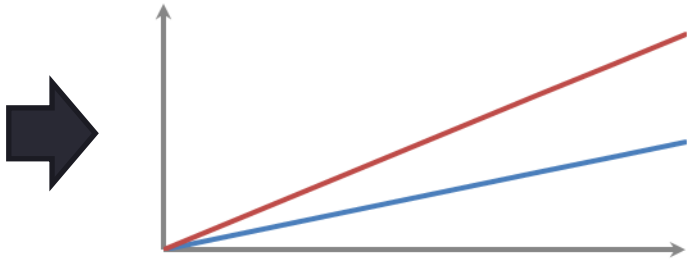
- Expressions on this slide only hold when $X^T X = I$, i.e. the design matrix is orthonormal
- **Subset selection of size k:**
  - Choose k largest coefficients in the absolute values and set the rest to zero

    $\hat{\beta}_j^{SS} = \beta_j^{OLS} \quad iff \quad |\hat{\beta}_j^{OLS}| > \lambda$



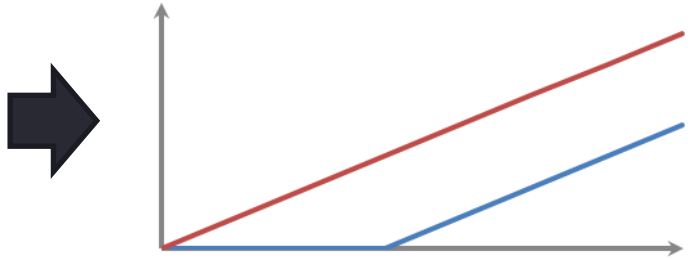
- **Ridge regression:**
  - Shrink all coefficients by a factor

    $\hat{\beta}_j^{Ridge} = \frac{1}{1+\xi}\hat{\beta}_j^{OLS}$



- **LASSO:**
  - Translate and truncate

    $\hat{\beta}_j^{LASSO} = \text{sign}(\hat{\beta}_j^{OLS}) \cdot \left(|\hat{\beta}_j^{OLS}| - \eta\right)^+$



LASSO summary

- Uses L^1 penalty function
- Sets some coefficients to zero and shrinks the rest of the coefficients
- If high correlations among predictors exist, the performance of the LASSO is dominated by Ridge regression (Tibshirani, 1996)
- If there is a group of variables among which the pairwise correlations are very high, then the LASSO tends to select only one variable from the group and does not care which one is selected.

Is there a compromise between Ridge regression and LASSO?

First attempt to compromise between Ridge and LASSO

- Use L^q penalty function for $1 < q < 2$

$$\widehat{\boldsymbol{\beta}}_{L^q} = \arg\min \left\{ \sum_{i=1}^{N} \left(y_i - \alpha - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\}$$

subject to

$$\sum_{j=1}^{p} |\beta_j|^q \leq t$$

"Naive" Elastic Net

- Introduced by Zou and Hastie (2005) with a sum of L^1 and L^2 penalty function

$$\hat{\boldsymbol{\beta}}_{\boldsymbol{Naive\ ENet}} = \arg\min \left\{ \sum_{i=1}^{N} \left(y_i - \alpha - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda_1 \cdot \sum_{j=1}^{p} |\beta_j| + \lambda_2 \cdot \sum_{j=1}^{p} {\beta_j}^2 \right\}$$

- The linear term (L^1) of the penalty forces certain variables to be zero
- The quadratic term (L^2) of the penalty:
  - Decreases the limitation on the number of selected variables
  - Encourages grouping effect
  - Stabilizes the L^1 regularization path and hence improves the prediction

"Naive" Elastic Net

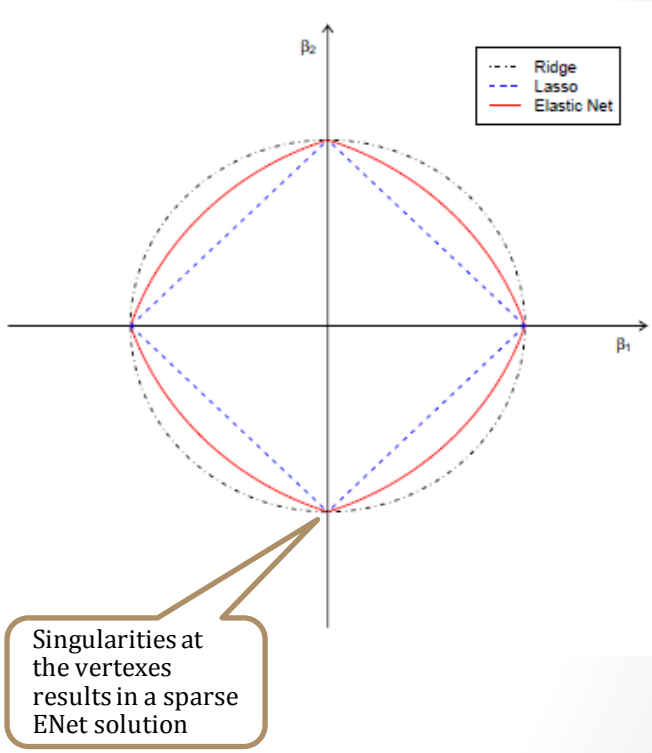
- Equivalent way to write the Elastic Net problem:

$$\widehat{\boldsymbol{\beta}}_{\boldsymbol{Naive}\ \mathbf{ENet}} = \arg\min \left\{ \sum_{i=1}^{N} \left(y_i - \alpha - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\}$$

subject to

$$(1-\alpha) \cdot \sum_{j=1}^{p} |\beta_j| + \alpha \cdot \sum_{j=1}^{p} {\beta_j}^2 \leq t$$

- Strict convexity guarantees the grouping effect even in the extreme situation of identical predictors



Deficiencies of naive Elastic Net

- While it overcomes the limitations of LASSO and Ridge regression, it does not perform satisfactorily unless it is close to Ridge or LASSO
- Naive Elastic Net is two stage-procedure:
  - Step 1: For each fixed λ_2, first find the ridge regression coefficients
  - Step 2: Do the LASSO type shrinkage along the LASSO solution path
- This amounts to incurring double shrinkage, which does not help to reduce the variance and introduces extra bias

Moving from naiveté

- Elastic Net scales the naive Elastic Net parameters

$$\widehat{\boldsymbol{\beta}}_{\boldsymbol{ENet}} = (1 + \lambda_2) \cdot \widehat{\boldsymbol{\beta}}_{\boldsymbol{Naive\ ENet}}$$

- Elastic Net:
  - Does automatic variable selection
  - Does continuous shrinkage
  - Handles multicollinearity
- Similar to the previous example, when $x_1 = x_2$; Elastic Net will include both variables
- Could include all the variables desired in the initial model without worrying about multicollinearity or near-aliasing

Similar to a fishing net, Elastic Net retains only all the "big fish"

A simple illustration*: Elastic Net vs. LASSO

- Two independent "hidden" variables: z_1 and z_2

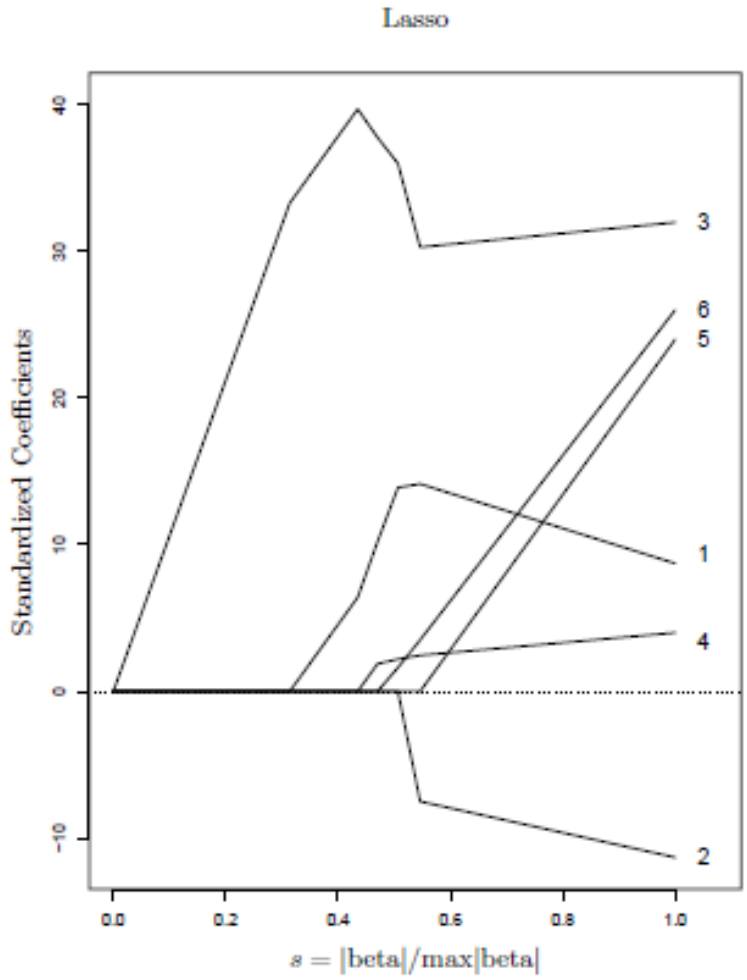
$$z_1 \sim U(0,20) \quad and \quad z_2 \sim U(0,20)$$

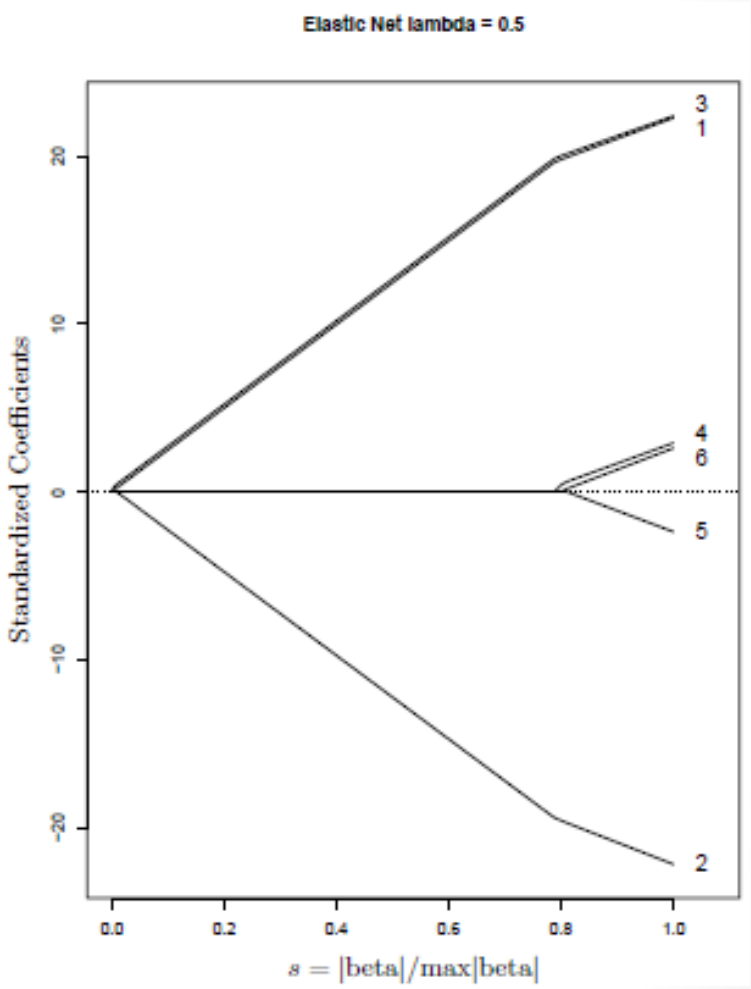
- Generate the response vector: $y = z_1 + 0.1\, z_2 + N(0,1)$
  - Suppose that the only predicators observed are:

$$x_1 = z_1 + \epsilon_1, \quad x_2 = -z_1 + \epsilon_2, \quad x_3 = z_1 + \epsilon_3$$
$$x_4 = z_2 + \epsilon_4, \quad x_5 = -z_2 + \epsilon_5, \quad x_6 = z_2 + \epsilon_6$$

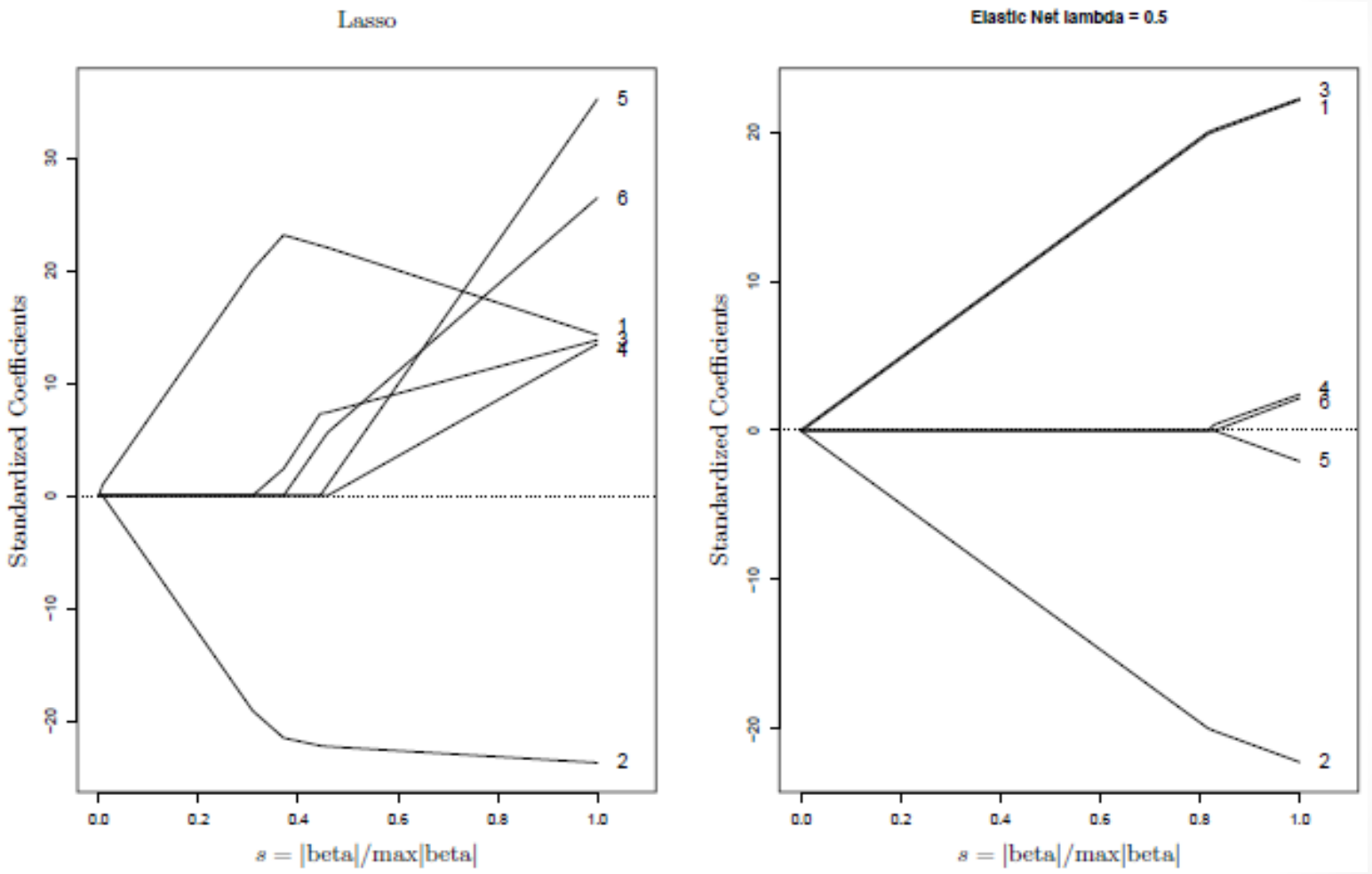
  - $\epsilon_1, \dots, \epsilon_6 \sim N(0, \frac{1}{16})$
  - Fit the model on (x,y)
- An "oracle" would identify x_1, x_2 and x_3 (the z_1 group) as the most important variables, but none of the z_2 group variables

Elastic Net vs. LASSO



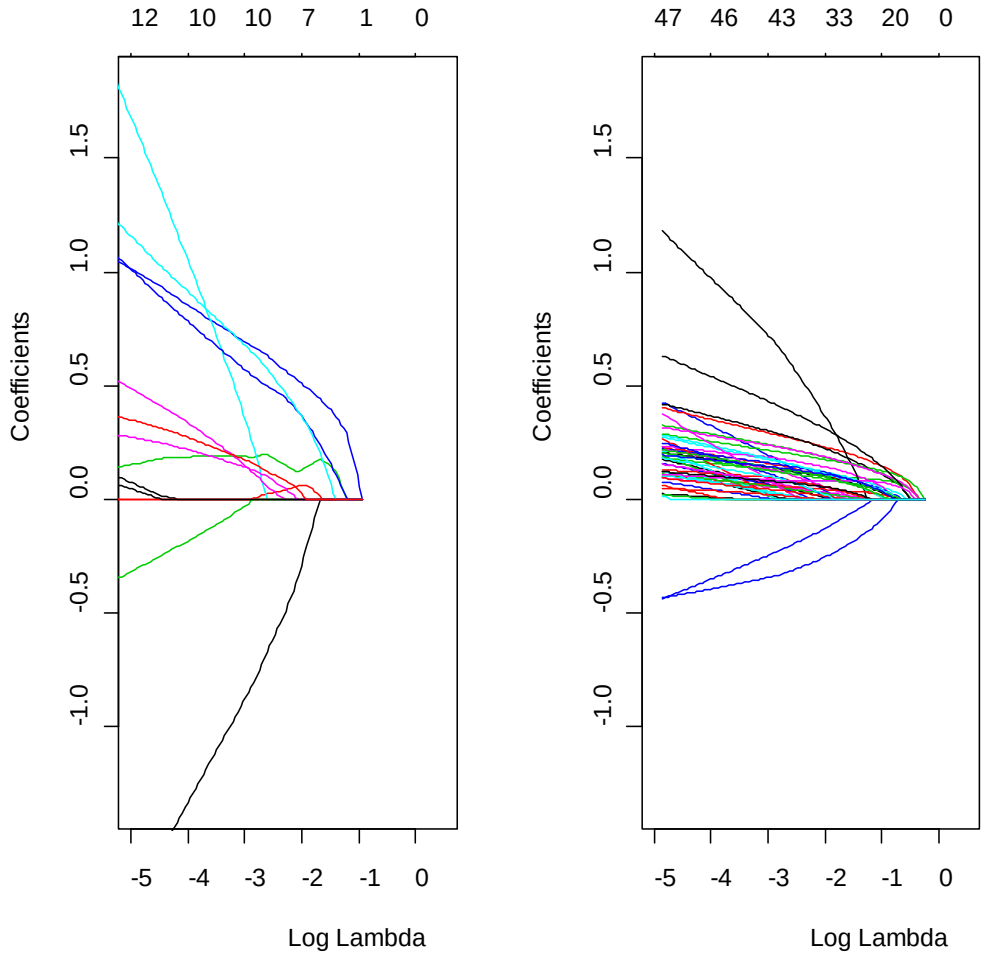


Elastic Net vs. LASSO



Elastic Net vs. LASSO

- The Elastic Net includes more non-zero coefficients than LASSO, but with smaller magnitudes



Extension to GLM

- GLM consists of three elements:
  - Dependent variable (y) assumed to come from a probability distribution from the **exponential family of distributions**
  - A **linear predictor** $\eta = X\boldsymbol{\beta}$
  - A **link function** g such that $E(Y) = \mu = g^{-1}(\eta)$
- Estimate the coefficients $\boldsymbol{\beta}$ by solving a set of equations to satisfy the maximum likelihood criterion:

$$\widehat{\boldsymbol{\beta}}_{GLM} = \arg\max\{L(y;\boldsymbol{\beta})\}$$

equivalently

$$\widehat{\boldsymbol{\beta}}_{GLM} = \arg\min[-\log\{L(y;\boldsymbol{\beta})\}]$$

Extension to GLM

- For penalized regression, the coefficients are obtained by solving the following equation:

$$\widehat{\boldsymbol{\beta}}_{Penalized} = \arg\min\left[-\log\{L(y;\boldsymbol{\beta})\} + \lambda \cdot J(\boldsymbol{\beta})\right]$$

- Optimization problem is harder and slower to solve
- The regularization path is piece-wise smooth rather than piece –wise linear
- Many algorithms are developed to solve this problem
  - Park and Hastie developed an algorithm to find the points where variables are added and then used a piece-wise linear approximation

Software to fit LASSO and Elastic Net

- Several packages are currently available in R including:
  - glmnet
  - elasticnet
  - LARS
  - penalized
- Models that are currently available:
  - Linear regression models
  - Logistic regression models
  - Multinomial regression models
  - Poisson regression models
  - Cox models
  - Alas, no gamma model; but may be coming soon!
- Currently not available in most other programs
  - SAS implemented LASSO for linear models
  - PROC GLMSELECT can be used to implement the Elastic Net for linear models

Appendix

A few extensions and close relatives to LASSO and Elastic Net

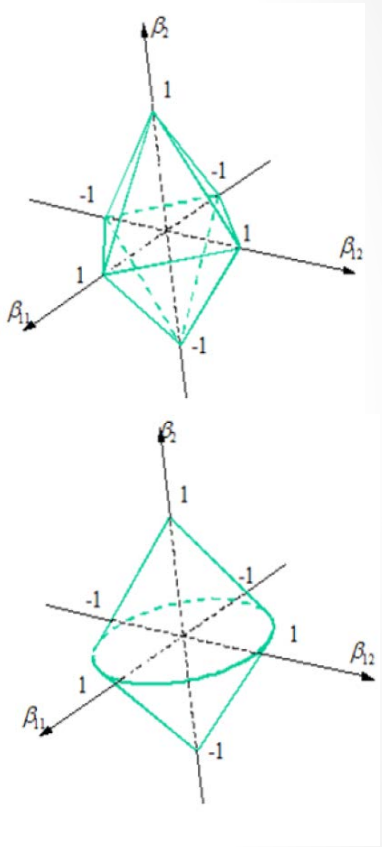
Some other extensions

- Group LASSO
- Sparse group LASSO
- Adaptive LASSO
- Adaptive Elastic Net

Group LASSO

- Introduced by Yuan & Lin (2007)
- Variables might come in groups, so need to include or exclude the entire group

$$\widehat{\boldsymbol{\beta}}_{\boldsymbol{Grp\ LASSO}} = \arg\min \left\{ \sum_{i=1}^{N} \left(y_i - \alpha - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \cdot \sum_{l=1}^{L} \sqrt{p_l} \cdot \|\boldsymbol{\beta}_l\|_2 \right\}$$

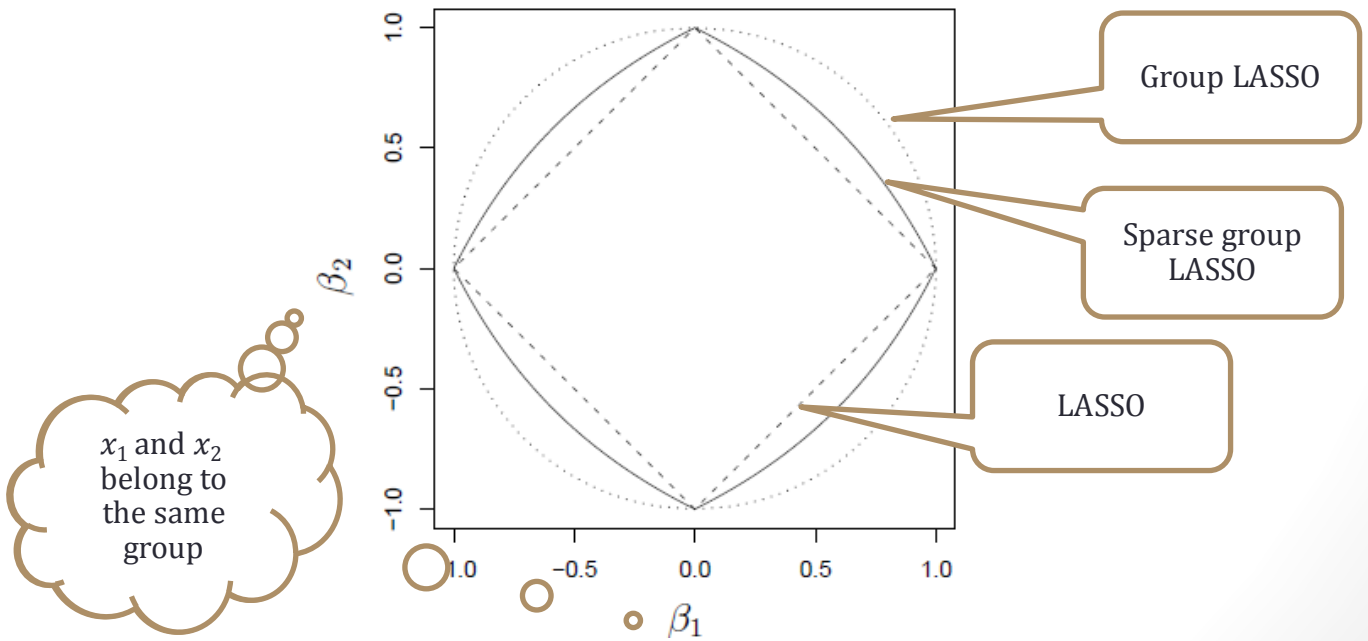


- **All or nothing approach.**
  - Does not allow for individual levels to have zero coefficients

Sparse group LASSO

- Introduced by Friedman, Hastie and Tibshirani (2010)
- A compromise between Group LASSO and LASSO

$$\widehat{\boldsymbol{\beta}}_{\boldsymbol{S\,G\,LASSO}} = \arg\min \left\{ \sum_{i=1}^{N} \left(y_i - \alpha - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda_1 \cdot \sum_{l=1}^{L} \sqrt{p_l} \cdot \|\boldsymbol{\beta_l}\|_2 + \lambda_2 \cdot \|\boldsymbol{\beta}\|_1 \right\}$$



Adaptive LASSO

- LASSO shrinks all the coefficients by the same magnitude
- More reasonable to shrink large coefficients more than small coefficients
- Adaptive LASSO does exactly that…

$$\hat{\beta}_{Ada\ LASSO} = \arg\min \left\{ \sum_{i=1}^{N} \left(y_i - \alpha - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \cdot \sum_{j=1}^{p} \hat{w}_j \left| \beta_j \right| \right\}$$

- Adaptive LASSO exhibits oracle properties

Adaptive Elastic Net

- Adaptive Elastic Net is a similar variation of the Adaptive LASSO

$$\hat{\beta}_{Ada\ ENet} = (1+\lambda_2)\cdot \arg\min\left\{\sum_{i=1}^{N}\left(y_i-\alpha-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2+\dot{\lambda}_1\cdot\sum_{j=1}^{p}\widehat{w}_j\,|\beta_j| + \lambda_2\cdot\sum_{j=1}^{p}{\beta_j}^2\right\}$$

where $\widehat{w}_j = \left(\left|\hat{\beta}_j(ENet)\right|\right)^{-\gamma}$

Bayesian interpretation of penalized regression

Bayes Theorem

- Bayes Rule:

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$$

- In the regression context:

$$P(\beta|y) \propto P(y|\beta)P(\beta)$$

- "Posterior is proportional to prior times likelihood"
- For OLS, we assume no prior knowledge about β

Bayesian interpretation of Ridge regression

- In the Ridge regression, we expect *a priori* that the parameters will be small
- A reasonable prior distribution is normal with mean value zero:

$$P(\beta) \propto e^{-\frac{1}{2\sigma^2}\|\beta\|_2^2}$$

- Then the posterior probability is:

$$P(\beta|y) \propto e^{-\frac{1}{2}\left[\|y-\beta X\|_2^2+\frac{1}{\sigma^2}\|\beta\|_2^2\right]}$$

- The mode is given by:

$$\|y-\beta X\|_2^2+\frac{1}{\sigma^2}\|\beta\|_2^2$$

which is the Ridge solution where $\lambda = \frac{1}{\sigma^2}$

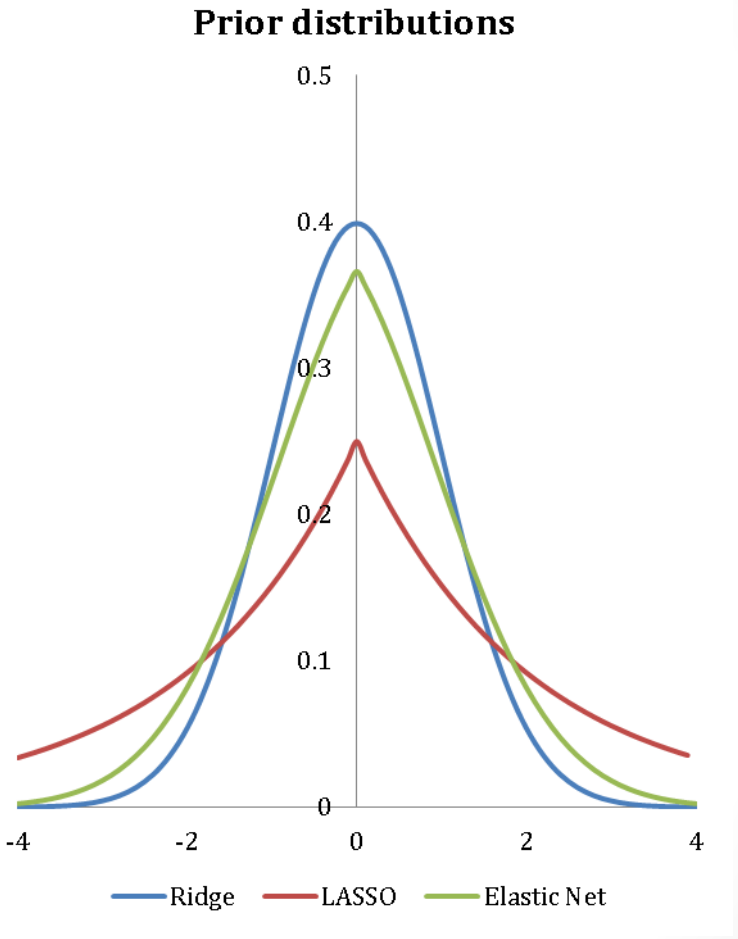
Bayesian interpretation of LASSO and Elastic Net

- For LASSO, the prior is given by:

$$P(\beta) \propto e^{-\frac{\lambda}{2}\|\beta\|_1}$$

- For Elastic Net, the prior is given by:

$$P(\beta) \propto e^{-\frac{1}{2}[\lambda_1\|\beta\|_1+\lambda_2\|\beta\|_2^2]}$$



Contact information

If you would like additional information or references for this presentation, please contact:

Mohamad Hindawi, PhD, FCAS
Towers Watson
175 Powder Forest Dr.
Weatogue, CT 06089
860.843.7134
Mohamad.Hindawi@towerswatson.com