Review of Ordinary and Penalized Linear Regression

Andrew Nobel

August, 2023

Preliminaries

Regression: Prediction with a Real-Valued Response

Setting: Jointly distributed pair (X, Y) with $X \in \mathbb{R}^p$ and $Y \in \mathbb{R}$

- X is a feature vector, often high dimensional
- Y is a real-valued response

Goals

- Predict Y from X
- Identify the components of X that most affect Y

Regression: Prediction with a Real-Valued Response

Ex 1: Marketing

- $X =$ money spent on different components of marketing campaign
- $Y =$ gross profits from sales of marketed item

Ex 2: Housing

- $X =$ geographic and demographic features of a neighborhood
- $Y =$ median home price

Regression: Statistical Framework

1. Jointly distributed pair $(X, Y) \in \mathbb{R}^p \times \mathbb{R}$

2. Prediction rule $\varphi : \mathbb{R}^p \to \mathbb{R}$. Regard $\varphi(X)$ as an estimate of Y

3. Squared loss $\ell(y', y) = (y' - y)^2$, error when y' used to predict y

4. Risk of prediction rule φ is its expected loss

$$R(\varphi) = \mathbb{E}\,\ell(\varphi(X), Y) = \mathbb{E}(\varphi(X) - Y)^2$$

Overall goal: Find a prediction rule φ with small risk $R(\varphi)$

Optimal Prediction and the Regression Function

Fact: Under the squared loss the risk of any fixed rule φ is

$$R(\varphi) \;=\; \mathbb{E}[\varphi(X) - \mathbb{E}(Y|X)]^2 \;+\; \mathbb{E}[\mathbb{E}(Y|X) - Y]^2$$

Optimal prediction rule is *regression function* $f(x) = \mathbb{E}(Y|X = x)$

Signal Plus Noise Model: Assume for some function $f : \mathbb{R}^p \to \mathbb{R}$

$$Y = f(X) + \varepsilon \;\text{ where }\; \mathbb{E}\,\varepsilon = 0 \;\text{ and }\; \varepsilon \perp\!\!\!\perp X$$

In this case f is the regression function, and for every prediction rule φ

$$R(\varphi) \;=\; \mathbb{E}(\varphi(X) - f(X))^2 \;+\; \mathrm{Var}(\varepsilon)$$

Regression Procedures and Empirical Risk

Observations: $D_n = (X_1, Y_1), \ldots, (X_n, Y_n) \in \mathbb{R}^p \times \mathbb{R}$ iid copies of (X, Y)

Definition

- A *regression procedure* is a map $\varphi_n : \mathbb{R}^p \times (\mathbb{R}^p \times \mathbb{R})^n \to \mathbb{R}$
- Let $\hat{\varphi}_n(x) := \varphi_n(x : D_n)$ be the prediction rule based on D_n

Definition: The *empirical risk* or *training error* of a rule φ is given by

$$\hat{R}_n(\varphi) \;=\; \frac{1}{n} \sum_{i=1}^{n} (Y_i - \varphi(X_i))^2$$

Linear Regression

Encompasses assumptions about data generation and prediction

- Linear models: How data is generated
- Linear prediction rules: How data is fit

Linear Regression Model

Model: For some coefficient vector $\beta = (\beta_0, \beta_1, \ldots, \beta_p)^t \in \mathbb{R}^{p+1}$

$$Y \;=\; \beta_0 + \sum_{j=1}^{p} X_j \beta_j + \varepsilon \;=\; \langle \beta, X \rangle + \varepsilon$$

where we assume that

- ε is independent of augmented feature vector $X = (1, X_1, \ldots, X_p)^t$
- $\mathbb{E}\varepsilon = 0$ and $\mathrm{Var}(\varepsilon) = \sigma^2$

Note: *No assumption* about distribution of feature vector X

Flexibility of Linear Model (from ESL)

Flexibility arises from latitude in *defining the features* of $X = (1, X_1, \ldots, X_p)^t$

Features can include

- Any numerical quantity (possibly taking a finite number of values)
- Transformations (square root, log, square) of numerical quantities
- Polynomial ($X_2 = X_1^2$, $X_3 = X_1^3$) or basis expansions of other features
- Dummy variables to code qualitative inputs
- Variable interactions, e.g., $X_3 = X_1 \cdot X_2$ or $X_3 = \mathbb{I}(X_1 \geq 0, X_2 \geq 0)$

Linear Rules and Procedures

Definition

- *Linear prediction rule* has form $\varphi_\beta(x) = x^t\beta$ for some $\beta \in \mathbb{R}^{p+1}$
- *Linear procedure* φ_n produces linear rules from observations D_n

Notation: Linear rule φ_β fully determined by coefficient vector β. Write

- $R(\beta) \;=\; \mathbb{E}(Y - X^t\beta)^2$
- $\hat{R}_n(\beta) \;=\; n^{-1}\sum_{i=1}^n (Y_i - X_i^t\beta)^2$

Different Settings, Different Assumptions

Fitting: Fitting linear models

- Data $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$ is fixed, non-random
- No assumption about underlying distribution(s)

Inference: Concerning coefficients from OLS, Ridge, LASSO

- $y_i = \mathbf{x}_i^t \beta + \varepsilon_i$ with $\mathbf{x}_j$ fixed and ε_j iid $\sim \mathcal{N}(0, \sigma^2)$
- Conditions on feature vectors $\mathbf{x}_j$ (design matrix)

Assessment: Test error, cross-validation

- Observations (X_i, Y_i) are iid copies of (X, Y)

Ordinary Least Squares (OLS)

Ordinary Least Squares (OLS)

Given: Paired observations $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n) \in \mathbb{R}^{p+1} \times \mathbb{R}$ define

- Response vector $\mathbf{y} = (y_1, \ldots, y_n)^t$
- Design matrix $\mathbf{X} \in \mathbb{R}^{n \times (p+1)}$ with ith row $\mathbf{x}_i^t$

OLS: Identify the vector $\hat{\beta}$ minimizing the residual sum of squares (RSS)

$$n\,\hat{R}_n(\beta) \;=\; \sum_{i=1}^{n} (y_i - \mathbf{x}_i^t \beta)^2 \;=\; ||\mathbf{y} - \mathbf{X}\beta||^2$$

Interpretation: Projecting $\mathbf{y}$ onto subspace of $\mathbb{R}^n$ spanned by columns of $\mathbf{X}$, which correspond to features of the data

Least Squares Estimation of Coefficient Vector

Fact: If $\text{rank}(\mathbf{X}) = p$ then $\hat{R}_n(\beta)$ is strictly convex and has unique minimizer

$$\hat{\beta} = (\mathbf{X}^t\mathbf{X})^{-1}\mathbf{X}^t\mathbf{y} \quad \text{(normal equations)}$$

- Minimization problem has closed form solution
- Assumption $\text{rank}(\mathbf{X}) = p$ ensures $\mathbf{X}^t\mathbf{X}$ is invertible, requires $n \geq p$
- Solution $\hat{\beta}$ yields linear prediction rule $\varphi_{\hat{\beta}}(x) = \langle \hat{\beta}, x \rangle$
- Fitted value of the response $\mathbf{y}$ is the projection $\hat{\mathbf{y}} = \mathbf{X}\hat{\beta}$

Gaussian Linear Model

Assume feature vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n$ are fixed and that responses y_i follow linear model with normal errors

$$y_i = \mathbf{x}_i^t \beta + \varepsilon_i \text{ with } \varepsilon_i \text{ iid } \mathcal{N}(0, \sigma^2)$$

Model can be written in vector form $\mathbf{y} = \mathbf{X}\beta + \varepsilon$ with $\varepsilon \sim \mathcal{N}_n(0, \sigma^2 I)$

Fact: Estimate $\hat{\beta} = (\mathbf{X}^t\mathbf{X})^{-1}\mathbf{X}^t\mathbf{y}$ has following properties

1. $\mathbb{E}\hat{\beta} = \beta$ and $\mathrm{Var}(\hat{\beta}) = (\mathbf{X}^t\mathbf{X})^{-1}\sigma^2$
2. $\hat{\beta}$ is multivariate normal

Inference for Gaussian Linear Model

1. Prediction error $||\mathbf{y} - \mathbf{X}\hat{\beta}||^2 \sim \sigma^2 \chi^2_{n-p-1}$. Estimate noise variance σ^2 by

$$\hat{\sigma}^2 = \frac{||\mathbf{y} - \mathbf{X}\hat{\beta}||^2}{n-p-1}$$

2. Let $v_j = (\mathbf{X}^t\mathbf{X})^{-1}_{jj}$. If $\beta_j = 0$ then

$$T_j = \frac{\hat{\beta}_j}{\hat{\sigma}\sqrt{v_j}} \sim t_{n-p-1}$$

We can use T_j to test if $\beta_j = 0$. Approximate 95% confidence interval for β_j is

$$(\hat{\beta}_j - 1.96\sqrt{v_j}\,\hat{\sigma}, \hat{\beta}_j + 1.96\sqrt{v_j}\,\hat{\sigma})$$

Ridge Regression

Penalized Linear Regression

Recal: OLS estimate $\hat{\beta}$ depends directly on $(\mathbf{X}^t\mathbf{X})^{-1}$

- Inverse does not exist if $p > n$
- Small eigenvalues resulting from (near) collinearity among features can lead to unstable estimates, unreliable predictions

Alternative: Penalized regression

- Regularize OLS cost function by adding a term that penalizes large coefficients, shrinking estimates towards zero

Ridge Regression

Setting: Paired observations $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n) \in \mathbb{R}^p \times \mathbb{R}$

- Centering: assume $\sum_{i=1}^n \mathbf{x}_i = 0$ and $\sum_{i=1}^n y_i = 0$
- Design matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$ and response vector $\mathbf{y} \in \mathbb{R}^n$

Penalized cost function: For each $\lambda \geq 0$ define

$$\hat{R}_{n,\lambda}(\beta) \;=\; ||\mathbf{y} - \mathbf{X}\beta||^2 \;+\; \lambda\, ||\beta||^2$$

- $||\mathbf{y} - \mathbf{X}\beta||^2$ measures fit of the linear model
- $||\beta||^2$ measures magnitude of coefficient vector
- λ controls tradeoff between fit and magnitude: OLS is case $\lambda = 0$

Ridge Regression, cont.

Fact: If $\lambda > 0$ then $\hat{R}_{n,\lambda}(\beta)$ is strictly convex and has unique minimizer

$$\hat{\beta}_\lambda = (\mathbf{X}^t\mathbf{X} + \lambda I_p)^{-1}\mathbf{X}^t\mathbf{y}$$

1. Eigenvalues of $\mathbf{X}^t\mathbf{X} + \lambda I_p$ = eigenvalues of $\mathbf{X}^t\mathbf{X}$ plus λ
2. If $\lambda > 0$ then $\mathbf{X}^t\mathbf{X} + \lambda I_p > 0$ is invertible so $\hat{\beta}_\lambda$ is well defined
3. If $\lambda_1 \leq \lambda_2$ then $||\hat{\beta}_{\lambda_2}|| \leq ||\hat{\beta}_{\lambda_1}||$: penalty shrinks $\hat{\beta}_\lambda$ towards zero

Note

- Ridge procedure yields linear rule $\varphi_{\hat{\beta}_\lambda}(\mathbf{x}) = \langle \mathbf{x}, \hat{\beta}_\lambda \rangle$
- Ridge regression is really a *family* of procedures, one for each λ

Selecting Penalty Parameter

Issue: Different parameters λ give different solutions $\hat{\beta}_\lambda$. How to choose λ?

- Fix "grid" $\Lambda = \{\lambda_1, \ldots, \lambda_N\}$ of parameter values

Approach 1. Independent training set D_n and test set D_m

- Find vectors $\hat{\beta}_{\lambda_1}, \ldots, \hat{\beta}_{\lambda_N}$ using training set D_n with different λ
- Select vector $\hat{\beta}_{\lambda_\ell}$ minimizing test error $\hat{R}_m(\beta) = m^{-1} \sum_{j=1}^{m} (Y_j - X_j^t \beta)^2$

Approach 2. Cross-validation

- For each $1 \leq \ell \leq N$ evaluate cross-validated risk $\hat{R}^{\text{k-CV}}(\text{Ridge}(\lambda_\ell))$
- Select vector $\hat{\beta}_{\lambda_\ell}$ for which λ_ℓ minimizes cross-validated risk

Ridge Regression and Linear Model

Setting: Suppose $\mathbf{y} = \mathbf{X}\beta + \varepsilon$ with $\mathbf{X}$ fixed and $\varepsilon \sim \mathcal{N}_n(0, \sigma^2 I)$

Ridge estimate $\hat{\beta}_\lambda$ shrinks OLS estimate $\hat{\beta}$ towards zero. For $\lambda > 0$

- Increased bias $\mathbb{E}\hat{\beta}_\lambda \neq \beta$
- Reduced variance $\mathrm{Var}(\hat{\beta}_\lambda) < \mathrm{Var}(\hat{\beta})$

Appropriate choice of λ can reduce overall mean-squared error, that is,

$$\mathbb{E}||\hat{\beta}_\lambda - \beta||^2 < \mathbb{E}||\hat{\beta} - \beta||^2$$

Model Assessment and Prediction Error

Setting and Assumptions

Additive model: Assume response $Y = f(X) + \varepsilon$

- $f : \mathbb{R}^p \to \mathbb{R}$ fixed but unknown function
- $X \in \mathbb{R}^p$ distribution unspecified
- $\mathbb{E}\varepsilon = 0$ and $\mathrm{Var}(\varepsilon) = \sigma_\varepsilon^2$ unknown

Observations $D_n = \{(X_i, Y_i)\}_{i=1}^n$ with $Y_i = f(X_i) + \varepsilon_i$

Regression procedure $f_n : \mathbb{R}^p \times (\mathbb{R}^p \times \mathbb{R})^n \to \mathbb{R}$ gives rise to estimate

$$\hat{f}_n(x) := f_n(x : D_n)$$

Issue: How to measure the performance of $\hat{f}_n$?

Risky Business

Recall: Common measures of performance

- Conditional risk $R(\hat{f}_n) = \mathbb{E}[(Y - \hat{f}_n(X))^2 \mid D_n]$
- Expected risk $\mathbb{E}R(\hat{f}_n) = \mathbb{E}(Y - \hat{f}_n(X))^2$
- Empirical risk $\hat{R}_n(f_0) = n^{-1} \sum_{i=1}^n (Y_i - f_0(X_i))^2$ of fixed rule f_0
- Training error $\hat{R}_n(\hat{f}_n) =$ empirical risk of estimated rule $\hat{f}_n$

In practice

- Conditional risk $R(\hat{f}_n)$ of interest, but unknown
- Training error $\hat{R}_n(\hat{f}_n)$ is available, but tends to be optimistic, underestimates $R(\hat{f}_n)$

Bias-Variance Decomposition (fixed covariate)

Definition: Expected prediction error of procedure f_n at input $X = x_0$

$$\mathbb{E}R(\hat{f}_n : x_0) \;=\; \mathbb{E}[(Y - \hat{f}_n(X))^2 \,|\, X = x_0]$$

Fact: Quantity $\overline{R}(f_n : x_0)$ can be decomposed as

$$\mathbb{E}R(\hat{f}_n : x_0) \;=\; \sigma_\varepsilon^2 \;+\; (f(x_0) - \mathbb{E}\hat{f}_n(x_0))^2 \;+\; \mathbb{E}(\hat{f}_n(x_0) - \mathbb{E}\hat{f}_n(x_0))^2$$

- $\sigma_\varepsilon^2 =$ irreducible error arising from noise
- $(f(x_0) - \mathbb{E}\hat{f}_n(x_0))^2 =$ squared bias of $\hat{f}_n$ at x_0
- $\mathbb{E}(\hat{f}_n(x_0) - \mathbb{E}\hat{f}_n(x_0))^2 =$ variance of $\hat{f}_n$ at x_0

Generally, increasing complexity of f_n increases variance, decreases bias

Replicate Response Variables

Goal: Quantify and correct optimism of the training error to better assess performance of a regression procedure

Idea: Consider auxiliary data set with with same feature vectors X_i, but *independent responses* Y_i'

$$D_n' \;=\; (X_1, Y_1'), \ldots, (X_n, Y_n')$$

In detail, conditional on $X_1, \ldots, X_n$, the random variables

$$Y_1', \ldots, Y_n', Y_1, \ldots, Y_n \;\text{ are independent with }\; Y_i', Y_i \sim \mathcal{L}(Y|X = X_i)$$

Note: Under the additive noise model we may define $Y_i' = f(X_i) + \varepsilon_i'$ where the noise ε_i' is independent of ε_i.

Replicate Response: Empirical Risk, Performance, Optimism

Definition 1: Empirical risk of rule $\hat{f}_n$ under replicate responses

$$\hat{R}'_n(\hat{f}_n) = n^{-1} \sum_{i=1}^{n} (Y'_i - \hat{f}_n(X_i))^2$$

Definition 2: Overall performance of the procedure f_n on $X_1, \ldots, X_n$

$$\mathrm{Err}'(f_n \mid X_1^n) = \mathbb{E}(\hat{R}'_n(\hat{f}_n) \mid X_1^n)$$

Definition 3: Expected optimism of the procedure f_n on $X_1, \ldots, X_n$

$$\Delta(X_1^n) \;=\; \mathbb{E}\left(\hat{R}'_n(\hat{f}_n) - \hat{R}_n(\hat{f}_n) \mid X_1^n\right)$$

Properties of Optimism $\Delta(X_1^n)$

Fact A: Under the additive model $Y_i = f(X_i) + \varepsilon_i$

1. $\Delta(X_1^n) = 2n^{-1} \sum_{i=1}^n \mathrm{Cov}(\varepsilon_i, \hat{f}_n(X_i) \,|\, X_1^n)$
2. $\Delta(X_1^n) = 2n^{-1} \sum_{i=1}^n \mathrm{Cov}(Y_i, \hat{f}_n(X_i) \,|\, X_1^n)$

Fact B: If $\hat{f}_n(x_i) = (Sy)_i$ where S depends only on $x_1, \ldots, x_n$ then

$$\Delta(X_1^n) = \frac{2\,\mathrm{trace}(S)\,\sigma_\varepsilon^2}{n}$$

Interpret trace(S) as the effective number of parameters of the linear fit

Fact C: If $\hat{f}_n(x) = \langle \hat{\beta}_n, x \rangle$ is the OLS estimator, then $\Delta(X_1^n) = 2p\,\sigma_\varepsilon^2/n$

Using Optimism $\Delta(X_1^n)$

Note: By definition, overall performance of f_n on $X_1, \ldots, X_n$ is

$$\mathrm{Err}'(f_n \mid X_1^n) = \mathbb{E}(\hat{R}_n(\hat{f}_n) \mid X_1^n) + \Delta(X_1^n)$$

Idea: Estimate overall error by

$$\widehat{\mathrm{Err}}'(f_n \mid X_1^n) = \hat{R}_n(\hat{f}_n) + \hat{\Delta}(X_1^n)$$

where $\hat{\Delta}(X_1^n)$ is an estimate of the optimism $\Delta(X_1^n)$. For OLS this gives

$$\widehat{\mathrm{Err}}'(f_n \mid X_1^n) = \hat{R}_n(\hat{f}_n) + \frac{p\,\hat{\sigma}_\varepsilon^2}{n}$$

where $\hat{\sigma}_\varepsilon^2$ is an estimate of the error variance

The LASSO

High Dimensional Linear Regression

Data: Feature-response pairs $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n) \in \mathbb{R}^p \times \mathbb{R}$

- Centering: assume $\sum_{i=1}^n \mathbf{x}_i = 0$ and $\sum_{i=1}^n y_i = 0$
- Design matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$ and response vector $\mathbf{y} \in \mathbb{R}^n$
- Interested in fitting linear model $\mathbf{y} = \mathbf{X}\,\beta + \varepsilon$

Common situation: More features than variables, that is, $p >> n$

- Common in genomics, biomedicine, climatology
- Requires regularization

Sparsity

Assumption: Only a small number s of the p available features are related to the response; the other features are unimportant

- Sparsity assumption approximately true for many data sets
- Implies true coefficient vector β has only s non-zero components
- Shift in focus: value and *identity* of non-zero coefficients of interest
- Number s referred to as the sparsity of the model

Sparse Linear Regression

Common goals

- Prediction: find sparse $\hat{\beta}$ so that $\mathbf{x}^t\hat{\beta}$ close to y for new pair $(\mathbf{x}, y)$
- Feature selection: identify the "true" features, i.e., $\{j : \beta_j \neq 0\}$

Issue: For OLS and Ridge all estimated coefficients are non-zero

LASSO: Least absolute shrinkage and selection operator

- Replace ridge penalty $\sum_{j=1}^{p} \beta_j^2$ by ℓ_1-penalty $\sum_{j=1}^{p} |\beta_j|$
- The ℓ_1 penalty enforces sparsity but preserves convexity

LASSO Regression

Procedure: Given design matrix $\mathbf{X}$, response vector $\mathbf{y}$, and parameter $\lambda \geq 0$, find coefficient vector $\hat{\beta}_\lambda^{\text{LASSO}}$ minimizing

$$\tilde{R}_{n,\lambda}(\beta) \;=\; \frac{1}{2}||\mathbf{y} - \mathbf{X}\beta||^2 \;+\; \lambda\,||\beta||_1$$

- $||\mathbf{y} - \mathbf{X}\beta||^2$ measures fit of linear model
- $||\beta||_1 = \sum_{j=1}^{p} |\beta_j|$ measures magnitude of coefficient vector
- Parameter λ controls tradeoff between fit and magnitude

Key fact: The ℓ_1-penalty forces some coefficients $\hat{\beta}_\lambda^{\text{LASSO}}$ to be *exactly* zero

- Increasing λ tends to increase number of zero coefficients in $\hat{\beta}_\lambda^{\text{LASSO}}$

LASSO Estimation as a Convex Program

Fact A: For every $\lambda \geq 0$ objective $\tilde{R}_{n,\lambda}(\beta)$ is a convex function of β

Fact B: Minimizing $\tilde{R}_{n,\lambda}(\beta)$ is Lagrangian form of the mathematical program

$$\min f(\beta) = ||\mathbf{y} - \mathbf{X}\beta||^2 \text{ subject to } ||\beta||_1 \leq t$$

where t depends on λ. Objective function and constraint set are convex.

Upshot: Zero-ing property follows from *geometry* of the ℓ_1-penalty

Geometry of the L_1 Penalty (from ESL)

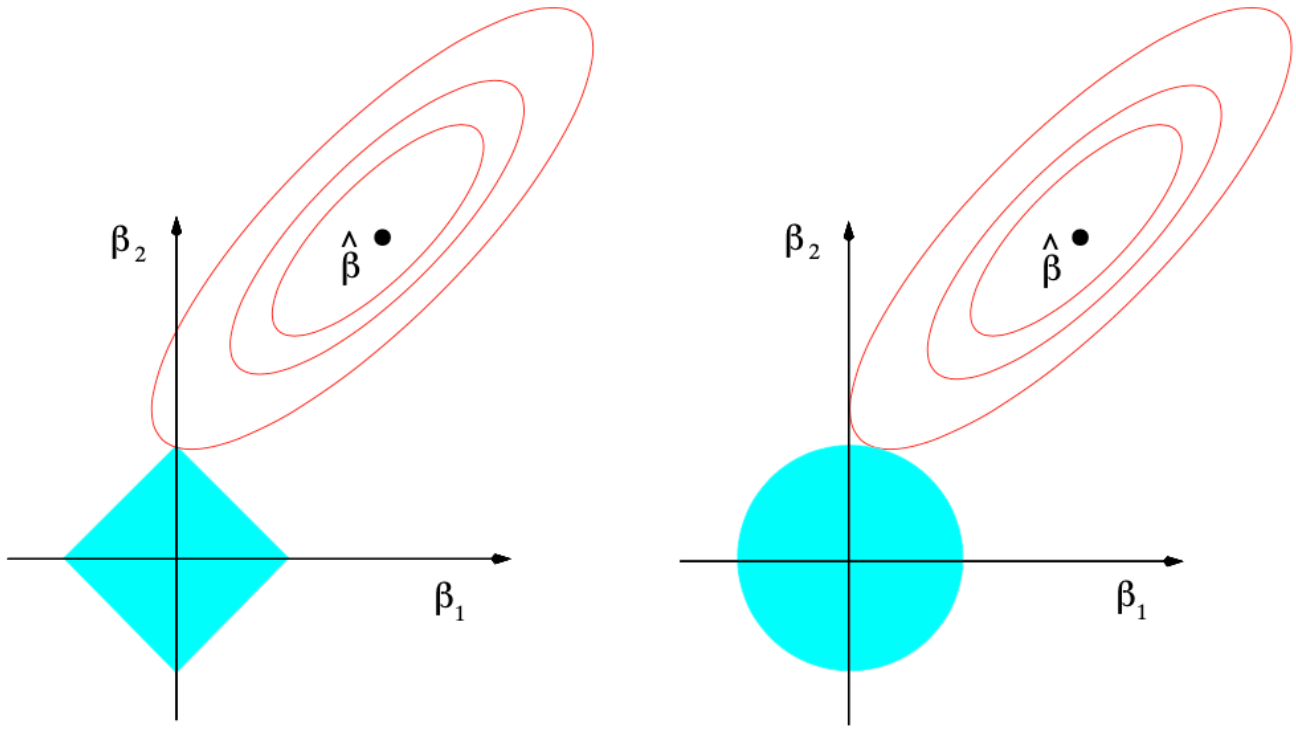


FIGURE 3.11. *Estimation picture for the lasso (left) and ridge regression (right). Shown are contours of the error and constraint functions. The solid blue areas are the constraint regions* $|\beta_1| + |\beta_2| \leq t$ *and* $\beta_1^2 + \beta_2^2 \leq t^2$*, respectively, while the red ellipses are the contours of the least squares error function.*

Selecting Penalty Parameter

Note: Different parameters λ give different solutions $\hat{\beta}_\lambda$. How to choose λ?

- Fix "grid" $\Lambda = \{\lambda_1, \ldots, \lambda_N\}$ of parameter values

Approach 1. Independent training set D_n and test set D_m

- Find vectors $\hat{\beta}_{\lambda_1}, \ldots, \hat{\beta}_{\lambda_N}$ using training set D_n
- Select vector $\hat{\beta}_{\lambda_\ell}$ minimizing test error $\hat{R}_m(\beta)$

Approach 2. Cross-validation

- For each $1 \leq \ell \leq N$ evaluate cross-validated risk $\hat{R}^{\text{k-CV}}(\text{LASSO}(\lambda_\ell))$
- Select vector $\hat{\beta}_{\lambda_\ell}$ for which λ_ℓ minimizes cross-validated risk

Estimating the Penalty Parameter, cont

Idea: If response vector $\mathbf{y}$ is independent of $\mathbf{X}$ then β should be $\mathbf{0}$

Procedure: Do the following 20-30 times

1. Randomly permute the components of $\mathbf{y}$ to get a "dummy response" $\tilde{\mathbf{y}}$
2. Apply LASSO procedure to $(\mathbf{X}, \tilde{\mathbf{y}})$ with different values of λ
3. Let $\tilde{\lambda} =$ smallest λ such that $\hat{\beta}_{\lambda}^{\text{LASSO}}(\mathbf{X}, \tilde{\mathbf{y}}) = 0$

Estimate the penalty parameter $\hat{\lambda}$ by median of the $\tilde{\lambda}$'s

Example: B-cell gene expression data

Background: Data from Basso et al. 2005, Affymetrix microarrays

1. Samples: Samples of $n = 211$ normal and tumor tissue
2. Feature vector: Expression measurements of $p = 6,249$ genes
3. Response: Expression of single `ADA` gene

Question: How does the expression of `ADA` depend on the expression of the 6248 other genes?

OLS Solution

```
R > summary(my_model)

Call:
lm(formula = ADA ~ ., data = gene_expressions)

Residuals:
ALL 211 residuals are 0: no residual degrees of freedom!

Coefficients: (6038 not defined because of singularities)
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)    -412.40983         NA      NA       NA
CDH2             -1.86356         NA      NA       NA
MED6              7.10850         NA      NA       NA
NR2E3            -1.40334         NA      NA       NA
ACOT8             3.48331         NA      NA       NA
ABI1             -5.88529         NA      NA       NA
GNPDA1            0.28055         NA      NA       NA
TANK             -6.02434         NA      NA       NA
HGC6.3           -0.79016         NA      NA       NA
C1orf68          -1.21752         NA      NA       NA
LOC100129361      0.20853         NA      NA       NA
OLFM1                  NA         NA      NA       NA
TIMM17A                NA         NA      NA       NA
N4BP2L2                NA         NA      NA       NA
MCRS1                  NA         NA      NA       NA
 [ reached getOption("max.print") -- omitted 6229 rows ]

Residual standard error: NaN on 0 degrees of freedom
Multiple R-squared:      1, Adjusted R-squared:    NaN
F-statistic:   NaN on 210 and 0 DF,  p-value: NA
```

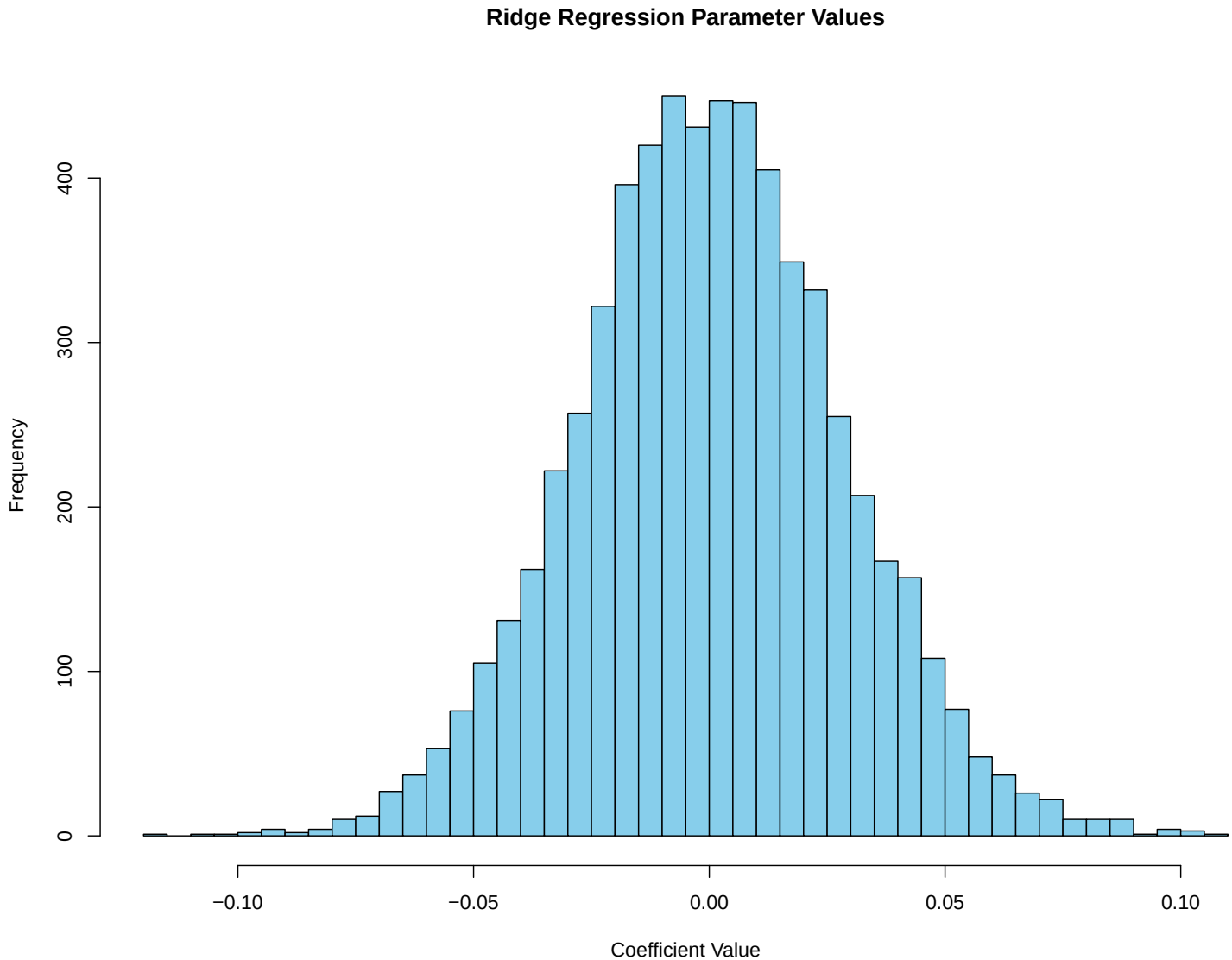
Ridge Solution

- R function selects penalty parameter λ based on the variance explained by the first 8 PCs. [1]
- Note: coefficient estimates for every feature are non-zero

```
R > ridge.fit = linearRidge(ADA~., data = gene_expressions)
R > ridge.fit$coef[,"nPCs8"]
          CDH2            MED6           NR2E3           ACOT8
 1.390812e-02 -3.920405e-02  2.380735e-02 -1.577109e-02

  ABI1         GNPDA1           TANK         HGC6.3
 1.902280e-04 -5.952662e-03  1.141530e-02  5.133231e-02

       C1orf68  LOC100129361           CD24           HDAC5
-5.509269e-02 -3.030931e-02 -4.909134e-02 -5.526016e-03

PDCD6         BCL2L11          SH2B3            GNE
1.990365e-02  2.167638e-02 -3.561387e-02 -1.047401e-01
 [ reached getOption("max.print") -- omitted 6232 entries ]

 R > length(which(coef(ridge.fit) == 0))
[1] 0
```

[1] From Cule & De Iorio (2012). A semi-automatic method to guide the choice of ridge parameter in ridge regression

Histogram of Ridge Coefficients



LASSO Solution

- `R` function selects penalty parameter λ using 10-fold CV

```
R > lasso.fit = Lasso(as.matrix(gene_expressions)[,-1], as.matrix(gene_
    expressions)[,1], fix.lambda = FALSE)
R > lasso.fit
$beta0
[1] 11.85041
$beta
   [1]  0.000000000  0.000000000  0.000000000  0.000000000  0.000000000
        0.000000000  0.000000000  0.000000000  0.000000000
  [10]  0.000000000  0.000000000  0.000000000  0.000000000  0.000000000
        0.000000000 -0.143888739  0.000000000  0.000000000
  [19]  0.113587487  0.000000000  0.000000000  0.000000000  0.000000000
        0.000000000  0.000000000  0.000000000  0.000000000
  [28]  0.000000000  0.000000000  0.000000000  0.000000000  0.000000000
        0.000000000  0.000000000  0.000000000  0.000000000
  [ reached getOption("max.print") -- omitted 6212 entries ]
$lambda
[1] 0.09383066
```

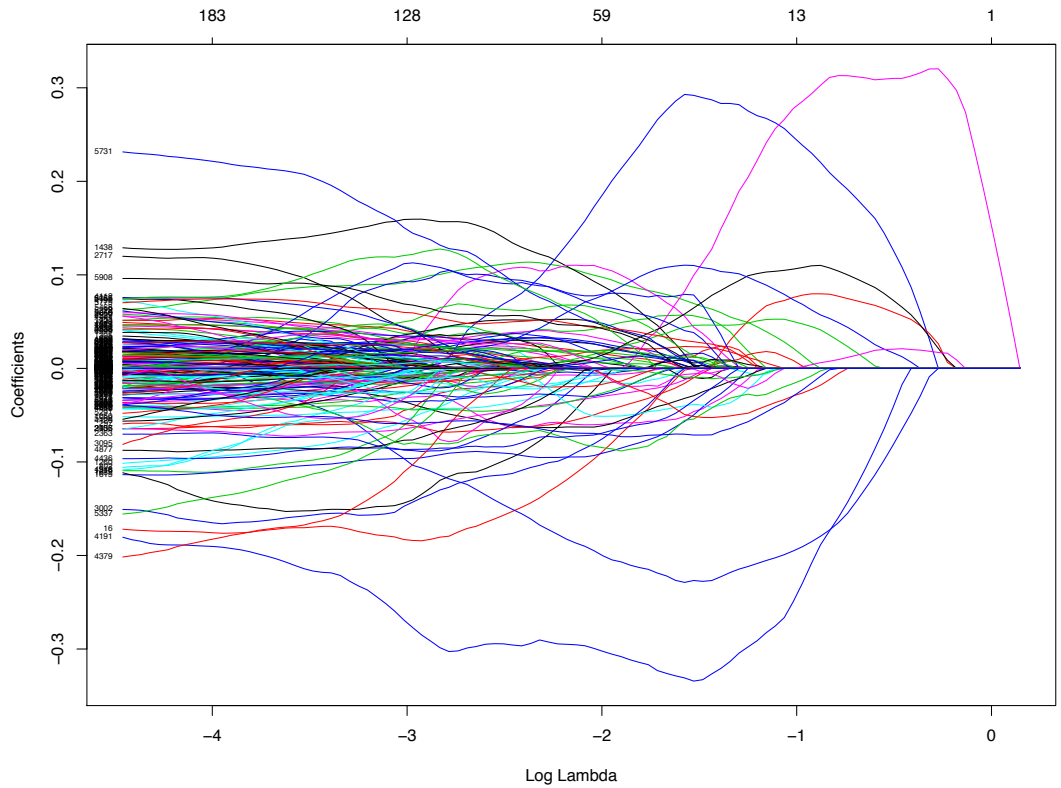
LASSO Solution Cont.

- LASSO sets most coefficients to zero. Only 84 are non-zero.

```
1  R > length(which(lasso.fit$beta != 0))
2  [1] 84
3  R > colnames(gene_expressions)[which(lasso.fit$beta != 0)]
4    [1] "SH2B3"      "PIGK"       "ACTR2"      "MBNL2"      "POP7"       "RRAGB"
                "RBM14"      "FBLN5"      "RAD51AP1"   "RALBP1"
5   [11] "GLMN"       "FILIP1L"    "AP2S1"      "CLCN4"      "ZNF384"     "DLG1"
                "AGXT"       "EPHA7"      "F12"        "FABP4"
6   [21] "FCN1"       "ABCF1"      "TMCC1"      "PDS5B"      "ZHX3"       "SEPT6"
                "RRS1"       "SCFD1"      "MCF2L"      "KHNYN"
7   [31] "COG4"       "ODZ4"       "GCG"        "PELP1"      "AHDC1"      "RNF115"
                "GNAT2"      "ANGPT2"     "GUCA2A"     "GZMB"
8   [41] "HBD"        "HLA.DPA1"   "HSD17B1"    "IDH3B"      "ACADS"      "AQP1"
                "ITGA1"      "L1CAM"      "ST20"       "MSMB"
9   [51] "MYO6"       "NFATC1"     "KRT76"      "FAM8A1"     "PIK3C2B"    "SSH1"
                "ZNF821"     "PSG11"      "PTHLH"      "GATAD1"
10  [61] "RAD52"      "RGS16"      "BCL9"       "RPS4X"      "RPS27"      "CCL5"
                "SLC4A3"     "SNAPC1"     "BTG1"       "UBE2E1"
11  [71] "VRK1"       "ZNF23"      "ZNF76"      "DDX39B"     "ACTL6A"     "VNN2"
                "WASF1"      "CD1D"       "MS4A3"      "NRXN1"
12  [81] "TMPRSS11D"  "POLR1C"     "MDC1"       "TMED10"
```

LASSO Solution Cont.

```
1 R > model <- cv.glmnet(as.matrix(gene_expressions)[,-1], as.matrix(gene_
      expressions)[,1], standardize=TRUE)
2 R > plot(model$glmnet.fit, "lambda", label=TRUE)
```



Some Theory: Error Bounds for the LASSO

MGF Bound on Expected Maxima. Gaussian and General Cases

Fact A: If $X_1, \ldots, X_n \sim \mathcal{N}(0, \sigma^2)$, not necessarily independent, then

$$\mathbb{E} \max(X_1, \ldots, X_n) \leq \sigma \sqrt{2 \log n}$$

Bound continues to hold if $X_i \sim \mathcal{N}(0, \sigma_i^2)$ with $\sigma_i^2 \leq \sigma^2$

Fact B: If $X_1, \ldots, X_n$ are such that $M_{X_i}(s) \leq M(s)$ for all $s \geq 0$ then

$$\mathbb{E} \max(X_1, \ldots, X_n) \ \leq \ \inf_{s : M(s) \geq 1} \frac{\log n + \log M(s)}{s}$$

Projections, Convex Sets, and Oblique Angles

Fact: Let $C \subseteq \mathbb{R}^d$ be a compact, convex set. Let $y \in \mathbb{R}^d$ and let

$$\hat{y} \;=\; \operatorname*{argmin}_{u \in C} ||u - y||$$

be the projection of y onto C. Then $\langle u - \hat{y}, y - \hat{y} \rangle \leq 0$ for every $u \in C$

Cor: Let C, y, and $\hat{y}$ be as above. Then for every $u \in C$

$$||u - \hat{y}||^2 \;\leq\; \langle u - y, u - \hat{y} \rangle$$

Setting for LASSO Error Bounds

Observations: $(X_1, Y_1), \ldots, (X_n, Y_n)$ iid copies of $(X, Y) \in \mathbb{R}^p \times \mathbb{R}$

- Each component of X bounded in absolute value by $M < \infty$
- $Y = \langle \beta^*, X \rangle + \varepsilon$ where $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ is independent of X
- Let $\mathbf{X} \in \mathbb{R}^{n \times p}$ be the design matrix, $\mathbf{Y} \in \mathbb{R}^n$ the response vector

Consider the constrained version of the LASSO regression estimate

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} ||\mathbf{Y} - \mathbf{X}\beta|| \text{ subject to } ||\beta||_1 \leq K$$

where $||\beta||_1 = \sum_{j=1}^{p} |\beta_j|$ is the usual ℓ_1-norm

Finite Sample LASSO Error Bound

Theorem: Let $\hat{\beta}_n$ be LASSO coefficient estimate with constraint K. If true coefficient vector β^* satisfies $||\beta^*||_1 \leq K$ then

$$\mathbb{E}||\mathbf{X}\beta^* - \mathbf{X}\hat{\beta}_n||^2 \ \leq \ 2KM\sigma\sqrt{2n\log(2p)}$$

Cor: Let $\hat{\mathbf{Y}} := \mathbf{X}\hat{\beta}_n$ be the predicted response vector based on $\hat{\beta}_n$. If true coefficient vector β^* satisfies $||\beta^*||_1 \leq K$ then

$$\frac{1}{n}\,\mathbb{E}||\mathbf{Y} - \hat{\mathbf{Y}}||^2 \ \leq \ 2KM\sigma\sqrt{\frac{2\log(2p)}{n}} \ + \ \sigma^2$$

Finite Sample LASSO Error Bound

Note: The theorem and corollary give finite sample bounds that hold for every sample size n, dimension p, and constraint K.

In principle, dimension p and constraint K can grow with sample size n. If $K = o((n/\log p)^{1/2})$ then as n tends to infinity

$$\frac{1}{n}\,\mathbb{E}||\mathbf{Y} - \hat{\mathbf{Y}}||^2 \;\rightarrow\; \text{minimum value } (\sigma^2)$$