

Econ 722 – Advanced Econometrics IV, Part II

Francis J. DiTraglia

University of Pennsylvania

Lecture #8 – High-Dimensional Regression I

The James-Stein Estimator

QR Decomposition

Singular Value Decomposition

Review of Principal Component Analysis (PCA)

Recall: Gauss-Markov Theorem

Linear Regression Model

$$\mathbf{y} = X\beta + \epsilon, \quad \mathbb{E}[\epsilon|X] = \mathbf{0}$$

Best Linear Unbiased Estimator

- ▶ $\text{Var}(\epsilon|X) = \sigma^2 I \Rightarrow$ then OLS has lowest variance among linear, unbiased estimators of β .
- ▶ $\text{Var}(\epsilon|X) \neq \sigma^2 I \Rightarrow$ then GLS gives a lower variance estimator.

What if we consider biased estimators?

Dominance and Admissibility

Notation

Let R be a risk function, e.g. MSE, and $\hat{\theta}$ and $\tilde{\theta}$ be estimators of θ .

Dominance

We say that $\hat{\theta}$ **dominates** $\tilde{\theta}$ with respect to R if $R(\hat{\theta}, \theta) \leq R(\tilde{\theta}, \theta)$ for all $\theta \in \Theta$ and the inequality is strict for at least one value of θ .

Admissibility

We say that $\hat{\theta}$ is **admissible** if no other estimator dominates it.

Inadmissibility

To prove that an estimator $\tilde{\theta}$ is **inadmissible** it suffices to find an estimator $\hat{\theta}$ that dominates it.

A Very Simple Example: $X \sim N(\theta, I)$

Goal

Estimate the p -vector of unknown parameters θ using X .

Maximum Likelihood Estimator $\hat{\theta}$

MLE = sample mean, but only one observation: $\hat{\theta} = X$.

MSE of $\hat{\theta}$

$$(\hat{\theta} - \theta)' (\hat{\theta} - \theta) = (X - \theta)' (X - \theta) = \sum_{i=1}^p (X_i - \theta_i)^2 \sim \chi_p^2$$

Since $\mathbb{E}[\chi_p^2] = p$, we have $MSE(\hat{\theta}) = p$.

A Very Simple Example: $X \sim N(\theta, I)$

James-Stein Estimator

$$\hat{\theta}^{JS} = \hat{\theta} \left(1 - \frac{p-2}{\hat{\theta}'\hat{\theta}} \right) = X - \frac{(p-2)X}{X'X}$$

- ▶ Shrinks components of sample mean vector towards zero
- ▶ More elements in $\theta \Rightarrow$ more shrinkage
- ▶ MLE close to zero ($\hat{\theta}'\hat{\theta}$ small) gives more shrinkage

MSE of James-Stein Estimator

$$\begin{aligned}MSE(\hat{\theta}^{JS}) &= \mathbb{E} \left[(\hat{\theta}^{JS} - \theta)' (\hat{\theta}^{JS} - \theta) \right] \\&= \mathbb{E} \left[\left\{ (X - \theta) - \frac{(p-2)X}{X'X} \right\}' \left\{ (X - \theta) - \frac{(p-2)X}{X'X} \right\} \right] \\&= \mathbb{E} [(X - \theta)' (X - \theta)] - 2(p-2) \mathbb{E} \left[\frac{X'(X - \theta)}{X'X} \right] \\&\quad + (p-2)^2 \mathbb{E} \left[\frac{1}{X'X} \right] \\&= p - 2(p-2) \mathbb{E} \left[\frac{X'(X - \theta)}{X'X} \right] + (p-2)^2 \mathbb{E} \left[\frac{1}{X'X} \right]\end{aligned}$$

Using fact that $MSE(\hat{\theta}) = p$

Simplifying the Second Term

Writing Numerator as a Sum

$$\mathbb{E} \left[\frac{X'(X - \theta)}{X'X} \right] = \mathbb{E} \left[\frac{\sum_{i=1}^p X_i (X_i - \theta_i)}{X'X} \right] = \sum_{i=1}^p \mathbb{E} \left[\frac{X_i (X_i - \theta_i)}{X'X} \right]$$

For $i = 1, \dots, p$

$$\mathbb{E} \left[\frac{X_i (X_i - \theta_i)}{X'X} \right] = \mathbb{E} \left[\frac{X'X - 2X_i^2}{(X'X)^2} \right]$$

Not obvious: integration by parts, expectation as a p -fold integral, $X \sim N(\theta, I)$

Combining

$$\begin{aligned} \mathbb{E} \left[\frac{X'(X - \theta)}{X'X} \right] &= \sum_{i=1}^p \mathbb{E} \left[\frac{X'X - 2X_i^2}{(X'X)^2} \right] = p \mathbb{E} \left[\frac{1}{X'X} \right] - 2 \mathbb{E} \left[\frac{\sum_{i=1}^p X_i^2}{(X'X)^2} \right] \\ &= p \mathbb{E} \left[\frac{1}{X'X} \right] - 2 \mathbb{E} \left[\frac{X'X}{(X'X)^2} \right] = (p - 2) \mathbb{E} \left[\frac{1}{X'X} \right] \end{aligned}$$

The MLE is Inadmissible when $p \geq 3$

$$\begin{aligned} \text{MSE}(\hat{\theta}^{JS}) &= p - 2(p-2) \left\{ (p-2) \mathbb{E} \left[\frac{1}{X'X} \right] \right\} + (p-2)^2 \mathbb{E} \left[\frac{1}{X'X} \right] \\ &= p - (p-2)^2 \mathbb{E} \left[\frac{1}{X'X} \right] \end{aligned}$$

- ▶ $\mathbb{E}[1/(X'X)]$ exists and is positive whenever $p \geq 3$
- ▶ $(p-2)^2$ is always positive
- ▶ Hence, second term in the MSE expression is *negative*
- ▶ First term is MSE of the MLE

Therefore James-Stein strictly dominates MLE whenever $p \geq 3$!

James-Stein More Generally

- ▶ Our example was specific, but the result is general:
 - ▶ MLE is inadmissible under quadratic loss in regression model with at least three regressors.
 - ▶ Note, however, that this is MSE for the *full parameter vector*
- ▶ James-Stein estimator is also inadmissible!
 - ▶ Dominated by “positive-part” James-Stein estimator:

$$\hat{\beta}^{JS} = \hat{\beta} \left[1 - \frac{(p-2)\hat{\sigma}^2}{\hat{\beta}'X'X\hat{\beta}} \right]_+$$

- ▶ $\hat{\beta} = \text{OLS}$, $(x)_+ = \max(x, 0)$, $\hat{\sigma}^2 = \text{usual OLS-based estimator}$
- ▶ Stops us from shrinking *past* zero to get a negative estimate for an element of β with a small OLS estimate.
- ▶ Positive-part James-Stein isn't admissible either!

QR Decomposition

Result

Any $n \times k$ matrix A with full column rank can be decomposed as $A = QR$, where R is an $k \times k$ upper triangular matrix and Q is an $n \times k$ matrix with orthonormal columns.

Notes

- ▶ Columns of A are *orthogonalized* in Q via Gram-Schmidt.
- ▶ Since Q has orthogonal columns, $Q'Q = I_k$.
- ▶ It is *not* in general true that $QQ' = I$.
- ▶ If A is square, then $Q^{-1} = Q'$.

Different Conventions for the QR Decomposition

Thin aka Economical QR

Q is an $n \times k$ with orthonormal columns (`qr_econ` in Armadillo).

Thick QR

Q is an $n \times n$ *orthogonal* matrix.

Relationship between Thick and Thin

Let $A = QR$ be the “thick” QR and $A = Q_1 R_1$ be the “thin” QR:

$$A = QR = Q \begin{bmatrix} R_1 \\ 0 \end{bmatrix} = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \begin{bmatrix} R_1 \\ 0 \end{bmatrix} = Q_1 R_1$$

My preferred convention is the thin QR...

Least Squares via QR Decomposition

Let $X = QR$

$$\begin{aligned}\hat{\beta} &= (X'X)^{-1}X'y = [(QR)'(QR)]^{-1}(QR)'y \\ &= [R'Q'QR]^{-1}R'Q'y = (R'R)^{-1}R'Qy \\ &= R^{-1}(R')^{-1}R'Q'y = R^{-1}Q'y\end{aligned}$$

In other words, $\hat{\beta}$ solves $R\beta = Q'y$.

Why Bother?

Much easier and faster to solve $R\beta = Q'y$ than the normal equations $(X'X)\beta = X'y$ since R is **upper triangular**.

Back-Substitution to Solve $R\beta = Q'y$

The product $Q'y$ is a vector, call it v , so the system is simply

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} & \cdots & r_{1,n-1} & r_{1k} \\ 0 & r_{22} & r_{23} & \cdots & r_{2,n-1} & r_{2k} \\ 0 & 0 & r_{33} & \cdots & r_{3,n-1} & r_{3k} \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & r_{k-1,k-1} & r_{k-1,k} \\ 0 & 0 & \cdots & 0 & 0 & r_k \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \vdots \\ \beta_{k-1} \\ \beta_k \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_{k-1} \\ v_k \end{bmatrix}$$

$\beta_k = v_k/r_k \Rightarrow$ substitute this into $\beta_{k-1}r_{k-1,k-1} + \beta_k r_{k-1,k} = v_{k-1}$
to solve for β_{k-1} , and so on.

Calculating the Least Squares Variance Matrix $\sigma^2(X'X)^{-1}$

- ▶ Since $X = QR$, $(X'X)^{-1} = R^{-1}(R^{-1})'$
- ▶ Easy to invert R : just apply **repeated** back-substitution:
 - ▶ Let $A = R^{-1}$ and \mathbf{a}_j be the j th column of A .
 - ▶ Let \mathbf{e}_j be the j th standard basis vector.
 - ▶ Inverting R is equivalent to solving $R\mathbf{a}_1 = \mathbf{e}_1$, followed by $R\mathbf{a}_2 = \mathbf{e}_2, \dots, R\mathbf{a}_k = \mathbf{e}_k$.
- ▶ If you enclose a matrix in `trimatu()` or `trimatl()`, and request the inverse \Rightarrow Armadillo will carry out backward or forward substitution, respectively.

QR Decomposition for Orthogonal Projections

Let X have full column rank and define $P_X = X(X'X)^{-1}X'$

$$P_X = QR(R'R)^{-1}R'Q' = QRR^{-1}(R')^{-1}R'Q' = QQ'$$

It is *not* in general true that $QQ' = I$ even though $Q'Q = I$ since Q need not be square in the economical QR decomposition.

The Singular Value Decomposition (SVD)

Any $m \times n$ matrix A of arbitrary rank r can be written

$$X = UDV' = (\text{orthogonal})(\text{diagonal})(\text{orthogonal})$$

- ▶ $U = m \times m$ orthog. matrix whose cols contain e-vectors of AA'
- ▶ $V = n \times n$ orthog. matrix whose cols contain e-vectors of $A'A$
- ▶ $D = m \times n$ matrix whose first r main diagonal elements are the *singular values* d_1, \dots, d_r . All other elements are zero.
- ▶ The singular values d_1, \dots, d_r are the square roots of the non-zero eigenvalues of $A'A$ and AA' .
- ▶ (E-values of $A'A$ and AA' could be zero but not negative)

SVD for Symmetric Matrices

If A is **symmetric** then $A = Q\Lambda Q'$ where Λ is a diagonal matrix containing the e-values of A and Q is an orthonormal matrix whose columns are the corresponding e-vectors. Accordingly:

$$AA' = (Q\Lambda Q')(Q\Lambda Q')' = Q\Lambda Q'Q\Lambda Q' = Q\Lambda^2 Q'$$

and similarly

$$A'A = (Q\Lambda Q')'(Q\Lambda Q') = Q\Lambda Q'Q\Lambda Q' = Q\Lambda^2 Q'$$

using the fact that Q is orthogonal and Λ diagonal. Thus, when A is symmetric the SVD reduces to $U = V = Q$ and $D = \sqrt{\Lambda^2}$ so that *negative* eigenvalues become *positive* singular values.

The Economical SVD

- ▶ Number of singular values is $r = \text{Rank}(A) \leq \max\{m, n\}$
- ▶ Some cols of U or V multiplied by zeros in D
- ▶ Economical SVD: only keep columns in U and V that are multiplied by non-zeros in D (Armadillo: `svd_econ`)
- ▶ Summation form: $A = \sum_{i=1}^r d_i \mathbf{u}_i \mathbf{v}_i'$ where $d_1 \leq d_2 \leq \dots \leq d_r$
- ▶ Matrix form:
$$\underset{(n \times p)}{A} = \underset{(n \times r)}{U} \underset{(r \times r)}{D} \underset{(r \times p)}{V'}$$

In the economical SVD, U and V may no longer be square, so they are not orthogonal matrices but their *columns* are still orthonormal.

Principal Component Analysis (PCA)

Notation

Let \mathbf{x} be a $p \times 1$ random vector with variance-covariance matrix Σ .

Optimization Problem

$$\alpha_1 = \arg \max_{\alpha} \text{Var}(\alpha' \mathbf{x}) \quad \text{subject to} \quad \alpha' \alpha = 1$$

First Principal Component

The linear combination $\alpha_1' \mathbf{x}$ is the **first principal component** of \mathbf{x} .

It is the direction along with \mathbf{x} has **maximal variation**

Solving for α_1

Lagrangian

$$\mathcal{L}(\alpha_1, \lambda) = \alpha' \Sigma \alpha - \lambda(\alpha' \alpha - 1)$$

First Order Condition

$$2(\Sigma \alpha_1 - \lambda \alpha_1) = 0 \iff (\Sigma - \lambda I_p) \alpha_1 = 0 \iff \Sigma \alpha_1 = \lambda \alpha_1$$

Variance of 1st PC

α_1 is an e-vector of Σ but which one? Substituting,

$$\text{Var}(\alpha'_1 \mathbf{x}) = \alpha'_1 (\Sigma \alpha_1) = \lambda \alpha'_1 \alpha_1 = \lambda$$

Solution

Var. of 1st PC equals λ and this is what we want to **maximize**, so

α_1 is the e-vector corresponding to the largest e-value.

Subsequent Principal Components

Additional Constraint

Construct 2nd PC by solving the same problem as before with the additional constraint that $\alpha'_2 \mathbf{x}$ is uncorrelated with $\alpha'_1 \mathbf{x}$.

j th Principal Component

The linear combination $\alpha'_j \mathbf{x}$ where α_j is the e-vector corresponding to the j th largest e-value of Σ .

Sample PCA

Notation

$X = (n \times p)$ **centered** data matrix – columns are mean zero.

SVD

$$X = UDV', \text{ thus } X'X = VDU'UDV' = VD^2V'$$

Sample Variance Matrix

$S = n^{-1}X'X$ has same e-vectors as $X'X$ – the columns of V !

Sample PCA

Let \mathbf{v}_j be the j th column of V . Then,

\mathbf{v}_j = PC loadings for j th PC of S

$\mathbf{v}_j' \mathbf{x}_i$ = PC score for individual/time period i

Sample PCA

PC scores for j th PC

$$\mathbf{z}_j = \begin{bmatrix} z_{j1} \\ \vdots \\ z_{jn} \end{bmatrix} = \begin{bmatrix} \mathbf{v}_j' \mathbf{x}_1 \\ \vdots \\ \mathbf{v}_j' \mathbf{x}_n \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1' \mathbf{v}_j \\ \vdots \\ \mathbf{x}_n' \mathbf{v}_j \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1' \\ \vdots \\ \mathbf{x}_n' \end{bmatrix} \mathbf{v}_j = X \mathbf{v}_j$$

Getting PC Scores from SVD

Since $X = UDV'$ and $V'V = I$, $XV = UD$, i.e.

$$\begin{bmatrix} \mathbf{x}_1' \\ \vdots \\ \mathbf{x}_n' \end{bmatrix} \begin{bmatrix} \mathbf{v}_1 & \cdots & \mathbf{v}_p \end{bmatrix} = \begin{bmatrix} \mathbf{u}_1 & \cdots & \mathbf{u}_r \end{bmatrix} \begin{bmatrix} d_1 & \cdots & 0 \\ & \ddots & \\ 0 & \cdots & d_r \end{bmatrix}$$

Hence we see that $\mathbf{z}_j = d_j \mathbf{u}_j$

Properties of PC Scores \mathbf{z}_j

Since X has been de-meaned:

$$\bar{z}_j = \frac{1}{n} \sum_{i=1}^n \mathbf{v}_j' \mathbf{x}_i = \mathbf{v}_j' \left(\frac{1}{n} \sum_{i=1}^n \mathbf{x}_i \right) = \mathbf{v}_j' \mathbf{0} = 0$$

Hence, since $X'X = VD^2V'$

$$\frac{1}{n} \sum_{i=1}^n (z_{ji} - \bar{z}_j)^2 = \frac{1}{n} \sum_{i=1}^n z_{ji}^2 = \frac{1}{n} \mathbf{z}_j' \mathbf{z}_j = \frac{1}{n} (X\mathbf{v}_j)' (X\mathbf{v}_j) = \mathbf{v}_j' S \mathbf{v}_j = d_j^2 / n$$

Lecture #9 – High-Dimensional Regression I

Ridge Regression

LASSO

Ridge Regression – OLS with an L_2 Penalty

$$\hat{\beta}_{Ridge} = \arg \min_{\beta} (\mathbf{y} - X\beta)'(\mathbf{y} - X\beta) + \lambda \beta' \beta$$

- ▶ Add a penalty for large coefficients
- ▶ λ = non-negative constant we choose: strength of penalty
- ▶ X and \mathbf{y} assumed to be **de-meaned** (don't penalize intercept)
- ▶ Unlike OLS, Ridge Regression is **not scale invariant**
 - ▶ In OLS if we replace \mathbf{x}_1 with $c\mathbf{x}_1$ then β_1 becomes β_1/c .
 - ▶ The same is not true for ridge regression!
 - ▶ Typical to **standardize** X before carrying out ridge regression

Alternative Formulation of Ridge Regression Problem

$$\hat{\beta}_{Ridge} = \arg \min_{\beta} (\mathbf{y} - X\beta)'(\mathbf{y} - X\beta) \quad \text{subject to} \quad \beta'\beta \leq t$$

- ▶ Ridge Regression is like least squares “on a budget.”
- ▶ Make one coefficient larger \Rightarrow must make another one smaller.
- ▶ One-to-one mapping from t to λ (data-dependend)

Ridge as Bayesian Linear Regression

If we ignore the intercept, which is unpenalized), Ridge Regression gives the **posterior mode** from the Bayesian regression model:

$$\begin{aligned}y|X, \beta, \sigma^2 &\sim N(X\beta, \sigma^2 I_n) \\ \beta &\sim N(\mathbf{0}, \tau^2 I_p)\end{aligned}$$

where σ^2 is assumed known and $\lambda = \sigma^2/\tau^2$. (In this example, the posterior is normal so the mode equals the mean)

Explicit Solution to the Ridge Regression Problem

Objective Function:

$$\begin{aligned}Q(\beta) &= (\mathbf{y} - \mathbf{X}\beta)'(\mathbf{y} - \mathbf{X}\beta) + \lambda\beta'\beta \\&= \mathbf{y}'\mathbf{y} - \beta'\mathbf{X}\mathbf{y} - \mathbf{y}'\mathbf{X}\beta + \beta'\mathbf{X}'\mathbf{X}\beta + \lambda\beta' \mathbf{I}_p\beta \\&= \mathbf{y}'\mathbf{y} - 2\mathbf{y}'\mathbf{X}\beta + \beta'(\mathbf{X}'\mathbf{X} + \lambda\mathbf{I}_p)\beta\end{aligned}$$

Recall the following facts about matrix differentiation

$$\partial(\mathbf{a}'\mathbf{x})/\partial\mathbf{x} = \mathbf{a}, \quad \partial(\mathbf{x}'\mathbf{A}\mathbf{x})/\partial\mathbf{x} = (\mathbf{A} + \mathbf{A}')\mathbf{x}$$

Thus, since $(\mathbf{X}'\mathbf{X} + \lambda\mathbf{I}_p)$ is symmetric,

$$\frac{\partial}{\partial\beta}Q(\beta) = -2\mathbf{X}'\mathbf{y} + 2(\mathbf{X}'\mathbf{X} + \lambda\mathbf{I}_p)\beta$$

Explicit Solution to the Ridge Regression Problem

Previous Slide:

$$\frac{\partial}{\partial \beta} Q(\beta) = -2X'\mathbf{y} + 2(X'X + \lambda I_p)\beta$$

First order condition:

$$X'\mathbf{y} = (X'X + \lambda I_p)\beta$$

Hence,

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

But is $(X'X + \lambda I_p)$ guaranteed to be invertible?

Ridge Regression via OLS with “Dummy Observations”

Ridge regression solution is identical to

$$\arg \min_{\beta} \left(\tilde{\mathbf{y}} - \tilde{X}\beta \right)' \left(\tilde{\mathbf{y}} - \tilde{X}\beta \right)$$

where

$$\tilde{\mathbf{y}} = \begin{bmatrix} \mathbf{y} \\ \mathbf{0}_p \end{bmatrix}, \quad \tilde{X} = \begin{bmatrix} X \\ \sqrt{\lambda} I_p \end{bmatrix}$$

since:

$$\begin{aligned} \left(\tilde{\mathbf{y}} - \tilde{X}\beta \right)' \left(\tilde{\mathbf{y}} - \tilde{X}\beta \right) &= \begin{bmatrix} (\mathbf{y} - X\beta)' & (-\sqrt{\lambda}\beta)' \end{bmatrix} \begin{bmatrix} (\mathbf{y} - X\beta) \\ -\sqrt{\lambda}\beta \end{bmatrix} \\ &= (\mathbf{y} - X\beta)'(\mathbf{y} - X\beta) + \lambda\beta'\beta \end{aligned}$$

Ridge Regression Solution is Always Unique

Ridge solution is **always unique**, even if there are more regressors than observations! This follows from the preceding slide:

$$\hat{\beta}_{Ridge} = \arg \min_{\beta} \left(\tilde{\mathbf{y}} - \tilde{X}\beta \right)' \left(\tilde{\mathbf{y}} - \tilde{X}\beta \right)$$

$$\tilde{\mathbf{y}} = \begin{bmatrix} \mathbf{y} \\ \mathbf{0}_p \end{bmatrix}, \quad \tilde{X} = \begin{bmatrix} X \\ \sqrt{\lambda} I_p \end{bmatrix}$$

Columns of $\sqrt{\lambda} I_p$ are linearly independent, so columns of \tilde{X} are also linearly independent, **regardless** of whether the same holds for the columns of X .

Efficient Calculations for Ridge Regression

QR Decomposition

Write Ridge as OLS with “dummy observations” with $\tilde{X} = QR$ so

$$\hat{\beta}_{Ridge} = (\tilde{X}'\tilde{X})^{-1}\tilde{X}'\tilde{\mathbf{y}} = R^{-1}Q'\tilde{\mathbf{y}}$$

which we can obtain by back-solving the system $R\hat{\beta}_{Ridge} = Q'\tilde{\mathbf{y}}$.

Singular Value Decomposition

If $p \gg n$, it's much faster to use the SVD rather than the QR decomposition because the rank of X will be n . For implementation details, see Murphy (2012; Section 7.5.2).

Comparing Ridge and OLS

Assumption

Centered data matrix $X_{(n \times p)}$ with rank p so OLS estimator is unique.

Economical SVD

- ▶ $X_{(n \times p)} = U_{(n \times p)} D_{(p \times p)} V'_{(p \times p)}$ with $U'U = V'V = I_p$, D diagonal
- ▶ Hence: $X'X = (UDV')'(UDV') = VDU'UDV' = VD^2V'$
- ▶ Since V is square it is an orthogonal matrix: $VV' = I_p$

Comparing Ridge and OLS – The “Hat Matrix”

Using $X = UDV'$ and the fact that V and U are square orthogonal,

$$\begin{aligned} H(\lambda) &= X (X'X + \lambda I_p)^{-1} X' = UDV' (VD^2V + \lambda VV')^{-1} VDU' \\ &= UDV' (VD^2V' + \lambda VV')^{-1} VDU' \\ &= UDV' [V(D^2 + \lambda I_p)V']^{-1} VDU' \\ &= UDV' (V')^{-1} (D^2 + \lambda I_p)^{-1} (V)^{-1} VDU' \\ &= UDV' V (D^2 + \lambda I_p)^{-1} V' VDU' \\ &= UD (D^2 + \lambda I_p)^{-1} DU' \end{aligned}$$

Model Complexity of Ridge Versus OLS

OLS Case

Number of free parameters equals number of parameters p .

Ridge is more complicated

Even though there are p parameters they are **constrained!**

Idea: use trace of $H(\lambda)$

$$\text{df}(\lambda) = \text{tr} \{H(\lambda)\} = \text{tr} \{X(X'X + \lambda I_p)^{-1}X'\}$$

Why? Works for OLS: $\lambda = 0$

$$\text{df}(0) = \text{tr} \{H(0)\} = \text{tr} \{X(X'X)^{-1}X'\} = p$$

Effective Degrees of Freedom for Ridge Regression

Using cyclic permutation property of trace:

$$\begin{aligned}\text{df}(\lambda) &= \text{tr} \{H(\lambda)\} = \text{tr} \{X(X'X + \lambda I_p)^{-1}X'\} \\&= \text{tr} \{UD (D^2 + \lambda I_p)^{-1} DU'\} \\&= \text{tr} \{DU'UD (D^2 + \lambda I_p)^{-1}\} \\&= \text{tr} \{D^2 (D^2 + \lambda I_p)^{-1}\} \\&= \sum_{j=1}^p \frac{d_j^2}{d_j^2 + \lambda}\end{aligned}$$

- ▶ $\text{df}(\lambda) \rightarrow 0$ as $\lambda \rightarrow 0$
- ▶ $\text{df}(\lambda) = p$ when $\lambda = 0$
- ▶ $\text{df}(\lambda) < p$ when $\lambda > 0$

Comparing OLS and Ridge Predictions

$$\begin{aligned}\hat{y}(\lambda) &= X\hat{\beta}(\lambda) = X(X'X + \lambda I_p)^{-1}X' \\ &= H(\lambda) = \left[UD(D^2 + \lambda I_p)^{-1}DU'\right] \mathbf{y} \\ &= \left[\sum_{j=1}^p \mathbf{u}_j \left(\frac{d_j^2}{d_j^2 + \lambda} \right) \mathbf{u}_j' \right] \mathbf{y} = \sum_{j=1}^p \left(\frac{d_j^2}{d_j^2 + \lambda} \right) \mathbf{u}_j \mathbf{u}_j' \mathbf{y}\end{aligned}$$

Comparing OLS and Ridge Predictions

$$\hat{y}(\lambda) = \sum_{j=1}^p \left(\frac{d_j^2}{d_j^2 + \lambda} \right) \mathbf{u}_j \mathbf{u}_j' \mathbf{y}$$

- ▶ Since X is centered, $\mathbf{z}_j = d_j \mathbf{u}_j$ is the j th sample PC
- ▶ d_j^2 is proportional to the **variance** of the j th sample PC
- ▶ Prediction from regression of \mathbf{y} on \mathbf{z}_j is:

$$\mathbf{z}_j (\mathbf{z}_j' \mathbf{z}_j)^{-1} \mathbf{z}_j' \mathbf{y} = d_j \mathbf{u}_j (d_j^2 \mathbf{u}_j' \mathbf{u}_j)^{-1} d_j \mathbf{u}_j' \mathbf{y} = \mathbf{u}_j \mathbf{u}_j' \mathbf{y}$$

- ▶ Ridge equivalent to regressing y on sample PCs of X but shrinking predictions to zero: higher variance PCs are shrunk less.
- ▶ OLS doesn't shrink.

Principal Components Regression (PCR)

Instead of “smooth weights” as in Ridge, truncate the PCs:

1. Calculate SVD $X = UDV'$ of **centered** data matrix X
2. Construct the sample principal components: $\mathbf{z}_j = d_j \mathbf{u}_j$.
3. Throw away all but first M principal components, where $M < p$.
4. Regress \mathbf{y} on $\mathbf{z}_1, \dots, \mathbf{z}_k$.

PCR versus Ridge

- ▶ PCR is a much less smooth version of Ridge
- ▶ Conventional wisdom is that PCR will perform worse since it shrinks low variance directions too much and doesn't shrink high variance directions at all.
- ▶ However, Dhillon et al. (2013) show that the MSE risk of PCR is always within a constant factor of that of Ridge Regression while there are situations in which Ridge can be arbitrarily worse than PCR in terms of MSE.
- ▶ In practice, which is better depends on the DGP

Least Absolute Shrinkage and Selection Operator (LASSO)

Bühlmann & van de Geer (2011); Hastie, Tibshirani & Wainwright (2015)

Assume that X has been centered: don't penalize intercept!

Notation

$$\|\beta\|_2^2 = \sum_{j=1}^p \beta_j^2, \quad \|\beta\|_1 = \sum_{j=1}^p |\beta_j|$$

Ridge Regression – L_2 Penalty

$$\hat{\beta}_{Ridge} = \arg \min_{\beta} (\mathbf{y} - X\beta)'(\mathbf{y} - X\beta) + \lambda \|\beta\|_2^2$$

LASSO – L_1 Penalty

$$\hat{\beta}_{Lasso} = \arg \min_{\beta} (\mathbf{y} - X\beta)'(\mathbf{y} - X\beta) + \lambda \|\beta\|_1$$

Other Ways of Thinking about LASSO

Constrained Optimization

$$\arg \min_{\beta} (\mathbf{y} - X\beta)'(\mathbf{y} - X\beta) \quad \text{subject to} \quad \sum_{j=1}^p |\beta_j| \leq t$$

Data-dependent, one-to-one mapping between λ and t .

Bayesian Posterior Mode

Ignoring the intercept, LASSO is the posterior model for β under

$$\mathbf{y}|X, \beta, \sigma^2 \sim N(X\beta, \sigma^2 I_n), \quad \beta \sim \prod_{j=1}^p \text{Lap}(\beta_j|0, \tau)$$

where $\lambda = 1/\tau$ and $\text{Lap}(x|\mu, \tau) = (2\tau)^{-1} \exp \{-\tau^{-1}|x - \mu|\}$

Comparing Ridge and LASSO – Bayesian Posterior Modes

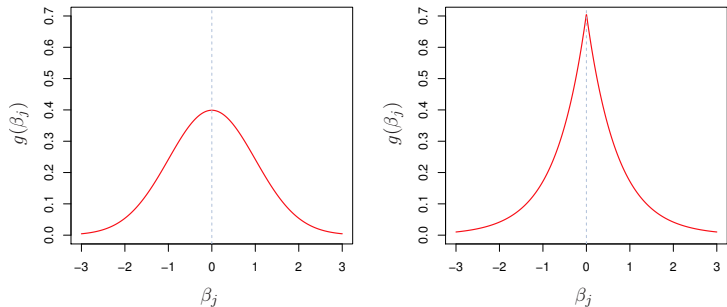


Figure: Ridge, at left, puts a normal prior on β while LASSO, at right, uses a Laplace prior, which has fatter tails and a taller peak at zero.

Comparing LASSO and Ridge – Constrained OLS

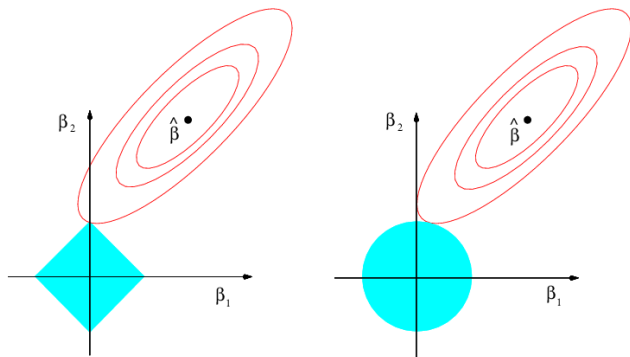


Figure: $\hat{\beta}$ denotes the MLE and the ellipses are the contours of the likelihood. LASSO, at left, and Ridge, at right, both shrink β away from the MLE towards zero. Because of its diamond-shaped constraint set, however, LASSO favors a **sparse solution** while Ridge does not

No Closed-Form for LASSO!

Simple Special Case

Suppose that $X'X = I_p$

Maximum Likelihood

$$\hat{\beta}_{MLE} = (X'X)^{-1}X'y = X'y, \quad \hat{\beta}_j^{MLE} = \sum_{i=1}^n x_{ij}y_i$$

Ridge Regression

$$\hat{\beta}_{Ridge} = (X'X + \lambda I_p)^{-1}X'y = [(1 + \lambda)I_p]^{-1}\hat{\beta}_{MLE}, \quad \hat{\beta}_j^{Ridge} = \frac{\hat{\beta}_j^{MLE}}{1 + \lambda}$$

So what about LASSO?

LASSO when $X'X = I_p$ so $\hat{\beta}_{MLE} = X'y$

Want to Solve

$$\hat{\beta}_{LASSO} = \arg \min_{\beta} (\mathbf{y} - X\beta)'(\mathbf{y} - X\beta) + \lambda \|\beta\|_1$$

Expand First Term

$$\begin{aligned}(\mathbf{y} - X\beta)'(\mathbf{y} - X\beta) &= \mathbf{y}'\mathbf{y} - 2\beta'X'\mathbf{y} + \beta'X'X\beta \\ &= (\text{constant}) - 2\beta'\hat{\beta}_{MLE} + \beta'\beta\end{aligned}$$

Hence

$$\begin{aligned}\hat{\beta}_{LASSO} &= \arg \min_{\beta} (\beta'\beta - 2\beta'\hat{\beta}_{MLE}) + \lambda \|\beta\|_1 \\ &= \arg \min_{\beta} \sum_{j=1}^p \left(\beta_j^2 - 2\beta_j\hat{\beta}_j^{MLE} + \lambda |\beta_j| \right)\end{aligned}$$

LASSO when $X'X = I_p$

Preceding Slide

$$\hat{\beta}_{LASSO} = \arg \min_{\beta} \sum_{j=1}^p \left(\beta_j^2 - 2\beta_j \hat{\beta}_j^{MLE} + \lambda |\beta_j| \right)$$

Key Simplification

Equivalent to solving j independent optimization problems:

$$\hat{\beta}_j^{Lasso} = \arg \min_{\beta_j} \left(\beta_j^2 - 2\beta_j \hat{\beta}_j^{MLE} + \lambda |\beta_j| \right)$$

- ▶ Sign of β_j^2 and $\lambda |\beta_j|$ unaffected by $\text{sign}(\beta_j)$
- ▶ $\hat{\beta}_j^{MLE}$ is a function of data only – outside our control
- ▶ Minimization requires **matching** $\text{sign}(\beta_j)$ to $\text{sign}(\hat{\beta}_j^{MLE})$

LASSO when $X'X = I_p$

Case I: $\hat{\beta}^{MLE} > 0 \implies \beta_j > 0 \implies |\beta_j| = \beta_j$

Optimization problem becomes

$$\hat{\beta}_j^{Lasso} = \arg \min_{\beta_j} \beta_j^2 - 2\beta_j \hat{\beta}_j^{MLE} + \lambda \beta_j$$

Interior solution:

$$\hat{\beta}_j = \hat{\beta}_j^{MLE} - \frac{\lambda}{2}$$

Can't have $\beta_j < 0$: corner solution sets $\beta_j = 0$

$$\hat{\beta}_j^{Lasso} = \max \left\{ 0, \hat{\beta}_j^{MLE} - \frac{\lambda}{2} \right\}$$

LASSO when $X'X = I_p$

Case II: $\hat{\beta}^{MLE} \leq 0 \implies \beta_j \leq 0 \implies |\beta_j| = -\beta_j$

Optimization problem becomes

$$\hat{\beta}_j^{Lasso} = \arg \min_{\beta_j} \beta_j^2 - 2\beta_j \hat{\beta}_j^{MLE} - \lambda \beta_j$$

Interior solution:

$$\hat{\beta}_j = \hat{\beta}_j^{MLE} + \frac{\lambda}{2}$$

Can't have $\beta_j > 0$: corner solution sets $\beta_j = 0$

$$\hat{\beta}_j^{Lasso} = \min \left\{ 0, \hat{\beta}_j^{MLE} + \frac{\lambda}{2} \right\}$$

Ridge versus LASSO when $X'X = I_p$

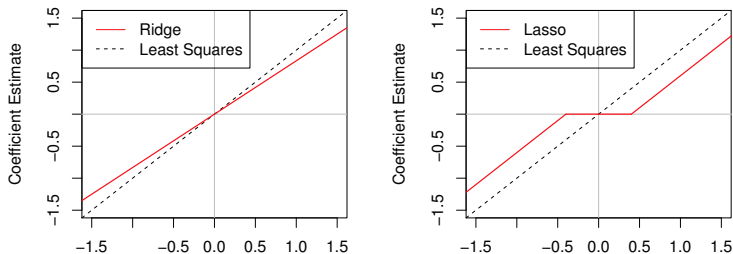


Figure: Horizontal axis in each plot is MLE

$$\hat{\beta}_j^{Ridge} = \left(\frac{1}{1 + \lambda} \right) \hat{\beta}_j^{MLE}$$

$$\hat{\beta}_j^{Lasso} = \text{sign} \left(\hat{\beta}_j^{MLE} \right) \max \left\{ 0, \left| \hat{\beta}_j^{MLE} \right| - \frac{\lambda}{2} \right\}$$

Calculating LASSO – The Shooting Algorithm

Cyclic Coordinate Descent

Data: \mathbf{y} , X , $\lambda \geq 0$, $\varepsilon > 0$

Result: LASSO Solution

$\beta \leftarrow \text{ridge}(X, \mathbf{y}, \lambda)$

repeat

$\beta^{\text{prev}} \leftarrow \beta$

for $j = 1, \dots, p$ **do**

$a_j \leftarrow 2 \sum_{i=1}^n x_{ij}^2$

$c_j \leftarrow 2 \sum_{i=1}^n x_{ij}(y_i - \mathbf{x}_i' \beta + \beta_j x_{ij})$

$\beta_j \leftarrow \text{sign}(c_j/a_j) \max \{0, |c_j/a_j| - \lambda/a_j\}$

end

until $\sum_{j=1}^p |\beta_j^{\text{prev}} - \beta_j| < \varepsilon;$