2018-10-10, SML 2018, Lec 13

Ridge regression: scaling of predictors

- The standard least squares coefficient estimates are *scale equivariant*: multiplying X_j by a constant c simply leads to a scaling of the least squares coefficient estimates by a factor of $1/c$. In other words, regardless of how the jth predictor is scaled, $X_j \hat{\beta}_j$ will remain the same.
- In contrast, the ridge regression coefficient estimates can change *substantially* when multiplying a given predictor by a constant, due to the sum of squared coefficients term in the penalty part of the ridge regression objective function.
- Therefore, it is best to apply ridge regression after *standardizing the predictors*, using the formula

$$\tilde{x}_{ij} = \frac{x_{ij} - \overline{x}_j}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{ij} - \overline{x}_j)^2}}$$

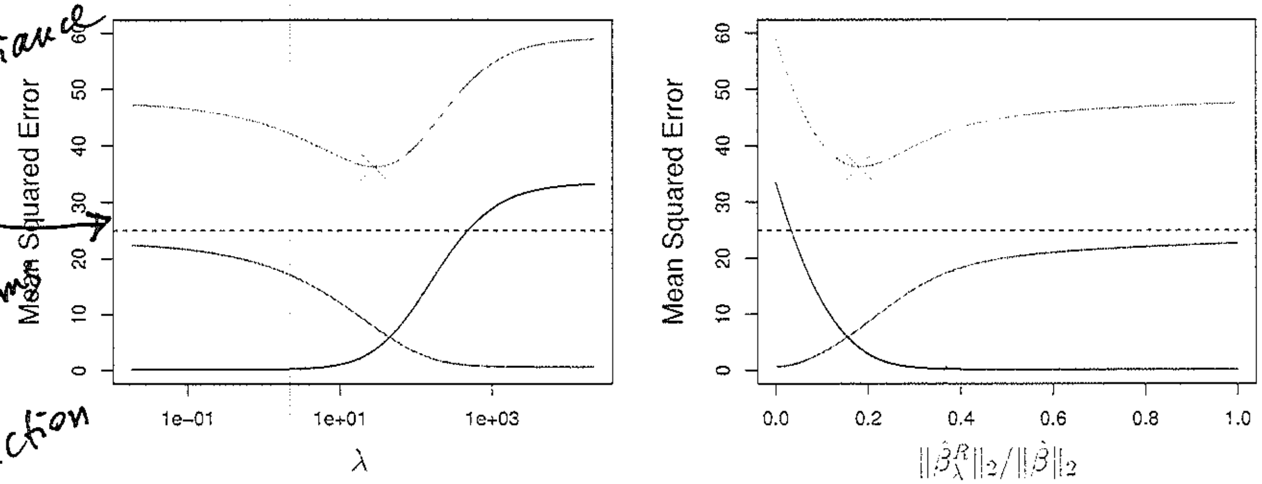
typically, X_j is also centered to have mean equal to 0.

This centering and scaling of the predictors is a common preprocessing step not only in ridge regression, but also in lasso and other ML algorithms.

Why Does Ridge Regression Improve Over Least Squares?

The Bias-Variance tradeoff

this is variance of irreducible error when ① true model with true params ② is used for out-of-sample prediction

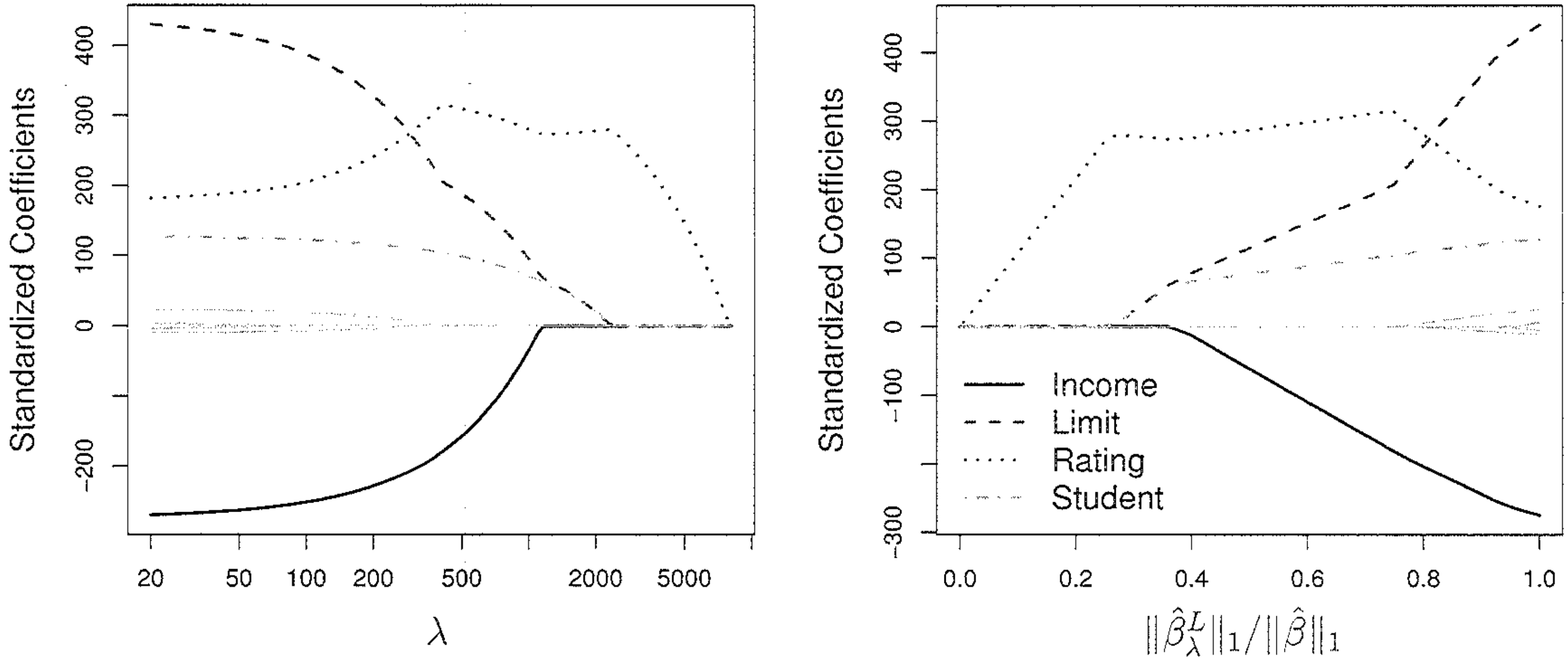


Simulated data with $n = 50$ observations, $p = 45$ predictors, all having nonzero coefficients. Squared bias (black), variance (green), and test mean squared error (purple) for the ridge regression predictions on a simulated data set, as a function of λ and $\|\hat{\beta}^R_\lambda\|_2/\|\hat{\beta}\|_2$. The horizontal dashed lines indicate the minimum possible MSE. The purple crosses indicate the ridge regression models for which the MSE is smallest.

Example: Credit dataset LASSO.

Q: does LASSO produce sparse solutions for every $\lambda > 0$?

A: no. e.g., see $\lambda = 20$ below.



The Variable Selection Property of the Lasso

Why is it that the lasso, unlike ridge regression, results in coefficient estimates that are exactly equal to zero?

One can show that the lasso and ridge regression coefficient estimates solve the problems

$$\underset{\beta}{\text{minimize}} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \quad \text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \leq s$$

and

$$\underset{\beta}{\text{minimize}} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \quad \text{subject to} \quad \sum_{j=1}^{p} \beta_j^2 \leq s,$$

respectively.

⊛ y has been centered; x_j's have been centered and scaled.

2.9 ℓ_q Penalties and Bayes Estimates

For a fixed real number $q \geq 0$, consider the criterion

$$\underset{\beta \in \mathbb{R}^p}{\text{minimize}} \left\{ \frac{1}{2N} \sum_{i=1}^{N} (y_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j|^q \right\}^{\circledast}. \tag{2.21}$$

This is the lasso for $q = 1$ and ridge regression for $q = 2$. For $q = 0$, the term $\sum_{j=1}^{p} |\beta_j|^q$ counts the number of nonzero elements in β, and so solving (2.21) amounts to best-subset selection. Figure 2.6 displays the constraint regions corresponding to these penalties for the case of two predictors ($p = 2$). Both

ℓ_q balls in $\mathbb{R}^2$ for different q.

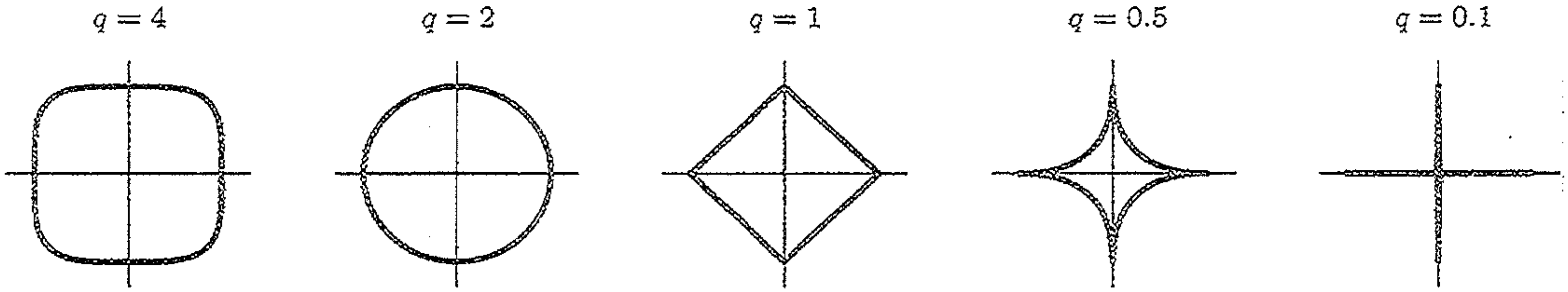
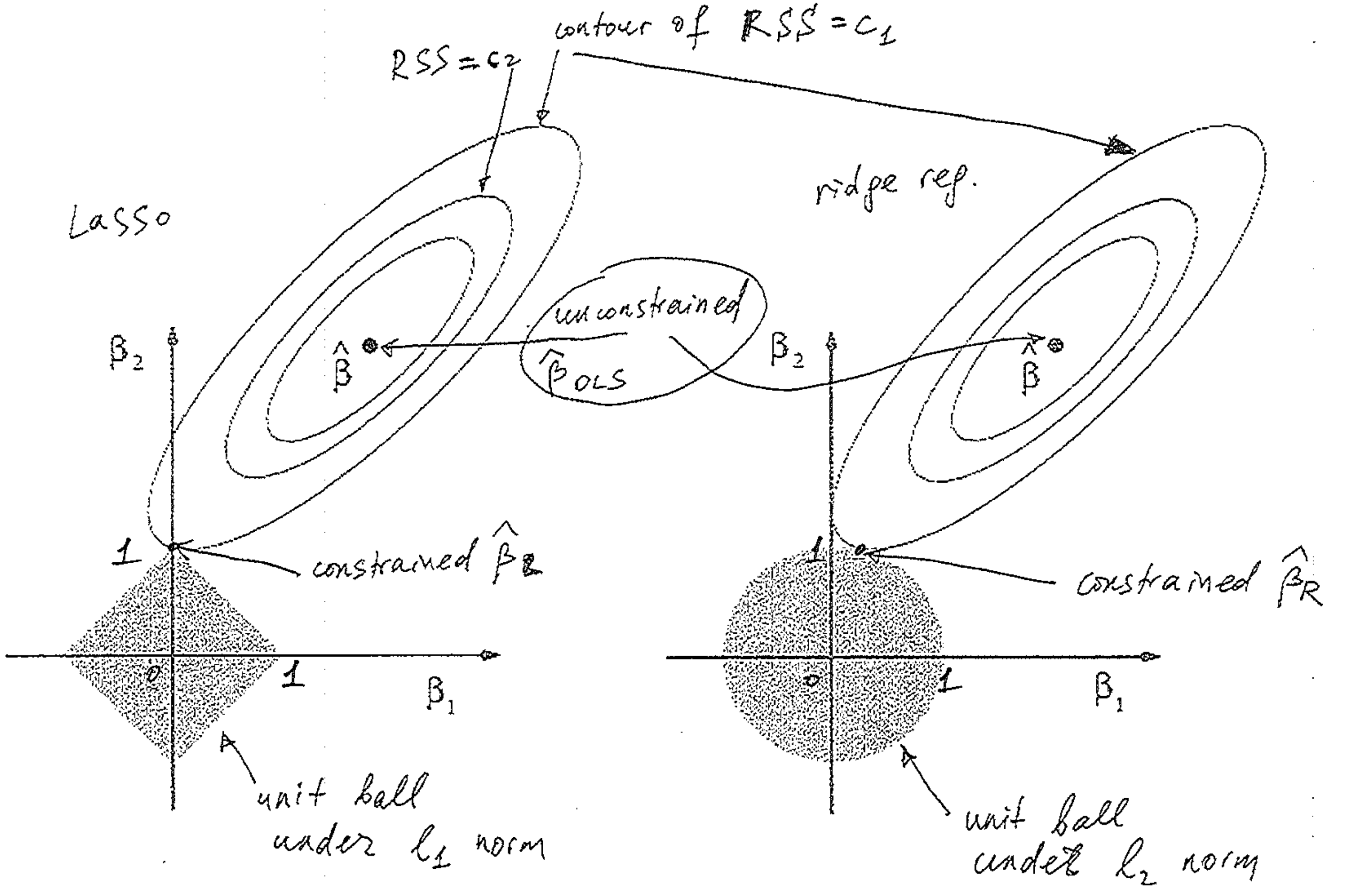


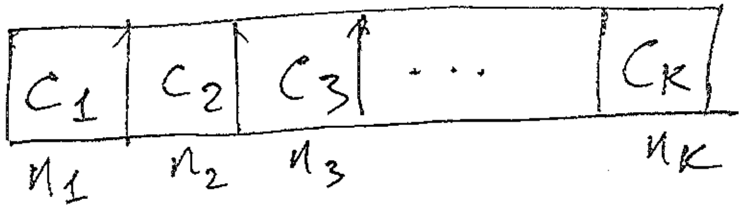
Figure 2.6 *Constraint regions $\sum_{j=1}^{p} |\beta_j|^q \leq 1$ for different values of q. For $q < 1$, the constraint region is nonconvex.*

The Lasso Picture


contour of RSS = c_1
RSS = c_2
Lasso
ridge reg.
unconstrained $\hat{\beta}_{OLS}$
β_2
$\hat{\beta}$
β_2
$\hat{\beta}$
1
constrained $\hat{\beta}_L$
1
constrained $\hat{\beta}_R$
0
1
β_1
0
1
β_1
unit ball under l_1 norm
unit ball undet l_2 norm

Cross-validation (K-fold): recap

Whole data training set D :



C_i : labels# of our data points corresponding to the ith subset

$|C_i|$ = "cardinality of set C_i" = # of elements in C_i ; n_i

$D = \bigcup_{i=1}^{K} C_i$; $C_i \cap C_j = \emptyset$ (subsets are disjoint).

$T_i = D \setminus C_i = \bigcup_{\substack{j=1 \\ j \neq i}}^{K} C_j$; T_i is ith training set

↑ "set difference"

$V_i = C_i$: the ith validation set

① For $i = 1, \dots, K$

(a) "train"/ fit our model on T_i , validate it on V_i.

(b) get a "discrepancy" $\mathrm{Discr}(\hat{\underset{\sim}{Y}}^{(i)}, \underset{\sim}{Y}^{(i)})$ between predicted values $\hat{\underset{\sim}{Y}}^{(i)}$ (based on T_i) for $\underset{\sim}{Y}^{(i)}$ (from V_i).

"Discr": MSE or misclassification rate.

② Aggregate the discrepancies from $i = 1, \dots, K$ into a single measure. obj. fun. $F(\tau) = \sum_{i=1}^{K} w_i \cdot \mathrm{Discr}(\hat{\underset{\sim}{Y}}^{(i)}(\tau), \underset{\sim}{Y}^{(i)})$: depends on τ since $\hat{Y}^{(i)}$ does.

How to calibrate/estimate tuning parameters using K-fold CV?

Examples:

1) polynomial regression: tuning par. is degree of polynomial.

2) GAM, ridge regression, lasso: need to choose the penalty parameter λ (for "regularization").

3) model selection: need to pick a subset of covariates.

In practice, the "aggregated discrepancy" (AD) depends on the tuning parameters (τ).

$\Rightarrow$ optimize/minimize AD with respect to τ.