Predictive Analytics (ISE529)

Dimension Reduction (II)

Dr. Tao Ma
ma.tao@usc.edu

Tue/Thu, Aug 26 - Dec 6, 2024, Fall

USC Viterbi
School of Engineering
Daniel J. Epstein Department of Industrial and Systems Engineering

USC University of Southern California

Outline

- Ridge Regression
- The Lasso
- Partial Least Squares

Shrinkage Methods

We can fit a model containing all *p* predictors using a technique that *constrains* or *regularize*s the coefficient estimates, or equivalently, that shrinks the coefficient estimates towards zero.

It may not be immediately obvious why such a constraint should improve the fit, but it turns out that shrinking the coefficient estimates can significantly reduce their variance.

The two best-known techniques for shrinking the regression coefficients towards zero are *ridge regression* and *the lasso*.

RIDGE REGRESSION

Ridge regression

Recall that the least squares fitting procedure estimates $\beta_0, \beta_1, \ldots, \beta_p$ using the values that minimize

$$\text{RSS} = \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2$$

Ridge regression is very similar to least squares, except that the coefficients are estimated by minimizing a slightly different quantity.

Ridge regression

Ridge regression shrinks the regression coefficients by imposing a penalty on their size in the objective function. In particular, the ridge regression coefficient estimates $\hat{\beta}^{ridge}$ are the values that minimize a penalized residual sum of squares.

$$\hat{\beta}^{ridge} = \arg\min_{\beta} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{P} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{P} \beta_j^2 \right\} \qquad (1)$$

$$= \arg\min_{\beta} \left\{ RSS + \lambda \sum_{j=1}^{P} \beta_j^2 \right\}$$

Here $\lambda \geq 0$ is a tuning parameter that controls the amount of shrinkage: the larger the value of λ, the greater the amount of shrinkage. The coefficients are shrunk toward zero (and each other). Selecting a good value for λ is critical and can be determined separately by cross-validation.

Ridge regression

Equation (1) trades off two different criteria.

- As with least squares, ridge regression seeks coefficient estimates that fit the data well, by making the RSS small.
- However, the second term, $\lambda \sum_j \beta_j^2$, called a *shrinkage penalty*, is small when $\beta_0, \beta_1, \ldots, \beta_p$ are close to zero, and so it has the effect of the estimates of β_j towards zero.

The tuning parameter λ serves to control the relative impact of these two terms on the regression coefficient estimates.

- When $\lambda = 0$, the penalty term has no effect, and ridge regression will produce the least squares estimates.
- However, as $\lambda \rightarrow \infty$, the impact of the shrinkage penalty grows, and the ridge regression coefficient estimates will approach zero.

Unlike least squares, which generates only one set of coefficient estimates, ridge regression will produce a different set of coefficient estimates, $\hat{\beta}^{ridge}$, for each value of λ.

Ridge regression

An equivalent way to write the ridge problem is

$$\hat{\beta}^{ridge} = \arg\min_{\beta} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{P} \beta_j x_{ij} \right)^2 \qquad (2)$$

$$\text{subject to } \sum_{j=1}^{P} \beta_j^2 \le t$$

which makes explicit the size constraint on the parameters. There is a one-to-one correspondence between the parameters λ in (1) and t in (2).

When applying Lagrange Multiplier to (2), we can obtain (1)

Ridge regression

When there are many correlated variables in a linear regression model, their coefficients can become poorly determined and exhibit high variance. A wildly large positive coefficient on one variable can be canceled by a similarly large negative coefficient on its correlated cousin. By imposing a size constraint t on the coefficients, this problem is alleviated.

In addition, notice that the intercept β_0 has been left out of the penalty term. Penalization of the intercept would make the procedure depend on the origin chosen for Y ; that is, adding a constant C to each of the targets y_i would not simply result in a shift of the predictions by the same amount C.

Ridge regression

The ridge solutions are not equivariant under scaling of the inputs, and so one normally standardizes the inputs before solving (1).

The solution to (1) can be separated into **two parts**, after re-parametrization using centered inputs: each x_{ij} gets replaced By $x_{ij} - \overline{x}_j$.

We estimate β_0 by $\bar{y} = 1/N \sum_1^N y_i$. The remaining coefficients get estimated by a ridge regression without intercept, using the centered x_{ij} . Henceforth we assume that this centering has been done, so that the input matrix $\boldsymbol{X}$ has p (rather than $p + 1$) columns.

Ridge regression

Equation (1) in matrix form:

$$L(\lambda) = (y - X\beta)^T (y - x\beta) + \lambda\beta^T\beta$$

The ridge regression solutions are

$$\hat{\beta}^{ridge} = \left(X^T X + \lambda I\right)^{-1} X^T y$$

where $\boldsymbol{I}$ is the $p \times p$ identity matrix. Notice that with the choice of quadratic penalty $\beta^T\beta$, the ridge regression solution is again a linear function of y. The solution adds a positive constant to the diagonal of $\mathbf{X^TX}$ before inversion. This makes the problem non-singular, even if $\mathbf{X^TX}$ is not of full rank, and was the main motivation for ridge regression when it was first introduced in statistics (Hoerl and Kennard, 1970).

Ridge regression

The *singular value decomposition* (SVD) of the centered input matrix **X** gives us some additional insight into the nature of ridge regression. The SVD of the $N \times p$ matrix **X** has the form

$$\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^T.$$

Here **U** and **V** are $N \times p$ and $p \times p$ orthogonal matrices, with the columns of **U** spanning the column space of **X**, and the columns of **V** spanning the row space. **D** is a $p \times p$ diagonal matrix, with diagonal entries $d_1 \geq d_2 \geq \cdots \geq d_p \geq 0$ called the singular values of **X**.

Ridge regression

Using the *singular value decomposition*, we can write the ridge solutions as

$$\begin{aligned}\mathbf{X}\hat{\beta}^{\text{ridge}} &= \mathbf{X}(\mathbf{X}^T\mathbf{X}+\lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y} \\ &= \mathbf{U}\,\mathbf{D}(\mathbf{D}^2+\lambda\mathbf{I})^{-1}\mathbf{D}\,\mathbf{U}^T\mathbf{y} \\ &= \sum_{j=1}^{p}\mathbf{u}_j\frac{d_j^2}{d_j^2+\lambda}\mathbf{u}_j^T\mathbf{y},\end{aligned}$$

where the $\boldsymbol{u_j}$ are the columns of **U**. Note that since $\lambda \geq 0$, we have $d_j^2/\left(d_j^2+\lambda\right) \leq 1$. Like linear regression, ridge regression computes the coordinates of $\boldsymbol{y}$ with respect to the orthonormal basis **U**. It then shrinks these coordinates by the factors $d_j^2/\left(d_j^2+\lambda\right)$. This means that a greater amount of shrinkage is applied to the coordinates of basis vectors with smaller d_j^2.

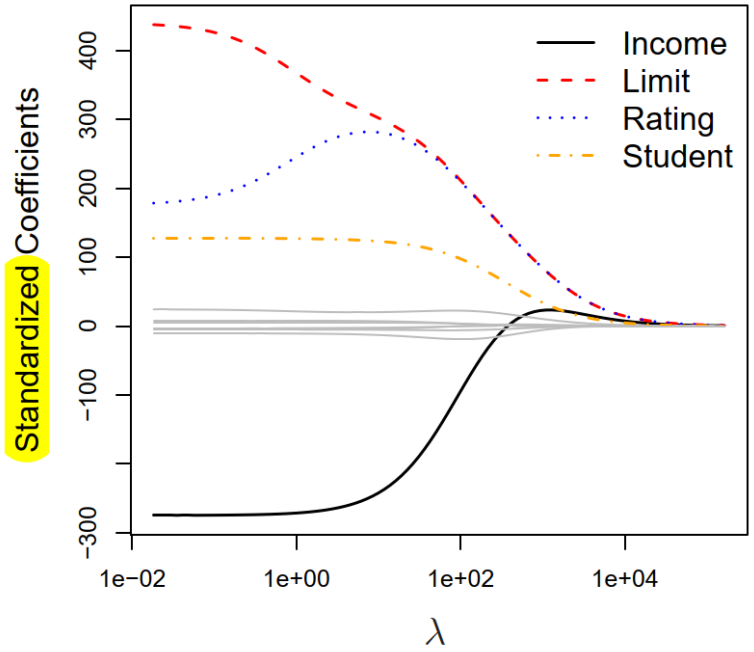
Ridge regression

This monotone decreasing function of λ is the *effective degrees of freedom* of the ridge regression fit.

$$\begin{aligned} \mathrm{df}(\lambda) &= \mathrm{tr}[\mathbf{X}(\mathbf{X}^T\mathbf{X}+\lambda\mathbf{I})^{-1}\mathbf{X}^T], \\ &= \mathrm{tr}(\mathbf{H}_\lambda) \\ &= \sum_{j=1}^{p} \frac{d_j^2}{d_j^2+\lambda}. \end{aligned}$$

Usually in a linear-regression fit with p variables, the degrees-of-freedom of the fit is p, the number of free parameters. The idea is that although all p coefficients in a ridge fit will be non-zero, they are fit in a restricted fashion controlled by λ. Note that df(λ) = p when $\lambda = 0$ (no regularization) and df(λ) $\rightarrow 0$ as $\lambda \rightarrow \infty$. Of course there is always an additional one degree of freedom for the intercept, which was removed *apriori*.

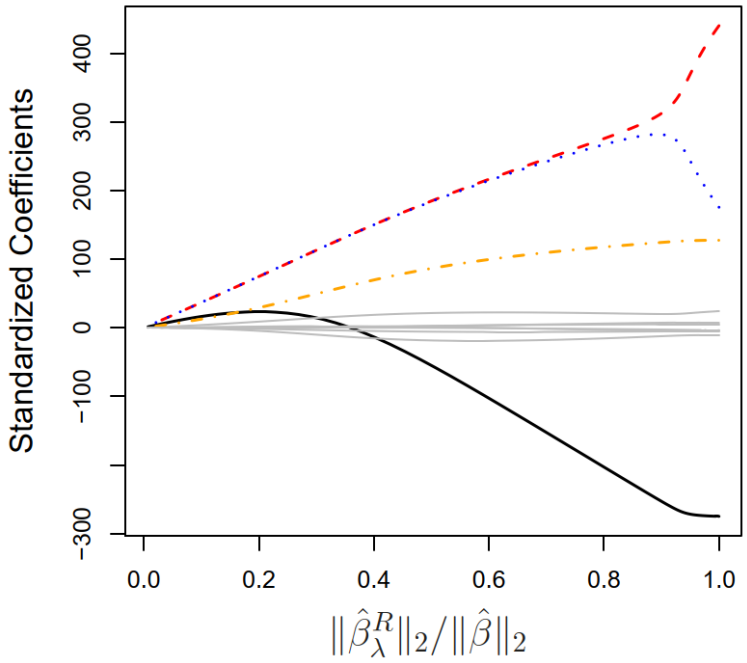
Example: Credit Data



The panel displays each curve corresponds to the ridge regression coefficient estimate for one of the ten variables, as a function of λ.

As λ equals zero, the corresponding ridge coefficient estimates are the same as the usual least squares esti mates. But as λ increases, the ridg coefficient estimates shrink towards zero.

Example: Credit Data



The panel displays the same ridge coefficient estimates as a function of $\|\hat{\beta}_\lambda^R\|_2/\|\hat{\beta}\|_2$

As λ equals zero, the ratio equals zero, corresponding ridge coefficient estimates are the same as the usual least squares estimates. But as λ increases, the ratio reduced to zero, ridge coefficient estimates shrink towards zero.

Input Data Standardization

The standard least squares coefficient estimates discussed in early Chapter are scale equivariant: multiplying X_j by a constant $\boldsymbol{c}$ simply leads to a scale of the least squares coefficient estimates by a factor of $1/\boldsymbol{c}$. 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 ridge regression objective function.

Therefore, it is best to apply ridge regression after standardizing the predictors, using the formula

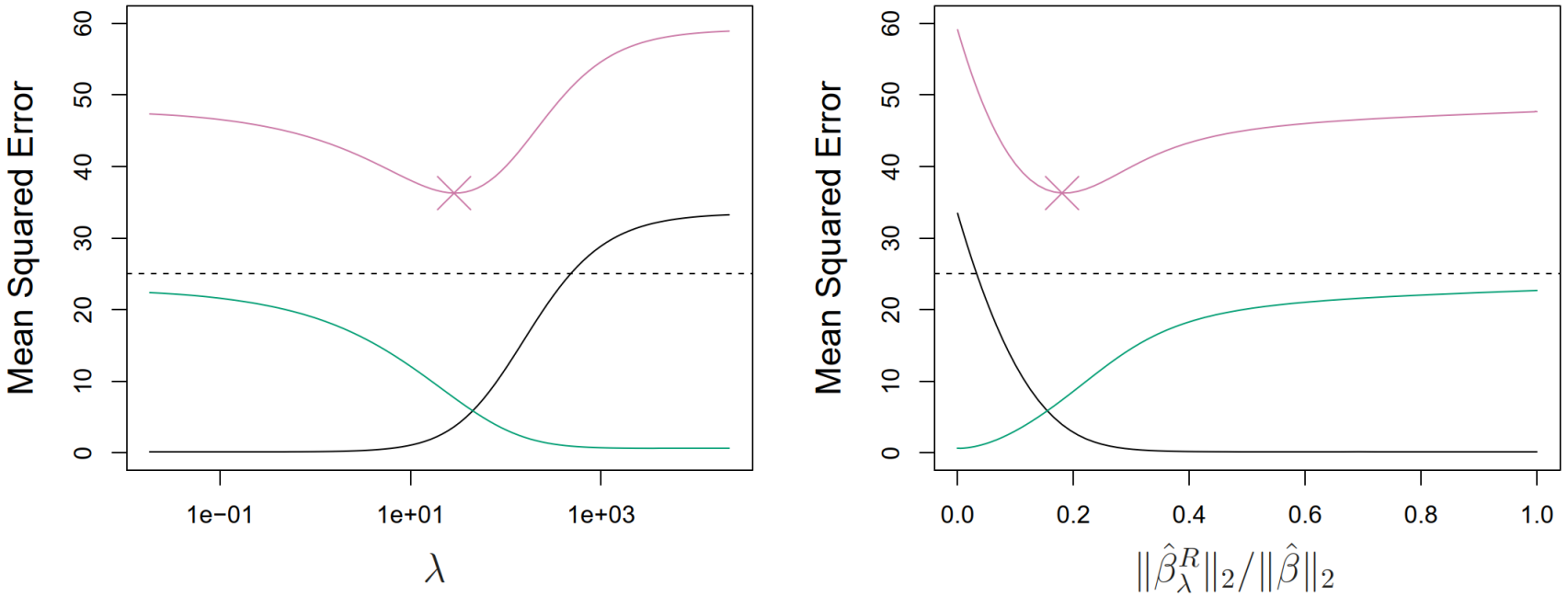
$$\tilde{x}_{ij} = \frac{x_{ij}}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{ij} - \overline{x}_j)^2}}$$

As a result, the final model coefficients will not depend on the scale on which the predictors are measured.

Bias-Variance Trade-Off

USC Viterbi School of Engineering

Why Does Ridge Regression Improve Over Least Squares?
As λ increases, the flexibility of the ridge regression fit decreases, leading to decreased variance but increased bias.



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. The horizontal dashed lines indicate the minimum possible MSE. The purple crosses indicate the ridge regression models for which the MSE is smallest.

Bias-Variance Trade-Off

USC Viterbi School of Engineering

Why Does Ridge Regression Improve Over Least Squares?

- In situations where the relationship between the response and the predictors is close to linear, the least squares estimates will have low bias but may have high variance.
- This means that a small change in the training data can cause a large change in the least squares coefficient estimates.
- In particular, when the number of variables *p* is almost as large as the number of observations *n* the least squares estimates will be extremely variable. And if p > n, then the least squares estimates do not even have a unique solution, whereas ridge regression can still perform well by trading off a small increase in bias for a large decrease in variance. Hence, ridge regression works best in situations where the least squares estimates have high variance.

THE LASSO

Lasso Regression

Issue: Ridge regression will include all *p* predictors in the final model. The penalty $\lambda \sum \beta j2$ in its objective function will shrink all of the coefficients towards zero, but it will not (exclude) set any of them exactly to zero (unless $\lambda = \infty$).

It can create a challenge in model interpretation in settings in which the number of variables *p* is quite large.

The lasso is an alternative to ridge regression that overcomes this disadvantage. The lasso coefficients, $\beta\hat{}\lambda L$, minimize the objective function:

$$\hat{\beta}^{lasso} = \arg\min_{\beta} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{P} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{P} |\beta_j| \right\} \quad (3)$$

$$= \arg\min_{\beta} \left\{ RSS + \lambda \sum_{j=1}^{P} |\beta_j| \right\}$$

Lasso Regression

It is also equivalent to the following Lagrangian form of optimization problem.

$$\hat{\beta}^{lasso} = \arg\min_{\beta} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{P} \beta_j x_{ij} \right)^2 \quad (4)$$

$$\text{subject to } \sum_{j=1}^{P} |\beta_j| \le t$$

Just as in ridge regression, we can re-parameterize the constant β_0 by standardizing the predictors; the solution for $\hat{\beta}_0$ is $\bar{y}$, and thereafter we fit a model without an intercept.

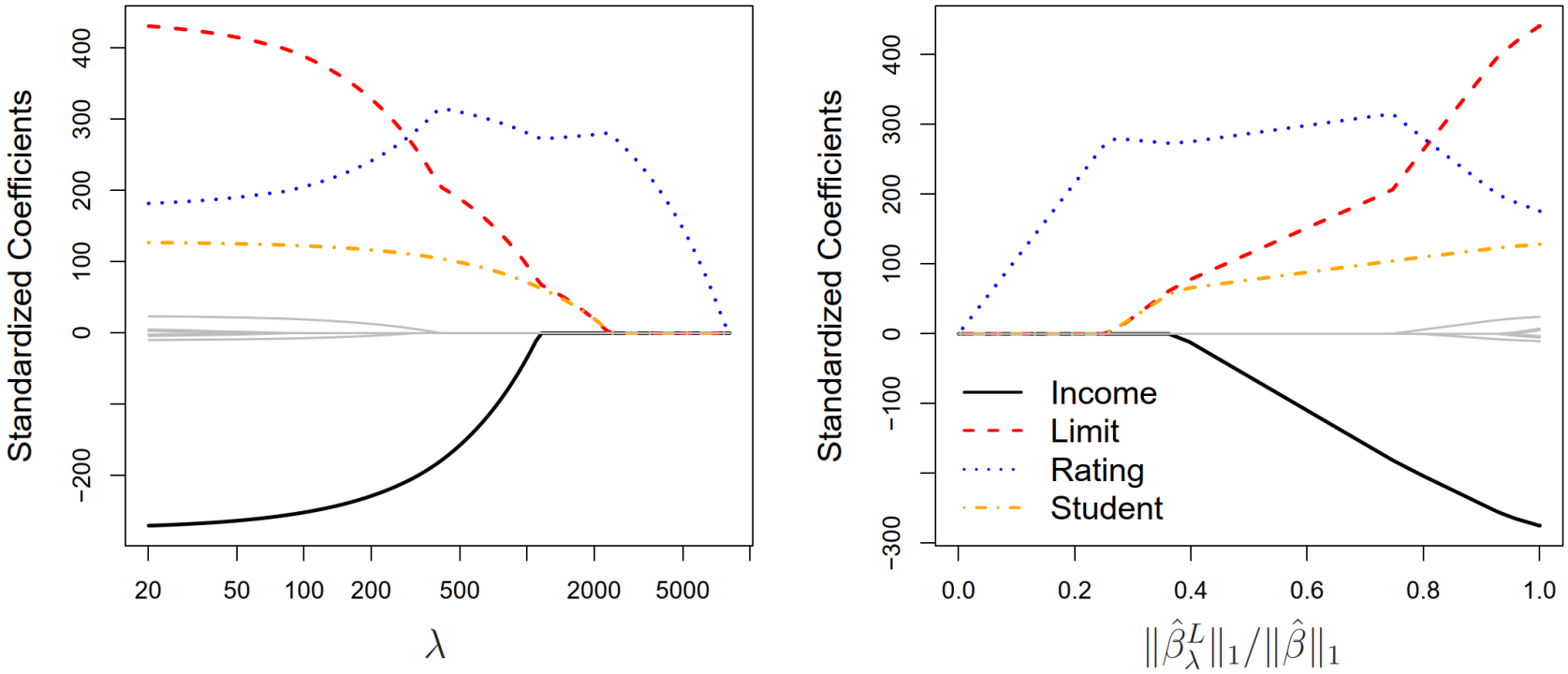
Note the similarity to the ridge regression equation (1) and (2): the L_2 ridge penalty $\sum_1^p \beta_j^2$ is replaced by the L_1 lasso penalty $\sum_1^p |\beta_j|$ This latter constraint makes the solutions nonlinear in the y_i, and there is no closed form expression as in ridge regression.

Lasso Regression

- As with ridge regression, the lasso shrinks the coefficient estimates towards zero. However, in the case of the lasso, the $\ell 1$ penalty has the effect of forcing some of the coefficient estimates to be exactly equal to zero when the tuning parameter λ is sufficiently large. In contrast, ridge regression will always include all of the variables in the model.

- As a result, models generated from the lasso are generally much easier to interpret than those produced by ridge regression. We say that the lasso yields sparse models — that is, sparse models that involve only a subset of the variables.

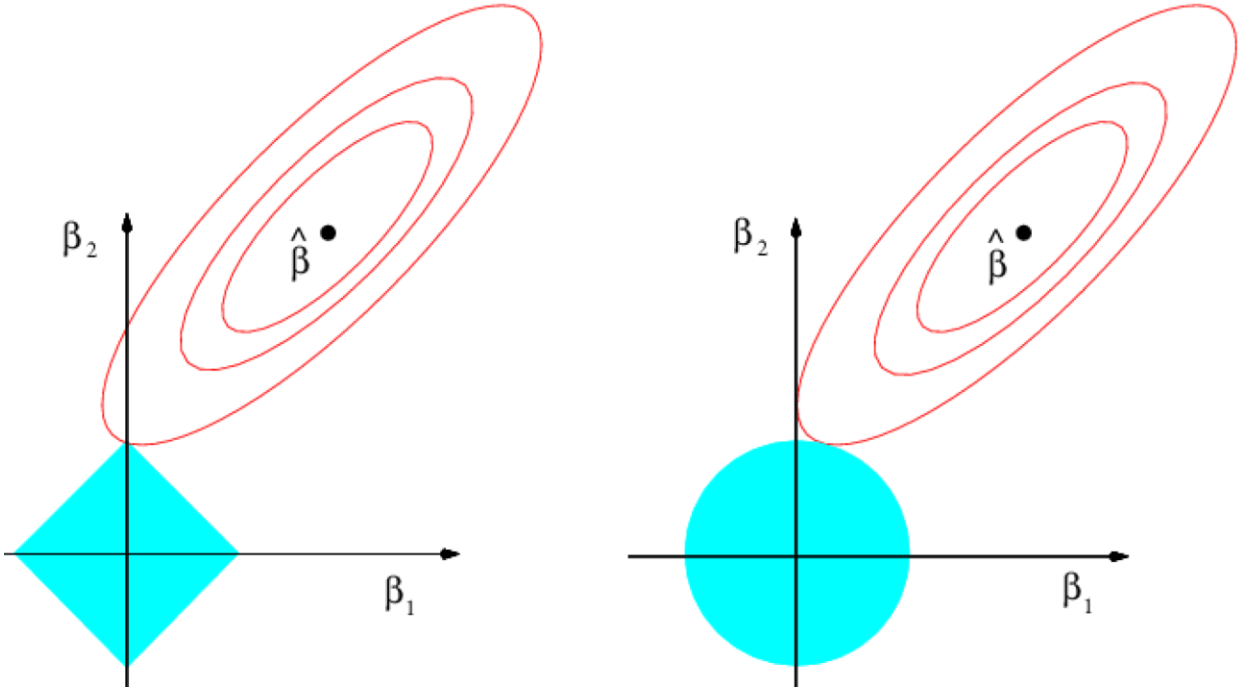
Example: Credit data



The plot shows applying the lasso to the Credit data set.

When $\lambda = 0$, then the lasso simply gives the least squares fit, and when λ becomes sufficiently large, the lasso gives the null model in which all coefficient estimates equal zero.

Example: Credit data



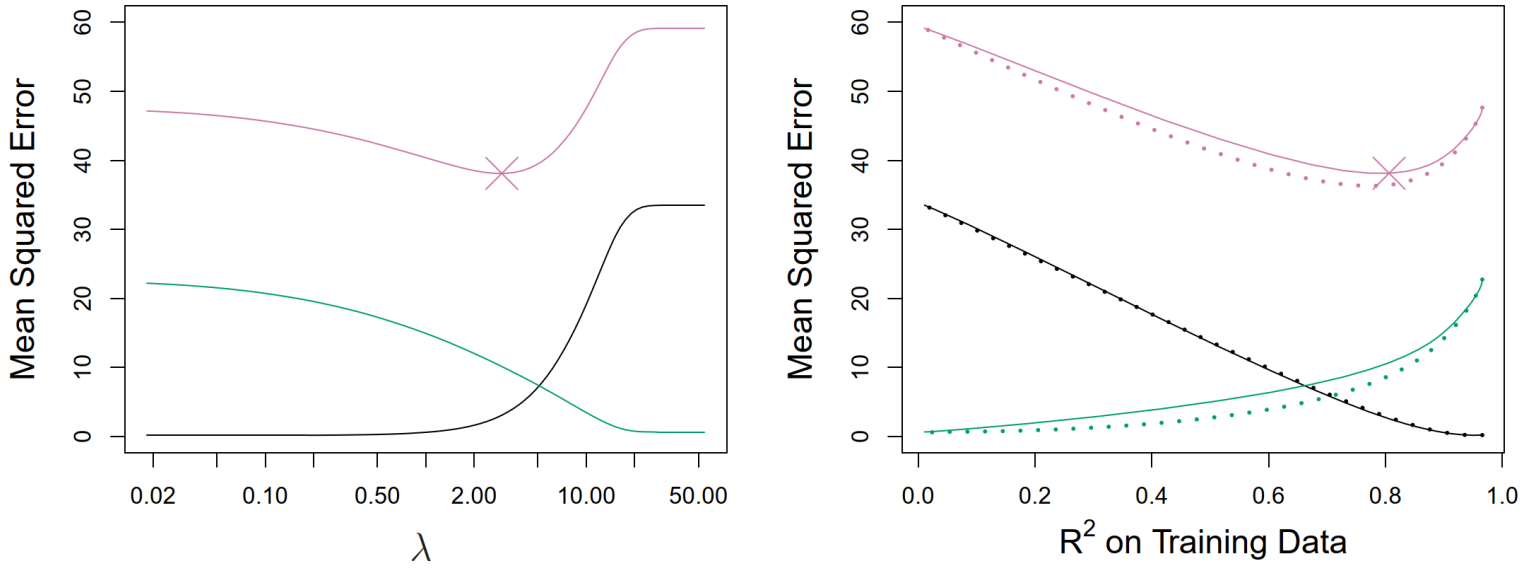
This shows the lasso (left) and ridge regression (right). The solid blue areas are the constraint regions, $|\beta_1|+|\beta_2| \leq$ t and $\beta_1^2 + \beta_2^2 \leq$ t, while the red ellipses are the contours of the RSS.

The lasso constraint has corners at each of the axes, and so the ellipse will often intersect the constraint region at an axis. When this occurs, one of the coefficients will equal zero.

Lasso vs. Ridge

Which method leads to better prediction accuracy?

The lasso has a major advantage over ridge regression, in that it produces simpler and more interpretable models that involve only a subset of the predictors. Some the coefficients will equal zero. However, the example illustrates that **neither** ridge regression **nor** the lasso will universally dominate the other.



Left: Plots of squared bias (black), variance (green), and test MSE (purple) for the lasso on a simulated data set. Right: Comparison of squared bias, variance, and test MSE between lasso (solid) and ridge (dotted).

Elastic-net Model

Partly for this reason as well as for computational tractability, Zou and Hastie (2005) introduced the *elastic-net* penalty

$$\lambda \sum_{j=1}^{p} \left(\alpha \beta_j^2 + (1-\alpha)|\beta_j| \right), \qquad (6)$$

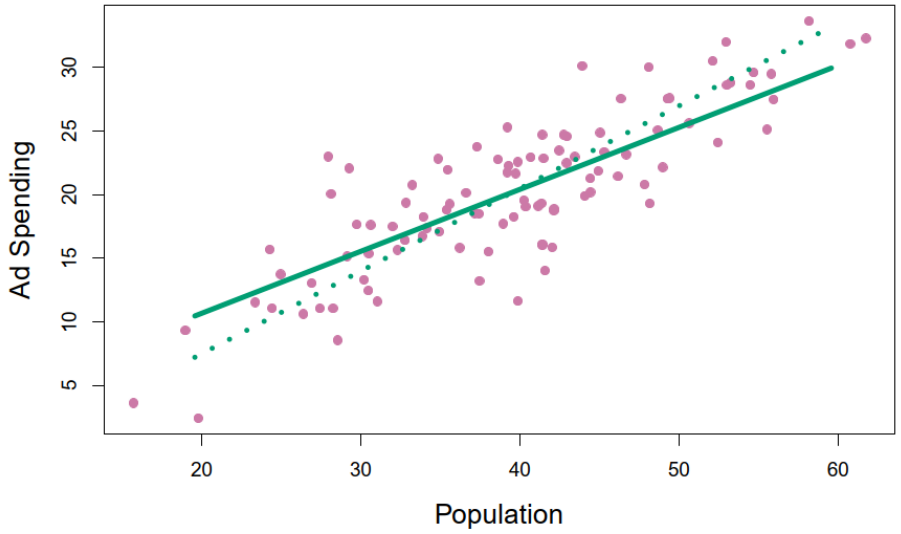
a different compromise between ridge and lasso. The elastic-net selects variables like the lasso, and shrinks together the coefficients of correlated predictors like ridge. It also has considerable computational advantages over the L_q penalties.

PARTIAL LEAST SQUARES

Issue with PC Regression

The PCR approach involves identifying linear combinations, or directions, that best represent the predictors $X_1, \ldots, X_p$. These directions are identified in an *unsupervised* way, since the response Y is not used to help determine the principal component directions. That is, *the response does not supervise the identification of the principal components*.

Consequently, PCR suffers from a drawback: there is *no guarantee* that the directions that *best explain the predictors* will also be the best directions to use for predicting the *response*.



The graph shows the advertising data, the first PLS direction (solid line) and first PCR direction (dotted line).

Partial Least Squares

1. Standardize each $\mathbf{x}_j$ to have mean zero and variance one. Set $\hat{\mathbf{y}}^{(0)} = \bar{y}\mathbf{1}$, and $\mathbf{x}_j^{(0)} = \mathbf{x}_j,\ j = 1, \ldots, p$.

2. For $m = 1, 2, \ldots, p$

   (a) $\mathbf{z}_m = \sum_{j=1}^{p} \hat{\varphi}_{mj}\mathbf{x}_j^{(m-1)}$, where $\hat{\varphi}_{mj} = \langle \mathbf{x}_j^{(m-1)}, \mathbf{y} \rangle$.

   (b) $\hat{\theta}_m = \langle \mathbf{z}_m, \mathbf{y} \rangle / \langle \mathbf{z}_m, \mathbf{z}_m \rangle$.

   (c) $\hat{\mathbf{y}}^{(m)} = \hat{\mathbf{y}}^{(m-1)} + \hat{\theta}_m \mathbf{z}_m$.

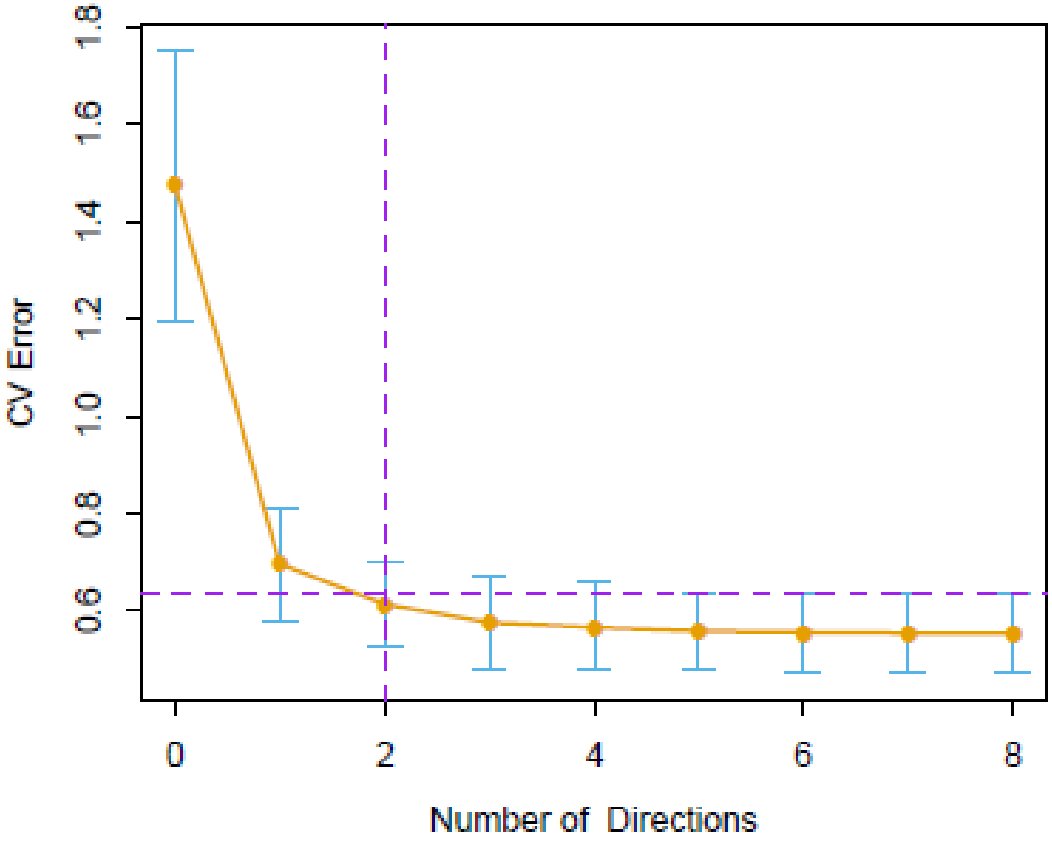
   (d) Orthogonalize each $\mathbf{x}_j^{(m-1)}$ with respect to $\mathbf{z}_m$: $\mathbf{x}_j^{(m)} = \mathbf{x}_j^{(m-1)} - [\langle \mathbf{z}_m, \mathbf{x}_j^{(m-1)} \rangle / \langle \mathbf{z}_m, \mathbf{z}_m \rangle]\mathbf{z}_m,\ j = 1, 2, \ldots, p$.

3. Output the sequence of fitted vectors $\{\hat{\mathbf{y}}^{(m)}\}_1^p$. Since the $\{\mathbf{z}_\ell\}_1^m$ are linear in the original $\mathbf{x}_j$, so is $\hat{\mathbf{y}}^{(m)} = \mathbf{X}\hat{\beta}^{\text{pls}}(m)$. These linear coefficients can be recovered from the sequence of PLS transformations.

Partial Least Squares

- This technique also constructs a set of linear combinations of the inputs for regression, but unlike principal components regression it uses $\boldsymbol{y}$ (in addition to $\mathbf{X}$) for this construction.
- Like principal component regression, partial least squares (PLS) is not scale invariant, so we assume that each $\boldsymbol{x_j}$ is standardized to have mean 0 and variance 1.
- In the construction of the derived input $\boldsymbol{z_m}$, the inputs are weighted by the strength of their univariate effect on y.
- The outcome $\boldsymbol{y}$ is regressed on $\boldsymbol{z_m}$ giving coefficient $\hat{\theta}_m$.
- Since it uses the response $\boldsymbol{y}$ to construct its directions, its solution path is a nonlinear function of $\boldsymbol{y}$. The partial least squares seeks directions that have high variance and high correlation with the response, in contrast to principal components regression which is only on high variance.

Partial Least Squares



Summary and Comparison

- The tuning parameters for ridge and lasso vary over a *continuous* range, while PLS and PCR take just two *discrete* steps to the least squares solution, they all finally converges to least squares.
- PLS and PCR show similar behavior to ridge, roughly track the ridge path, although are discrete and more extreme. The behavior of the lasso is intermediate to the other methods.
- Ridge regression shrinks all directions, but shrinks low-variance directions more. Principal components regression leaves M high-variance directions alone, and discards the rest. Partial least squares also tends to shrink the low-variance directions, but can actually inflate some of the higher variance directions.

Summary and Comparison

$$\hat{\beta}(\lambda) = \text{argmin}_{\beta} \left[R(\beta) + \lambda J(\beta) \right],$$

where

$$R(\beta) = \sum_{i=1}^{N} L(y_i, \beta_0 + \sum_{j=1}^{p} x_{ij}\beta_j),$$

where ***L*** is the loss function and ***J*** is the penalty function. The following are sufficient conditions for the solution ***β(λ)***

1. ***R*** is quadratic or piecewise-quadratic as a function of ***β***.
2. ***J*** is piecewise linear in ***β***.

A full study is given in Frank and Friedman (1993). These authors conclude that for minimizing prediction error, ridge regression is generally preferable to principal components regression and partial least squares. However the improvement over the latter two methods was only slight. The behavior of the lasso is intermediate to the other methods.