Why are Constraining and Adding a Penalty Equivalent?

We can minimize $\text{RSS}(\beta)$ subject to $\text{Penalty}(\beta) \leq t$ using *Lagrange multipliers*.

One possibility: The minimum of RSS satisfies the constraint. This is the same as minimizing $\text{RSS}(\beta) + \lambda\text{Penalty}(\beta)$ with $\lambda = 0$.

Otherwise: The constrained minimum occurs where $\text{Penalty}(\beta) = t$, and here the gradient of $\text{RSS}(\beta)$ must point opposite to the gradient of $\text{Penalty}(\beta)$ — so we can't decrease RSS without violating the constraint.

In other words, for some positive λ:

$$\nabla\text{RSS}(\hat{\beta}) \;=\; -\lambda\nabla\text{Penalty}(\hat{\beta})$$

This is equivalent to

$$\nabla\left(\text{RSS}(\hat{\beta}) + \lambda\text{Penalty}(\hat{\beta})\right) \;=\; 0$$

which happens at the point where $\text{RSS}(\beta) + \lambda\text{Penalty}(\beta)$ is minimized.

This can be extended to penalties like $\sum|\beta_i|$ that aren't differentiable everywhere. The converse is true as well.

Why the Lasso May Set Some Coefficients to Zero

The lasso, but not ridge regression, may force some of the β_i to be exactly zero.

For a geometric view of this, see Figure 3.12 in the text.

We can also see this from the fact that the derivative of the lasso penalty w.r.t. β_i is constant for all $\beta_i > 0$ — so there's a continuing force towards zero.

In contrast, the derivative of the ridge penalty w.r.t. β_i is proportional to β_i — so the effect of the penalty declines as the β_i approach zero.

Setting some β_i to exactly zero will be good if some of the inputs really are completely irrelevant. But in many problems, we think that all the inputs are at least a little bit relevant.

Principal Component Directions

The principal component directions in the input space, represented by unit vectors $v_1, \ldots, v_p$, are defined by

- v_1 is the direction of largest sample variance over the training set.
- v_2 is the direction of largest variance, subject to v_2 begin orthogonal to v_1.
- Generally, v_k is the direction of largest variance, subject to v_k being orthogonal to $v_1, \ldots, v_{k-1}$.

If $\mathbf{X}$ is the matrix of centered inputs (with no first column of 1's), the principal component directions are the (unit length) eigenvectors of $\mathbf{S} = \mathbf{X}^T\mathbf{X}/N$, ordered by decreasing eigenvalue:

$$\rho_k v_k \;=\; \mathbf{S} v_k, \quad \text{for } k = 1, \ldots, p, \text{ with } \rho_1 \geq \cdots \geq \rho_p$$

The projection of the training inputs in the k'th principal component is $\mathbf{z}_k = \mathbf{X} v_k$. The sample variance of $\mathbf{z}_k$ is $(1/N)\mathbf{z}_k^T\mathbf{z}_k = (1/N)v_k^T\mathbf{X}^T\mathbf{X}v_k = v_k^T\mathbf{S}v_k = \rho_k$.

We might sometimes decide to rescale inputs to have standard deviation one before finding principal component directions (or otherwise rescale them).

Why do These Eigenvectors Give the Maximum Variance?

Let v be a unit vector pointing in the direction of maximum variance. The sample variance in this direction is

$$\text{Var}(\mathbf{X}v) \;=\; (1/N)\,(\mathbf{X}v)^T(\mathbf{X}v) \;=\; (1/N)\,v^T\mathbf{X}^T\mathbf{X}v \;=\; v^TSv$$

We can maximize this subject to v being a unit vector (ie, $v^Tv = 1$) using a Lagrange multiplier. At the maximum, for some ρ:

$$\begin{aligned} \nabla(v^TSv) &= \rho\nabla(v^Tv) \\ 2Sv &= 2\rho v \\ Sv &= \rho v \end{aligned}$$

So v is an eigenvector of S with eigenvalue ρ. The maximum variance occurs when v is the eigenvalue for which ρ is biggest.

So the first principal component, v_1, points in the direction of maximum variance. If we project all points into the space orthogonal to v_1, we can then see that the second principal component is the direction of next highest variance, etc.

Principal Components Regression

For some $M < p$, we can use $z_1 = Xv_1, \ldots, z_M = Xv_M$ as inputs for a least squares linear regression model of Y. If we first centred the X_i, we can write this model as

$$Y \;=\; \bar{y} \;+\; \theta_1 z_1 \;+\; \cdots \;+\; \theta_M z_M \;+\; \text{noise}$$

The predictions of this model are the same as those of a linear regression model using the original inputs with the following coefficients:

$$\beta \;=\; \sum_{m=1}^{M} \theta_m v_m$$

When will this be a good way of creating a regression model? When the directions of high variance in the input space are also the directions in which Y changes.

There's no guarantee that this will be the case — conceivably, the directions of *low* variance might be the ones that matter. But it often works well in practice.

If $p \gg n$, the first few principal components can be found much faster from $\mathbf{X}\mathbf{X}^T$ than from $\mathbf{X}^T\mathbf{X}$. Because of this, principal components regression is feasible as long as *at least one* of n and p is less than about 3000.