Ridge Regression Explained

James Taylor

In the multivariate regression setting, the estimated coefficients of correlated predictor variables are very unstable - a large positive coefficient for one variable can be offset by a large negative coefficient for a correlated variable. Ridge regression mitigates this problem by introducing an (L^2) regularization term which penalizes large values of the model's (non-intercept) coefficients.

The setup. Suppose we are in the traditional multivariate regression setting with one quantitative response Y and p predictors $X_1, X_2, \ldots, X_p$, with an (approximately) linear relationship between them:

$$Y \approx \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p$$

Say we have training data consisting of n observations. We want to use these data to find approximations b_i for β_i; we will then be able to predict future values of Y given $(X_1, \ldots, X_p) = (x_1, \ldots, x_p)$ via the formula

$$\hat{y} = b_0 + b_1 x_1 + \cdots + b_p x_p$$

Our estimates for the model coefficients are going to be

$$\text{argmin}_b \left[RSS + \alpha \sum_{i=1}^{p} b_i^2 \right]$$

where $\alpha \geq 0$ is some user-supplied constant. For $\alpha = 0$ ridge regression is just ordinary least squares regression. But as α gets larger, the regularization term becomes an increasingly larger part of the optimization function, thus forcing the optimal $(b_1, \ldots, b_p)$ to have smaller L^2 norm. Clearly, as $\alpha \to \infty$ the L^2 norm of $(b_1, \ldots, b_p)$ goes to 0.

Note: Traditionally, one centers the data, estimates β_0 with $b_0 = \frac{1}{n} \sum_{i=1}^{n} y_i$, and the performs ridge regression without intercept to derive the equation $b = (X^T X + \alpha I)^{-1} X^T y$, where X is the $n \times p$ matrix containing our observed predictor data. We are not going to go this route.

Minimizing the ridge regression equation. Letting our n observations be (x_i, y_i) for $i = 1, \ldots, n$ where $x_i = (x_{i1}, x_{i2}, \ldots, x_{in})$, we want to minimize

$$\begin{aligned} RSS + \alpha \sum_{i=1}^{p} b_i^2 &= \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \alpha \sum_{i=1}^{p} b_i^2 \\ &= \sum_{i=1}^{n} (y_i - (b_0 + b_1 x_{i1} + \cdots + b_p x_{ip}))^2 + \alpha \sum_{i=1}^{p} b_i^2 \end{aligned}$$

We minimize this function by differentiating with respect to b_j, setting this expression equal to zero, and simplifying the resulting equation to be of the form $\rho_j \cdot b = \gamma_j$. We do this for each j, and then solve the following matrix equation for b:

$$\rho b = \gamma$$

where ρ is the $(p+1) \times (p+1)$ matrix whose jth row is ρ_{j-1}, and γ is the $(p+1)$-vector with jth entry γ_{j-1}.

More explicitly, for $j = 0$ we have

$$\begin{aligned}
\frac{\partial}{\partial b_0}\left(RSS + \alpha \sum_{i=1}^{p} b_i^2\right) &= \sum_{i=1}^{n} -2(y_i - (b_0 + b_1 x_{i1} + \cdots + b_p x_{ip})) = 0 \\
&\implies \sum_{i=1}^{n}(y_i - (b_0 + b_1 x_{i1} + \cdots + b_p x_{ip})) = 0 \\
&\implies n b_0 + b_1 \sum_{i=1}^{n} x_{i1} + \cdots + b_p \sum_{i=1}^{n} x_{ip} = \sum_{i=1}^{n} y_i
\end{aligned}$$

and for $j = 1, \ldots, p$ we have

$$\begin{aligned}
\frac{\partial}{\partial b_j}\left(RSS + \alpha \sum_{i=1}^{p} b_i^2\right) &= \sum_{i=1}^{n} -2x_{ij}(y_i - (b_0 + b_1 x_{i1} + \cdots + b_p x_{ip})) + 2\alpha b_j = 0 \\
&\implies \sum_{i=1}^{n} x_{ij}(y_i - (b_0 + b_1 x_{i1} + \cdots + b_p x_{ip})) - \alpha b_j = 0 \\
&\implies \alpha b_j + \sum_{i=1}^{n} x_{ij}(b_0 + b_1 x_{i1} + \cdots + b_p x_{ip}) = \sum_{i=1}^{n} x_{ij} y_i \\
&\implies b_0 \sum_{i=1}^{n} x_{ij} + b_1 \sum_{i=1}^{n} x_{ij} x_{i1} + \cdots + b_j(\alpha + \sum_{i=1}^{n} x_{ij}^2) + \cdots + b_p \sum_{i=1}^{n} x_{ij} x_{ip} = \sum_{i=1}^{n} x_{ij} y_i
\end{aligned}$$

These equations give rise to the following matrix equation:

$$\begin{pmatrix}
n & \sum_{i=1}^n x_{i1} & \sum_{i=1}^n x_{i2} & \cdots & \sum_{i=1}^n x_{ip} \\
\sum_{i=1}^n x_{i1} & \alpha + \sum_{i=1}^n x_{i1}^2 & \sum_{i=1}^n x_{i1}x_{i2} & \cdots & \sum_{i=1}^n x_{i1}x_{ip} \\
\sum_{i=1}^n x_{i2} & \sum_{i=1}^n x_{i2}x_{i1} & \alpha + \sum_{i=1}^n x_{i2}^2 & \cdots & \sum_{i=1}^n x_{i2}x_{ip} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
\sum_{i=1}^n x_{ip} & \sum_{i=1}^n x_{ip}x_{i1} & \sum_{i=1}^n x_{ip}x_{i2} & \cdots & \alpha + \sum_{i=1}^n x_{ip}^2
\end{pmatrix}
\begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ b_p \end{pmatrix}
=
\begin{pmatrix} \sum_{i=1}^n y_i \\ \sum_{i=1}^n x_{i1}y_i \\ \sum_{i=1}^n x_{i2}y_i \\ \vdots \\ \sum_{i=1}^n x_{ip}y_i \end{pmatrix}$$

Let x_{*j} be the n-vector consisting of all n observations of the jth predictor X_j. If we set x_{*0} to be the n-vector containing all 1's, then this equation becomes

$$\begin{pmatrix}
x_{*0} \cdot x_{*0} & x_{*0} \cdot x_{*1} & x_{*0} \cdot x_{*2} & \cdots & x_{*0} \cdot x_{*p} \\
x_{*1} \cdot x_{*0} & \alpha + x_{*1} \cdot x_{*1} & x_{*1} \cdot x_{*2} & \cdots & x_{*1} \cdot x_{*p} \\
x_{*2} \cdot x_{*0} & x_{*2} \cdot x_{*1} & \alpha + x_{*2} \cdot x_{*2} & \cdots & x_{*2} \cdot x_{*p} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
x_{*p} \cdot x_{*0} & x_{*p} \cdot x_{*1} & x_{*p} \cdot x_{*2} & \cdots & \alpha + x_{*p} \cdot x_{*p}
\end{pmatrix} b =
\begin{pmatrix} x_{*0} \cdot y \\ x_{*1} \cdot y \\ x_{*2} \cdot y \\ \vdots \\ x_{*p} \cdot y \end{pmatrix}$$

Notice that the first matrix is $X^T X + diag(0, \alpha, \ldots, \alpha)$, and the RHS is $X^T y$, where X is the $n \times (p+1)$ matrix

$$\begin{pmatrix}
1 & x_{11} & \cdots & x_{1p} \\
1 & x_{21} & \cdots & x_{2p} \\
\vdots & \vdots & \ddots & \vdots \\
1 & x_{n1} & \cdots & x_{np}
\end{pmatrix}$$

So our matrix equation is now

$$(X^T X + diag(0, \alpha, \ldots, \alpha))b = X^T y$$

and thus

$$b = (X^T X + diag(0, \alpha, \ldots, \alpha))^{-1} X^T y$$

Note that, depending on how large X is, inverting $X^T X + diag(0, \alpha, \ldots, \alpha)$ could be extremely resource intensive. A way around this is to compute the QR decomposition of $X^T X + diag(0, \alpha, \ldots, \alpha)$; that is, we find an orthogonal matrix Q and an upper triangular matrix R such that $X^T X + diag(0, \alpha, \ldots, \alpha) = QR$. (Note: A QR decomposition exists for every matrix.) We'll still need to solve $QRb = X^T y$, but since Q is orthogonal we have $Q^{-1} = Q^T$, and thus we only need to solve

$$b = R^{-1} Q^T X^T y = R^{-1} (XQ)^T y$$

So now the only matrix we need to invert is the triangular matrix R, which is a much simpler task.