Linear & Generalized Linear Models/ Advanced Regression for Independent Data

Andrew J. Spieker, Ph.D.

Associate Professor of Biostatistics
Vanderbilt University

Set 17 (Supplement 3): Regularization

Version: 07/03/2026

Table of Contents

1 Penalized regression

2 The ridge penalty

3 The LASSO penalty

Motivation:

- So far, we have assumed that our $N \times K$ design matrix, $\mathbf{X}$, is such that its K predictors could be supported by N observations ($N \gg K$).
- However, sometimes the covariate space that we're seeking to understand is extremely high-dimensional relative to any sample size we can reasonably obtain.
  - We estimate that humans have about 20,000-25,000 genes.
  - ...and about 3.2 billion base pairs.
- We can handle non-full rank $\mathbf{X}$ for $N \gg K$.
- However, OLS is not equipped to handle $N \leq K$, and the case where $N \not\gg K$ is far from optimal.
  - Some rules of thumb call for at least ten independent observations for each one degree of freedom used.
- In these notes, consider $\mathbf{X}$ fixed for simplicity of calculations.

Degrees of freedom:

- One way to define degrees of freedom used by a model (in the case of constant variance) is through the relationship between the observed and fitted values:

$$\text{df}(\widehat{\mathbf{y}}) = \frac{1}{\sigma^2}\sum_{i=1}^{N}\text{Cov}[Y_i, \widehat{Y}_i]$$

- We can use this to show that OLS uses K degrees of freedom:

$$\begin{aligned}
\text{df}(\widehat{\mathbf{y}}) &= \frac{1}{\sigma^2}\sum_{i=1}^{N}\text{Cov}[Y_i, \mathbf{x}_i^\intercal(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{x}_i Y_i] \\
&= \frac{1}{\sigma^2}\sum_{i=1}^{N}\mathbf{x}_i^\intercal(\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{x}_i\text{Cov}[Y_i, Y_i] \\
&= \sum_{i=1}^{N}\mathbf{x}_i^\intercal(\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{x}_i = \text{tr}(\mathbf{P}) = \sum_{i=1}^{N}h_i = K.
\end{aligned}$$

Degrees of freedom:

- Intuitively, the more degrees of freedom used by a model, the more closely correlated the fitted and observed values. Adding more degrees of freedom to a model improves the model's fit *to the data*.
- The problem is that our goal is not simply to predict observations in our own data, but to use the data to predict observations *external* to our data.
- Many machine-learning approaches focus on ways to balance the following trade-off:
  - Allowing the model to use available covariate information.
  - Controlling the model from overfitting the data.
- We'll focus in this set of notes specifically on penalized least squares.
- Note, however, that I am doing **nothing** in this set of notes to address proper *selection* of a penalty (e.g., through cross-validation, the subject of other courses).

Prediction error: Bias-variance decomposition

- Controlling degrees of freedom is central to being able to navigate the bias-variance trade-off.
- Consider the goal of estimating $\mathrm{E}[Y] = f(\mathbf{x})$.
- Estimate f with $\widehat{f}$.
- Consider random out-of-sample observation, $(\mathbf{x}_0, Y_0)$.
- Mean squared error at $\mathbf{x}_0$ is:

$$\begin{aligned}\mathrm{E}[(Y_0 - \widehat{f}(\mathbf{x}_0))^2] &= \mathrm{E}[(Y_0 - \mathrm{E}[Y_0])^2] + \mathrm{E}[(\mathrm{E}[Y_0] - \mathrm{E}[\widehat{f}(\mathbf{x}_0)])^2] \\ &\quad + \mathrm{E}[(\widehat{f}(\mathbf{x}_0) - \mathrm{E}[\widehat{f}(\mathbf{x}_0)])^2] \\ &= \sigma^2_{Y_0} + \mathrm{Bias}^2(\widehat{f}(\mathbf{x}_0)) + \mathrm{Var}(\widehat{f}(\mathbf{x}_0)).\end{aligned}$$

- Prediction error: trade-off between bias and variance.
  - Bias: Model correctly captures f on average?
  - Variance: Model precisely captures relationship?

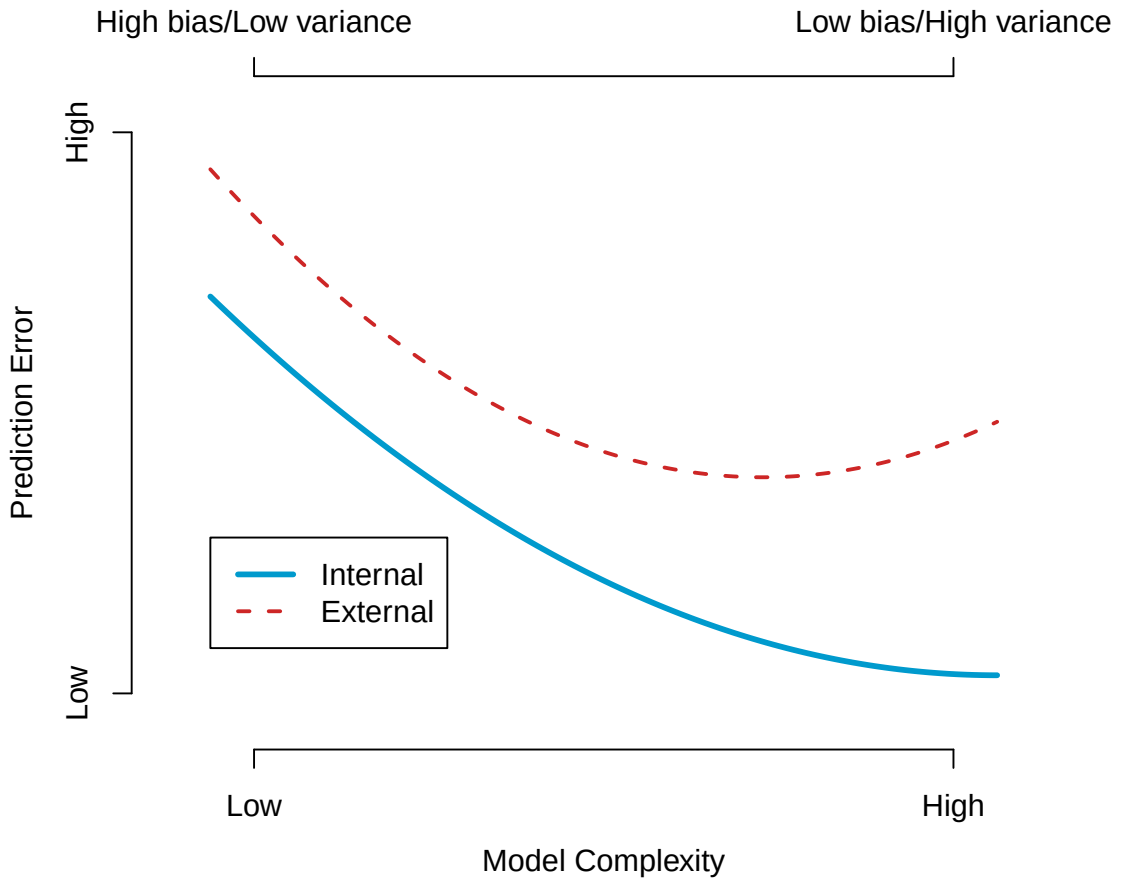

Figure S3.1: The bias-variance trade-off.

Table of Contents

1 Penalized regression

2 The ridge penalty

3 The LASSO penalty

The ridge penalty

Objective function:

- We understand OLS as the following optimization problem:

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\text{argmin}} \left\{ ||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2 \right\}.$$

- One way to control the degrees of freedom used by the model is to shrink some of the coefficients toward zero.
  - Including a predictor in a model in an unconstrained fashion uses one degree of freedom.
  - Leaving the predictor out of the model effectively treats the corresponding coefficient as zero and uses no degrees of freedom.
  - Intuitively, including a predictor in a model with a shrunken coefficient should use "somewhere between zero and one" degrees of freedom.

The ridge penalty

Objective function:

- Suppose we add a penalty to the objective function:

$$\widehat{\boldsymbol{\beta}}_\lambda = \underset{\boldsymbol{\beta}|\lambda}{\text{argmin}} \left\{ ||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2 + \lambda ||\boldsymbol{\beta}||^2 \right\}.$$

- The parameter λ is not "estimated" per se, but rather "chosen" by the researcher in some informed way (typically data-driven, but more on this later).
  - $\lambda = 0$: this is just the OLS problem (K degrees of freedom).
  - $\lambda = \infty$: $\widehat{\boldsymbol{\beta}} = \mathbf{0}$ (zero degrees of freedom).
  - $0 < \lambda < \infty$: $||\widehat{\boldsymbol{\beta}}||^2$ shrunken toward 0 (>0, $<K$ degrees of freedom).

Objective function:

- How do we obtain a solution to this optimization problem?

$$\widehat{\boldsymbol{\beta}}_\lambda = \underset{\boldsymbol{\beta}|\lambda}{\text{argmin}} \left\{ ||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2 + \lambda ||\boldsymbol{\beta}||^2 \right\}.$$

- This is a convex loss function; the calculus is similar to the OLS case:

$$\begin{aligned} \frac{\partial}{\partial \boldsymbol{\beta}} \left(||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2 + \lambda ||\boldsymbol{\beta}||^2 \right) &= -2\mathbf{X}^\intercal(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) + 2\lambda\boldsymbol{\beta} \\ &= -2\mathbf{X}^\intercal\mathbf{y} + 2\mathbf{X}^\intercal\mathbf{X}\boldsymbol{\beta} + 2\lambda\boldsymbol{\beta} \\ \\ \Rightarrow \widehat{\boldsymbol{\beta}}_\lambda &= (\mathbf{X}^\intercal\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\intercal\mathbf{y}. \end{aligned}$$

Solution to penalized least squares equations:

- For the ridge penalty, there is a closed-form solution:

$$\widehat{\boldsymbol{\beta}}_\lambda \quad = \quad (\mathbf{X}^\intercal\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\intercal\mathbf{y}.$$

- Letting $d_1, \ldots, d_K$ denote the singular values of $\mathbf{X}$, we have that:

$$\text{df}_\lambda(\widehat{\mathbf{y}}) \quad = \quad \sum_{k=1}^{K} \frac{d_k^2}{d_k^2 + \lambda},$$

  which you will prove for homework!
  - Note that d_k^2 is an eigenvalue of $\mathbf{X}^\intercal\mathbf{X}$.
- Quick checks of sensibility:
  - If $\lambda = 0$, then $\text{df}_\lambda(\widehat{\mathbf{y}}) = K$.
  - If $\lambda = \infty$, $\text{df}_\lambda(\widehat{\mathbf{y}}) = 0$.

The ridge penalty

Penalized estimates:

- It is computationally cleaner to center the variables to avoid estimating the intercept (though I can estimate it on the back end).
- The ridge regression is not scale-invariant!!
  - If one of the predictors is height, I get a different answer depending on whether I measure in inches, feet, meters, etc.
  - We can deal with this by scaling everything to have unit variance prior to estimation.
- If $\widehat{\boldsymbol{\beta}}_\lambda$ denotes the standardized (centered/scaled) estimates:
  - The penalized coefficients can be obtained as $\sigma_Y \text{diag}(1/\sigma_{x_k})\widehat{\boldsymbol{\beta}}_\lambda$.
  - $\widehat{\beta}_{\lambda,0} = \mu_y - \sigma_Y \text{diag}(1/\sigma_{x_k})\boldsymbol{\mu}_{\mathbf{X}}^{\mathsf{T}}\widehat{\boldsymbol{\beta}}_\lambda$ is the penalized intercept.

Simulation: Parameter setup

```
## Set seed for reproducibility
set.seed(7345)

## Set sample size
n <- 100

## Generate predictors
x <- rmvnorm(n, mean = rep(0, 4), sigma = 0.5*diag(4))

## Generate outcome
y <- 3 + 4*x[,1] - 2*x[,2] + 0.1*x[,3] - 1*x[,4] + rnorm(n, 0, 2)

## Center/scale cariables
means <- colMeans(x); sds <- apply(x, 2, sd)
meanY <- mean(y); sdY <- sd(y)
x1 <- (x[,1] - means[1])/sds[1]
x2 <- (x[,2] - means[2])/sds[2]
x3 <- (x[,3] - means[3])/sds[3]
x4 <- (x[,4] - means[4])/sds[4]
y <- (y - meanY)/sdY
```

Simulation: Fitting models; extracting results

```
## Generate design matrix
X <- cbind(x1, x2, x3, x4)

## Choose values for tuning parameter
lambda <- seq(0.1, 100000, 1)

## Create place to store results
results <- matrix(0, nrow = length(lambda), ncol = 4)
dfres <- matrix(0, nrow = length(lambda), ncol = 1)

for (j in 1:length(lambda))
{
  ## Compute estimate, re-scale, and store results
  bhat.lambda <- solve(t(X) %*% X + diag(lambda[j], 4)) %*% t(X) %*% y
  results[j,] <- sdY*bhat.lambda/sds

  ## Determine degrees of freedom
  sv <- svd(X)$d
  dfres[j,1] <- sum(sv^2/(sv^2 + lambda[j]))
}
```

The ridge penalty

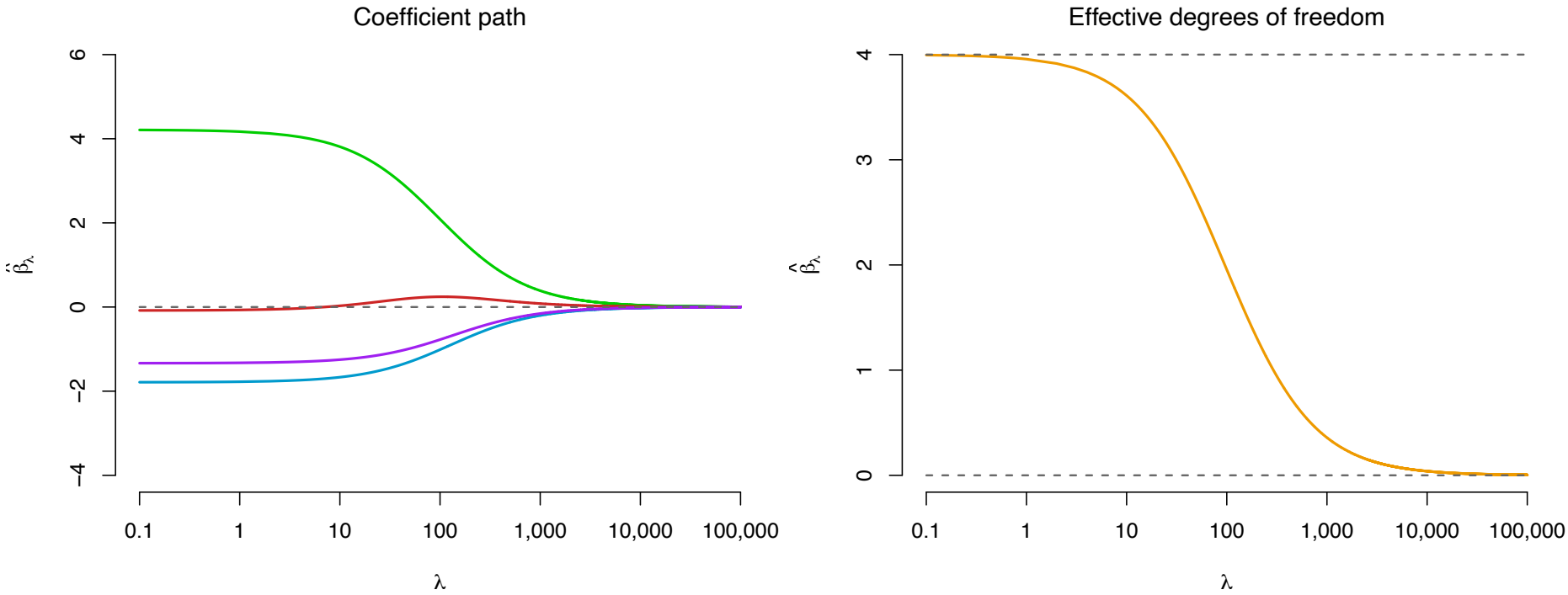

Figure S3.2: Results of a single simulation.

The ridge penalty

Solution to penalized least squares equations:

- Interestingly, $(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda\mathbf{I})$ is invertible for all choices of $\lambda > 0$, even if $\mathbf{X}^\mathsf{T}\mathbf{X}$ is singular.
- This has to do with the fact that λ serves to shift eigenvalues, the minimum of which can be zero for $\mathbf{X}^\mathsf{T}\mathbf{X}$.

The ridge penalty

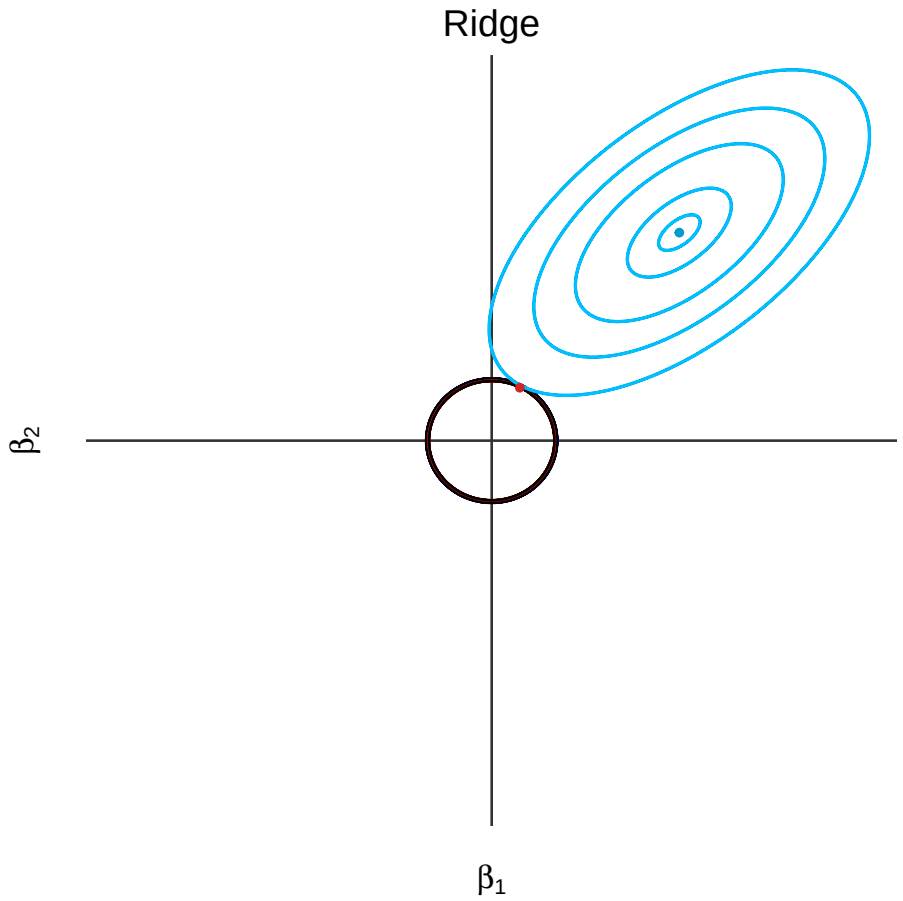

Figure S3.3: Understanding the ridge penalty.

Ridge and Bayes:

- Suppose that $\mathbf{y} \sim \mathcal{N}(\mathbf{X}\boldsymbol{\beta}, \sigma^2\mathbf{I})$.
- Consider the Gaussian prior $\boldsymbol{\beta} \sim \mathcal{N}(\mathbf{0}, \tau^2\mathbf{I})$.
- It is straightforward to show that the posterior mode, $\widehat{\boldsymbol{\beta}}_\pi$, is given by:

$$\widehat{\boldsymbol{\beta}} = \left(\mathbf{X}^\top\mathbf{X} + \frac{\sigma^2}{\tau^2}\mathbf{I}\right)^{-1}\mathbf{X}^\top\mathbf{y}.$$

- Letting $\lambda = \sigma^2/\tau^2$ makes clear the connection with the ridge penalty.
- Bayesian methods can be understood more generally as shrinkage approaches (it is arguably one of the best ways to think about Bayesian methods).

Table of Contents

1 Penalized regression

2 The ridge penalty

3 The LASSO penalty

The LASSO penalty

Objective function:

- There's no rule that says we *have* to penalize via the ℓ_2-norm. Suppose we instead penalize based on the ℓ_1-norm:

$$\widehat{\boldsymbol{\beta}}_\lambda = \underset{\boldsymbol{\beta}|\lambda}{\text{argmin}} \left\{ \frac{1}{2}||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2 + \lambda||\boldsymbol{\beta}||_1 \right\}.$$

  where $||\boldsymbol{\beta}||_1 = \sum_{k=1}^{K} |\beta_k|$.
- While a different choice of a penalty seems as if it should just be a technicality, the LASSO penalty actually produces very different behavior as compared to the ridge penalty.
- Geometrically, the fact that $||\boldsymbol{\beta}||^2$ is "circular" and $||\boldsymbol{\beta}||_1$ "has sharp corners" is what accounts for differences in behavior.

The LASSO penalty

Objective function:

- Rephrasing the ridge penalty problem:

$$\underset{\boldsymbol{\beta}}{\text{minimize}} \; ||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2 \text{ subject to constraint } ||\boldsymbol{\beta}||^2 \leq c.$$

- Rephrasing the LASSO penalty problem:

$$\underset{\boldsymbol{\beta}}{\text{minimize}} \; ||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2 \text{ subject to constraint } ||\boldsymbol{\beta}||_1 \leq c.$$

The LASSO penalty

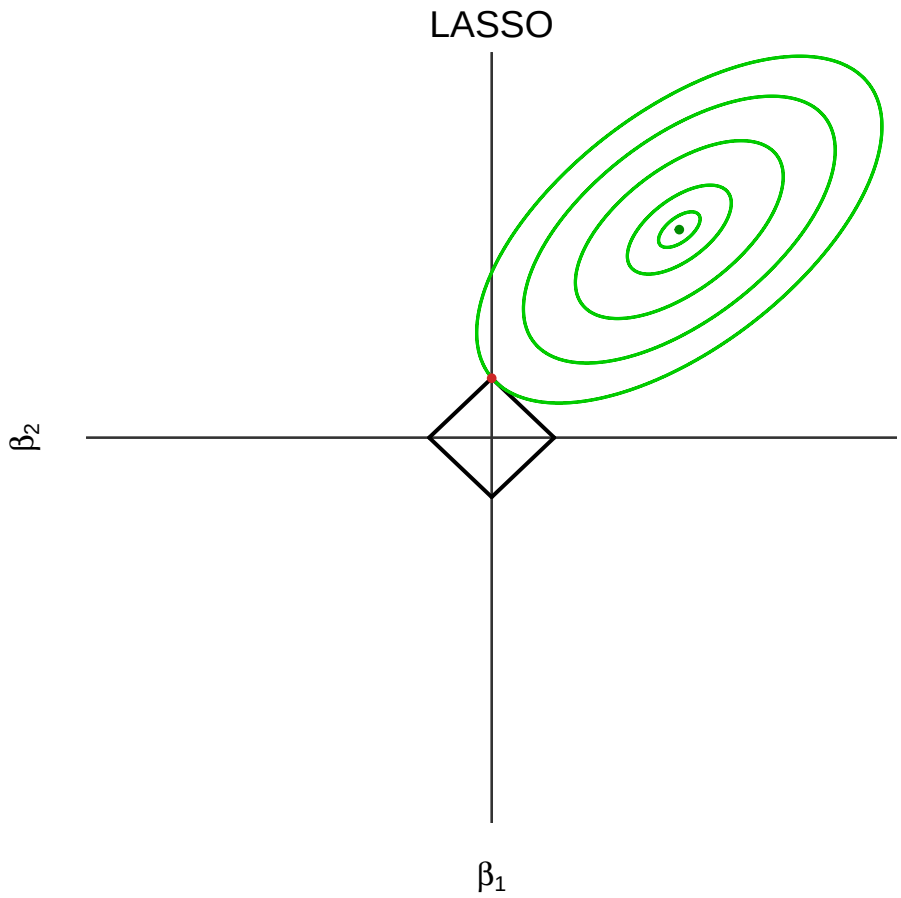

Figure S3.4: Understanding the LASSO penalty.

The LASSO penalty

Geometry:

- While the ridge penalty shrinks coefficients *toward* zero, the LASSO penalty shrinks coefficients *to* zero.
- LASSO is said to perform "feature selection," so that it allows you to determine a smaller set of predictors that must be measured to form a prediction on an external observation. This will have good performance when there are a large number of variables having no association with the outcome.
  - This is still a source of debate and controversy.
- On the other hand, ridge will perform better when most of the predictors are associated with the outcome.

The LASSO penalty

Estimation:

- Sadly, there is no closed-form expression for the LASSO-penalized least squares problem.
- Instead, we rely on coordinate-descent algorithms to obtain numerical solutions. These algorithms are lightning-fast (same computational cost as ridge), but a pain to code.
- Under certain assumptions, the number of degrees of freedom used by the LASSO model can be shown to be estimated as the number of non-zero predictors selected into the model.