Lecture 3: Regularization and Smoothing

Introduction to Time Series, Fall 2023

Ryan Tibshirani

1 Trouble in high dimensions?

- As in the regression lecture, let's suppose that we seek $\beta \in \mathbb{R}^p$ such that for given samples $x_i \in \mathbb{R}^p$ and $y_i \in \mathbb{R}$ (predictor and response pairs), $i = 1, \ldots, n$,

$$y_i \approx x_i^\mathsf{T} \beta, \quad i = 1, \ldots, n$$

  (As explained in that lecture, our notation omits the intercept from the model, but that is done without a loss of generality, since it can always be obtained by appending a coordinate value of 1 to the start of each x_i)

- Equivalently, we can write $y \in \mathbb{R}^n$ for the response vector (with i^{th} component y_i) and $X \in \mathbb{R}^{n \times p}$ for the feature matrix (with i^{th} row x_i), and say that we are seeking β such that $y \approx X\beta$

- Recall, the least squares estimates of the coefficients are given by solving

$$\min_{\beta \in \mathbb{R}^p} \sum_{i=1}^n (y_i - x_i^\mathsf{T} \beta)^2 \iff \min_{\beta \in \mathbb{R}^p} \|y - X\beta\|_2^2 \tag{1}$$

  where we use $\|\cdot\|_2$ for the Euclidean or ℓ_2 norm of a vector, defined for $a \in \mathbb{R}^d$ as $\|a\|_2^2 = \sum_{i=1}^d a_i^2$

- If $p \leq n$ and $\text{rank}(X) = p$ (here $\text{rank}(X)$ denotes the rank of the matrix X), then this produces the unique solution

$$\hat{\beta} = (X^\mathsf{T} X)^{-1} X^\mathsf{T} y \tag{2}$$

- But if $p > n$, which means we have more features than samples, which we often call the "high dimensional" (or "overparametrized") setting, then we are in trouble ... the matrix $X^\mathsf{T} X$ cannot be invertible, so the expression in (2) isn't even well-defined

- Moreover, the least squares optimization problem (1) does not have a unique solution in this case. Indeed, are you'll show on the homework, it $\tilde{\beta}$ is one solution, then any other vector of the form

$$\hat{\beta} = \tilde{\beta} + \eta, \quad \text{where } \eta \in \text{null}(X) \tag{3}$$

  also solves (1), where $\text{null}(X)$ is the null space of the matrix X:

$$\text{null}(X) = \{\eta \in \mathbb{R}^p : X\eta = 0\}$$

- When $\text{rank}(X) < p$, the null space $\text{null}(X)$ is nonempty, and since it is a linear space: $\eta \in \text{null}(X) \implies c\eta \in \text{null}(X)$ for any $c \in \mathbb{R}$, we see that from one least squares solution $\tilde{\beta}$, we can generate *infinitely many others* in (3)

- Furthermore, we can always take the "one least squares solution" to be:

$$\tilde{\beta} = (X^\mathsf{T} X)^+ X^\mathsf{T} y \tag{4}$$

- Here A^+ denotes the *generalized inverse* (also called the *Moore-Penrose pseudoinverse*) of a matrix A. If you don't know what that is, then it doesn't really matter for this lecture, but you can think of it precisely as follows: among all solutions in (1), the solution in (4) is the unique one having the smallest ℓ_2 norm $\|\tilde{\beta}\|_2$

- So, now we come to the discussion of specific *troubles*. There are actually two distinct troubles. The first trouble involves the interpretation of the coefficients themselves. If we are interested in such interpretations, then the $p = n$ barrier is the end of the road for least squares. Why? Once $p > n$, and we find any least squares solution $\tilde{\beta}$ with $\tilde{\beta}_j > 0$ for some j, then we can always find[1] another solution $\hat{\beta}$ of the form (3) with $\hat{\beta}_j < 0$. You will prove this on the homework. Thus we cannot even consistently interpret the sign of any of any estimated coefficient (let alone its magnitude)

- The second trouble involves prediction. The $p = n$ barrier is generally disastrous for least squares prediction. If $p < n$ and $\hat{y}_{\text{new}} = x_{\text{new}}^{\mathsf{T}}\hat{\beta}$ is the least squares prediction at a new predictor value x_{new} (for $\hat{\beta}$ the usual least squares coefficients in (2)), whose associated response is y_{new}, then under fairly standard conditions for regression theory, the prediction MSE behaves as:

$$\mathbb{E}\big[(y_{\text{new}} - \hat{y}_{\text{new}})^2\big] \approx \sigma^2 \frac{p}{n-p}$$

  for large n and p, where σ^2 is the error variance. What do we notice? This explodes as p approaches n. Big problem!

- (Aside: what happens with the prediction MSE when $p > n$? The answer may surprise you. The MSE associated with the particular solution in (4) is actually quite interesting and in some ways exotic when $p > n$. Typically we need p to be *much larger* than n (away from the $p = n$ barrier) in order for it to be well-behaved. This has been the topic of a recent flurry of research in statistics in machine learning ... we won't focus on it in this lecture and will instead talk about explicit regularization. But feel free to ask about it in office hours)

2 Regularization

- *Regularization* to the rescue! This will finesse both of the problems described above: it gives us a way to produce nontrivial coefficient estimates, and it often gives us more accurate predictions

- In the regression setting, a general approach for regularization moves us from (1) to solving:

$$\min_{\beta} \ \|y - X\beta\|_2^2 + h(\beta) \tag{5}$$

- Here $h : \mathbb{R}^p \to \mathbb{R}_+$ is some (typically convex) penalty function. Arguably the three canonical choices for penalties are based on the ℓ_0, ℓ_1, and ℓ_2 norms:

$$h(\beta) = \|\beta\|_0 = \sum_{j=1}^{p} 1\{\beta_j \neq 0\},$$

$$h(\beta) = \|\beta\|_1 = \sum_{j=1}^{p} |\beta_j|,$$

$$h(\beta) = \|\beta\|_2^2 = \sum_{j=1}^{p} \beta_j^2.$$

  This gives rise to what we call *best subset selection*, the *lasso*, and *ridge regression*, respectively

- (Aside: calling $\|\cdot\|_0$ the "ℓ_0 norm" is a misnomer, as it is not actually a norm: it does not satisfy positive homogeneity, i.e., $\|a\beta\|_0 = a\|\beta\|_0$ for all $a > 0$. It would be more accurate to call it the "ℓ_0 pseudonorm", but nearly everybody just calls it the "ℓ_0 norm")

- Critically, $\|\cdot\|_0$ is *not convex*, while $\|\cdot\|_1$ and $\|\cdot\|_2$ are convex (note that any norm is a convex function). This makes best subset selection a nonconvex problem, and one that is generally very hard to solve in practice except for small p. We won't focus on best subset selection further in this lecture. (Though it is itself the topic of a flurry of work in the operations research literature a few years ago ... which you can ask about in office hours if you are curious)

[1] Technically, this is only true if $\text{null}(X) \not\perp e_j$, where e_j is the j^{th} standard basis vector.

2.1 Ridge

- The *ridge* estimates of regression coefficients are given by solving

$$\min_{\beta \in \mathbb{R}^p} \sum_{i=1}^{n} (y_i - x_i^\mathsf{T} \beta)^2 + \lambda \sum_{j=1}^{p} \beta_j^2$$

  or equivalently

$$\min_{\beta \in \mathbb{R}^p} \|y - X\beta\|_2^2 + \lambda \|\beta\|_2^2 \tag{6}$$

- Here $\lambda \geq 0$ is a tuning parameter (also called the regularization parameter) which trades off the importance of the squared loss—first term in (6), with the ridge penalty—second term in (6). In other words, the larger the value of λ, the more weight we put on the ridge penalty, which penalizes large coefficient magnitudes very heavily. This results in estimates that we call more "regularized"
- The solution in (6) has an explicit form (derived by differentiating the criterion and setting the result equal to zero),

$$\hat{\beta} = (X^\mathsf{T} X + \lambda I)^{-1} X^\mathsf{T} y \tag{7}$$

  where I is in the $p \times p$ identity matrix. This *always exists* (the matrix $X^\mathsf{T} X + \lambda I$ is always invertible), regardless of the relative sizes of n, p

2.2 Lasso

- The *lasso* estimates of regression coefficients are given by solving

$$\min_{\beta \in \mathbb{R}^p} \sum_{i=1}^{n} (y_i - x_i^\mathsf{T} \beta)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

  or equivalently

$$\min_{\beta \in \mathbb{R}^p} \|y - X\beta\|_2^2 + \lambda \|\beta\|_1 \tag{8}$$

- Again, $\lambda \geq 0$ is a tuning parameter trading off the importance of the squared loss and the ℓ_1 penalty—first and second terms in (8). As before, larger λ gives us more regularized estimates
- There are a number of key differences between the ridge (6) and (8) problems (and estimator). First, the lasso problem does not have a closed-form solution (it does not even necessarily always admit a unique solution; though it is essentially always unique if we have continuously distributed features)
- A second key difference is that the lasso estimates of the regression coefficients are *sparse*. In other words, solving the lasso problem results in a vector $\hat{\beta}$ with many components exactly equal to zero, and a larger choice of λ will result in more zeros. This doesn't happen with ridge regression, whose coefficient estimates are generically *dense*. Figure 1 is the "classic" picture used to explain this, and we will talk through its interpretation in lecture
- Importantly, this allows the lasso to perform *variable selection* in the working linear model. By zeroing out some coefficients exactly, it discards some features from having any predictive influence in the fitted model. Which precise features it discards is chosen based on the data. Many people like sparsity because it leads to better interpretability

2.3 Discussion

- We should be clear that the lasso is not "better" than ridge, in any general sense, and neither is ridge "better" than the lasso. They each can help tremendously with stabilizing coefficient estimates so as to lead to improved predictive accuracy. They each do so by regularizing in different ways

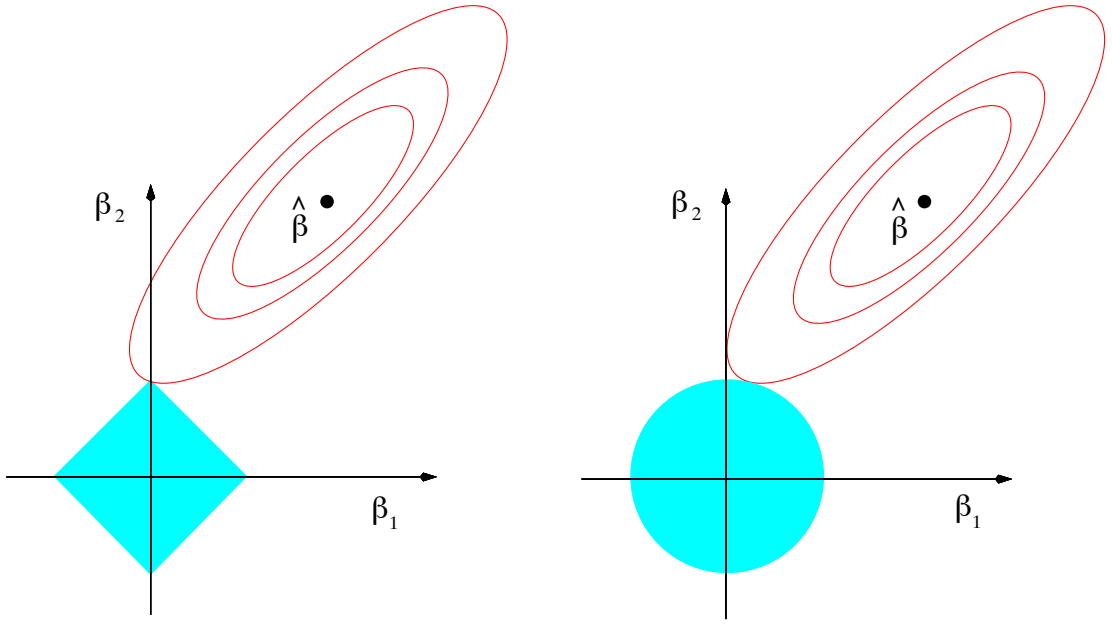

Figure 1: *The "classic" illustration comparing lasso and ridge, each in constrained form (from "Elements of Statistical Learning" by Hastie, Tibshirani, and Friedman)*

- The most basic question to ask: is a sparse linear model likely to be a good (or desirable) approximation to the true regression function? If so, then the lasso can outperform (or be preferable) to ridge. On the other hand, in problems where there are many underlying features that are relevant for prediction, ridge can outperform the lasso
- (And often times people combine the two penalties which gives rise to the *elastic net*)
- There is a lot more to say—in terms of connections to other ideas in statistics, extensions, and so on—but we won't be able to cover it in this class. We'll simply view ridge and lasso as tools that allow us to consider many more features than we would otherwise feel comfortable including in traditional regression models, and then regularize in order to control variance (stabilize estimates)
- In time series regression, in the vein of examples we studied in the last lecture, this would allow us to include *many lags* of a feature of interest, or a few lags of *many external covariates*, and so on, and then apply a ridge or lasso penalty. Then, you might wonder: how would we select λ? In fact, you already know the answer (for problems with a predictive focus): use time series cross-validation!
- That is, define a grid of λ values, fit ridge or lasso estimates for each λ, let each one make predictions, and select the value that yields the best CV error. You will practice this on the homework, where you'll also use the `glmnet` package to solve the ridge and lasso problems

3 Smoothing

3.1 Linear filters

3.2 Hodrick-Prescott filter

3.3 Trend filter