

# STATS305B – Lecture 10

Jonathon Taylor  
Scribed by Michael Howes

02/07/22

## Contents

<b>1 Regularized glms</b>	<b>1</b>
1.1 Fitting a ridge glm	1
1.2 Why regularize	2
<b>2 LASSO regularization</b>	<b>2</b>
2.1 The LASSO in one dimension	2
2.2 Why the LASSO?	5
2.3 Fitting the LASSO	5
2.3.1 Coordinate descent	5
2.3.2 Proximal gradient descent	7

## 1 Regularized glms

### 1.1 Fitting a ridge glm

The objective for fitting a glm can be written as

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \Lambda(X\beta) - \beta^T(X^TY) = \underset{\beta}{\operatorname{argmin}} -\log L(\beta|Y).$$

The penalized objective is

$$\Lambda(X\beta) - \beta^T(X^TY) + \mathcal{P}(\beta),$$

and the regularized estimator is

$$\hat{\beta}_{\mathcal{P}} = \underset{\beta}{\operatorname{argmin}} \Lambda(X\beta) - \beta^T(X^TY) + \mathcal{P}(\beta).$$

One class of penalties are the *ridge penalties*

$$\mathcal{P}(\beta) = \frac{\lambda}{2} \|\beta\|_2^2,$$

$$\mathcal{P}(\beta) = \frac{1}{2} \sum_{j=1}^p \lambda_j \beta_j^2,$$

$$\mathcal{P}(\beta) = \frac{1}{2} \beta^T Q \beta,$$

where  $\lambda, \lambda_j > 0$  and  $Q$  is symmetric and positive definite. The objective is the function

$$\beta \mapsto \Lambda(X\beta) - \beta^T(X^TY) + \frac{1}{2} \beta^T Q \beta.$$

To minimize this objective function we can do Newton–Raphson. Some calculus gives that the iterates are given by

$$\hat{\beta}^{(t+1)} = \hat{\beta}^{(t)} - (X^T W^{(t)} X + Q)^{-1} (X^T W^{(t)} g'(\hat{\mu}^{(t)}) (\hat{\mu}^{(t)} - Y) + Q \hat{\beta}^{(t)}).$$

There are other iterative fitting methods.

## 1.2 Why regularize

Suppose we are fit a model regularized with standard ridge regression so that  $\mathcal{P}(\beta) = \frac{\lambda}{2} \|\beta\|_2^2$ . Consider the simple Gaussian case with unit variance. This means that  $Y \sim N(X\beta, I_n)$ . The bias of the ridge estimator is

$$\text{Bias}(\hat{\beta}_\lambda) = \left\| \mathbb{E}[\hat{\beta}_\lambda] - \beta \right\|_2^2 = \lambda^2 \sum_{j=1}^p \frac{\alpha_j^2}{(d_j^2 + \lambda)^2}.$$

And the variance satisfies

$$\text{tr}(\text{Var}(\hat{\beta}_\lambda)) = \sum_{j=1}^p \frac{d_j^2}{(d_j^2 + \lambda)^2} \leq \sum_{j=1}^p \frac{1}{d_j^2}.$$

The values  $d_j$  are the singular values of  $X$  and  $\alpha_j = \beta^T v_j$  where  $v_j$  is the  $j^{\text{th}}$  singular vector of  $X = UDV^T$ . Thus, when compared to the OLS estimator, the ridge estimator has higher bias but lower variance. Combining these gives

$$\mathbb{E}[\|\hat{\beta}_\lambda - \beta\|_2^2] = \lambda^2 \sum_{j=1}^p \frac{\alpha_j^2}{(\alpha_j^2 + \lambda)^2} + \sum_{j=1}^p \frac{d_j^2}{(d_j^2 + \lambda)^2}.$$

By differentiating the above one can find  $\lambda^*$  which is the value of  $\lambda$  that minimizes the expected square error. It turns out that  $\lambda^* > 0$  and the expected square error is always decreasing at  $\lambda = 0$ . This means that doing a small amount of ridge regularization will decrease the expected square error. Unfortunately finding the optimal  $\lambda$  depends on  $\alpha_j$ , and we do not know  $\alpha_j$ . Thus, in practice, a value of  $\lambda$  is chosen based on cross validation.

## 2 LASSO regularization

### 2.1 The LASSO in one dimension

Consider the penalty

$$\mathcal{P}(\beta) = \lambda \|\beta\|_1 = \lambda \sum_{j=1}^p |\beta_j|.$$

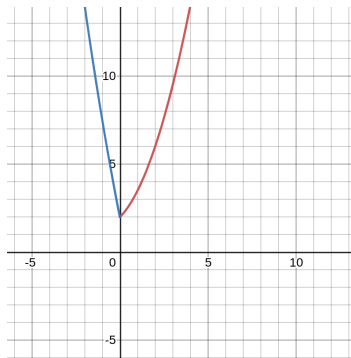
Let's see what this penalty does in a one dimensional problem. Consider

$$\hat{\beta} = \underset{\beta}{\text{argmin}} \frac{1}{2} (Z - \beta)^2 + \lambda |\beta|.$$

If  $\lambda = 3$ ,  $Z = 2$ , then the objective becomes

$$\begin{aligned} \frac{1}{2} (2 - \beta)^2 + 3|\beta| &= \frac{1}{2} \beta^2 - 2\beta + 2 + 3|\beta| \\ &= \begin{cases} \frac{1}{2} \beta^2 + \beta + 2 & \text{if } \beta \geq 0, \\ \frac{1}{2} \beta^2 - 5\beta + 2 & \text{if } \beta < 0. \end{cases} \end{aligned}$$

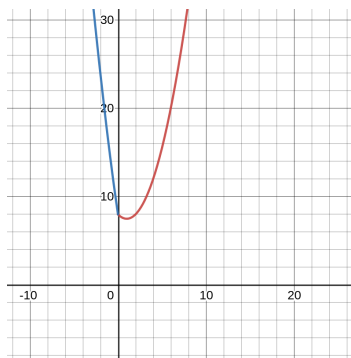
The objective looks like this



Thus, the minimum occurs at  $\lambda = 0$ . If  $\lambda = 3$  and  $Z = 4$ , then the objective is

$$\begin{aligned} \frac{1}{2}(4 - \beta)^2 + 3|\beta| &= \frac{1}{2}\beta^2 - 4\beta + 8 + 3|\beta| \\ &= \begin{cases} \frac{1}{2}\beta^2 - \beta + 8 & \text{if } \beta \geq 0, \\ \frac{1}{2}\beta^2 - 7\beta + 8 & \text{if } \beta < 0. \end{cases} \end{aligned}$$

So the objective looks like this



The minimum occurs at  $\beta = 1$ . In general, consider the objective

$$g_z(\beta) = \frac{1}{2}(\beta - z)^2 + \lambda|\beta|.$$

This objective function is differentiable at all points other than  $\beta = 0$  and for  $\beta \neq 0$ ,

$$\frac{d}{d\beta}g_z(\beta) = \beta - z + \lambda \operatorname{sign}(\beta),$$

where

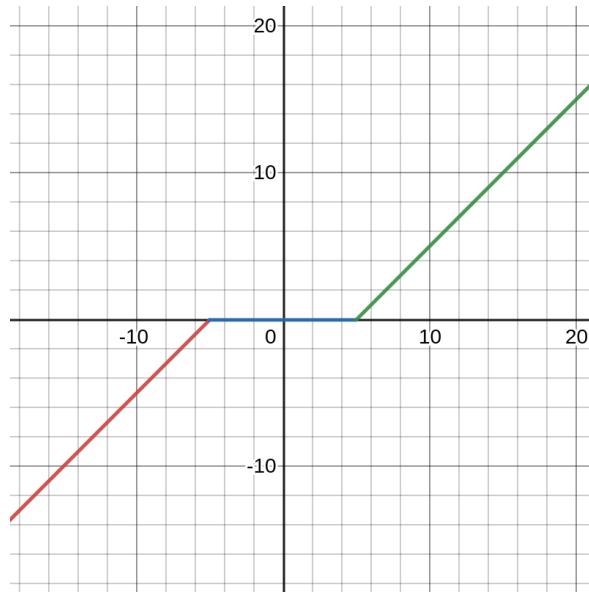
$$\operatorname{sign}(\beta) = \begin{cases} 1 & \text{if } \beta > 0, \\ -1 & \text{if } \beta < 0. \end{cases}$$

Solving  $\frac{d}{d\beta}g_z(\beta) = 0$  implies that  $\beta = z + \lambda \operatorname{sign}(\beta)$ . This equation has a solution if and only if  $|z| \geq \lambda$ . When  $z \geq \lambda$ , the solution is  $\hat{\beta}_\lambda(z) = z - \lambda$  and when  $z \leq -\lambda$ , the solution is  $\hat{\beta}_\lambda(z) = z + \lambda$ . When

the equation  $\frac{d}{d\beta}g_z(\beta) = 0$  has no solutions, the minimizer must occur at 0. Therefore, we have

$$\begin{aligned}\widehat{\beta}_\lambda(z) &= \operatorname{argmin}_{\beta} \frac{1}{2}(z - \beta)^2 + \lambda|\beta| \\ &= \begin{cases} z + \lambda & \text{if } z \leq -\lambda, \\ 0 & \text{if } -\lambda < z < \lambda, \\ z - \lambda & \text{if } z \geq \lambda. \end{cases} \\ &=: S_\lambda(z).\end{aligned}$$

The function  $S_\lambda(z)$  is called the soft-threshold function. For  $\lambda = 5$ , it looks like this



One way to write the soft threshold function is

$$S_\lambda(z) = \operatorname{sign}(z) \max(|z| - \lambda, 0).$$

Part of this argument generalizes to functions other than  $\frac{1}{2}(z - \beta)^2$ . Fix a smooth and convex function  $f: \mathbb{R} \rightarrow \mathbb{R}$  and consider the penalized problem,

$$\widehat{\beta}_\lambda = \operatorname{argmin}_{\beta} f(\beta) + \lambda|\beta|.$$

Note that for  $\beta \neq 0$ ,

$$\frac{d}{d\beta}(f(\beta) + \lambda|\beta|) = f'(\beta) + \lambda \operatorname{sign}(\beta).$$

We will use this to show that if  $|f'(0)| < \lambda$ , then  $\widehat{\beta}_\lambda = 0$ . Recall that since  $f$  is convex and smooth,  $f'(\beta)$  is an increasing function. Thus, if  $|f'(0)| < \lambda$ , then  $f'(0) < \lambda$  and so  $f'(\beta) < \lambda$  for all  $\beta < 0$ . Thus, if  $\beta < 0$ , then

$$\frac{d}{d\beta}(f(\beta) + \lambda|\beta|) = f'(\beta) - \lambda < 0.$$

By a similar argument, if  $f'(\beta) > -\lambda$ , then for all  $\beta > 0$ ,

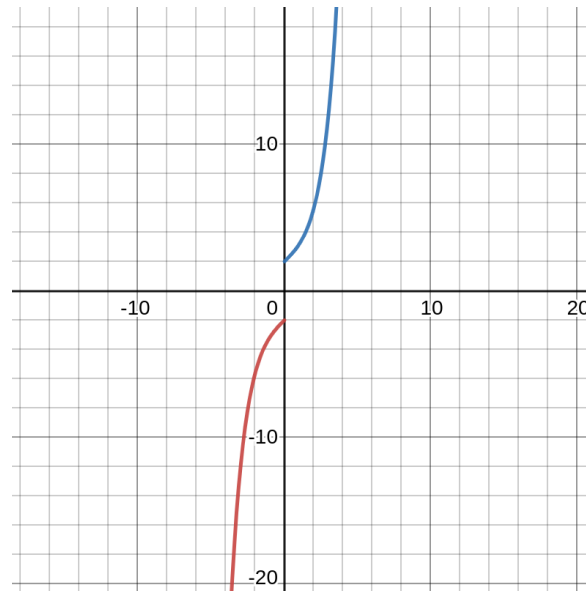
$$\frac{d}{d\beta}(f(\beta) + \lambda|\beta|) = f'(\beta) + \lambda > 0.$$

Thus,  $|f'(\beta)| < \lambda$  implies that the first order equation

$$\frac{d}{d\beta}(f(\beta) + \lambda|\beta|) = 0,$$

has no solutions. And thus we must have  $\hat{\beta}_\lambda = 0$  whenever  $|f'(0)| < \lambda$ . This provides some intuition about what the LASSO penalty sets coefficients equal to zero, and it is also helpful for fitting the LASSO. It provides a quick check for when a coefficient should be zero.

The graph of  $\frac{d}{d\beta}(f(\beta) + \lambda|\beta|)$  when  $|f'(0)| < \lambda$  would look something like this:



## 2.2 Why the LASSO?

There are other methods that induce sparsity, but the LASSO has the following nice properties.

- The minimization problem is convex.
- The LASSO minimization problem can be solved for high dimensions.
- Since the LASSO minimization problem is convex, the KKT conditions describe the solution. This lets us study LASSO solutions in a way that does not depend on the method used to optimize the LASSO.

## 2.3 Fitting the LASSO

We will talk about two iterative methods to fit the LASSO. They are coordinate descent and proximal gradient descent.

### 2.3.1 Coordinate descent

If the penalty  $\mathcal{P}$  was a smooth function, then we could simply use Newton–Raphson to fit the penalized regression. The LASSO penalty is not differentiable at points where one of more  $\beta$  coefficient is zero. But, the penalty is *separable*, meaning that

$$\mathcal{P}(\beta) = \sum_{j=1}^p \mathcal{P}_j(\beta_j).$$

For the LASSO we have,

$$\lambda \|\beta\|_1 = \sum_{j=1}^p \lambda |\beta_j|.$$

When the penalty is separable, we can solve penalized regression by iteratively solving a low dimensional problem. More concretely, consider the following procedure

1. Start at some  $\hat{\beta}^{(0)}$ .
2. At time step  $t$ , choose an index  $j$  and define the univariate objective function,

$$\beta_j \mapsto f^{(t)}(\beta_j) + \lambda |\beta_j| = \Lambda(X_{-j} \hat{\beta}_{-j}^{(t)} + X_j \beta_j) - \beta_j (X_j^T) + \lambda |\beta_j|.$$

That is,  $f^{(t)}(\beta_j)$  is the part of the likelihood that depends on  $\beta_j$  when the other coordinates are fixed to equal  $\hat{\beta}_{-j}^{(t)}$ .

3. Define  $\hat{\beta}_j^{(t+1)} = \operatorname{argmin}_{\beta_j} (f^{(t)}(\beta_j) + \lambda |\beta_j|)$ .
4. Return to step 2 and repeat until convergence.

Some comments

- The univariate problem

$$\hat{\beta}_j^{(t+1)} = \operatorname{argmin}_{\beta_j} (f^{(t)}(\beta_j) + \lambda |\beta_j|),$$

can be solved quickly. One timing saving method is our observation that if  $\left| \frac{d}{d\beta_j} f^{(t)}(\beta_j) \right| < \lambda$ , then  $\hat{\beta}_j^{(t+1)} = 0$ .

- If the solution  $\hat{\beta}_\lambda$  is sparse, then coordinate gradient descent will converge quickly.
- Coordinate descent can be thought of an optimization method that analogous to the Gibbs sampler.
- Version of coordinate gradient descent can also be used when the penalty  $\mathcal{P}$  is *group separable* meaning that it can be written as a sum of disjoint lower dimensional “groups”.

Consider the case of logistic regression. In this case we have

$$f^{(t)}(\beta_j) = -\beta_j X_j^T Y + \sum_{i=1}^n \log(1 + \exp(X_{i,j} \beta_j + Z_i)),$$

where  $Z_i = X_{i,-j}^T \hat{\beta}_{-j}^{(t)}$ . Thus,

$$\frac{d}{d\beta_j} f^{(t)}(\beta_j) = -X_j^T Y + \sum_{i=1}^n X_{i,j} \frac{\exp(X_{i,j} \beta_j + Z_i)}{1 + \exp(X_{i,j} \beta_j + Z_i)} = -X_j^T \left( Y - \frac{\exp(X[\cdot, j] \beta_j + Z)}{1 + \exp(X[\cdot, j] \beta_j + Z)} \right).$$

This means that if,

$$\left| -X_j^T \left( Y - \frac{\exp(Z)}{1 + \exp(Z)} \right) \right| < \lambda,$$

then  $\hat{\beta}_j^{(t+1)} = 0$ . If this does not hold, then we have to solve

$$X_j^T \left( Y - \frac{\exp(X[\cdot, j] \beta_j + Z)}{1 + \exp(X[\cdot, j] \beta_j + Z)} \right) = \lambda \operatorname{sign}(\beta_j).$$

### 2.3.2 Proximal gradient descent

Proximal gradient descent work by using the easier optimization problem

$$\hat{\beta} = \operatorname{argmin}_{\beta} \frac{L}{2} \|Z - \beta\|_2^2 + \lambda \|\beta\|_1.$$

By an analogous argument to the univariate example from earlier we have

$$\hat{\beta}_i = \max(|Z_i| - \lambda/L, 0) \operatorname{sign}(Z_i) = S_{\lambda/L}(Z_i),$$

where  $S_{\lambda/L}$  is the soft-threshold function from before. The map  $Z \mapsto \hat{\beta}$  is called the *proximal map*. This proximal map can be used to create an iterative method similar to quasi-Newton–Raphson. Consider the optimization problem

$$\hat{\beta} = \operatorname{argmin}_{\beta} \ell(\beta) + \lambda \|\beta\|_1,$$

where  $\ell$  is smooth and convex. Suppose that we have a current guess  $\hat{\beta}^{(t)}$ . Since  $\ell(\beta)$  is smooth and convex, we know that there exists a sufficiently large  $L$  so that

$$\ell(\beta) \leq \ell(\hat{\beta}^{(t)}) + \nabla \ell(\hat{\beta}^{(t)})^T (\beta - \hat{\beta}^{(t)}) + \frac{L}{2} \|\beta - \hat{\beta}^{(t)}\|_2^2.$$

And thus,

$$\ell(\beta) + \lambda \|\beta\|_1 \leq \ell(\hat{\beta}^{(t)}) + \nabla \ell(\hat{\beta}^{(t)})^T (\beta - \hat{\beta}^{(t)}) + \frac{L}{2} \|\beta - \hat{\beta}^{(t)}\|_2^2 + \lambda \|\beta\|_1 =: Q^L(\beta; \hat{\beta}^{(t)}).$$

If we define,

$$\hat{\beta}^{(t+1)} = \operatorname{argmin}_{\beta} Q^L(\beta; \hat{\beta}^{(t)}),$$

then

$$\ell(\hat{\beta}^{(t+1)}) \leq Q^L(\hat{\beta}^{(t+1)}; \hat{\beta}^{(t)}) \leq Q^L(\hat{\beta}^{(t)}; \hat{\beta}^{(t)}) = \ell(\beta) + \lambda \|\hat{\beta}^{(t)}\|_1.$$

So the iterative method is always decreasing the objective function, implying that we will get convergence. Now note that,

$$\begin{aligned} \hat{\beta}^{(t+1)} &= \operatorname{argmin}_{\beta} Q^L(\beta; \hat{\beta}^{(t)}) \\ &= \operatorname{argmin}_{\beta} \ell(\hat{\beta}^{(t)}) + \nabla \ell(\hat{\beta}^{(t)})^T (\beta - \hat{\beta}^{(t)}) + \frac{L}{2} \|\beta - \hat{\beta}^{(t)}\|_2^2 + \lambda \|\beta\|_1 \\ &= \operatorname{argmin}_{\beta} \nabla \beta^T \ell(\hat{\beta}^{(t)}) + \frac{L}{2} \beta^T \beta - L \beta^T \hat{\beta}^{(t)} + \frac{L}{2} (\hat{\beta}^{(t)})^T \hat{\beta}^{(t)} + \lambda \|\beta\|_1 \\ &= \operatorname{argmin}_{\beta} \frac{L}{2} \left( \beta^T \beta - 2\beta^T (\hat{\beta}^{(t)} - L^{-1} \nabla \ell(\hat{\beta}^{(t)})) + (\hat{\beta}^{(t)})^T \hat{\beta}^{(t)} \right) + \lambda \|\beta\|_1 \\ &= \operatorname{argmin}_{\beta} \frac{L}{2} \left( \beta^T \beta - 2\beta^T (\hat{\beta}^{(t)} - L^{-1} \nabla \ell(\hat{\beta}^{(t)})) + (\hat{\beta}^{(t)})^T \hat{\beta}^{(t)} \right) + \lambda \|\beta\|_1 \\ &= \operatorname{argmin}_{\beta} \frac{L}{2} \left\| \beta - \left( \hat{\beta}^{(t)} - L^{-1} \nabla \ell(\hat{\beta}^{(t)}) \right) \right\|_2^2 + \lambda \|\beta\|_1 \\ &= S_{\lambda/L} \left( \hat{\beta}^{(t)} - L^{-1} \nabla \ell(\hat{\beta}^{(t)}) \right), \end{aligned}$$

where  $S_{\lambda/L}$  is the proximal map from above. The constant  $L$  can be chosen in an adaptive way where  $L$  is increased whenever the proximal step doesn't actually decrease the objective function.