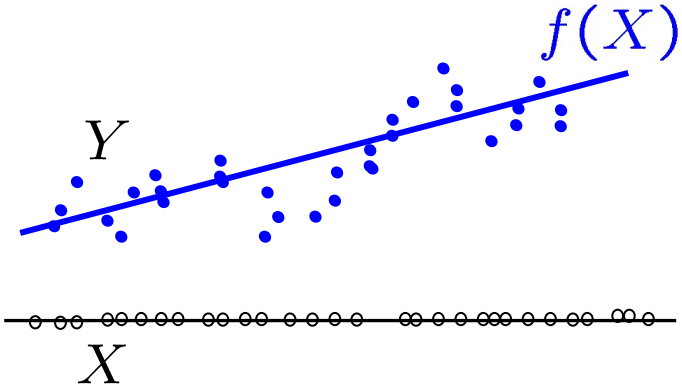
Regularized Linear Regression

Aarti Singh

Machine Learning 10-315
Sept 22, 2021

Linear Regression (Matrix-vector form)

$$\widehat{f}_n^L = \arg\min_{f \in \mathcal{F}_L} \frac{1}{n} \sum_{i=1}^{n} (f(X_i) - Y_i)^2$$



$$\widehat{\beta} = \arg\min_{\beta} \frac{1}{n} \sum_{i=1}^{n} (X_i\beta - Y_i)^2$$

$$= \arg\min_{\beta} \frac{1}{n} (\mathbf{A}\beta - \mathbf{Y})^T (\mathbf{A}\beta - \mathbf{Y})$$

$$\widehat{f}_n^L(X) = X\widehat{\beta}$$

$$\mathbf{A} = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix} = \begin{bmatrix} X_1^{(1)} & \dots & X_1^{(p)} \\ \vdots & \ddots & \vdots \\ X_n^{(1)} & \dots & X_n^{(p)} \end{bmatrix} \quad \mathbf{Y} = \begin{bmatrix} \mathbf{Y}_1 \\ \vdots \\ \mathbf{Y}_n \end{bmatrix}$$

Least Square solution satisfies Normal Equations

$$\left.\frac{\partial J(\beta)}{\partial \beta}\right|_{\widehat{\beta}} = 0 \qquad \text{gives} \qquad (\mathbf{A}^T\mathbf{A})\widehat{\beta} = \mathbf{A}^T\mathbf{Y}$$

p x p p x1 p x1

If $(\mathbf{A}^T\mathbf{A})$ is invertible,

1) If dimension p not too large, analytical solution:

$$\widehat{\beta} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{Y} \qquad \widehat{f}_n^L(X) = X\widehat{\beta}$$

2) If dimension p is large, computing inverse is expensive $O(p^3)$
Gradient descent since objective is convex ($\mathbf{A}^T\mathbf{A} \succeq 0$)

$$\begin{aligned}\beta^{t+1} &= \beta^t - \frac{\alpha}{2}\left.\frac{\partial J(\beta)}{\partial \beta}\right|_t \\ &= \beta^t - \alpha\ \mathbf{A}^T(\mathbf{A}\beta^t - Y)\end{aligned}$$

Linear regression solution satisfies Normal Equations

$$(\mathbf{A}^T\mathbf{A})\hat{\beta} = \mathbf{A}^T\mathbf{Y}$$

p x p p x 1 p x 1

When is $(\mathbf{A}^T\mathbf{A})$ invertible ?

Recall: Full rank matrices are invertible. What is rank of $(\mathbf{A}^T\mathbf{A})$?

If $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^\top$, then

S - r x r

normal equations $(\mathbf{S}\mathbf{V}^\top)\hat{\beta} = (\mathbf{U}^\top\mathbf{Y})$

r x p p x 1 r x 1

r equations in p unknowns.

Under-determined if r < p, hence no unique solution.

Regularized Least Squares

What if $(\mathbf{A}^T\mathbf{A})$ is not invertible ?

$r \leq \min(n,p)$

r equations , p unknowns – underdetermined system of linear equations
many feasible solutions

Need to constrain solution further

e.g. bias solution to "small" values of β (small changes in input don't translate to large changes in output)

$$\hat{\beta}_{\text{MAP}} = \arg\min_{\beta} \sum_{i=1}^{n} (Y_i - X_i\beta)^2 + \lambda\|\beta\|_2^2$$

Ridge Regression
(l2 penalty)

$$= \arg\min_{\beta} \; (\mathbf{A}\beta - \mathbf{Y})^T(\mathbf{A}\beta - \mathbf{Y}) + \lambda\|\beta\|_2^2 \qquad \lambda \geq 0$$

$$\hat{\beta}_{\text{MAP}} = (\mathbf{A}^\top\mathbf{A} + \lambda\mathbf{I})^{-1}\mathbf{A}^\top\mathbf{Y}$$

Regularized Least Squares

$\frac{\partial \beta^T M \beta}{\partial \beta} = (M + M^T)\beta$

$\beta^T A^T A \beta$

$$\hat{\beta}_{\text{MAP}} = \arg\min_{\beta} \sum_{i=1}^{n} (Y_i - X_i\beta)^2 + \lambda \|\beta\|_2^2$$

Ridge Regression (l2 penalty)

$$= \arg\min_{\beta} \frac{1}{n} (\mathbf{A}\beta - \mathbf{Y})^T (\mathbf{A}\beta - \mathbf{Y}) + \lambda \|\beta\|_2^2$$

J

$\lambda \geq 0$

$\frac{\partial J(\beta)}{\partial \beta} = 2A^TA\beta - 2A^TY + \lambda \frac{\partial \|\beta\|^2}{\partial \beta}$

$\|\beta\|_2^2 = \beta^T\beta = \sum_j \beta^{(j)2}$

$= 2\lambda\beta$

$= 2(A^TA + \lambda I)\beta - 2A^TY$

$= 0 \Rightarrow \hat{\beta} = (A^TA + \lambda I)^{-1} A^TY$

$\text{eval}(A^TA + \lambda I) = \text{eval}(A^TA) + \lambda$

$p \times p$

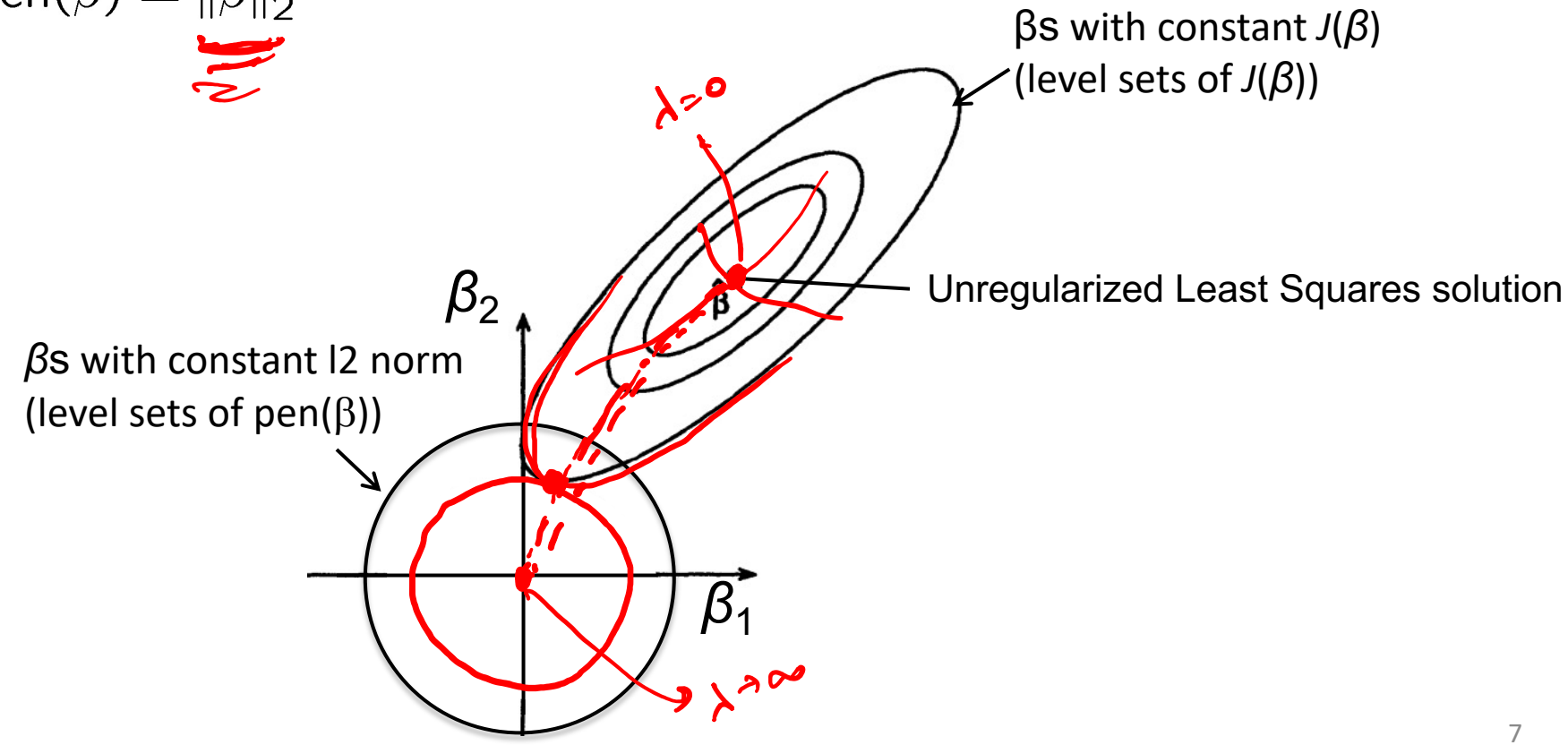
Is $(\mathbf{A}^\top \mathbf{A} + \lambda \mathbf{I})$ invertible ?

Understanding regularized Least Squares

$$\min_{\beta}(\mathbf{A}\beta - \mathbf{Y})^T(\mathbf{A}\beta - \mathbf{Y}) + \lambda \text{pen}(\beta) = \min_{\beta} J(\beta) + \lambda \text{pen}(\beta)$$

Ridge Regression:

$$\text{pen}(\beta) = \|\beta\|_2^2$$



Regularized Least Squares

What if $(\mathbf{A}^T\mathbf{A})$ is not invertible ?

r equations , p unknowns – underdetermined system of linear equations
many feasible solutions

Need to constrain solution further

e.g. bias solution to "small" values of β (small changes in input don't translate to large changes in output)

$$\widehat{\beta}_{\text{MAP}} = \arg\min_{\beta} \sum_{i=1}^{n} (Y_i - X_i\beta)^2 + \lambda\|\beta\|_2^2$$

Ridge Regression
(l2 penalty)

$$\widehat{\beta}_{\text{MAP}} = \arg\min_{\beta} \sum_{i=1}^{n} (Y_i - X_i\beta)^2 + \lambda\|\beta\|_1$$

$\lambda \geq 0$

Lasso
(l1 penalty)

Many β can be zero – many inputs are irrelevant to prediction in high-dimensional settings (typically intercept term not penalized)

Regularized Least Squares

What if $(\mathbf{A}^T\mathbf{A})$ is not invertible ?

r equations , p unknowns – underdetermined system of linear equations
many feasible solutions

Need to constrain solution further

e.g. bias solution to "small" values of β (small changes in input don't translate to large changes in output)

$$\hat{\beta}_{\text{MAP}} = \arg\min_{\beta} \sum_{i=1}^{n} (Y_i - X_i\beta)^2 + \lambda\|\beta\|_2^2$$

Ridge Regression
(l2 penalty)

$$\hat{\beta}_{\text{MAP}} = \arg\min_{\beta} \sum_{i=1}^{n} (Y_i - X_i\beta)^2 + \lambda\|\beta\|_1$$

$\lambda \geq 0$

Lasso
(l1 penalty)

$= \sum_j |\beta^j|$

No closed form solution, but can optimize using sub-gradient descent (packages available)

Ridge Regression vs Lasso

$$\min_{\beta}(\mathbf{A}\beta - \mathbf{Y})^T(\mathbf{A}\beta - \mathbf{Y}) + \lambda \text{pen}(\beta) = \min_{\beta} J(\beta) + \lambda \text{pen}(\beta)$$

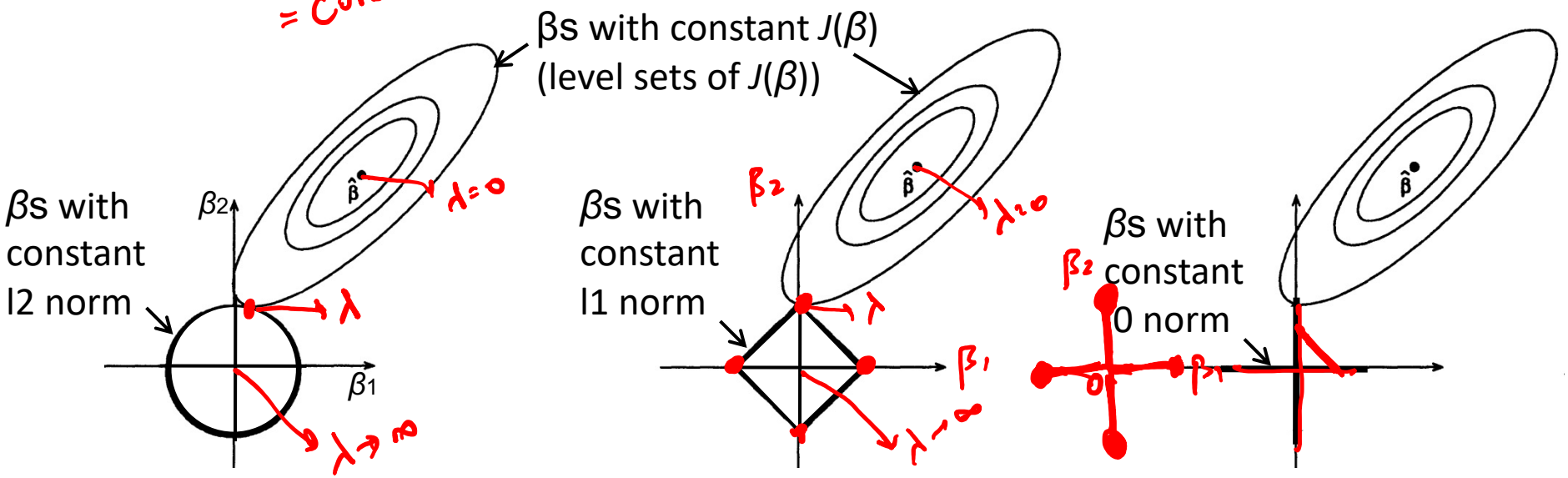
Ridge Regression:

$$\text{pen}(\beta) = \|\beta\|_2^2$$

Lasso:

$$\text{pen}(\beta) = \|\beta\|_1$$

Ideally l0 penalty, but optimization becomes non-convex



Lasso (l1 penalty) results in sparse solutions – vector with more zero coordinates
Good for high-dimensional problems – don't have to store all coordinates, interpretable solution!

Matlab example

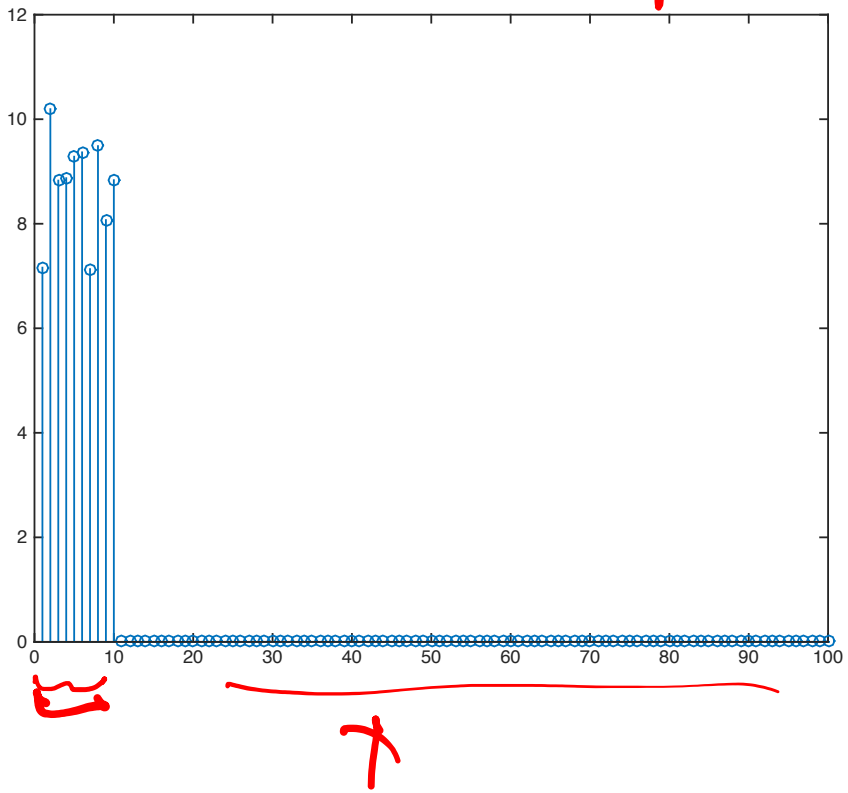
```
clear all
close all

n = 80;     % datapoints
p = 100;    % features
k = 10;     % non-zero features

rng(20);
X = randn(n,p);
weights = zeros(p,1);
weights(1:k) = randn(k,1)+10;
noise = randn(n,1) * 0.5;
Y = X*weights +  noise;

Xtest = randn(n,p);
noise = randn(n,1) * 0.5;
Ytest = Xtest*weights + noise;
```

```
lassoWeights = lasso(X,Y,'Lambda',1,
'Alpha', 1.0);
Ylasso = Xtest*lassoWeights;
norm(Ytest-Ylasso)

ridgeWeights = lasso(X,Y,'Lambda',1,
'Alpha', 0.0001);
Yridge = Xtest*ridgeWeights;
norm(Ytest-Yridge)

stem(lassoWeights)
pause
stem(ridgeWeights)
```

Matlab example

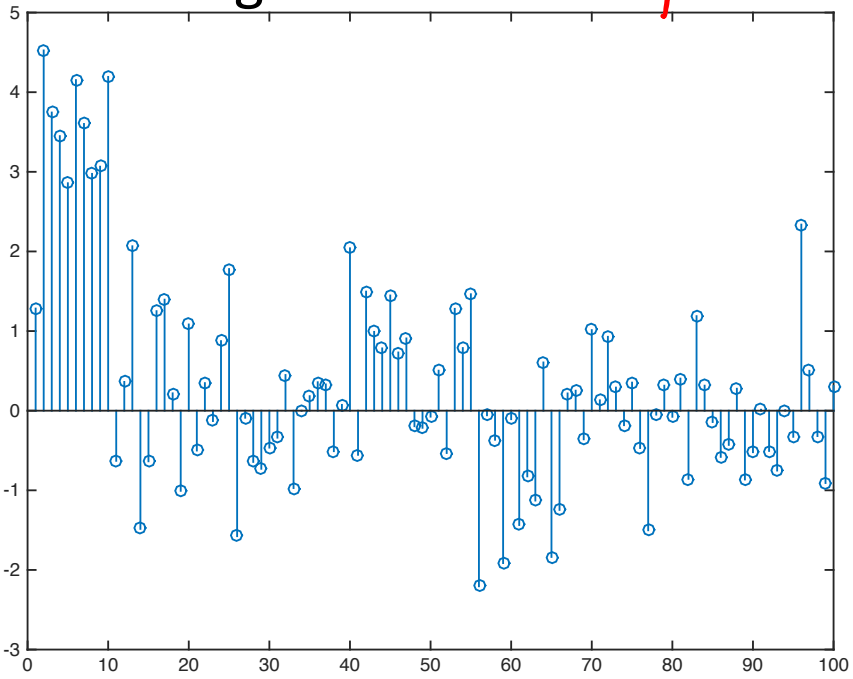
Test MSE = 33.7997

Lasso Coefficients $\hat{\beta}$



Test MSE = 185.9948

Ridge Coefficients $\hat{\beta}$

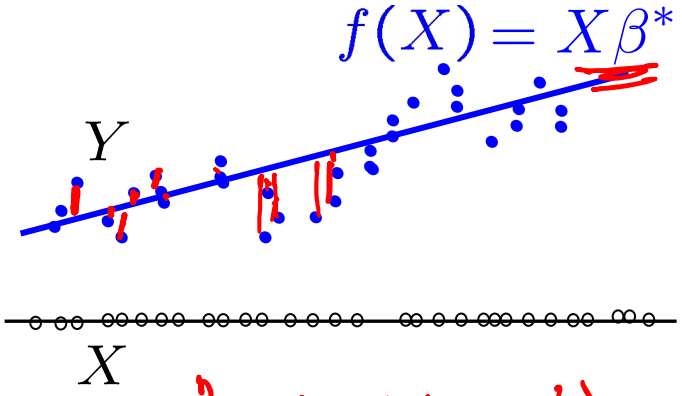


Least Squares and M(C)LE

Intuition: Signal plus (zero-mean) Noise model

$$Y = f^*(X) + \epsilon = X\beta^* + \epsilon$$

$$\epsilon \sim \mathcal{N}(0, \sigma^2 \mathbf{I}) \quad Y \sim \mathcal{N}(X\beta^*, \sigma^2 \mathbf{I})$$



$$\widehat{\beta}_{\mathsf{MLE}} = \arg\max_{\beta} \underbrace{\log p(\{Y_i\}_{i=1}^n | \beta, \sigma^2, \{X_i\}_{i=1}^n)}_{\text{Conditional log likelihood}}$$

$$= \arg\max_{\beta} \sum_{i=1}^n \log p(Y_i | X_i; \beta, \sigma^2)$$

$$= \arg\max_{\beta} \sum_{i=1}^n \log \frac{e^{-(Y_i - X_i\beta^*)^2/2\sigma^2}}{\sqrt{2\pi\sigma^2}}$$

$$= \arg\max_{\beta} -\sum_{i=1}^n \frac{(Y_i - X_i\beta^*)^2}{2\sigma^2}$$

$$= \arg\min_{\beta} \sum_{i=1}^n (Y_i - X_i\beta^*)^2$$

Least Square Estimate is same as Maximum Conditional Likelihood Estimate under a Gaussian noise model !