Linear Regression

Aarti Singh

Co-instructor: Barnabas Poczos

Machine Learning 10-401
Mar 29, 2016

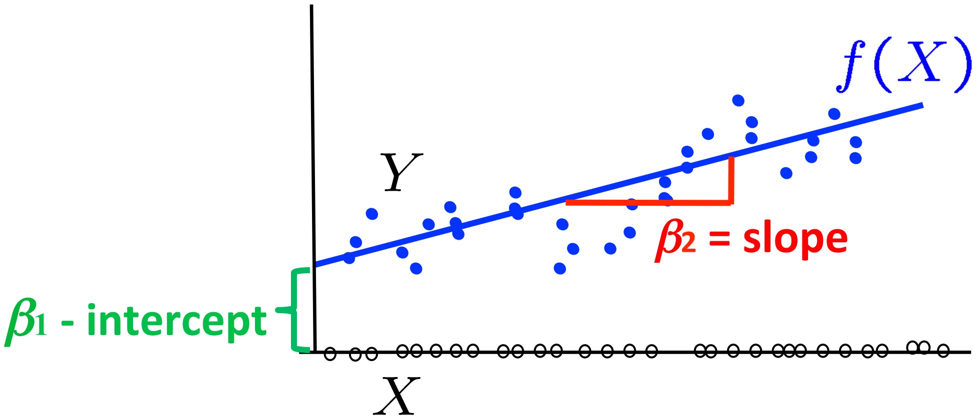
Linear Regression

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

Least Squares Estimator

$\mathcal{F}_L$ - Class of Linear functions

$$f(X_i) = X_i \beta$$



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

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

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

Least Square solution satisfies Normal Equations

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

p x p   p x1   p x1

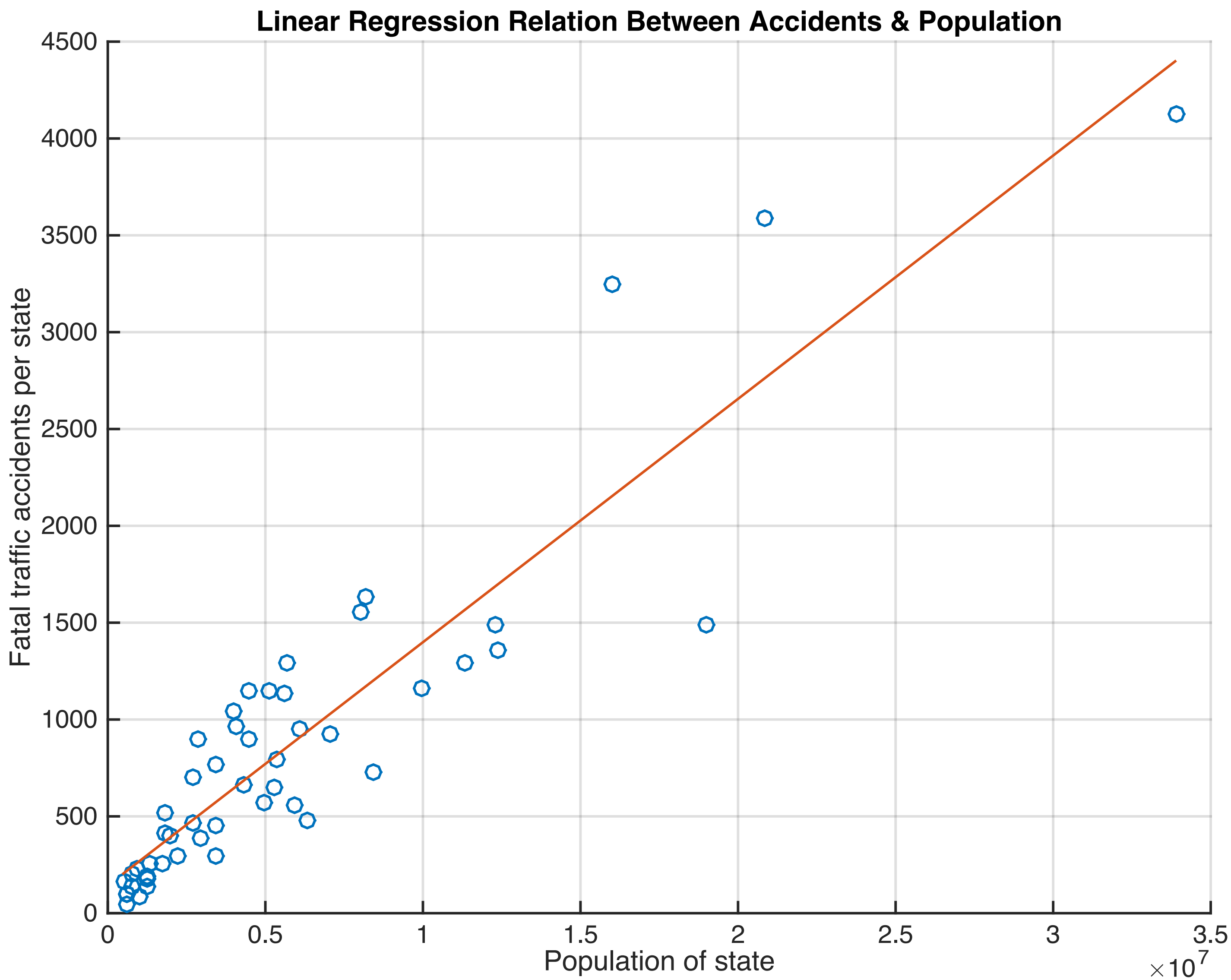
If $(\mathbf{A}^T\mathbf{A})$ is invertible,

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

Matlab example – linear regression

```
load accidents
x = hwydata(:,14);                %Population of states
y = hwydata(:,4);                 %Accidents per state
scatter(x,y)
hold on
X = [ones(length(x),1) x];

b = X\y;
yhat = X*b;
plot(x,yhat)
xlabel('Population of state')
ylabel('Fatal traffic accidents per state')
title('Linear Regression Relation Between Accidents &
Population')
```

Matlab example – linear regression



Least Square solution satisfies Normal Equations

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

p x p   p x1   p x1

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

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

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

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

Rank$(\mathbf{A}^T\mathbf{A})$ = number of non-zero eigenvalues of $(\mathbf{A}^T\mathbf{A})$ = number of non-zero singular values of **A** <= min(n,p) since **A** is n x p

So, rank$(\mathbf{A}^T\mathbf{A})$, r <= min(n,p)   not invertible if r < p (e.g. n < p i.e. high-dimensional setting)

Least Square solution satisfies Normal Equations

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

p x p   p x1   p x1

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

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

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

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

If $\mathbf{A} = \mathbf{USV}^\top$, then normal equations $(\mathbf{SV}^\top)\hat{\beta} = (\mathbf{U}^\top\mathbf{Y})$

S - r x r   r x p   p x 1   r x 1

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

Least Square solution satisfies Normal Equations

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

p x p   p x1   p x1

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

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

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

What if $(\mathbf{A}^T\mathbf{A})$ is not invertible ?
Constrain solution i.e. Regularization (later)

Now: What if $(\mathbf{A}^T\mathbf{A})$ is invertible but expensive (p very large)?

Gradient Descent

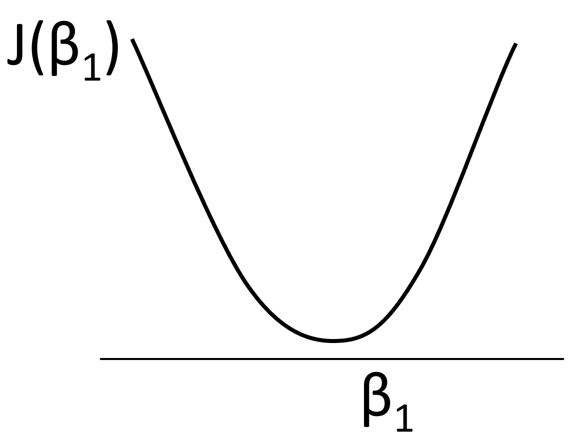
Even when $(\mathbf{A}^T\mathbf{A})$ is invertible, might be computationally expensive if **A** is huge.

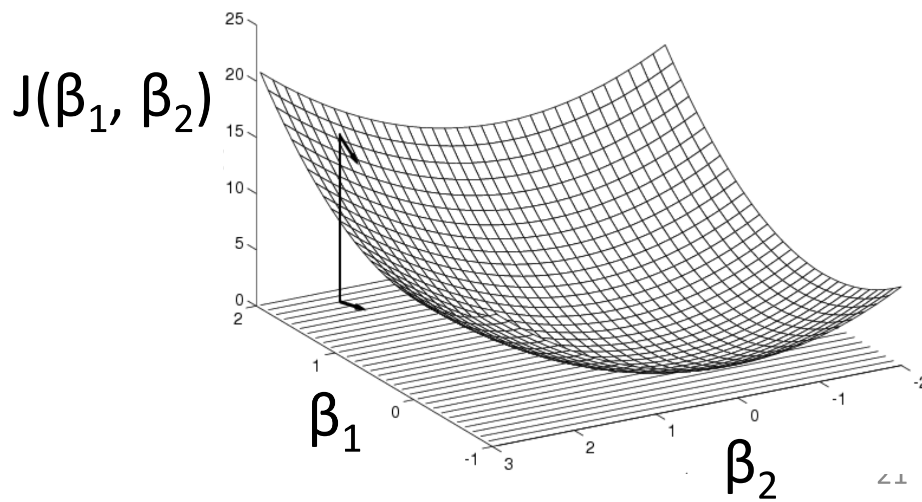
$$\widehat{\beta} = \arg\min_{\beta} \frac{1}{n}(\mathbf{A}\beta - \mathbf{Y})^T(\mathbf{A}\beta - \mathbf{Y}) = \arg\min_{\beta} J(\beta)$$

Treat as optimization problem

<u>Observation:</u> J(β) is convex in β.

How to find the minimizer?





Gradient Descent

Even when $(\mathbf{A}^T\mathbf{A})$ is invertible, might be computationally expensive if **A** is huge.

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

Since J(β) is convex, move along negative of gradient

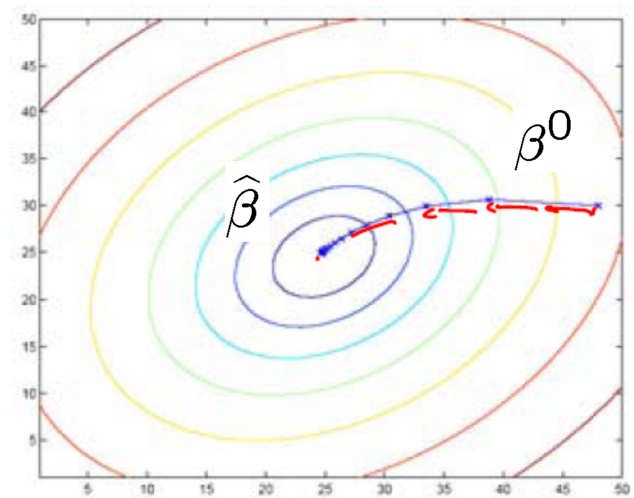
Initialize: β^0

step size

Update:

$$\beta^{t+1} = \beta^t - \frac{\alpha}{2}\frac{\partial J(\beta)}{\partial \beta}\bigg|_t$$

$$= \beta^t - \alpha\ \mathbf{A}^T\underbrace{(\mathbf{A}\beta^t - Y)}_{0 \text{ if } \widehat{\beta} = \beta^t}$$



Stop: when some criterion met e.g. fixed # iterations, or $\frac{\partial J(\beta)}{\partial \beta}\bigg|_{\beta^t} < \varepsilon$.

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)

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

$$\lambda \geq 0$$

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

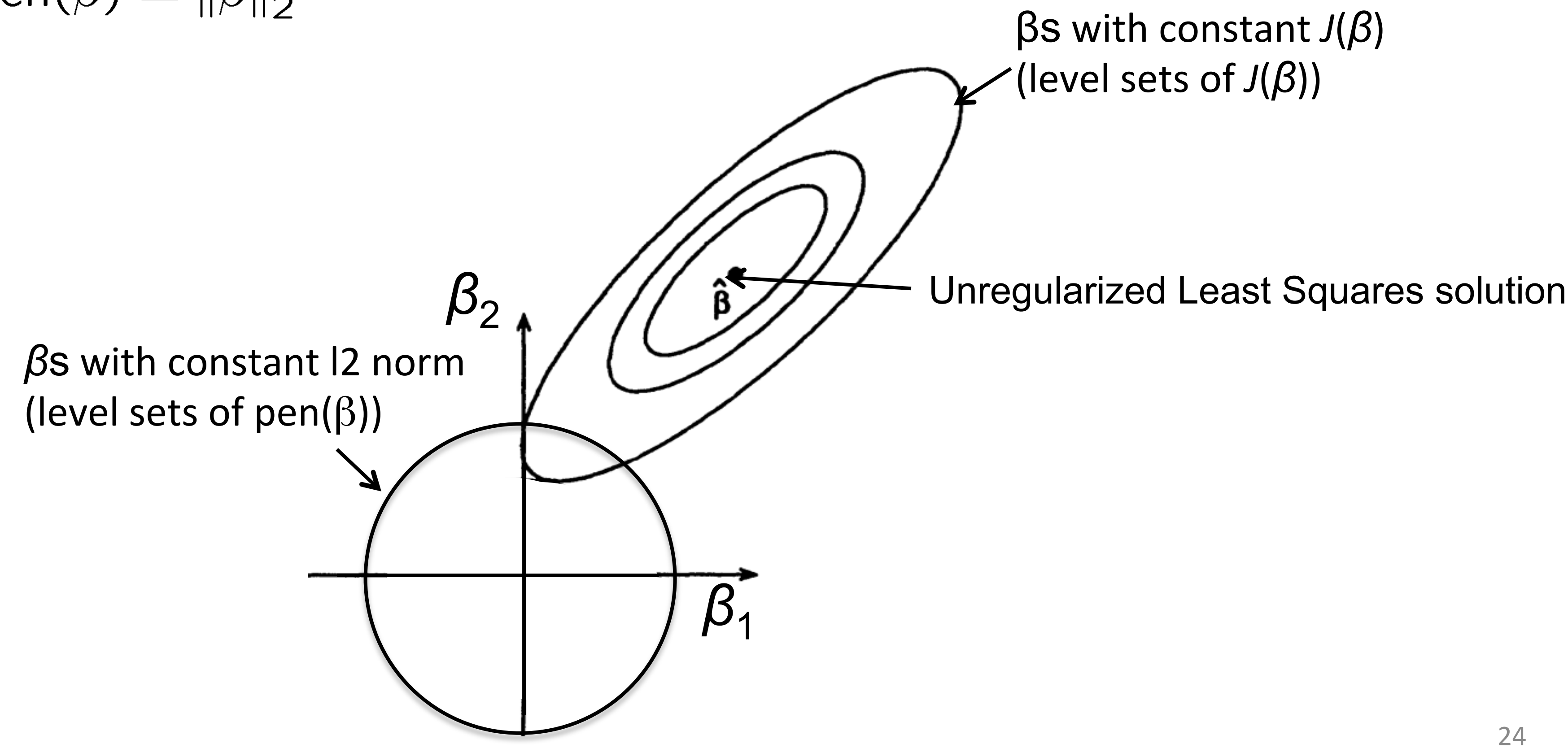
Is $(\mathbf{A}^\top\mathbf{A} + \lambda\mathbf{I})$ invertible ?

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)

$$\lambda \geq 0$$

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

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)

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

Ridge Regression
(l2 penalty)

$$\lambda \geq 0$$

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

Lasso
(l1 penalty)

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

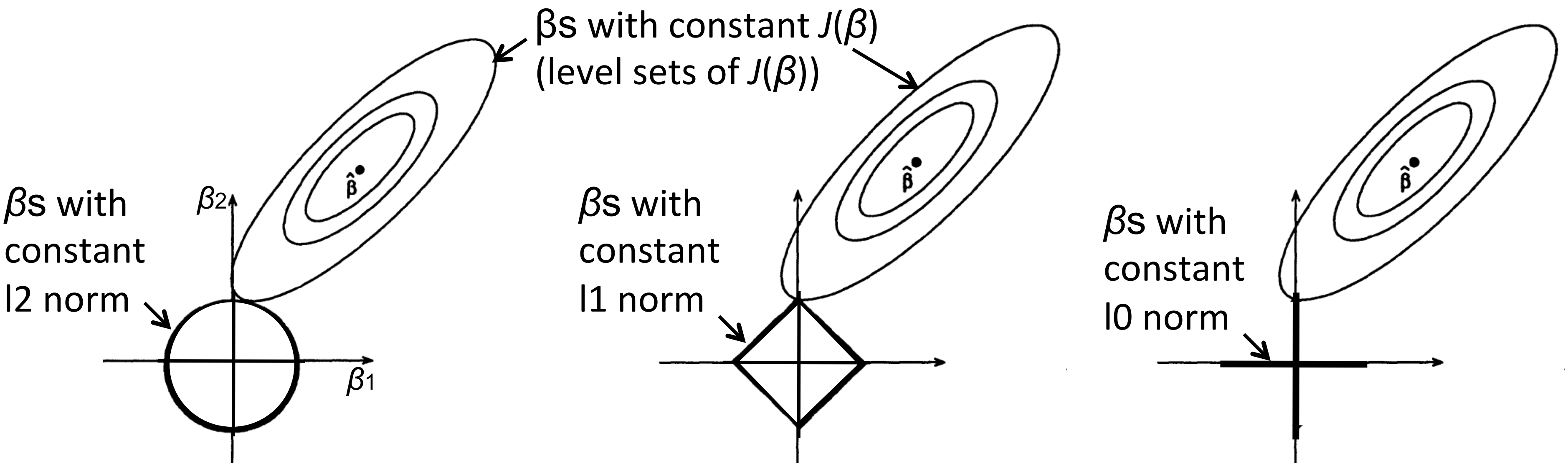
Ridge Regression vs Lasso

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

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!

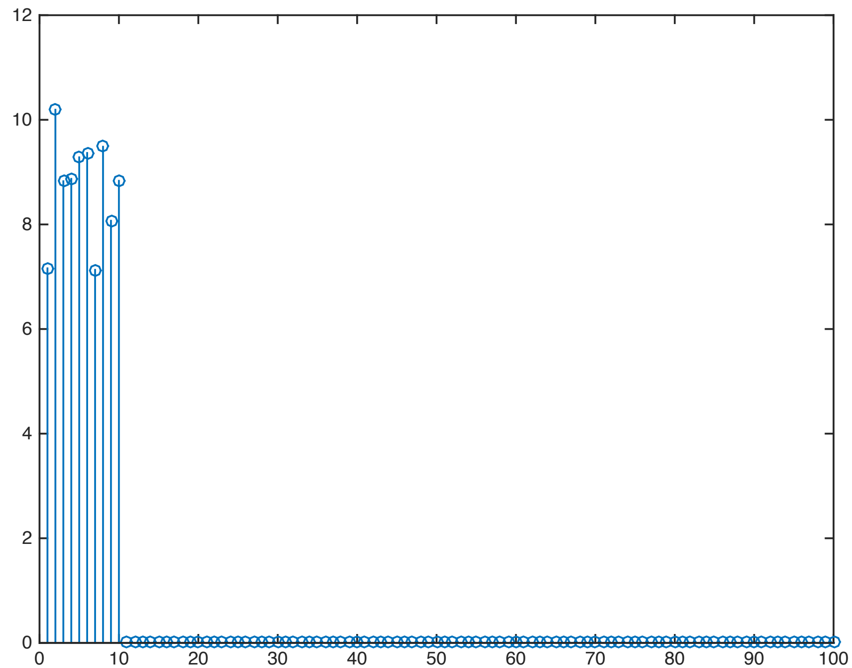
Matlab example

```
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



Test MSE = 185.9948

Ridge Coefficients