Linear Regression

Aarti Singh

Co-instructor: Pradeep Ravikumar

Machine Learning 10-701
Jan 30, 2017

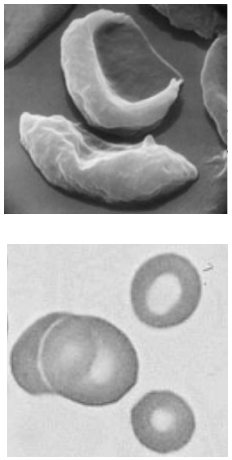
Discrete to Continuous Labels

Classification

X = Document

Sports
Science
News

Y = Topic

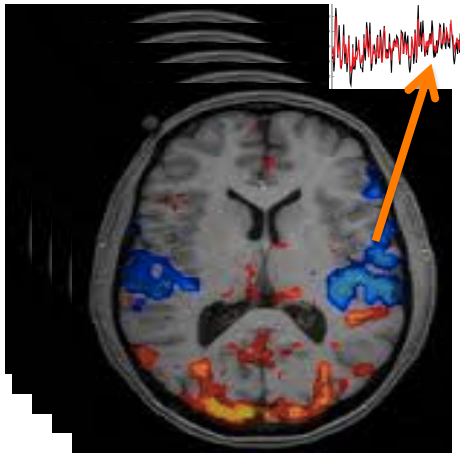


X = Cell Image

Anemic cell
Healthy cell

Y = Diagnosis

Regression

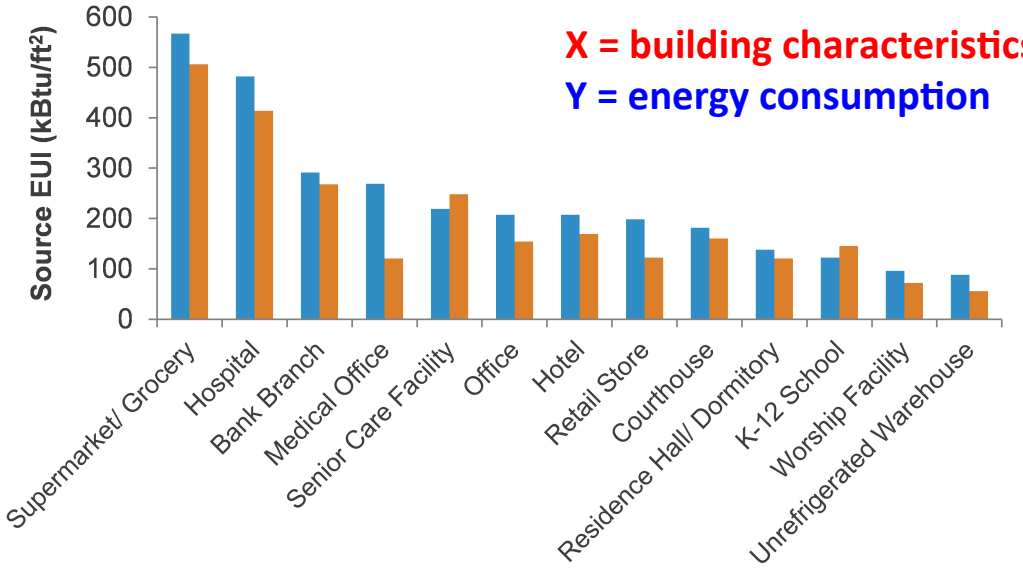


X = Brain Scan

Y = Age of a subject

Regression Tasks

Estimating Energy Usage



X = building characteristics
Y = energy consumption

energystar.gov

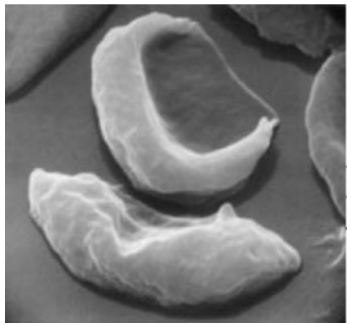
Estimating Contamination

X = new location
Y = sensor reading

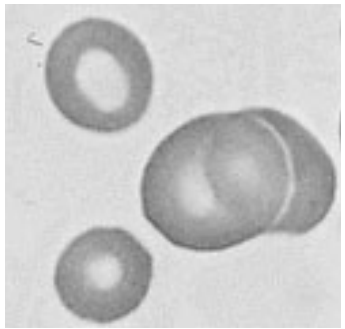
Supervised Learning Prediction Task

Task: Given $X \in \mathcal{X}$, predict $Y \in \mathcal{Y}$.

$\equiv$ Construct **prediction rule** $f : \mathcal{X} \to \mathcal{Y}$



"Anemic cell (0)"

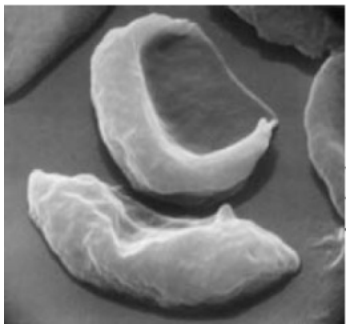


"Healthy cell (1)"

Performance Measures

Performance Measure: Quantifies knowledge gained

$\text{loss}(Y, f(X))$ - Measure of closeness between true label *Y* and prediction *f*(*X*)

X	*Y*	*f*(*X*)	$\text{loss}(Y, f(X))$
	"Anemic cell"	"Anemic cell"	0
		"Healthy cell"	1

$$\text{loss}(Y, f(X)) = 1_{\{f(X) \neq Y\}}$$ **0/1 loss**

Performance Measures

Performance Measure: Quantifies knowledge gained

$\text{loss}(Y, f(X))$ - Measure of closeness between true label *Y* and prediction *f*(*X*)

X	Share price, *Y*	*f*(*X*)	$\text{loss}(Y, f(X))$
Past performance, trade volume etc. as of Sept 8, 2010	"$24.50"	"$24.50"	0
		"$26.00"	1?
		"$26.10"	2?

$$\text{loss}(Y, f(X)) = (f(X) \neq Y)^2$$ **square loss**

Performance Measures

Performance Measure: Quantifies knowledge gained

$\text{loss}(Y, f(X))$ - Measure of closeness between true label *Y* and prediction *f*(*X*)

Don't just want label of one test data (cell image), but any cell image $X \in \mathcal{X}$

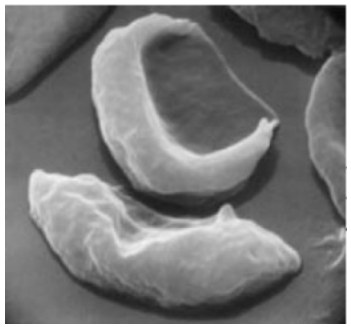
$$(X, Y) \sim P_{XY}$$

Given a cell image drawn randomly from the collection of all cell images, how well does the predictor perform on average?

$$\text{Risk } R(f) \equiv \mathbb{E}_{XY}\left[\text{loss}(Y, f(X))\right]$$

Performance Measures

Performance Measure: Risk $R(f) \equiv \mathbb{E}_{XY}\left[\text{loss}(Y, f(X))\right]$



	loss$(Y, f(X))$	Risk $R(f)$
"Anemic cell"	$1_{\{f(X) \neq Y\}}$	$P(f(X) \neq Y)$
	0/1 loss	**Probability of Error**
Share Price "$ 24.50"	$(f(X) \neq Y)^2$	$\mathbb{E}[(f(X) \neq Y)^2]$
	square loss	**Mean Square Error**

Bayes Optimal Rule

Ideal goal: Construct **prediction rule** $f^* : \mathcal{X} \to \mathcal{Y}$

$$f^* = \arg\min_f \mathbb{E}_{XY}\left[\text{loss}(Y, f(X))\right]$$

Bayes optimal rule

Best possible performance:

Bayes Risk $\quad R(f^*) \leq R(f)$ for all f

BUT… Optimal rule is not computable - depends on unknown P_{XY} !

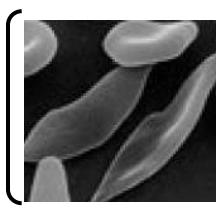
Experience - Training Data

Can't minimize risk since P_{XY} unknown!

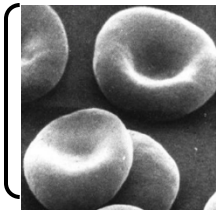
Training data (experience) provides a glimpse of P_{XY}

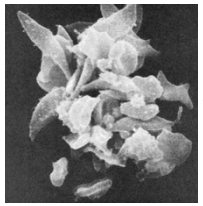
(observed) $\{(X_i, Y_i)\}_{i=1}^{n} \overset{i.i.d.}{\sim} P_{XY}$ **(unknown)**

independent, identically distributed


, Anemic cell

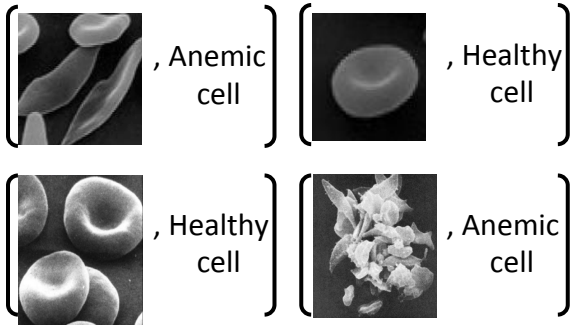

, Healthy cell


, Healthy cell


, Anemic cell

Provided by expert, measuring device, some experiment, …

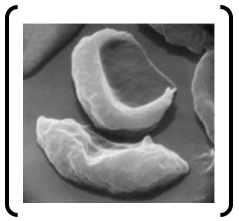
Machine Learning Algorithm



$\widehat{f}_n$

Data $\{(X_i, Y_i)\}_{i=1}^n$

$\widehat{f}_n$ is a mapping from $\mathcal{X} \to \mathcal{Y}$

$\widehat{f}_n$  = "Anemic cell"

Test data X

Model based approach: Use data to learn a model for P_{XY}

Model-free approach: Use data to learn mapping directly

Model-free approach: Empirical Risk Minimization

Optimal predictor: $$f^* = \arg\min_f \mathbb{E}[(f(X) - Y)^2]$$

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

Empirical mean

Law of Large Numbers:

$$\frac{1}{n} \sum_{i=1}^{n} [\text{loss}(Y_i, f(X_i))] \xrightarrow{n \to \infty} \mathbb{E}_{XY} [\text{loss}(Y, f(X))]$$

Restrict class of predictors

Optimal predictor: $$f^* = \arg\min_f \mathbb{E}[(f(X) - Y)^2]$$

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

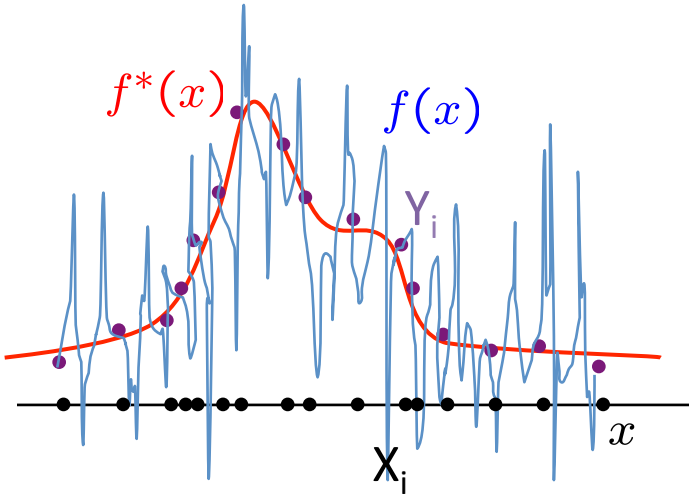
Class of predictors

Why?

Overfitting!

Empiricial loss minimized by any function of the form

$$f(x) = \begin{cases} Y_i, & x = X_i \text{ for } i = 1, \ldots, n \\ \text{any value}, & \text{otherwise} \end{cases}$$



Restrict class of predictors

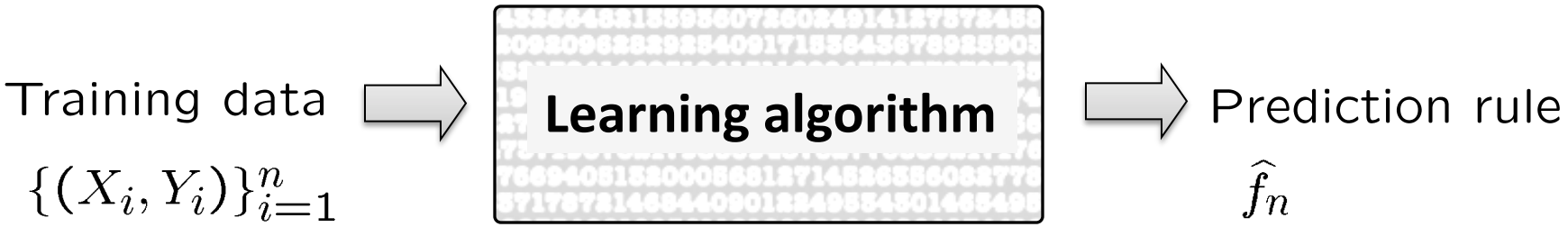
Optimal predictor: $$f^* = \arg\min_f \mathbb{E}[(f(X) - Y)^2]$$

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

Class of predictors

$\mathcal{F}$ -
- Class of Linear functions
- Class of Polynomial functions
- Class of nonlinear functions

Regression algorithms



Linear Regression
Regularized Linear Regression – Ridge regression, Lasso
Polynomial Regression
Kernelized Ridge Regression
Gaussian Process Regression
Kernel regression, Regression Trees, Splines, Wavelet estimators, …

Linear Regression

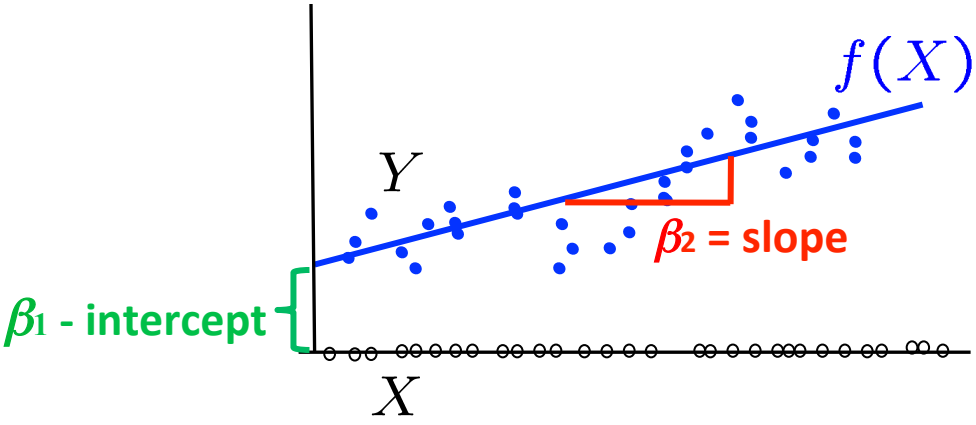
$$\hat{f}_n^L = \arg\min_{f \in \mathcal{F}_L} \frac{1}{n} \sum_{i=1}^{n} (f(X_i) - Y_i)^2$$

Least Squares Estimator

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

Uni-variate case:

$$f(X) = \beta_1 + \beta_2 X$$



Multi-variate case:

$$f(X) = f(X^{(1)}, \ldots, X^{(p)}) = \beta_1 X^{(1)} + \beta_2 X^{(2)} + \cdots + \beta_p X^{(p)}$$

$X^{(1)}$ → 1

$$= X\beta \qquad \text{where} \quad X = [X^{(1)} \ldots X^{(p)}], \quad \beta = [\beta_1 \ldots \beta_p]^T$$

Least Squares Estimator

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

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

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

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

Least Squares Estimator

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

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

$$\left.\frac{\partial J(\beta)}{\partial \beta}\right|_{\widehat{\beta}} = 0$$

Least Square solution satisfies Normal Equations

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

p x p p x1 p x1

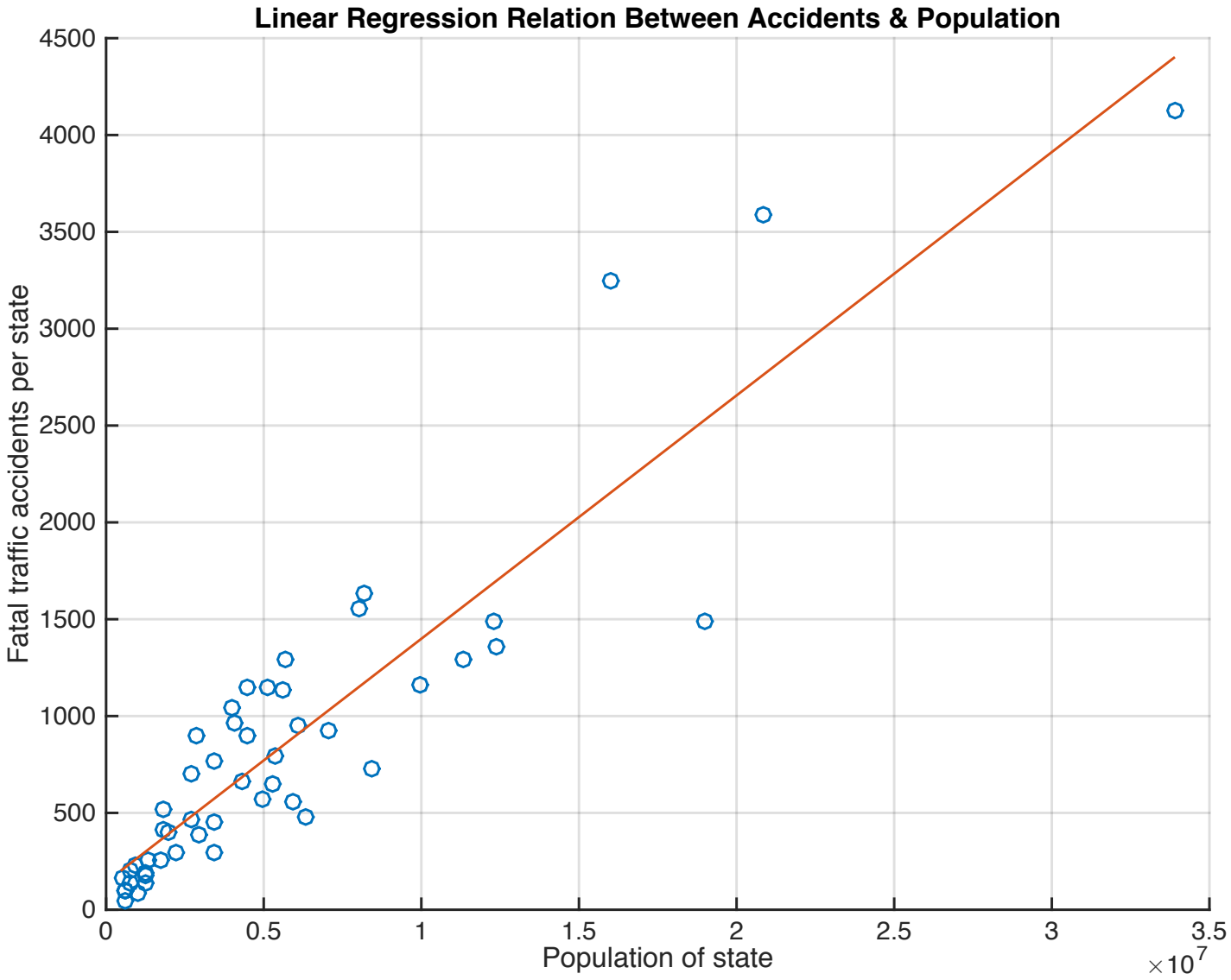
If $(\mathbf{A}^T\mathbf{A})$ is invertible,

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

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

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

p x p p x1 p x1

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

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

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

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

p x p p x1 p x1

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

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

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

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

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

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

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

p x p p x1 p x1

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

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

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)?

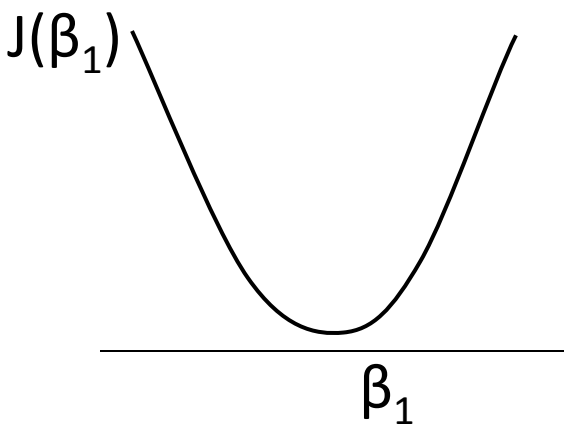
Gradient Descent

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

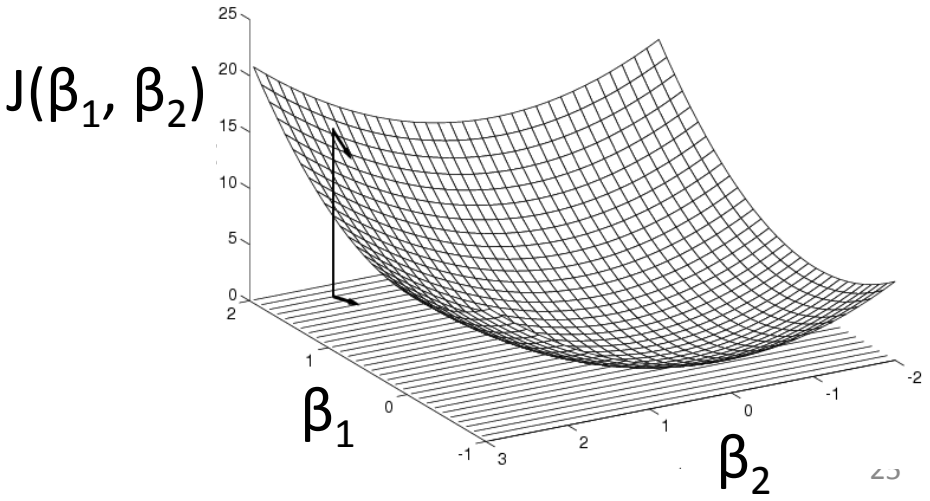
$$\widehat{\beta} = \arg\min_{\beta} \frac{1}{n}(\mathbf{A}\beta - \mathbf{Y})^T(\mathbf{A}\beta - \mathbf{Y}) = \arg\min_{\beta} J(\beta)$$

Treat as optimization problem

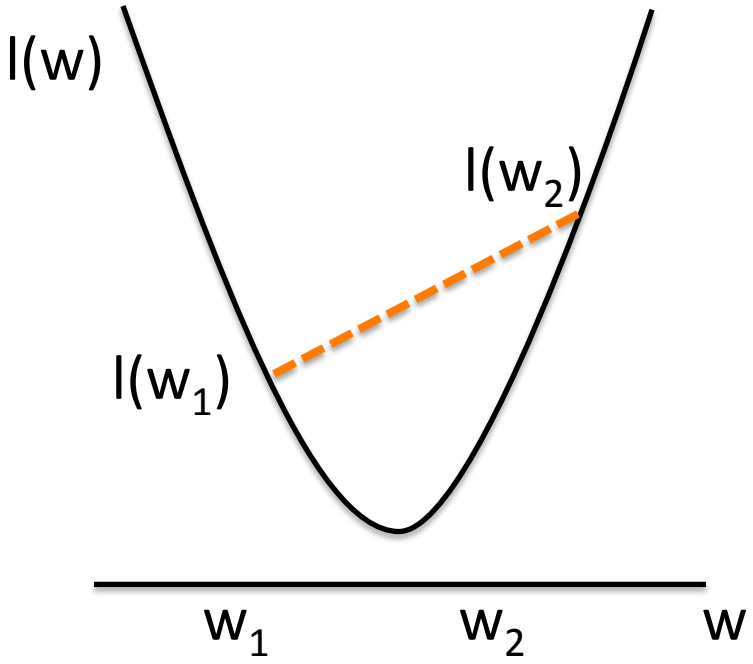
Observation: J(β) is convex in β.



How to find the minimizer?

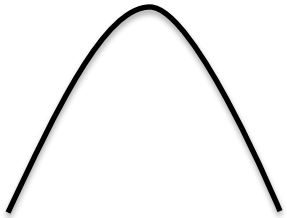


Convex function

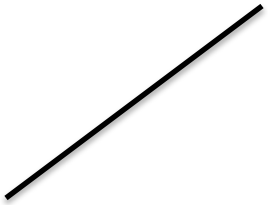


A function l(w) is called **convex** if the line joining two points $l(w_1), l(w_2)$ on the function does not go below the function on the interval $[w_1, w_2]$

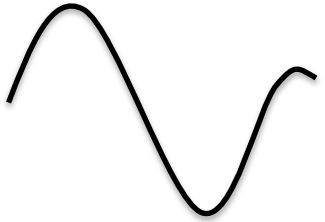
(Strictly) Convex functions have a unique minimum!



Concave



Both Concave & Convex

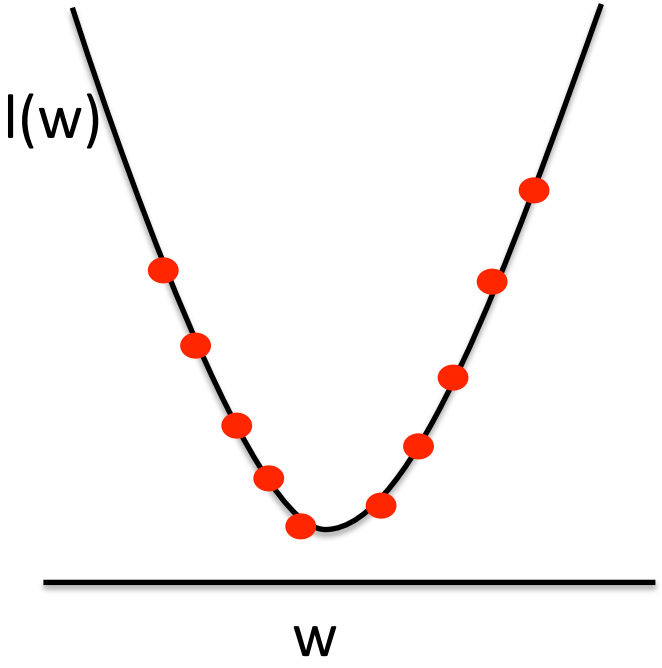


Neither

Optimizing convex functions

- Minimum of a concave function can be reached by

Gradient Descent Algorithm



Initialize: Pick **w** at random

Gradient:

$$\nabla_{\mathbf{w}} l(\mathbf{w}) = [\frac{\partial l(\mathbf{w})}{\partial w_0}, \ldots, \frac{\partial l(\mathbf{w})}{\partial w_d}]'$$

Learning rate, η>0

Update rule:

$$\Delta \mathbf{w} = \eta \nabla_{\mathbf{w}} l(\mathbf{w})$$

$$w_i^{(t+1)} \leftarrow w_i^{(t)} - \eta \left. \frac{\partial l(\mathbf{w})}{\partial w_i} \right|_t$$

Gradient Descent

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

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

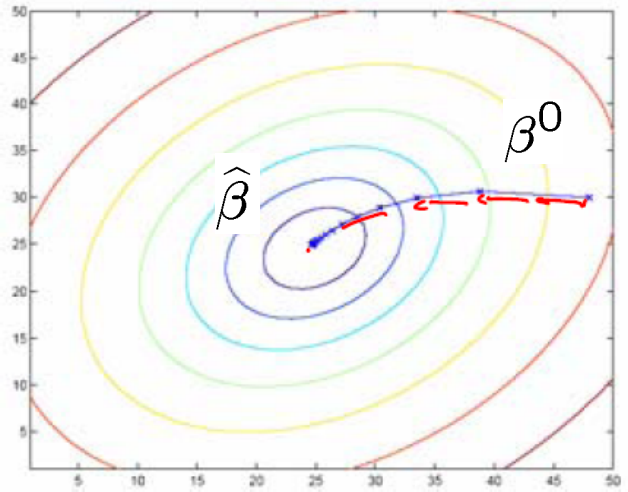
Since J(β) is convex, move along negative of gradient

Initialize: β^0

step size

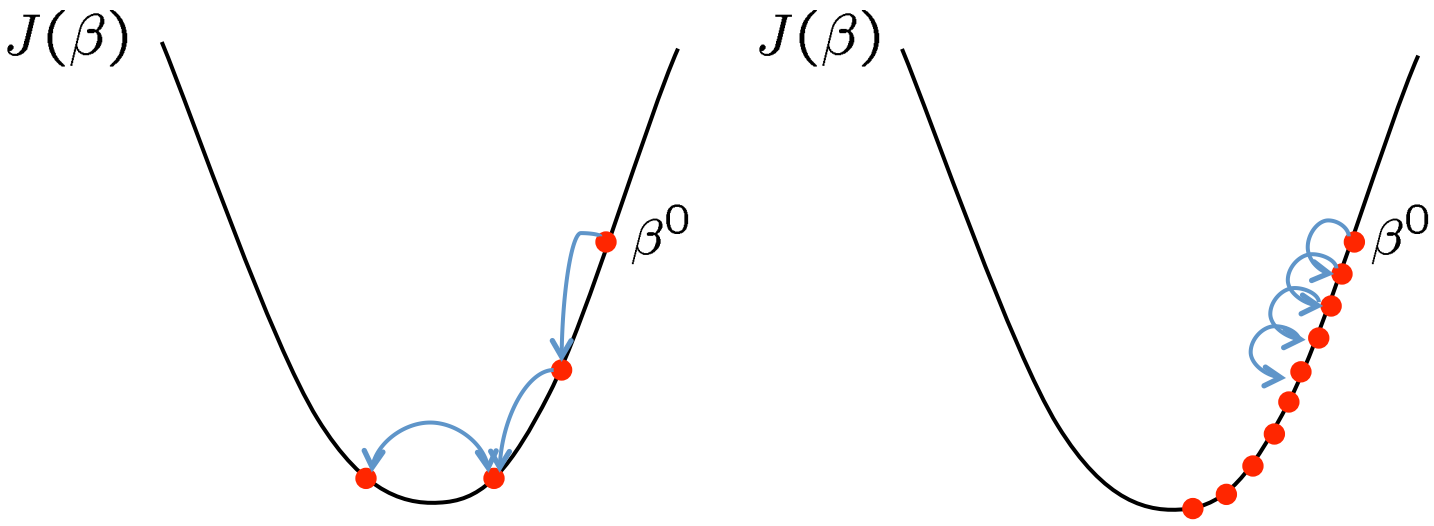
Update: $\beta^{t+1} = \beta^t - \frac{\alpha}{2}\frac{\partial J(\beta)}{\partial \beta}\Big|_t$

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



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

Effect of step-size α



Large α => Fast convergence but larger residual error
Also possible oscillations

Small α => Slow convergence but small residual error

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)

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

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

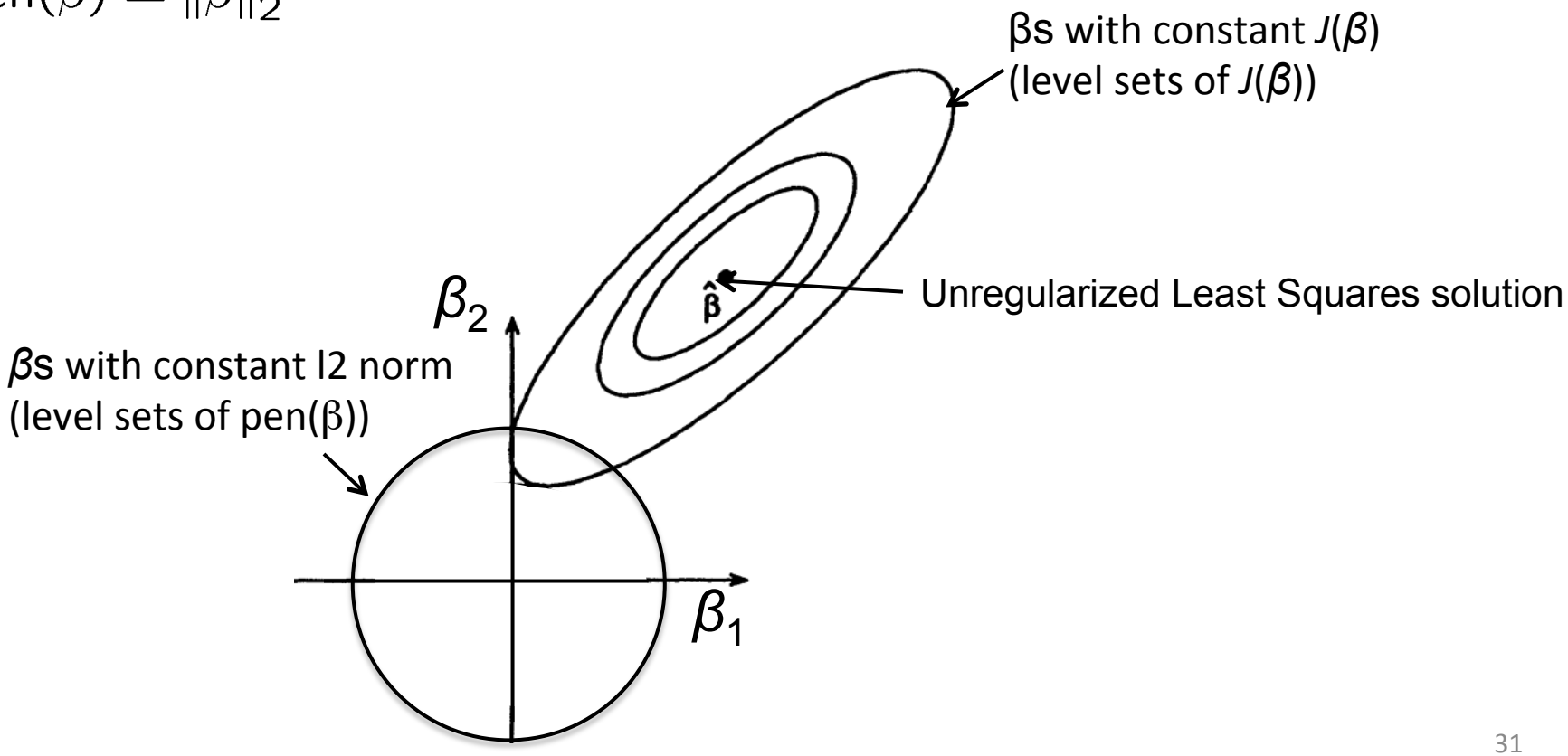
Is $(\mathbf{A}^\top\mathbf{A} + \lambda\mathbf{I})$ invertible ?

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)

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

Ridge Regression
(l2 penalty)

$$\hat{\beta}_{\text{MAP}} = \arg\min_{\beta} \sum_{i=1}^{n} (Y_i - X_i\beta)^2 + \lambda\|\beta\|_1 \qquad \lambda \geq 0$$

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)

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

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)

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

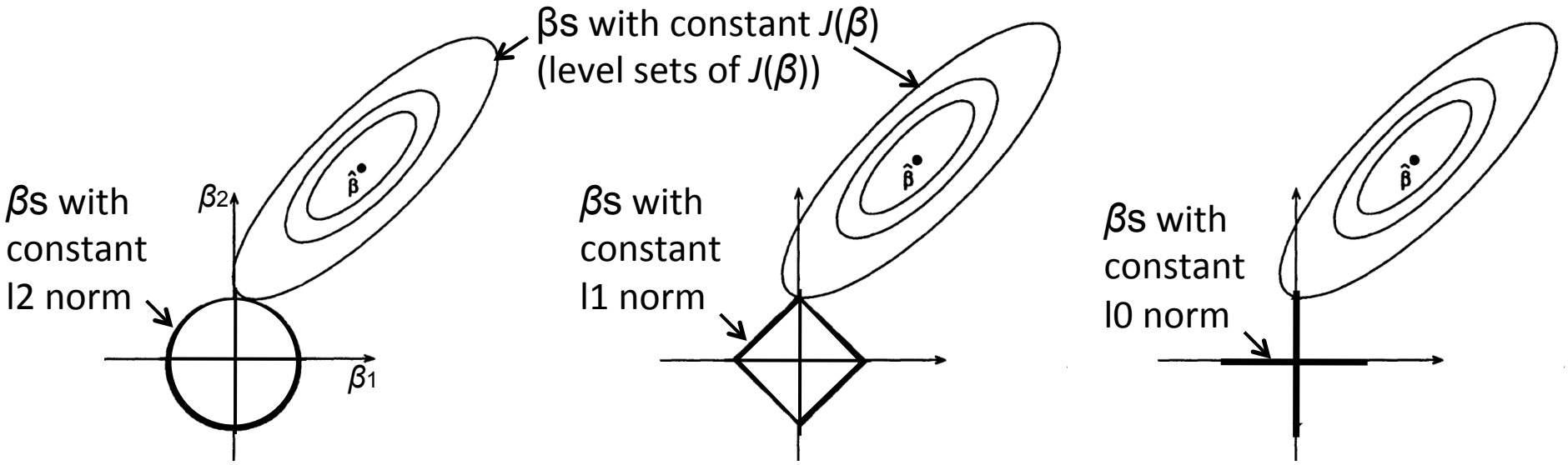
Ridge Regression vs Lasso

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

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

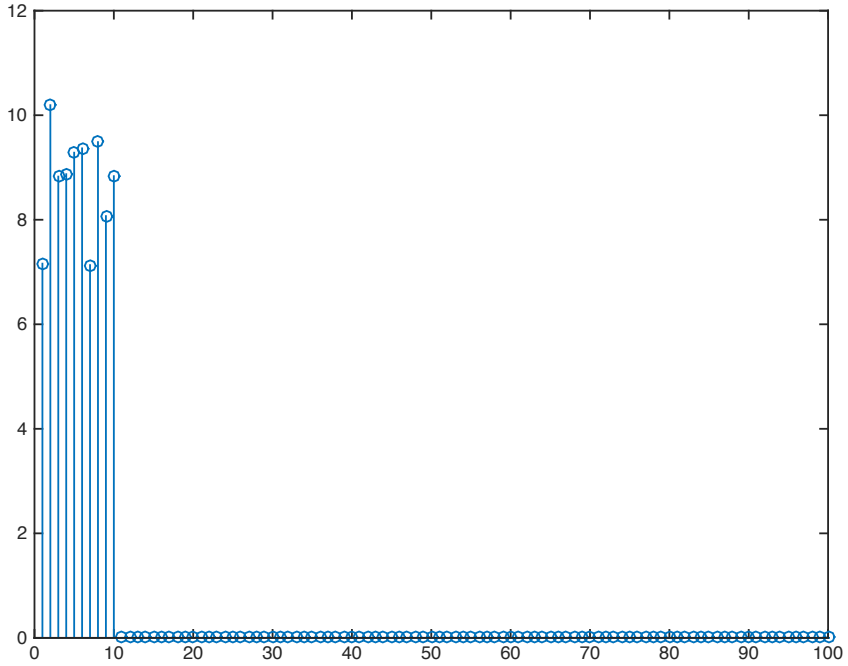
```
clear all
close all

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

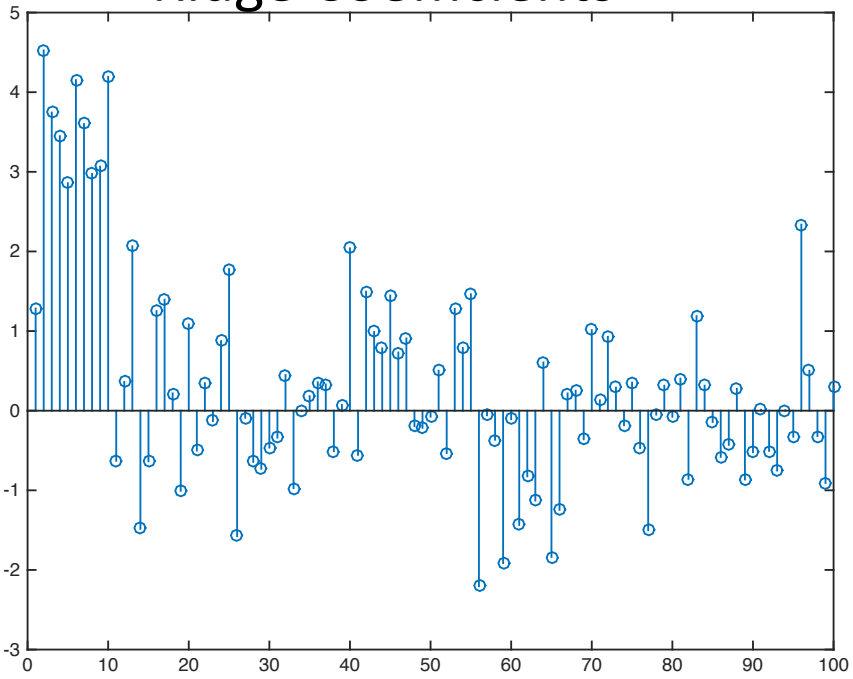
Test MSE = 33.7997

Lasso Coefficients



Test MSE = 185.9948

Ridge Coefficients

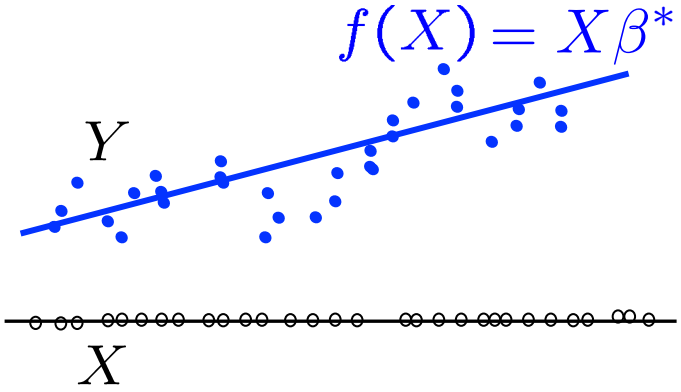


Regularized Least Squares – connection to MLE and MAP (Model-based approaches)

Least Squares and M(C)LE

Intuition: Signal plus (zero-mean) Noise model

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



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

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

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

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

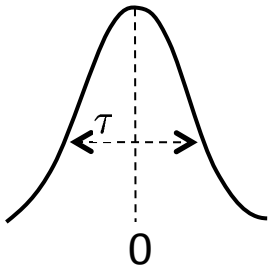
Regularized Least Squares and M(C)AP

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

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

I) Gaussian Prior

$$\beta \sim \mathcal{N}(0, \tau^2\mathbf{I}) \qquad p(\beta) \propto e^{-\beta^T\beta/2\tau^2}$$



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

$\lambda \downarrow$ constant(σ^2, τ^2)

Ridge Regression

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

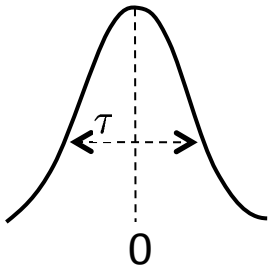
Regularized Least Squares and M(C)AP

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

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

I) Gaussian Prior

$$\beta \sim \mathcal{N}(0, \tau^2\mathbf{I}) \qquad p(\beta) \propto e^{-\beta^T\beta/2\tau^2}$$



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

$\downarrow$

$\text{constant}(\sigma^2, \tau^2)$

Ridge Regression

Prior belief that β is Gaussian with zero-mean biases solution to "small" β

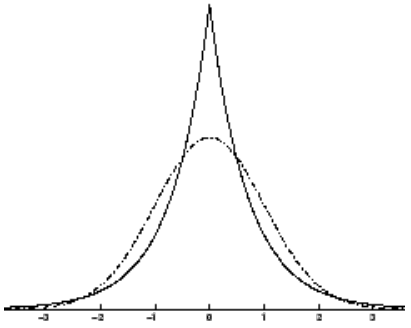
Regularized Least Squares and M(C)AP

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

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

II) Laplace Prior

$$\beta_i \overset{iid}{\sim} \text{Laplace}(0, t) \qquad p(\beta_i) \propto e^{-|\beta_i|/t}$$



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

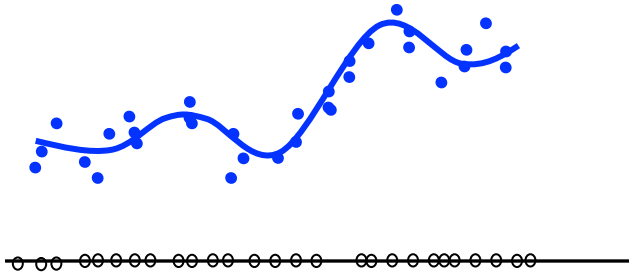
$\lambda \|\beta\|_1 \downarrow \text{constant}(\sigma^2, t)$

Lasso

Prior belief that β is Laplace with zero-mean biases solution to "sparse" β

Beyond Linear Regression

Polynomial regression
Regression with nonlinear features



Kernelized Ridge Regression (later)

Local Kernel Regression (later)

Polynomial Regression

degree m

Univariate (1-dim) case: $f(X) = \beta_0 + \beta_1 X + \beta_2 X^2 + \cdots + \beta_m X^m = \mathbf{X}\beta$

where $\mathbf{X} = [1 \; X \; X^2 \ldots X^m]$, $\beta = [\beta_1 \ldots \beta_m]^T$

$$\widehat{\beta} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{Y} \quad \text{or} \quad (\mathbf{A}^T\mathbf{A} + \lambda\mathbf{I})^{-1}\mathbf{A}^T\mathbf{Y} \qquad \widehat{f}_n(X) = \mathbf{X}\widehat{\beta}$$

where $\mathbf{A} = \begin{bmatrix} 1 & X_1 & X_1^2 & \ldots & X_1^m \\ \vdots & & & \ddots & \vdots \\ 1 & X_n & X_n^2 & \ldots & X_n^m \end{bmatrix}$

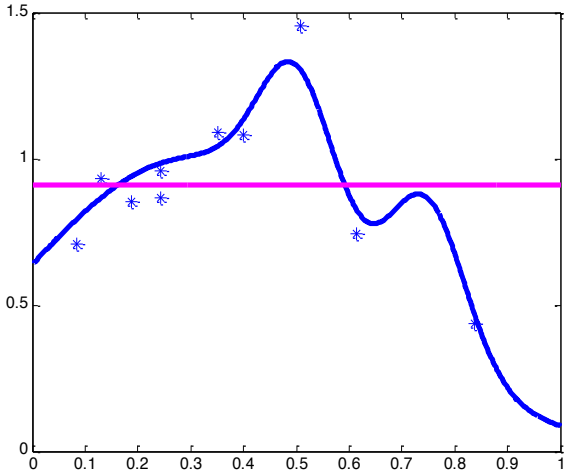
Multivariate (p-dim) case:

$$f(X) = \beta_0 + \beta_1 X^{(1)} + \beta_2 X^{(2)} + \cdots + \beta_p X^{(p)} + \sum_{i=1}^{p}\sum_{j=1}^{p} \beta_{ij} X^{(i)} X^{(j)} + \sum_{i=1}^{p}\sum_{j=1}^{p}\sum_{k=1}^{p} X^{(i)} X^{(j)} X^{(k)} + \ldots \text{terms up to degree m}$$

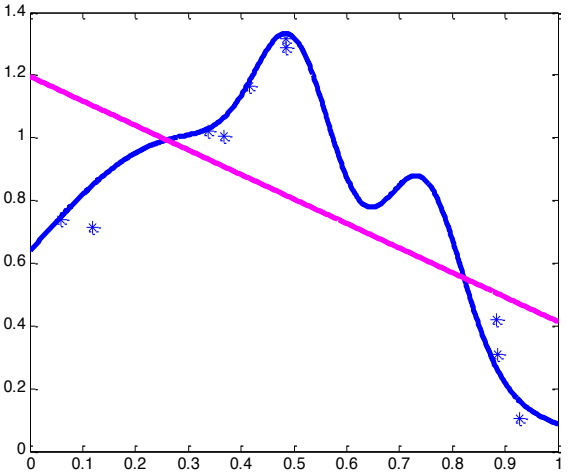
Polynomial Regression

Polynomial of order k, equivalently of degree up to k-1

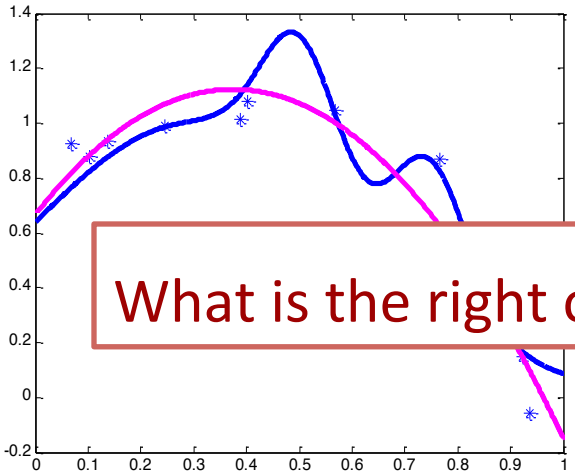
k=1



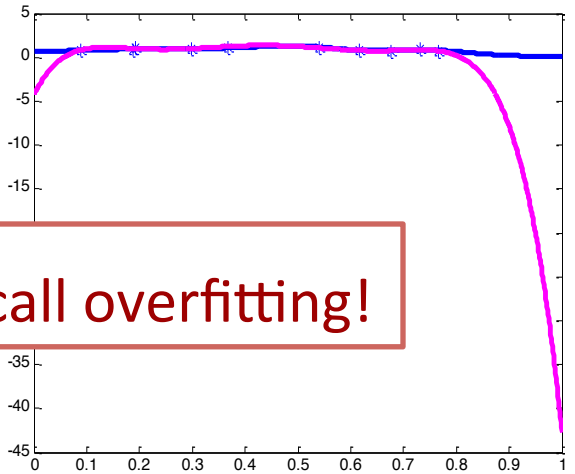
k=2



k=3



k=7

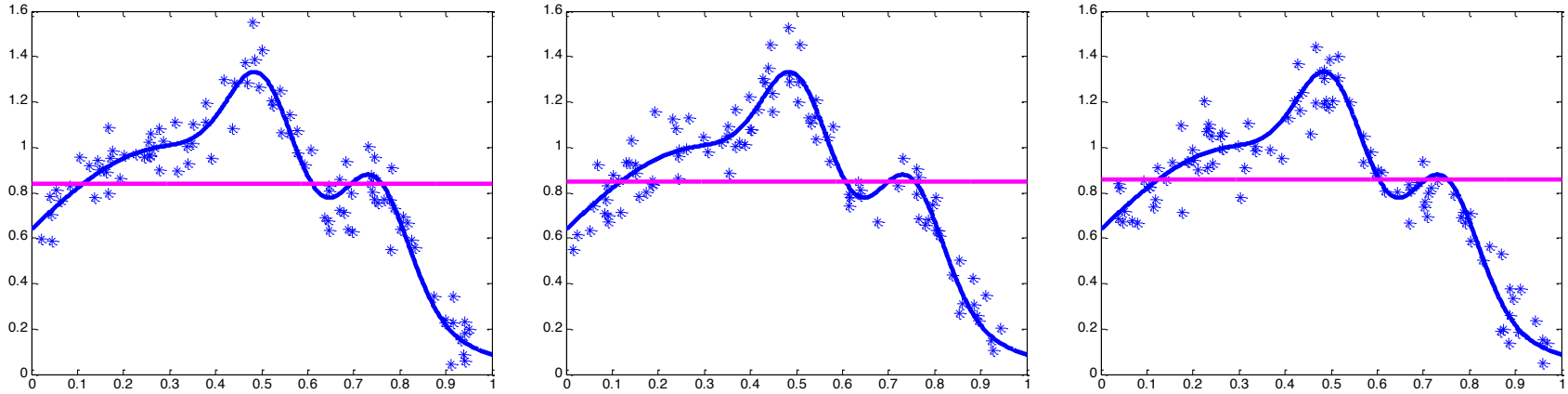


What is the right order? Recall overfitting!

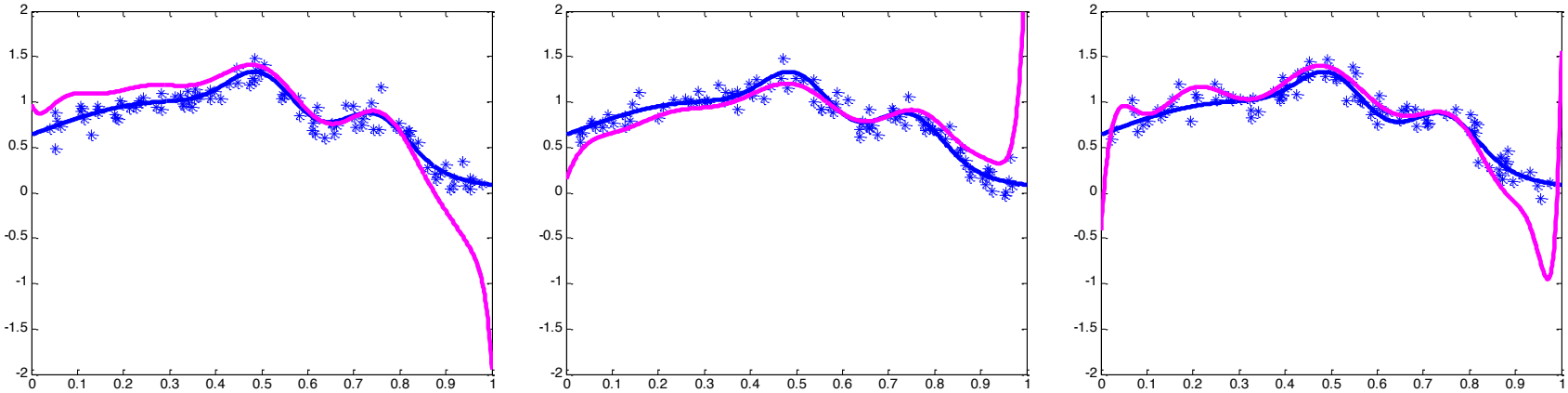
Bias – Variance Tradeoff

3 Independent training datasets

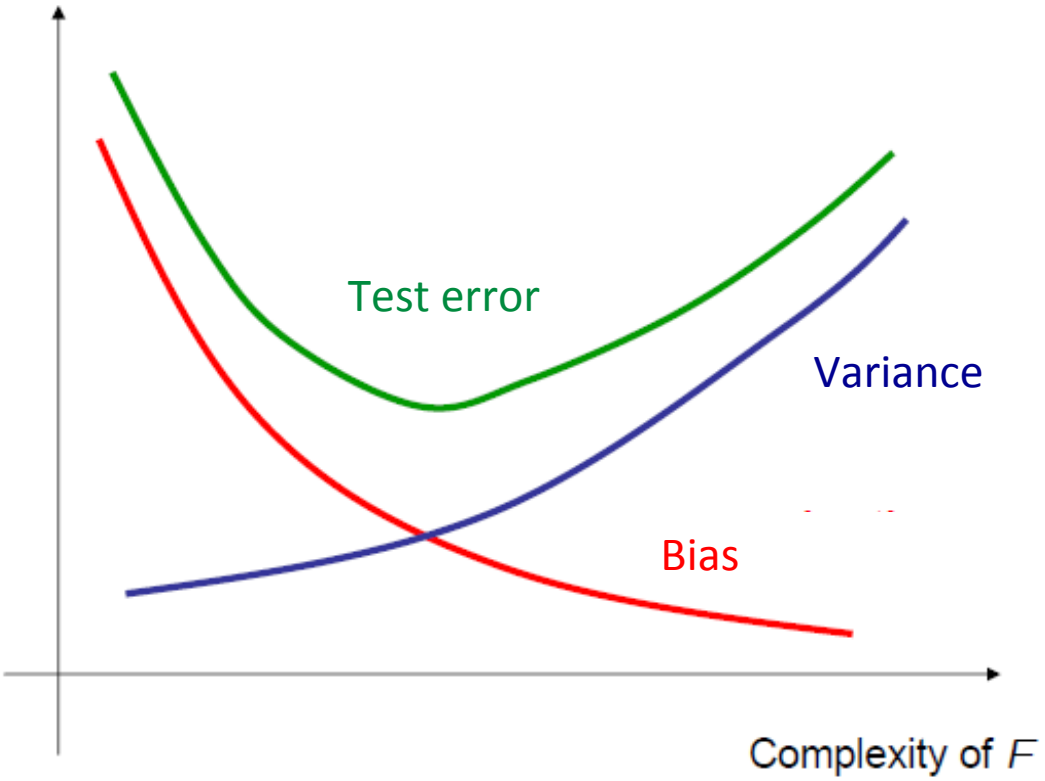
Large bias, Small variance – poor approximation but robust/stable



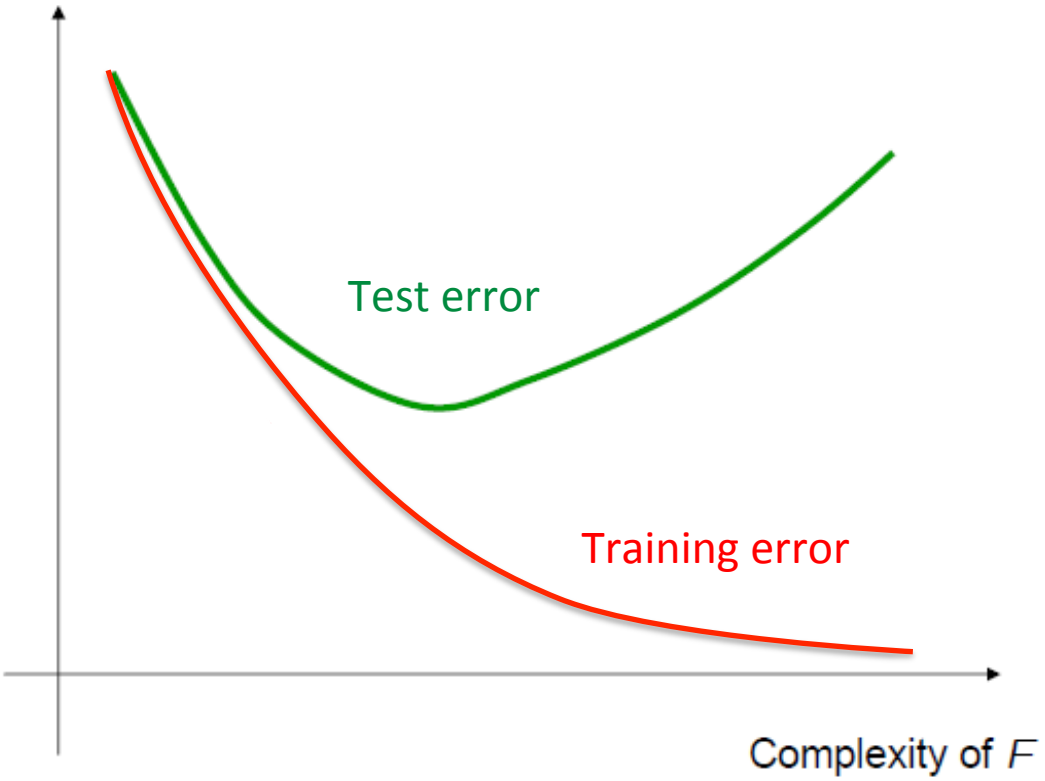
Small bias, Large variance – good approximation but unstable



Effect of Model Complexity



Effect of Model Complexity



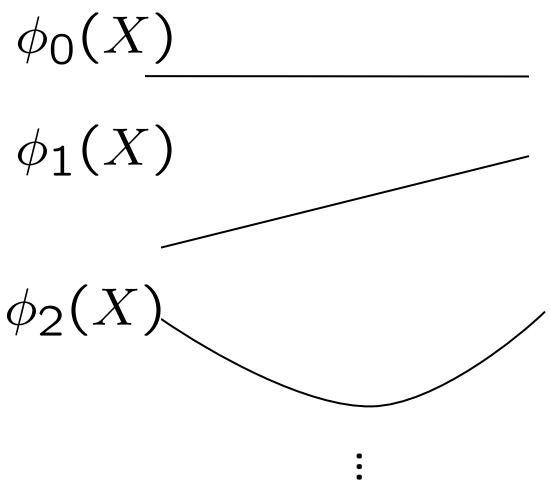
Regression with basis functions

$$f(X) = \sum_{j=0}^{m} \beta_j \phi_j(X)$$

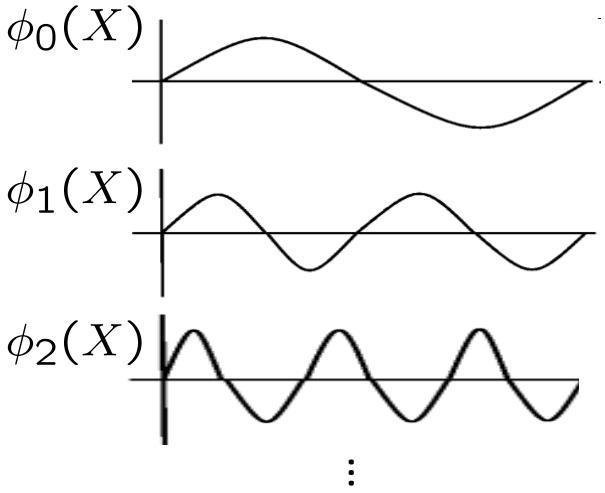
Basis coefficients

Basis functions (Linear combinations yield meaningful spaces)

Polynomial Basis

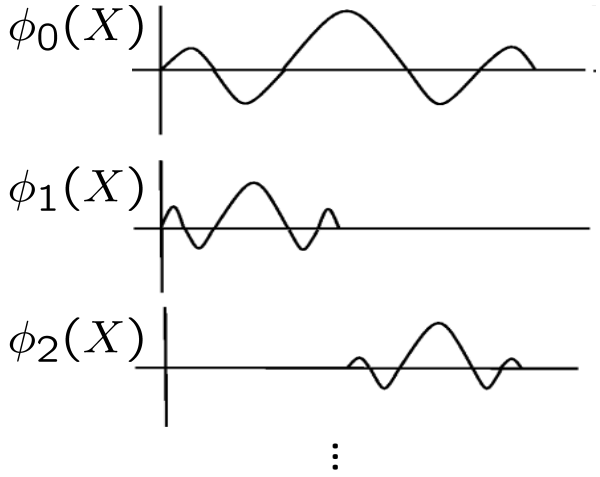


Fourier Basis



Good representation for periodic functions

Wavelet Basis



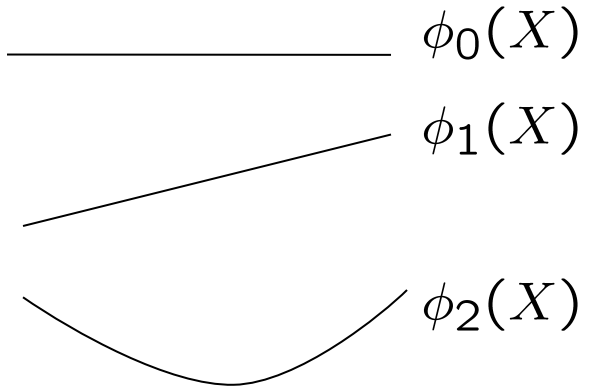
Good representation for local functions

Regression with nonlinear features

$$f(X) = \sum_{j=0}^{m} \beta_j X^j = \sum_{j=0}^{m} \beta_j \phi_j(X)$$

Weight of each feature

Nonlinear features



In general, use any nonlinear features

e.g. e^X, log X, 1/X, sin(X), …

$$\widehat{\beta} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{Y}$$

or

$$(\mathbf{A}^T\mathbf{A} + \lambda\mathbf{I})^{-1}\mathbf{A}^T\mathbf{Y}$$

$$\mathbf{A} = \begin{bmatrix} \phi_0(X_1) & \phi_1(X_1) & \dots & \phi_m(X_1) \\ \vdots & & \ddots & \vdots \\ \phi_0(X_n) & \phi_1(X_n) & \dots & \phi_m(X_n) \end{bmatrix}$$

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

$$\mathbf{X} = [\phi_0(X) \;\; \phi_1(X) \;\; \dots \;\; \phi_m(X)]$$