CS 66: Machine Learning

Prof. Sara Mathieson

Spring 2019

Admin

- Lab 2 due **THURSDAY** at midnight
- Lab 3 released today, due next Thursday
- Scribe notes for **extra credit!**
- Let me know of any partner issues (anytime)
- Reading posted (logistic regression & naïve bayes

Research Talk

Thursday, February 14, 2019

11:30-12:30 p.m.

Science Center 256

Anne Cocos

Department of Computer and Information Science

University of Pennsylvania

"Semantic Structure from Paraphrase Pairs"

Outline for February 13

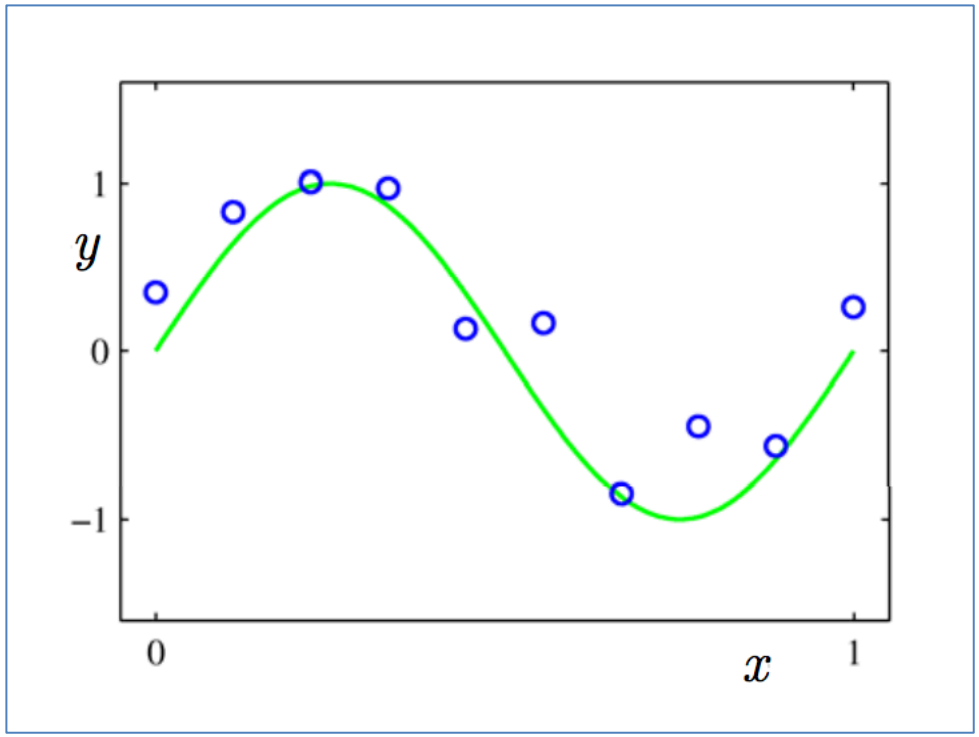
- Polynomial regression
- Regularization
- Linear regression for classification
- Begin: logistic regression

Outline for February 13

- Polynomial regression
- Regularization
- Linear regression for classification
- Begin: logistic regression

Polynomial Regression

- Can be thought of as regular linear regression with a change of basis



Plot: Jessica Wu

Polynomial Regression

before: $h_{\vec{b}}(\vec{x}) = \vec{b}^T \vec{x}$

ex

$$\vec{x} = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ \vdots \\ x_p \end{bmatrix} \rightarrow \begin{bmatrix} 1 \\ x_1 \\ x_1^2 \\ x_2 \\ x_2^2 \\ \vdots \\ x_p \\ x_p^2 \end{bmatrix}$$

find coefficient for each term

quadratic model

Generalized linear model

$$h_{\vec{b}}(\vec{x}) = \vec{b}^T \cdot \underbrace{\phi(\vec{x})}_{\text{some transformation}}$$

$$\Phi = \begin{bmatrix} - & \phi(\vec{x}_1) & - \\ - & \phi(\vec{x}_2) & - \\ & \vdots & \\ - & \phi(\vec{x}_n) & - \end{bmatrix}$$

replace X with Φ

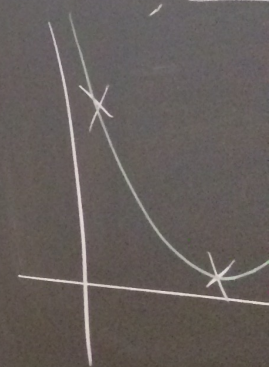
Analytic

$\vec{b} = ($

n data p

=> fit

$\boxed{n=3}$, $\boxed{d=}$



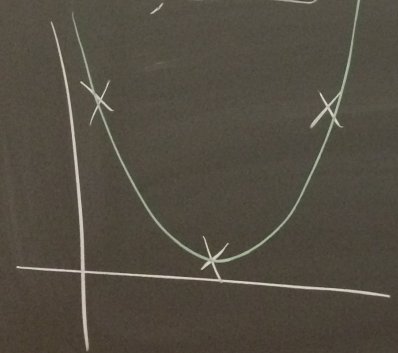
Analytic Solution

$$\vec{b} = \left(\Phi^T \Phi\right)^{-1} \left(\Phi^T \vec{y}\right)$$

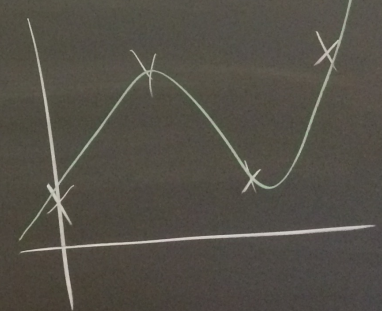
n data points, degree = d = n-1

$\Rightarrow$ fit perfectly!

$\boxed{n=3}$, $\boxed{d=2}$



$\boxed{n=4}$, $\boxed{d=3}$

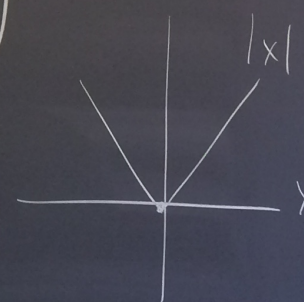


$\boxed{p=1}$, d = degree

$$\Rightarrow \boxed{h_{\vec{b}}(x) = b_0 + b_1 x + b_2 x^2 + \cdots b_d x^d}$$

in general: p features, degree d $\Rightarrow$ $\boxed{pd+1}$ coeffs

$$\phi(x) = \begin{bmatrix} 1 \\ x \\ x^2 \\ x^3 \\ \vdots \\ x_p \end{bmatrix} \qquad \boxed{\vec{b}^T \phi(x)}$$



SGD

$b_0 \leftarrow$

$b_j \leftarrow$

hyperparamet

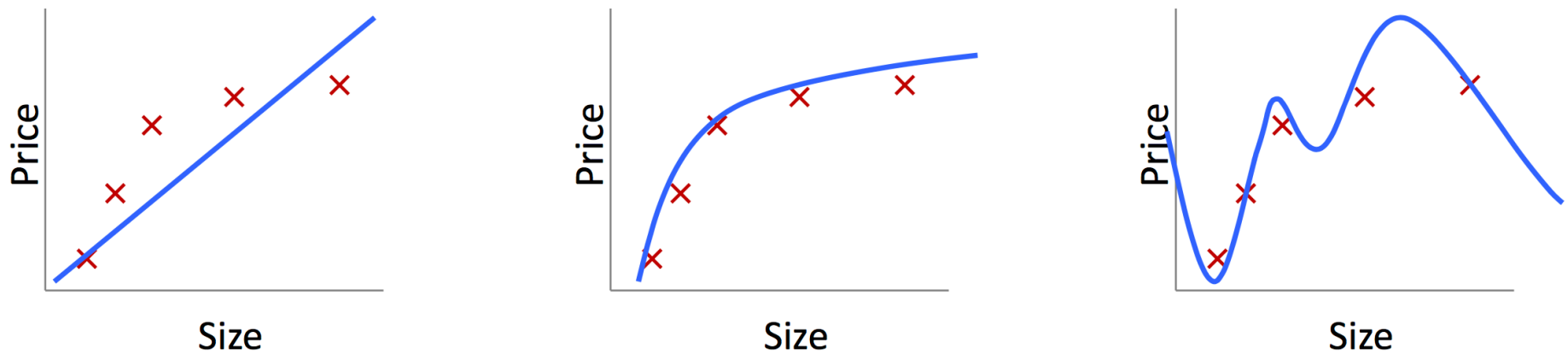
Outline for February 13

- Polynomial regression
- Regularization
- Linear regression for classification
- Begin: logistic regression

Adapted from slide by Jessica Wu
[example by Andrew Ng]

Generalization error

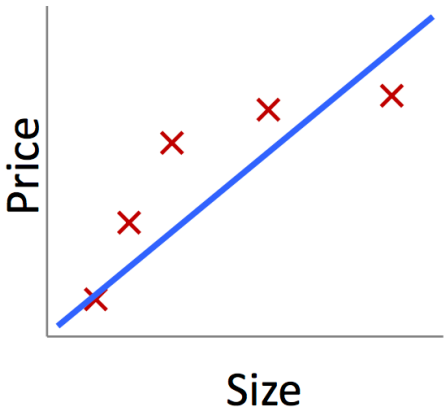
- Example: price vs. size (i.e. of a house or car)



Adapted from slide by Jessica Wu
[example by Andrew Ng]

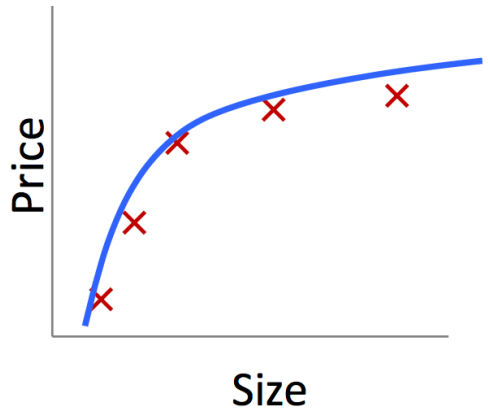
Generalization error

- Example: price vs. size (i.e. of a house or car)



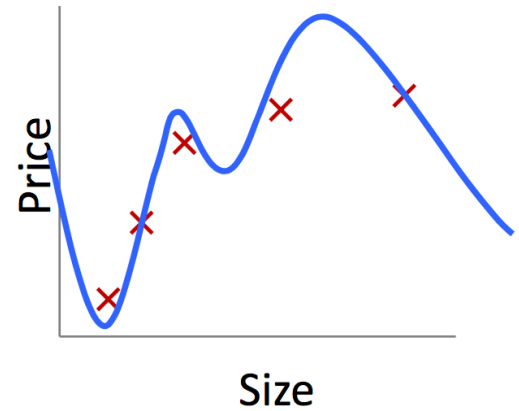
$b_0 + b_1 x$

underfitting
(high bias)



$b_0 + b_1 x + b_2 x^2$

correct fit

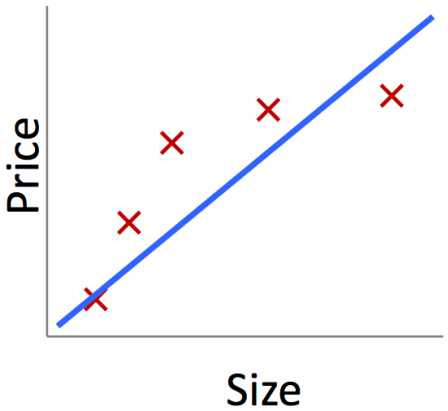


$b_0 + b_1 x + b_2 x^2 + b_3 x^3 + b_4 x^4$

overfitting
(high variance)

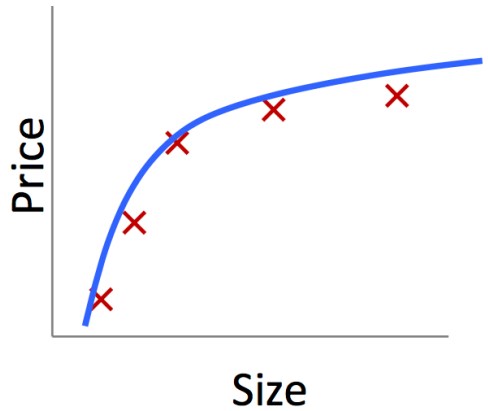
Generalization error

- Example: price vs. size (i.e. of a house or car)



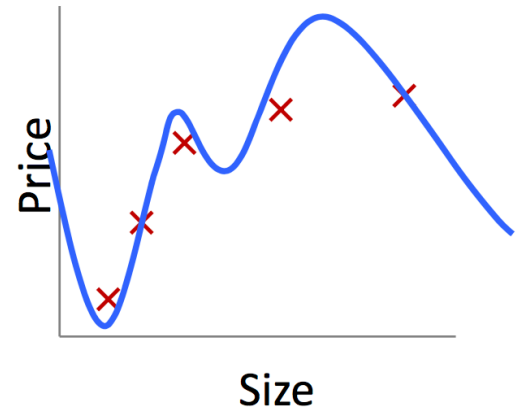
$b_0 + b_1 x$

underfitting
(high bias)



$b_0 + b_1 x + b_2 x^2$

correct fit



$b_0 + b_1 x + b_2 x^2 + b_3 x^3 + b_4 x^4$

overfitting
(high variance)

<u>Structural error:</u>
Hypothesis space cannot model true relationship

$\Rightarrow$ More data doesn't help
$\Rightarrow$ Need a more flexible model

balance
$\Leftrightarrow$

<u>Estimation (approximation) error:</u>
Hypothesis space *can* model true relationship, BUT hard to identify correct model due to large hypothesis space, small n, or noise
$\Rightarrow$ Reduce hypothesis space
$\Rightarrow$ Add more data

Regularization

What if ...

- we have a limited # of training examples ($n<p$), or
- we want to automatically control the complexity of the learned hypothesis?

Regularization

What if ...

- we have a limited # of training examples ($n<p$), or
- we want to automatically control the complexity of the learned hypothesis?

Idea: penalize large values of b_j

Why prefer small weights?

- if large weights, small change in feature can result in large change in prediction
- prevent giving too much weight to any one feature
- might prefer zero weight for useless features

Slide by Jessica Wu

Common Regularizers

$$||\boldsymbol{b}||_0 = \sum_{j:b_j \neq 0} 1$$

L_0 norm

- Number of non-zero entries
- Minimizing L_0 norm is NP hard

Slide adapted from Jessica Wu

Common Regularizers

$$||\boldsymbol{b}||_0 = \sum_{j:b_j \neq 0} 1$$

$$||\boldsymbol{b}||_1 = \sum_{j=1}^{p} |b_j|$$

L_0 norm

- Number of non-zero entries
- Minimizing L_0 norm is NP hard

L_1 norm

- Sum of magnitude of weights
- Not differentiable

Slide adapted from Jessica Wu

Common Regularizers

$$||\boldsymbol{b}||_0 = \sum_{j:b_j \neq 0} 1$$

$$||\boldsymbol{b}||_1 = \sum_{j=1}^{p} |b_j|$$

$$||\boldsymbol{b}||_2 = \sqrt{\sum_{j=1}^{p} b_j^2}$$

L_0 norm

- Number of non-zero entries
- Minimizing L_0 norm is NP hard

L_1 norm

- Sum of magnitude of weights
- Not differentiable

L_2 norm

- Sum of squared weights
- Differentiable

Slide adapted from Jessica Wu

Common Regularizers

$$||\boldsymbol{b}||_0 = \sum_{j:b_j \neq 0} 1$$

$$||\boldsymbol{b}||_1 = \sum_{j=1}^{p} |b_j|$$

$$||\boldsymbol{b}||_2 = \sqrt{\sum_{j=1}^{p} b_j^2}$$

L_0 norm

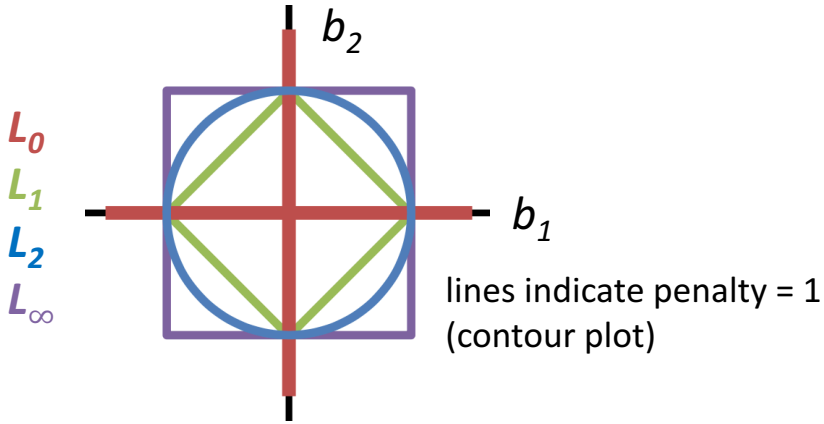
- Number of non-zero entries
- Minimizing L_0 norm is NP hard

L_1 norm

- Sum of magnitude of weights
- Not differentiable

L_2 norm

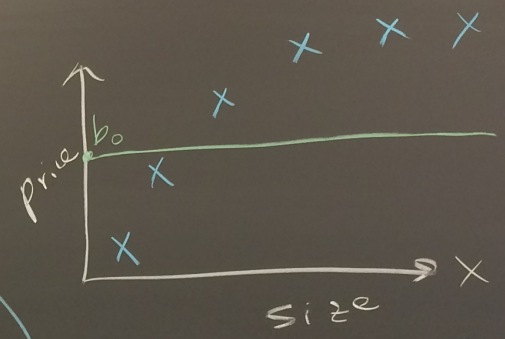
- Sum of squared weights
- Differentiable



Slide adapted from Jessica Wu

Regularization

$$J(\vec{b}) = \underbrace{\frac{1}{2}\sum_{i=1}^{n}(b^T\vec{x}_i - y_i)^2}_{\text{fit to training data}} + \underbrace{\frac{\lambda}{2}\sum_{j=1}^{p}(b_j)^2}_{\text{regularization.}}$$



$\lambda \geq 0$ is the regularization parameter

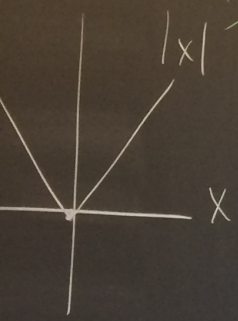
- Small λ: more fit to training data.
- large λ: keeps weights small
  more generalization.

do not regularize b_0

$$b_0 + \cancel{b_1}^{\,0} x + \cancel{b_2}^{\,0} x^2 + \cancel{b_3}^{\,0} x^3 + \cancel{b_4}^{\,0} x^4$$

Choose $\lambda = 10^{10}$ (very large)

$\boxed{d+1}$ coeffs



SGD

$$b_0 \leftarrow b_0 - \alpha(\vec{b}^T \vec{x}_i - y_i)$$

$$b_j \leftarrow b_j - \alpha\left[(b^T x_i - y_i)x_{ij} + \lambda b_j\right]$$

hyperparameters

$$= \underbrace{(1-\alpha\lambda)}_{\text{pulling } b \text{ terms toward } 0 \text{ each iteration.}} b_j - \alpha\left[(b^T x_i - y_i)x_{ij}\right]$$

Analytic

$$J(\vec{b}) = (X\vec{b} - \vec{y})^T(X\vec{b} - \vec{y}) + \underbrace{\lambda \vec{b}^T \vec{b}}$$

$$\downarrow$$

$$\vec{b} = (X^T X + \underbrace{\lambda I}) X^T \vec{y}$$

b_0

$$I = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

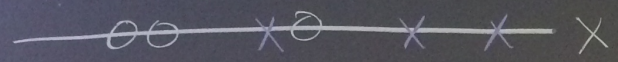
$$\underbrace{\begin{bmatrix} b_1 & \vec{b} & - \end{bmatrix} \begin{bmatrix} b_1 \\ \vec{b} \\ | \end{bmatrix}}_{\vec{b} \cdot \vec{b}} = \sum_{j=1}^{p} b_j^2$$

$\vec{b} \cdot \vec{b}$ dot product

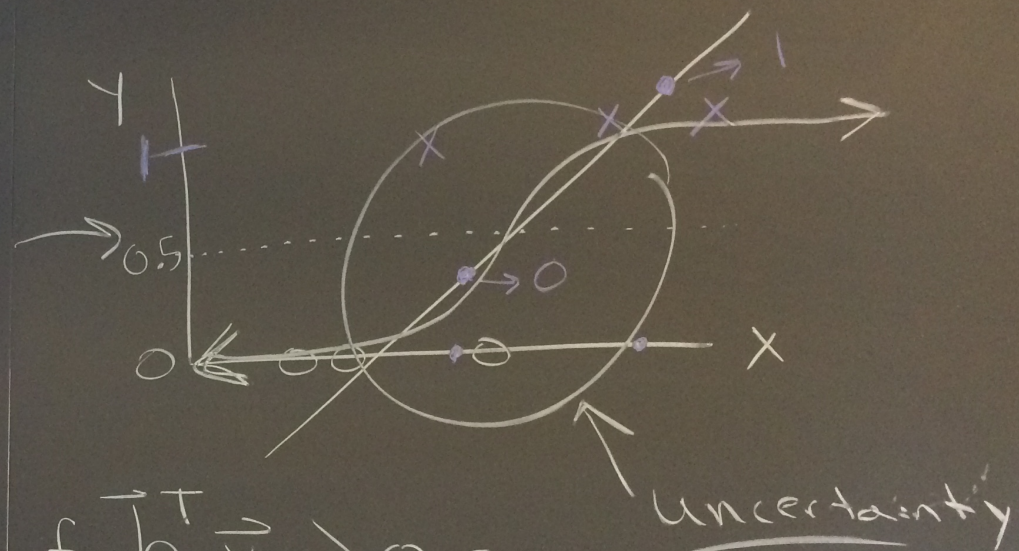
$= \vec{b}^T \vec{b}$

Outline for February 13

- Polynomial regression
- Regularization
- Linear regression for classification
- Begin: logistic regression



x	likes toy
8	Y
6	N
2	N
5	Y
10	Y
3	N



$$\hat{y} = \begin{cases} 1 & \text{if } \vec{b}^T \vec{x} \geq 0.5 \\ 0 & \text{o.w.} \end{cases}$$