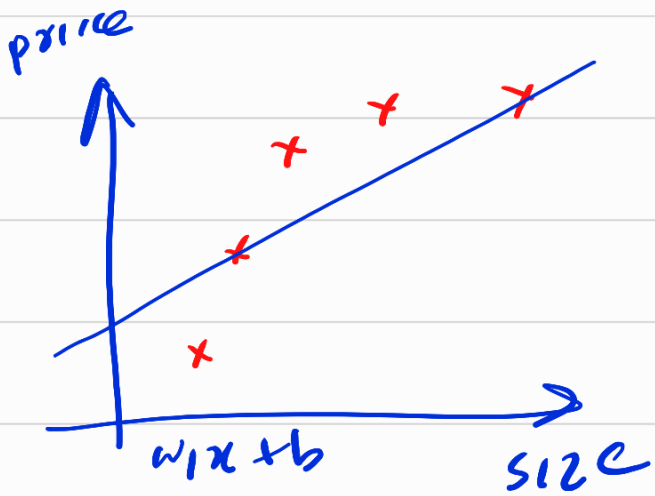
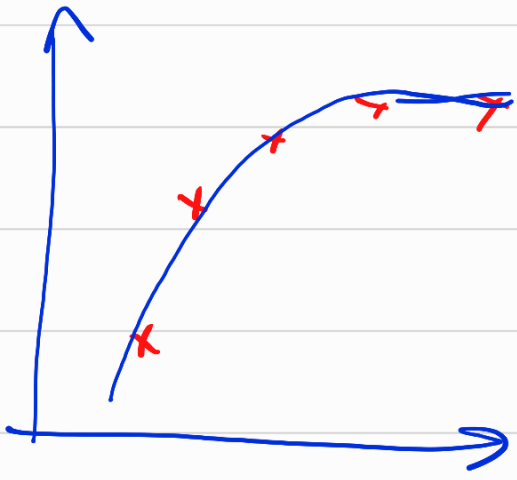


$w_1 x + b$

underfit

- high bias

✗

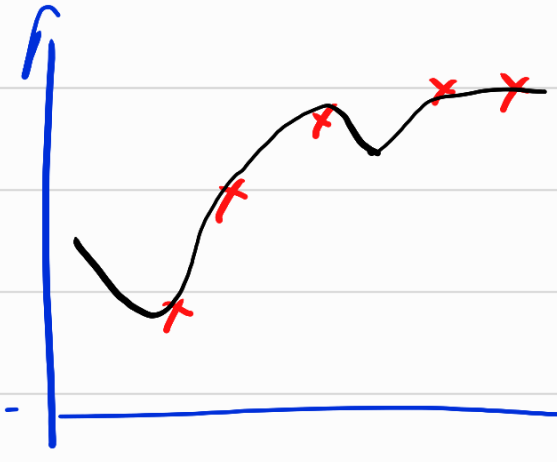


✓ $w_1 x + w_2 x^2 + b$

fits well

generalization

(well on brand new ex)



cost is 0 all fits to model

$w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4 + b$

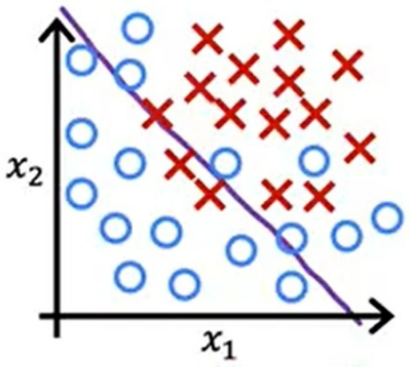
✗ * Overfit

not good for new examples

high variance

for classification — Overfitting

Classification

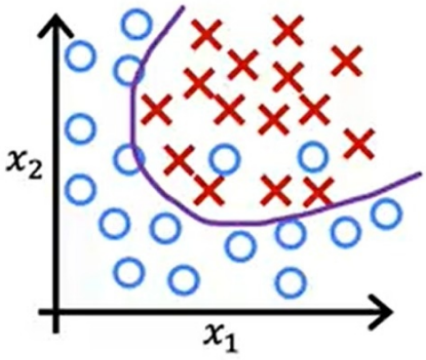


$z = w_1 x_1 + w_2 x_2 + b$

$f_{\vec{w},b}(\vec{x}) = g(z)$

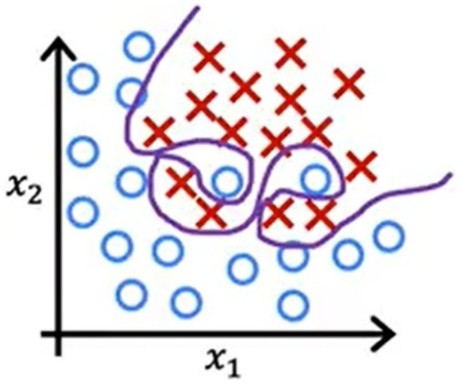
g is the sigmoid function

underfit high bias



$z = w_1 x_1 + w_2 x_2 + w_3 x_1^2 + w_4 x_2^2 + w_5 x_1 x_2 + b$

just right



$z = w_1 x_1 + w_2 x_2 + w_3 x_1^2 x_2 + w_4 x_1^2 x_2^2 + w_5 x_1^2 x_2^3 + w_6 x_1^3 x_2 + \cdots + b$

10:50

Address:

① Collect more training examples

② Select features to include/exclude

all features + insufficient data } overfit

③ Regularization

- reduce size of parameters
  (eliminate feature $x^q \to 0$)

↓

small values for w_j

reducing $b \to$ makes low diff.

Cost function with Regularization

$$\min \frac{1}{2m} \sum_{i=1}^{m} (f_{\vec{w},b}(x^i) - y^i)^2 + 1000 w_3 + 1000 w_4$$

to minimize w_3 & $w_4 \to 0$

$$J(\vec{w}, b) = \quad + \frac{\lambda}{2m} \sum_{j=1}^{n} w_j^2$$

lambda > 0 (regularization parameter)

to scale both similarly

Regularization

small values $w_1, w_2, \cdots, w_n, b$

simpler model

less likely to overfit

$w_3 \approx 0$

$w_4 \approx 0$

size	bedrooms	floors	age	avg income	...	distance to coffee shop	price
x_1	x_2	x_3	x_4	x_5		x_{100}	y

$n = 100$

n features

$w_1, w_1, w_2, \cdots, w_{100}, b$

regularization term

$$J(\vec{w}, b) = \frac{1}{2m}\left[\sum_{i=1}^{m}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)^2 + \frac{\lambda}{2m}\sum_{j=1}^{n} w_j^2 + \frac{\lambda}{2m} b^2\right]$$

"lambda" regularization parameter

$\lambda > 0$

can include or exclude b

not needed

if λ is very large

$f_{\vec{w},b}(\vec{x}) = w_1 x + w_2 x^2 + \cdots + b$

$w_1 \approx 0$, $w_2 \approx 0$

$f(x) = b \longrightarrow$ underfit

To implement Gradient Descent:

repeat {

$$w_j = w_j - \alpha \left[\frac{1}{m} \sum_{i=1}^{m} \left[\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)} \right) x_j^{(i)} \right] + \frac{\lambda}{m} w_j \right]$$

$$b = b - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)} \right)$$

} simultaneous update

same term for logistic reg.

More Derivative Examples

$w = 3$ $J(w) = w^2 = 9$ $w \uparrow 0.001$ $J(w) = J(3.001) = 9.006001$ $\frac{\partial}{\partial w} J(w) = 6$

$J(w) \uparrow 6 \times 0.001$

$w = 2$ $J(w) = w^2 = 4$ $w \uparrow 0.001$ $J(w) = J(2.001) = 4.004001$ $\frac{\partial}{\partial w} J(w) = 4$

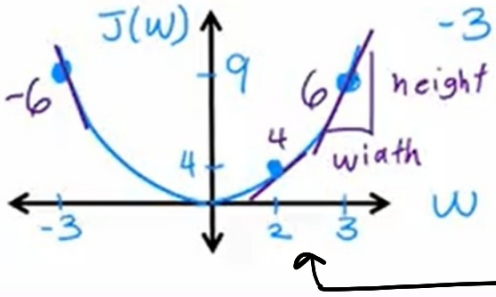
$J(w) \uparrow 4 \times 0.001$

$\downarrow 0.006$

$w = -3$ $J(w) = w^2 = 9$ $w \uparrow 0.001$ $J(w) = J(-2.999) = 8.994001$ $\frac{\partial}{\partial w} J(w) = -6$

$J(w) \downarrow 6 \times 0.001$

$J(w) \uparrow -6 \times 0.001$



$-3 + 0.001$

$\frac{\partial}{\partial w} J(w) = 2w$

$y = x^2$

$\frac{dy}{dx} = 2x$

$w = 2 \quad w^2 = 4 \quad 2w = 4$

$\rightarrow J(w) = 4.004001$

$\uparrow 4 \times \varepsilon$ //

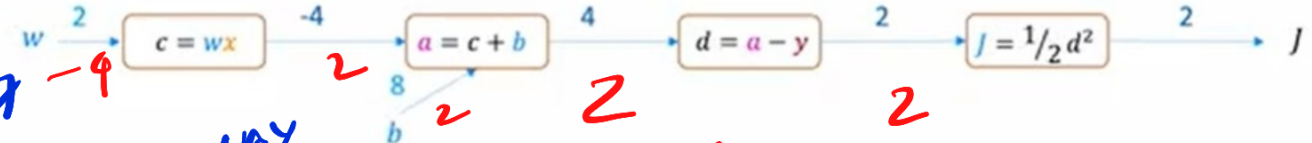
$\frac{\partial}{\partial w}$ → if w has many variables

sq error sum

Computing the Derivatives

Forward prop →

$w = 2 \quad b = 8 \quad x = -2 \quad y = 2 \quad a = wx + b \quad J = \frac{1}{2}(a - y)^2$



-4 $\quad 2 \quad 2 \quad 2$

w ↑ 0.001

same way $\frac{\partial J}{\partial c} = 2$ $\quad \frac{\partial J}{\partial b} = 2$ $\quad \frac{\partial J}{\partial d} = 2$

ε

d ↑ J ?

$\frac{\partial J}{\partial w} = \frac{\partial c}{\partial w} \times \frac{\partial J}{\partial c}$

$= -2 \times 2$

$\frac{\partial J}{\partial w} = -4$

Backward Propagation ←

↓ if w goes up by ε

$J \rightarrow +4\varepsilon$ + ves

if a ↑ 0.001

d → ↑ 0.001

but J → $\frac{\partial J}{\partial d} = 2$

J ↑ 0.002

B.P

compute derivatives in N+P steps

rather than N × P

$\frac{\partial J}{\partial a} = 2$ //