

# CS 260: Foundations of Data Science

Prof. Thao Nguyen

Fall 2024



**HAVERFORD**  
COLLEGE

# Outline for today

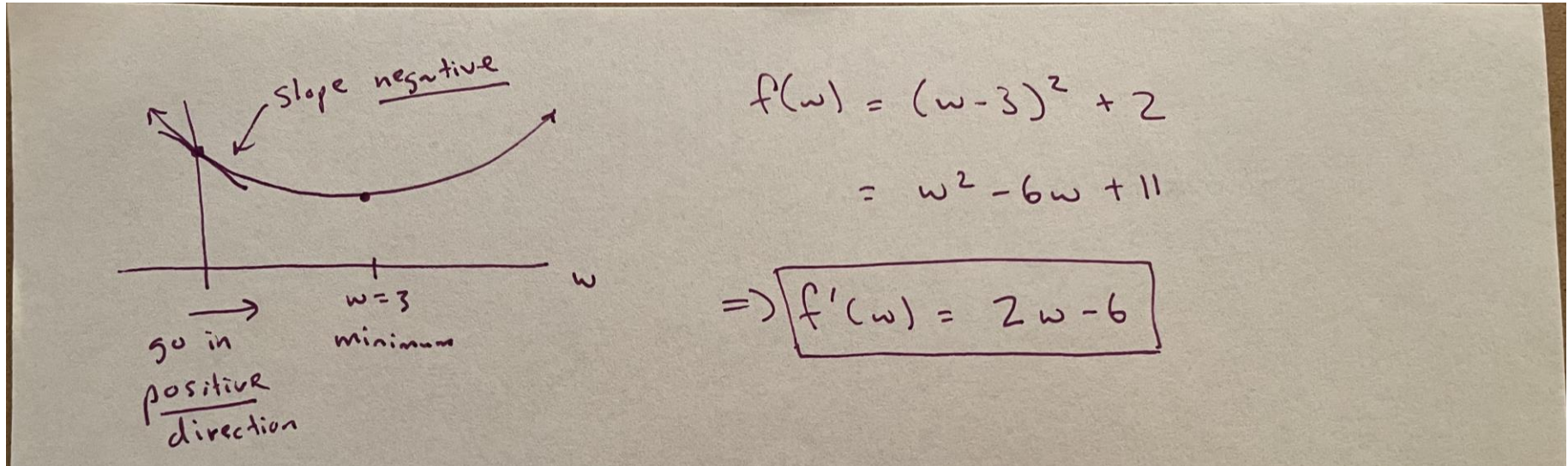
- SGD (Stochastic Gradient Descent)
- Handout 6 (SGD solution example)
- Analytic vs. SGD (pros and cons)
- (if time) Polynomial regression

# Outline for today

- SGD (Stochastic Gradient Descent)
- Handout 6 (SGD solution example)
- Analytic vs. SGD (pros and cons)
- (if time) Polynomial regression

# Stochastic gradient descent example

Goal: minimize the function  $f(w) = w^2 - 6w + 11$

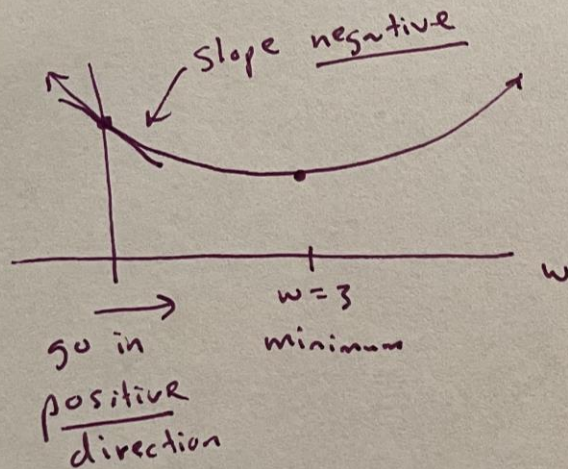


$$w \leftarrow w - \alpha f'(w)$$

↑  
step size

# Stochastic gradient descent example

Goal: minimize the function  $f(w) = w^2 - 6w + 11$



$$\begin{aligned} f(w) &= (w-3)^2 + 2 \\ &= w^2 - 6w + 11 \end{aligned}$$

$$\Rightarrow \boxed{f'(w) = 2w - 6}$$

$$\textcircled{1} \quad w \leftarrow 0 - 0.1(2 \cdot 0 - 6)$$

$$w \leftarrow 0 + 0.6$$

$$\boxed{w \leftarrow 0.6}$$

$$\textcircled{2} \quad w \leftarrow 0.6 - 0.1(2 \cdot 0.6 - 6)$$

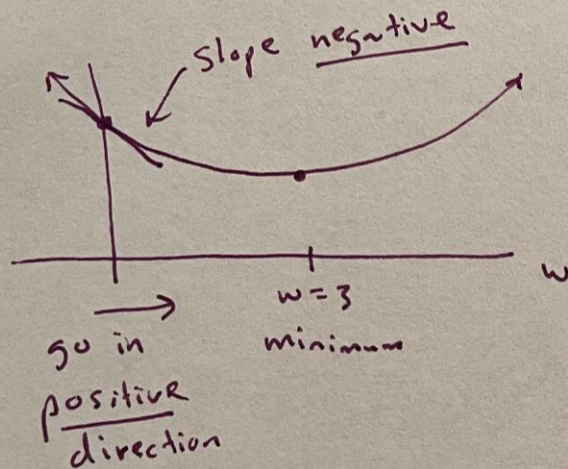
$$w \leftarrow 0.6 - 0.1(-4.8)$$

$$\boxed{w \leftarrow 1.08}$$



# Stochastic gradient descent example

Goal: minimize the function  $f(w) = w^2 - 6w + 11$



$$\begin{aligned} f(w) &= (w-3)^2 + 2 \\ &= w^2 - 6w + 11 \end{aligned}$$

$$\Rightarrow \boxed{f'(w) = 2w - 6}$$

$$\textcircled{1} \quad w \leftarrow 0 - 0.1(2 \cdot 0 - 6)$$

$$w \leftarrow 0 + 0.6$$

$$\boxed{w \leftarrow 0.6}$$

$$\textcircled{2} \quad w \leftarrow 0.6 - 0.1(2 \cdot 0.6 - 6)$$

$$w \leftarrow 0.6 - 0.1(-4.8)$$

$$\boxed{w \leftarrow 1.08}$$

stop when:

$$|f(w^t) - f(w^{t-1})| < \epsilon$$

$$\epsilon = 1 \times 10^{-8}$$

(for example)

# Stochastic Gradient Descent for Linear Regression

Key Idea: take the derivative of **one datapoint** at a time and use that to update  $w$

$$J(\vec{w}) = \frac{1}{2} \sum_{i=1}^n (\vec{w} \cdot \vec{x}_i - y_i)^2$$

gradient  
with respect to one datapoint: (i.e.  $\vec{x}_i$ )

$$\nabla J_{\vec{x}_i} = \frac{\partial J(\vec{w})}{\partial \vec{w}}_{\vec{x}_i} = (\vec{w} \cdot \vec{x}_i - y_i) \vec{x}_i$$



# Stochastic Gradient Descent for Linear Regression

for (epoch)  
iteration  $t$  :

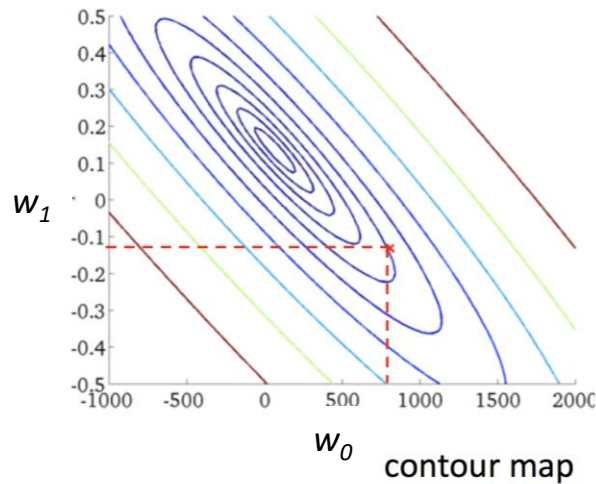
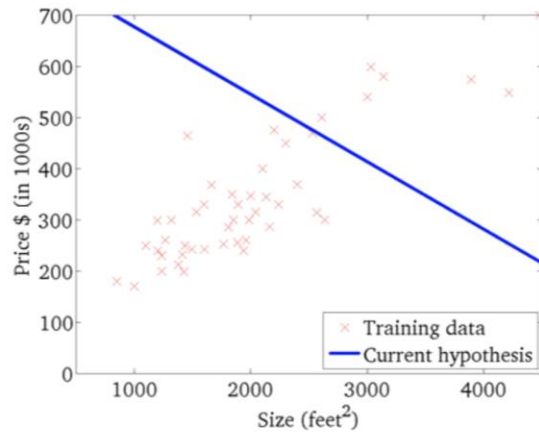
for  $i = 1, 2, 3 \dots n$  } usually shuffle

$$\vec{w} \leftarrow \vec{w} - \alpha (\vec{w} \cdot \vec{x}_i - y_i) \vec{x}_i$$

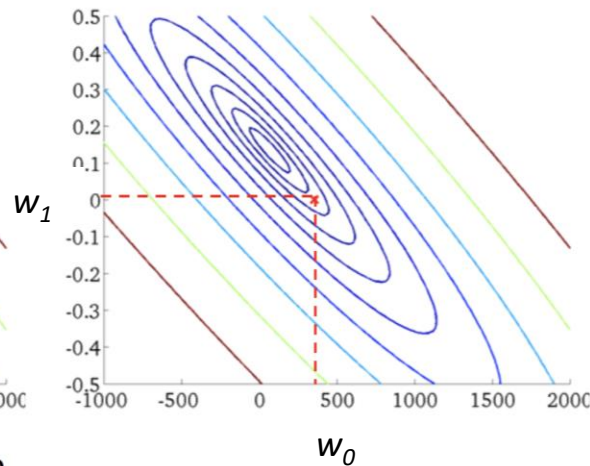
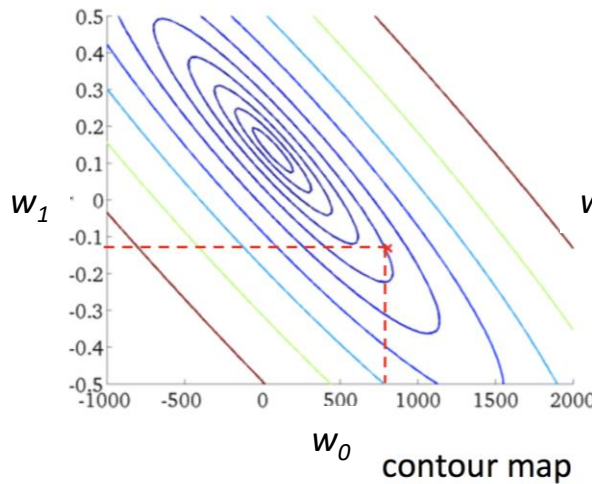
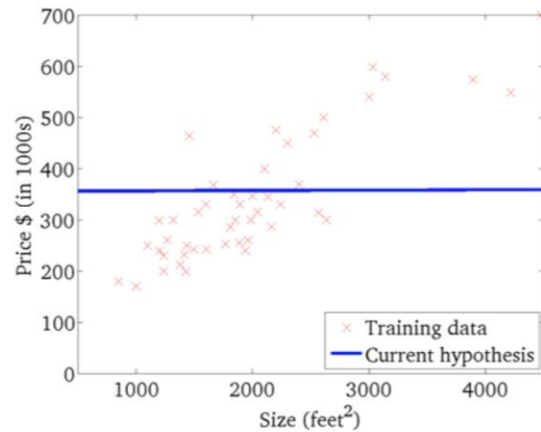
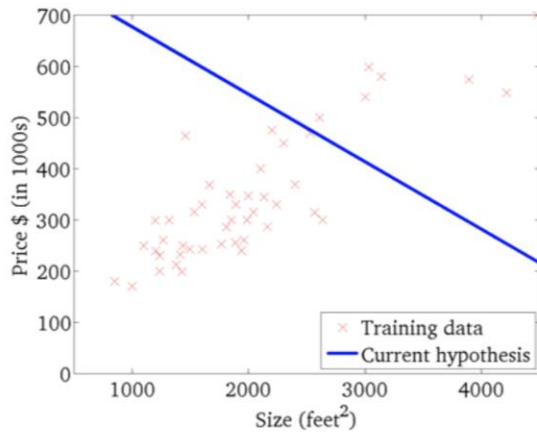
check for convergence :  $|\mathcal{J}(\vec{w}^t) - \mathcal{J}(\vec{w}^{t-1})| < \epsilon$



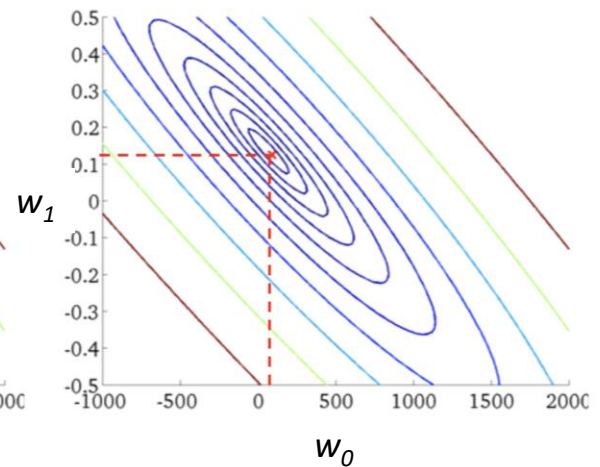
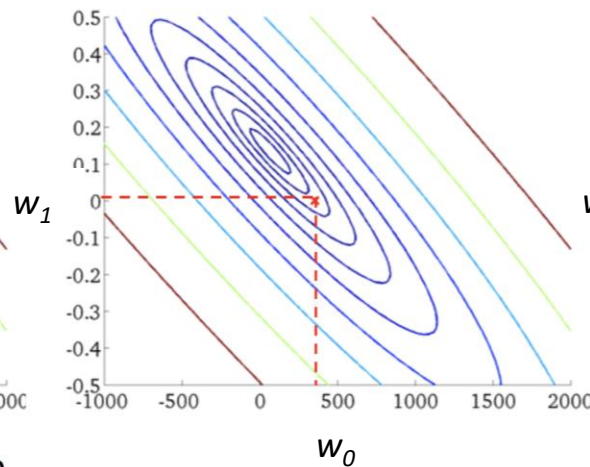
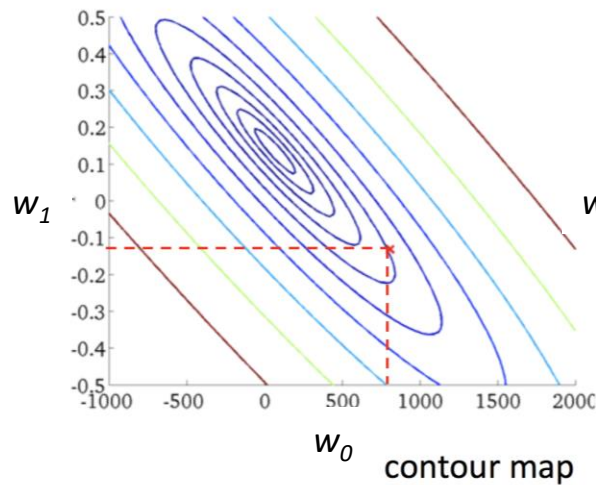
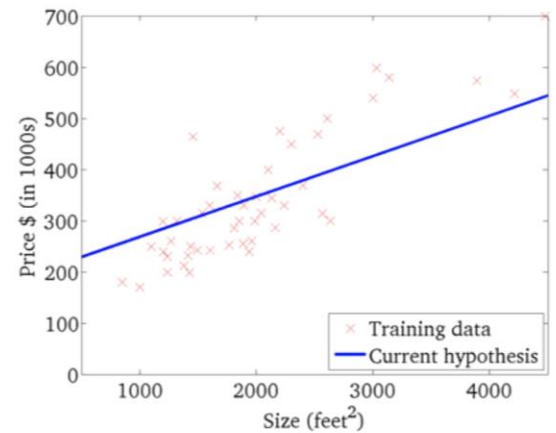
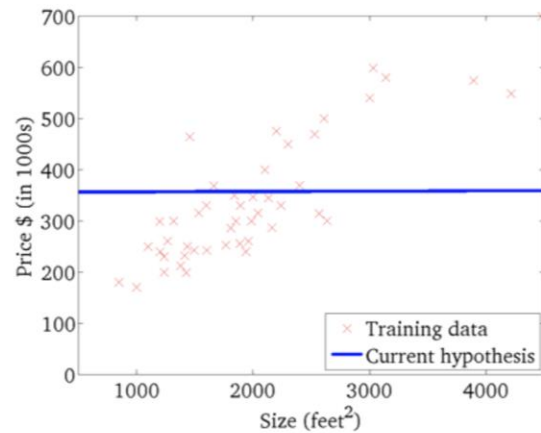
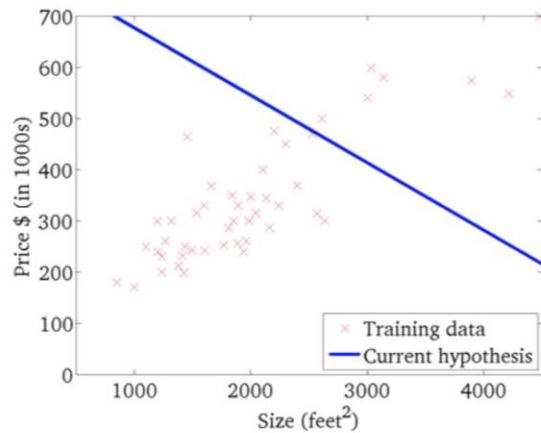
# Linear Model and Cost Function J



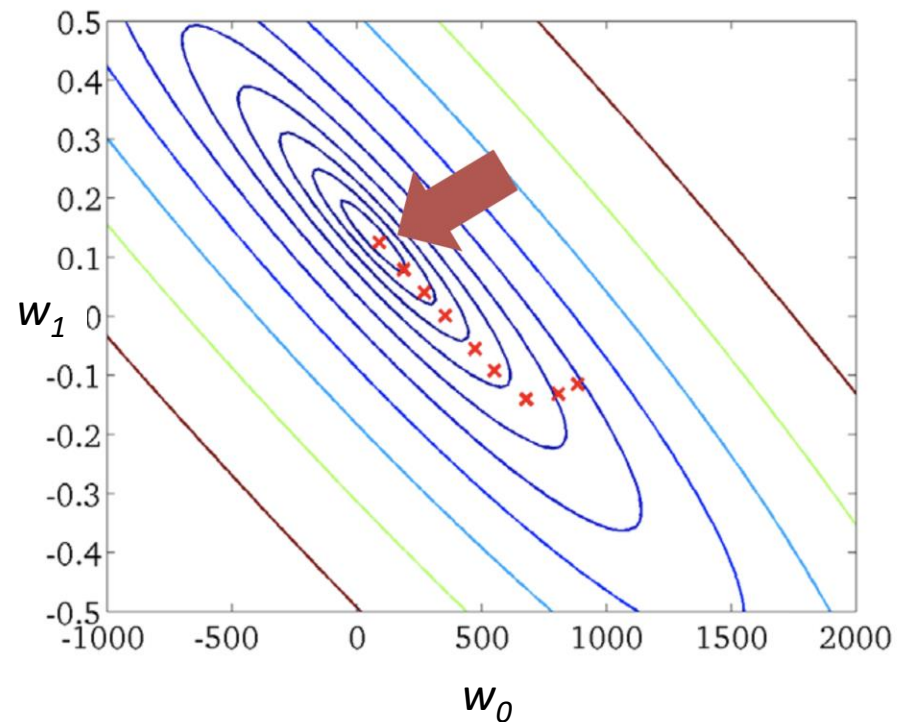
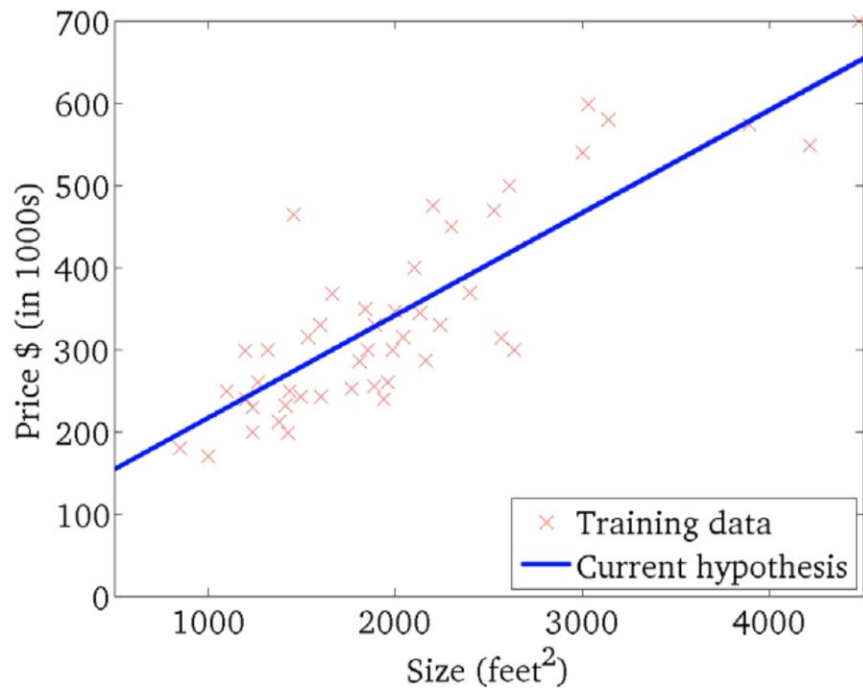
# Linear Model and Cost Function J



# Linear Model and Cost Function J



# Gradient Descent: walking toward the minimum

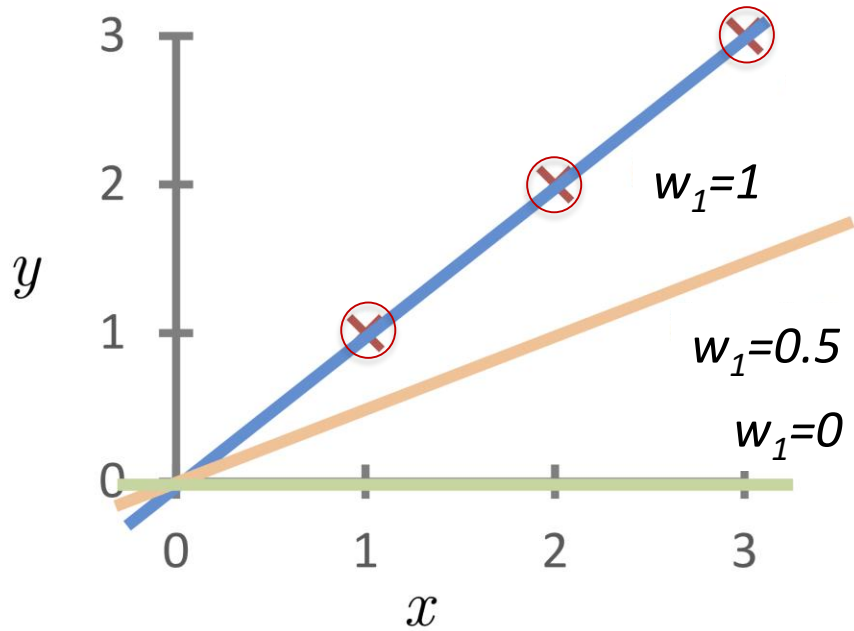




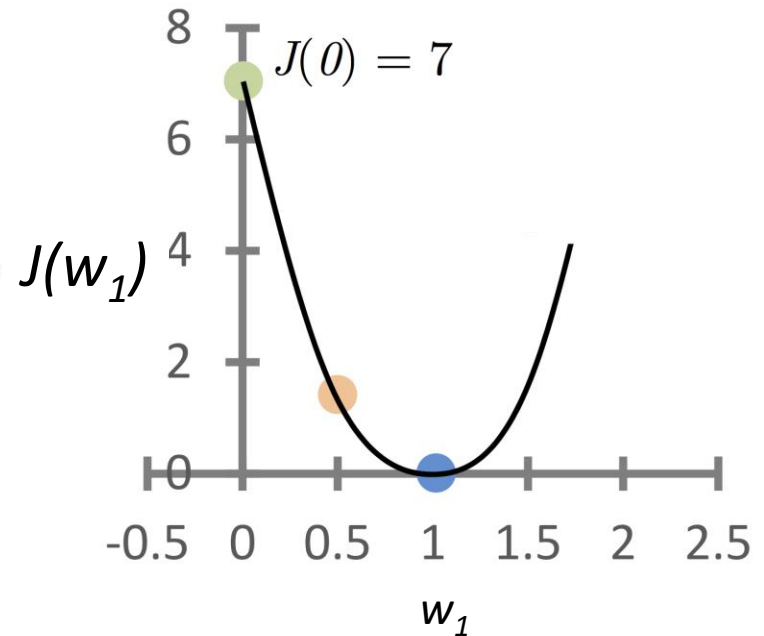
# Cost Function (extra practice)

$$h_w(x) = w_1 x$$

(assume  $w_0=0$  for this example)



$$J(w_1)$$

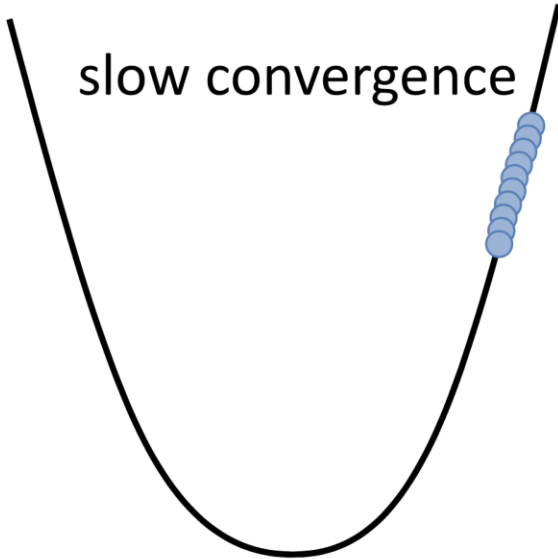


$$J(0.5) = \frac{1}{2} [(0.5 - 1)^2 + (1 - 2)^2 + (1.5 - 3)^2] = 1.75$$

# Choosing the step size alpha

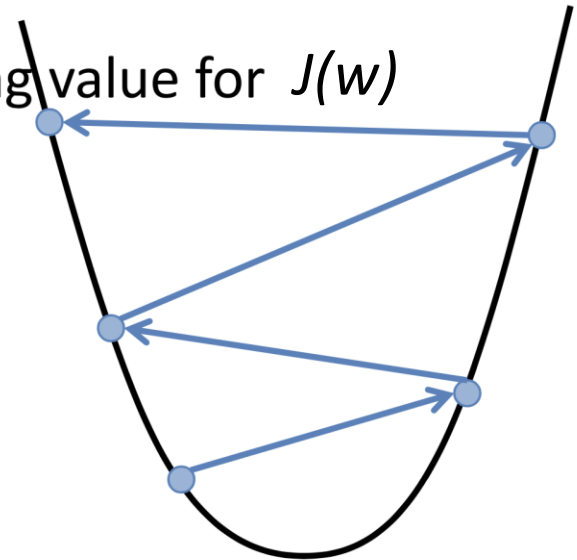
$\alpha$  too small

slow convergence



$\alpha$  too large

increasing value for  $J(w)$



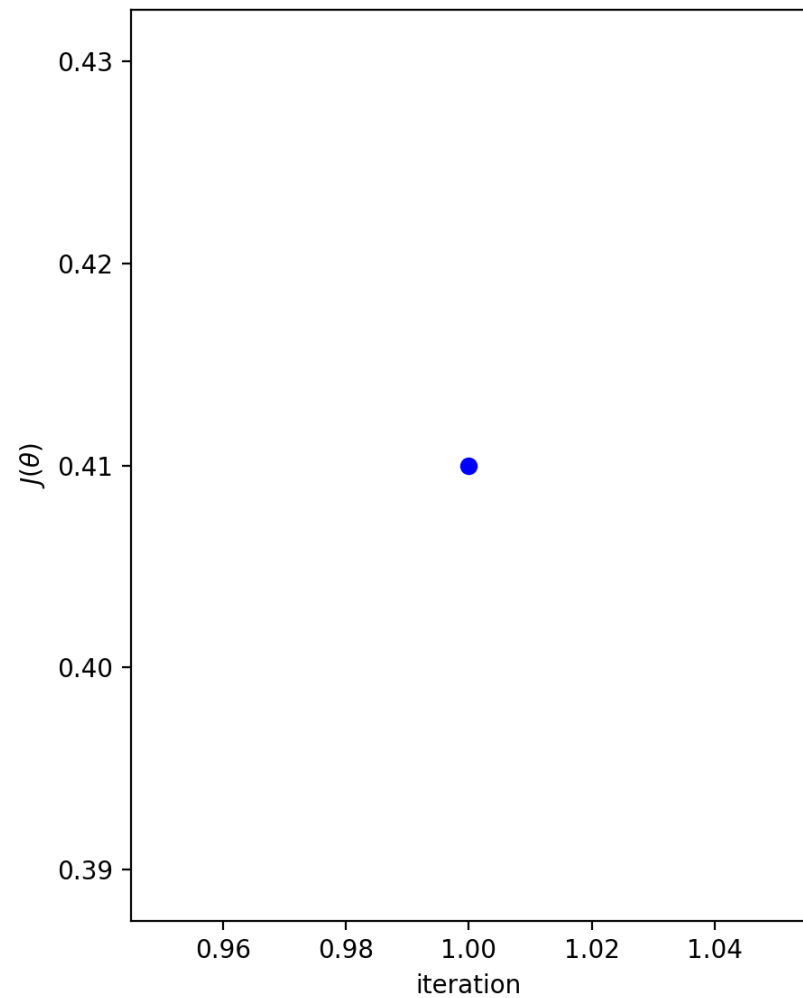
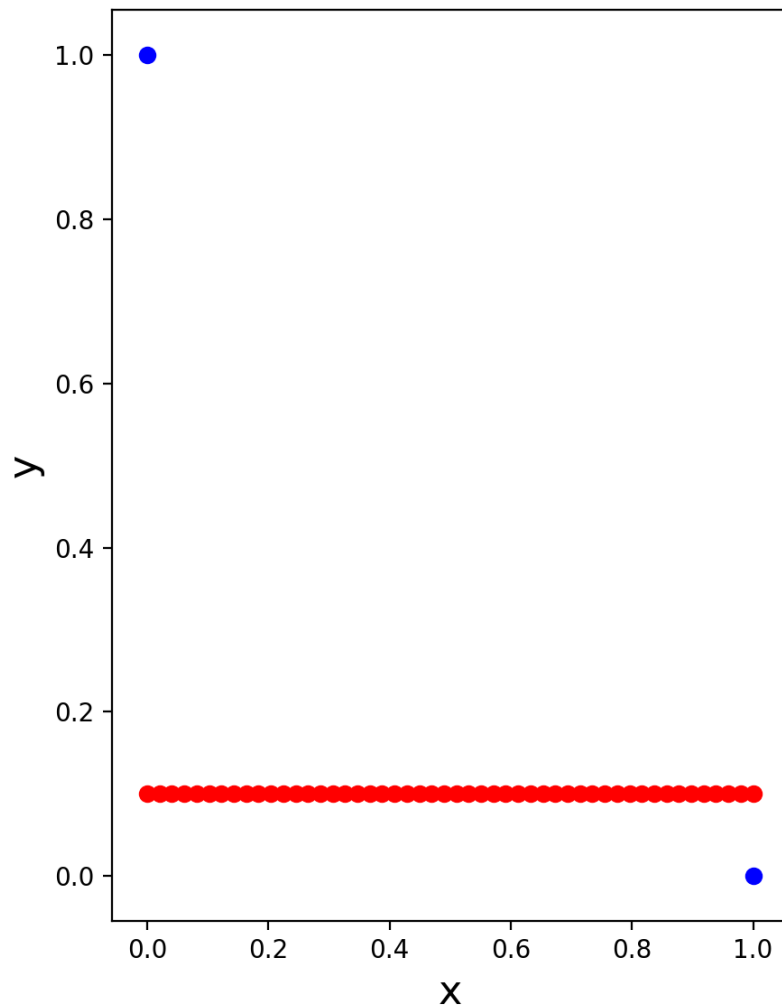
- may overshoot minimum
- may fail to converge (may even diverge)

# SGD with our small dataset from the handouts

Note: this is with the original order of the points

# Small example, iteration 1

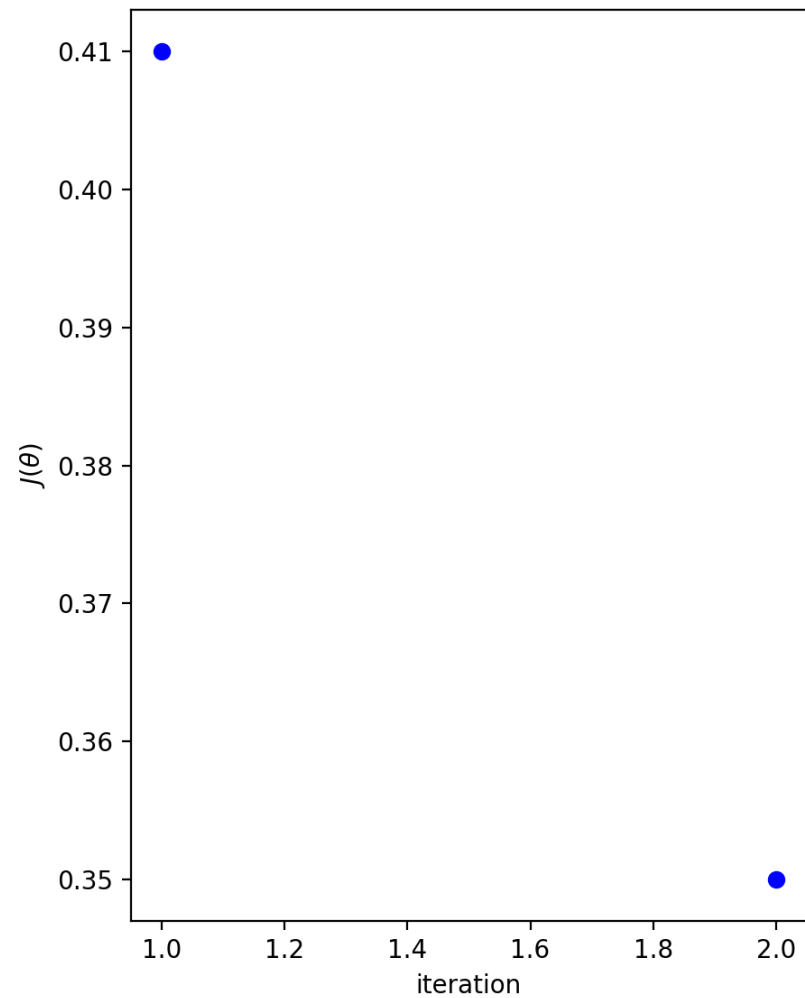
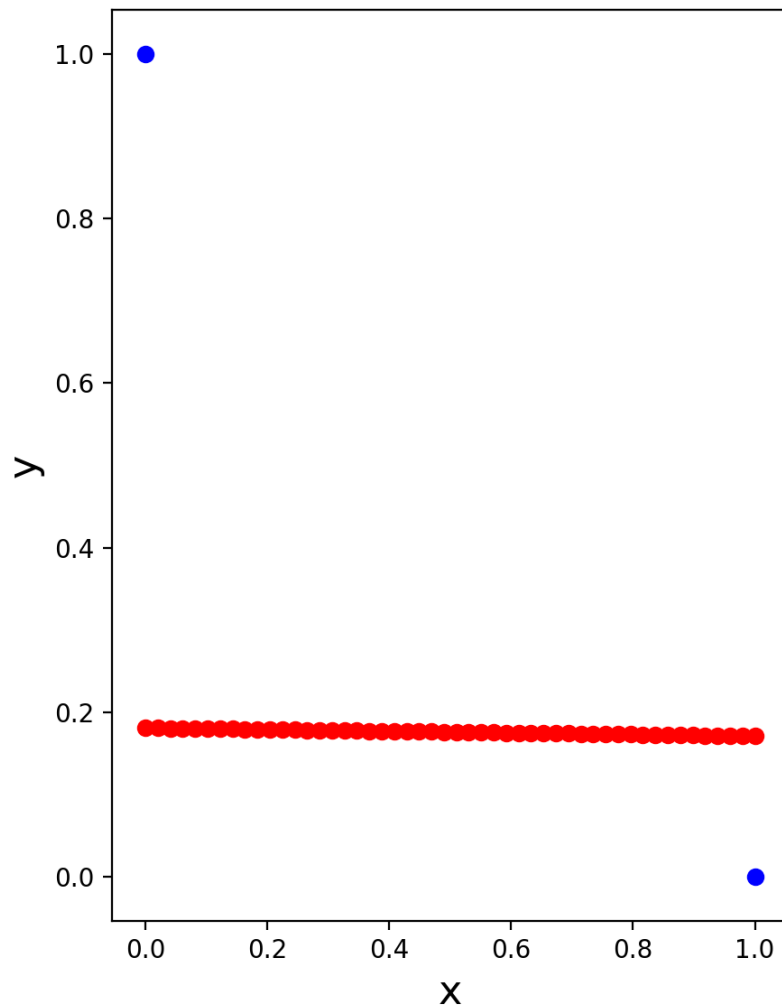
iteration: 1, cost: 0.410000





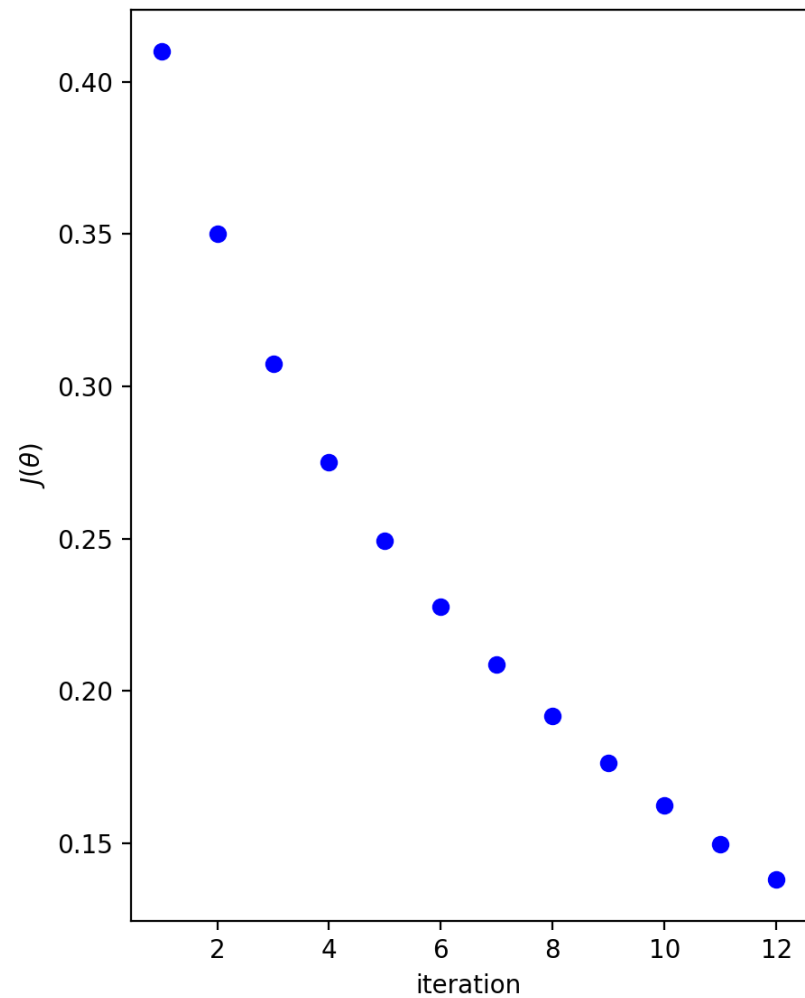
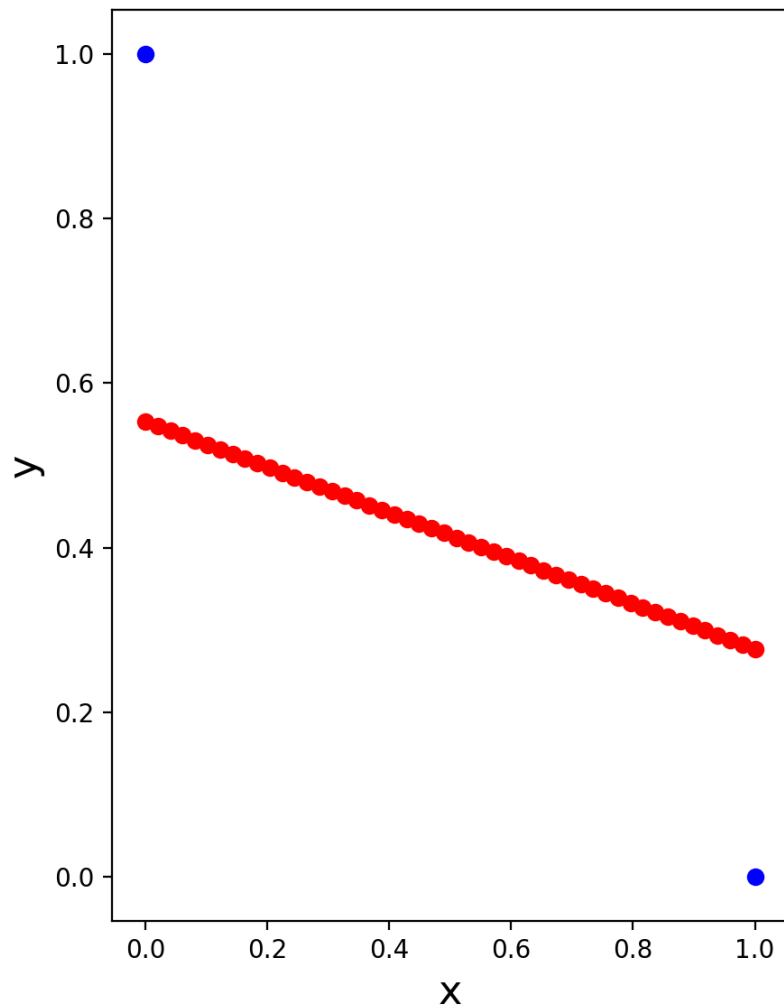
# Small example, iteration 2

iteration: 2, cost: 0.350001



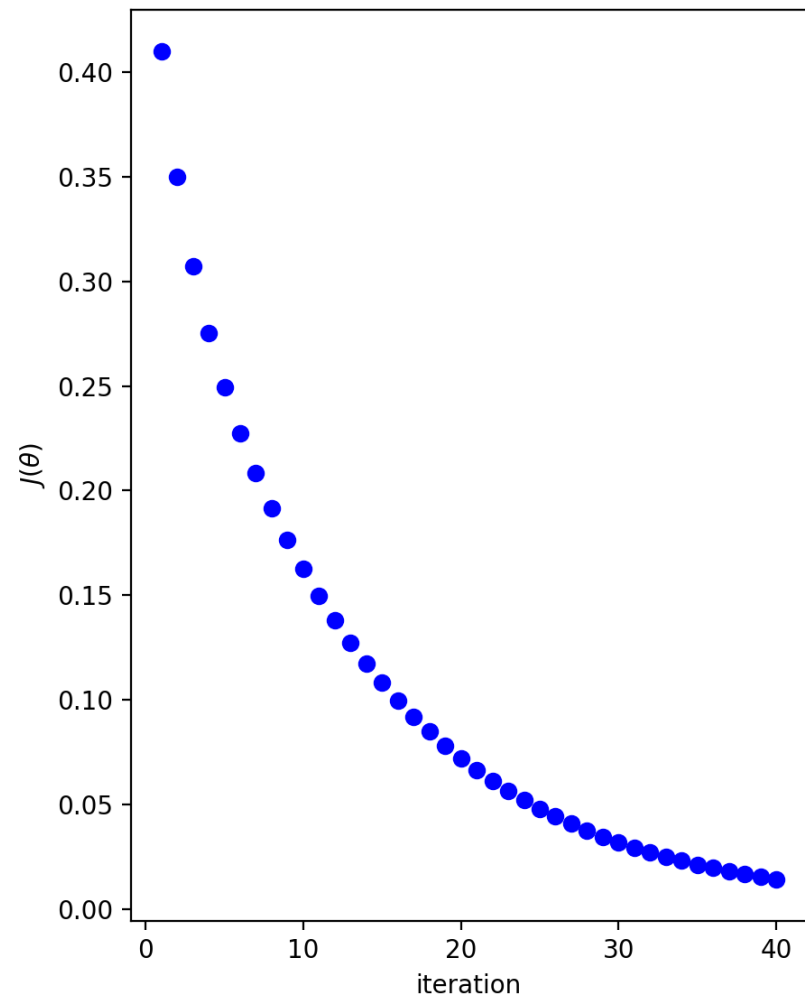
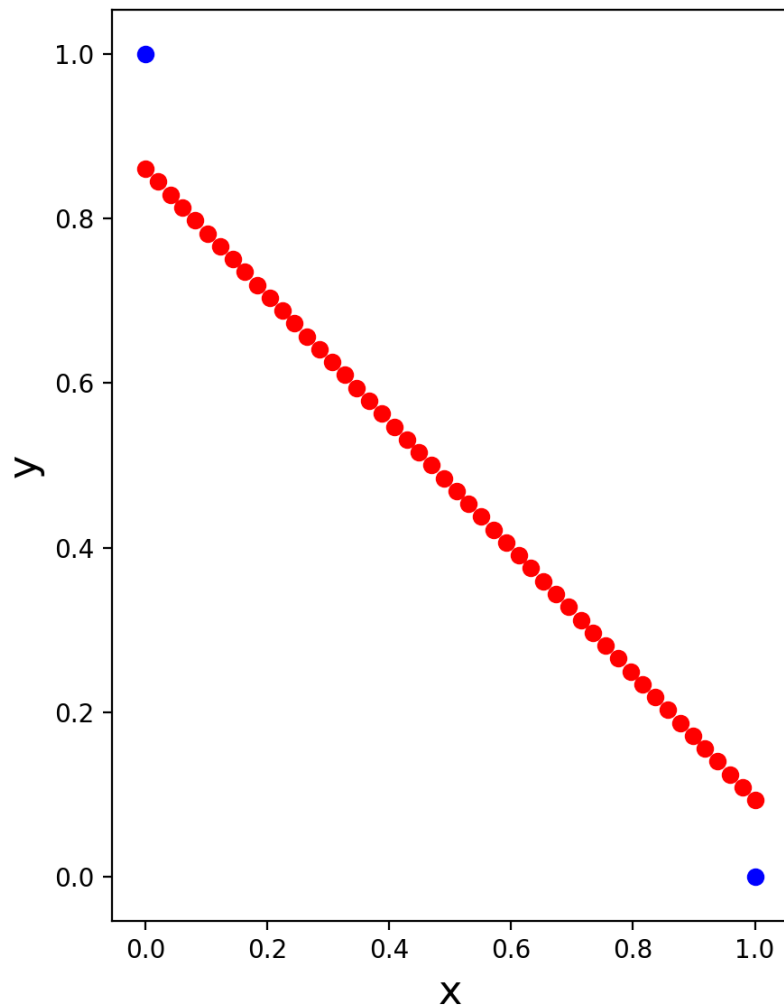
# Small example, iteration 12

iteration: 12, cost: 0.138047



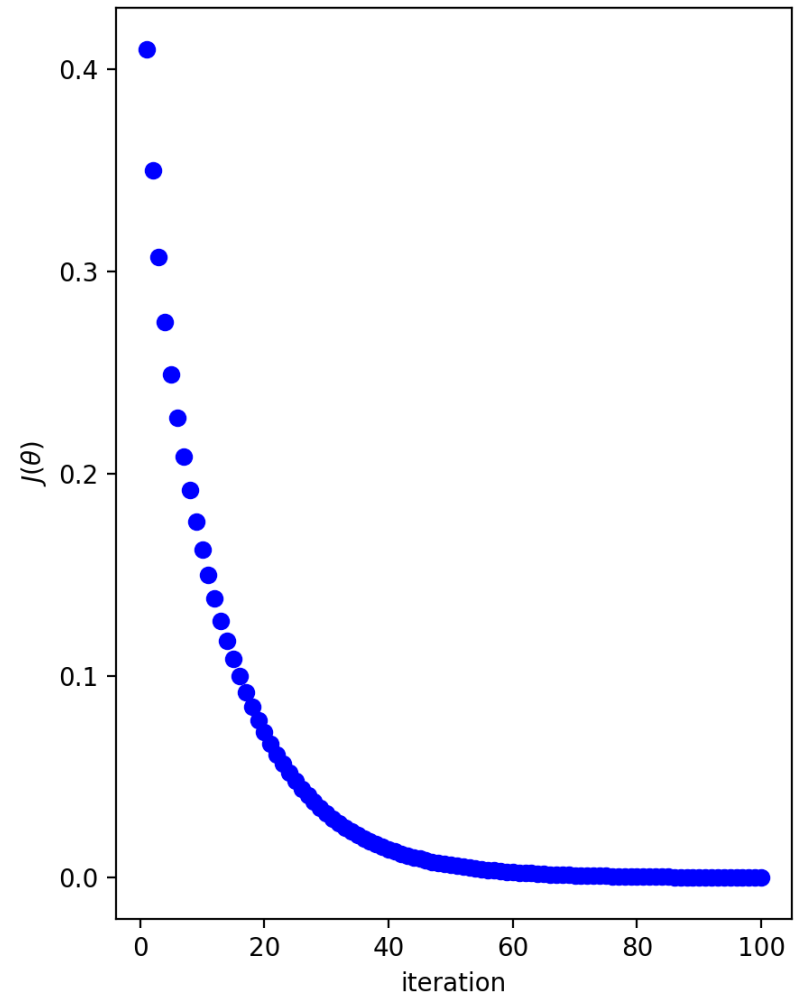
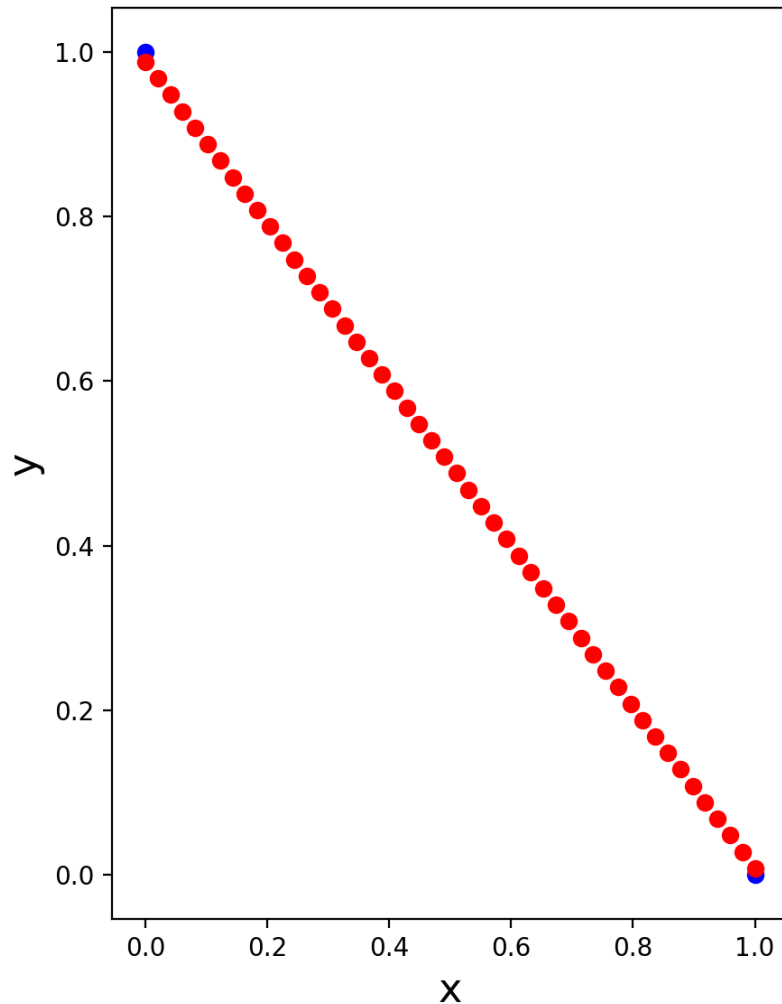
# Small example, iteration 40

iteration: 40, cost: 0.014064



# Small example, iteration 100

iteration: 100, cost: 0.000105





# Outline for today

- SGD (Stochastic Gradient Descent)
- Handout 6 (SGD solution example)
- Analytic vs. SGD (pros and cons)
- (if time) Polynomial regression

# Handout 6

## Linear Regression: SGD solution

*(find and work with a partner)*

In linear regression, we seek to minimize the sum of squared errors between the actual response and our prediction. We often call this RSS (residual sum of squares) or SSE (sum of squared errors). As an objective function, we often call it  $J$  and include a  $\frac{1}{2}$  in front to make the derivatives work out nicely.

$$J(\mathbf{w}) = \frac{1}{2} \sum_{i=1}^n (h_{\mathbf{w}}(\mathbf{x}_i) - y_i)^2$$

For linear regression in general, one iteration of stochastic gradient descent includes the following updates (usually with the data points shuffled):

for  $i = 1, 2, \dots, n$ :

$$\mathbf{w} \leftarrow \mathbf{w} - \alpha(\mathbf{w} \cdot \mathbf{x}_i - y_i)\mathbf{x}_i$$

We will begin with our same data from the previous two handouts:  $(x_1, y_1) = (0, 1)$  and  $(x_2, y_2) = (1, 0)$ , except we will reverse the order of the points to make the progress of gradient descent a bit clearer. So in this case our matrix/vector formulation is:

$$\mathbf{X} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

# Handout 6

$$\mathbf{X} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

1. Assuming  $\alpha = 0.1$  and our initial values are  $w_0 = 0$  and  $w_1 = 0$ , what are  $w_0$  and  $w_1$  after the just the first data point is used to update the gradient?

$$\vec{w} \leftarrow \begin{bmatrix} 0 \\ 0 \end{bmatrix} - 0.1 \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 0 \end{bmatrix} - 1 \right) \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\vec{w} \leftarrow \begin{bmatrix} 0 \\ 0 \end{bmatrix} + 0.1 \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\vec{w} \leftarrow \begin{bmatrix} 0.1 \\ 0 \end{bmatrix}$$

2. What are  $w_0$  and  $w_1$  after the second data point is used? Si

# Handout 6

$$X = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \quad y = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

1. Assuming  $\alpha = 0.1$  and our initial values are  $w_0 = 0$  and  $w_1 = 0$ , what are  $w_0$  and  $w_1$  after the just the first data point is used to update the gradient?

$$\vec{w} \leftarrow \begin{bmatrix} 0 \\ 0 \end{bmatrix} - 0.1 \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 0 \end{bmatrix} - 1 \right) \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\vec{w} \leftarrow \begin{bmatrix} 0 \\ 0 \end{bmatrix} + 0.1 \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\vec{w} \leftarrow \begin{bmatrix} 0.1 \\ 0 \end{bmatrix}$$

2. What are  $w_0$  and  $w_1$  after the second data point is used? Since we only have two examples here, your result would be the weight vector after the first iteration of SGD.

$$\vec{w} \leftarrow \begin{bmatrix} 0.1 \\ 0 \end{bmatrix} - 0.1 \left( \begin{bmatrix} 0.1 \\ 0 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \end{bmatrix} - 0 \right) \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$\vec{w} \leftarrow \begin{bmatrix} 0.09 \\ -0.01 \end{bmatrix}$$



# Handout 6

$$X = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \quad y = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

1. Assuming  $\alpha = 0.1$  and our initial values are  $w_0 = 0$  and  $w_1 = 0$ , what are  $w_0$  and  $w_1$  after the just the first data point is used to update the gradient?

$$\vec{w} \leftarrow \begin{bmatrix} 0 \\ 0 \end{bmatrix} - 0.1 \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 0 \end{bmatrix} - 1 \right) \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\vec{w} \leftarrow \begin{bmatrix} 0 \\ 0 \end{bmatrix} + 0.1 \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\vec{w} \leftarrow \begin{bmatrix} 0.1 \\ 0 \end{bmatrix}$$

2. What are  $w_0$  and  $w_1$  after the second data point is used? Since we only have two examples here, your result would be the weight vector after the first iteration of SGD.

$$\vec{w} \leftarrow \begin{bmatrix} 0.1 \\ 0 \end{bmatrix} - 0.1 \left( \begin{bmatrix} 0.1 \\ 0 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \end{bmatrix} - 0 \right) \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$\vec{w} \leftarrow \begin{bmatrix} 0.09 \\ -0.01 \end{bmatrix}$$

3. What is the value of the objective function (cost) after this initial iteration?

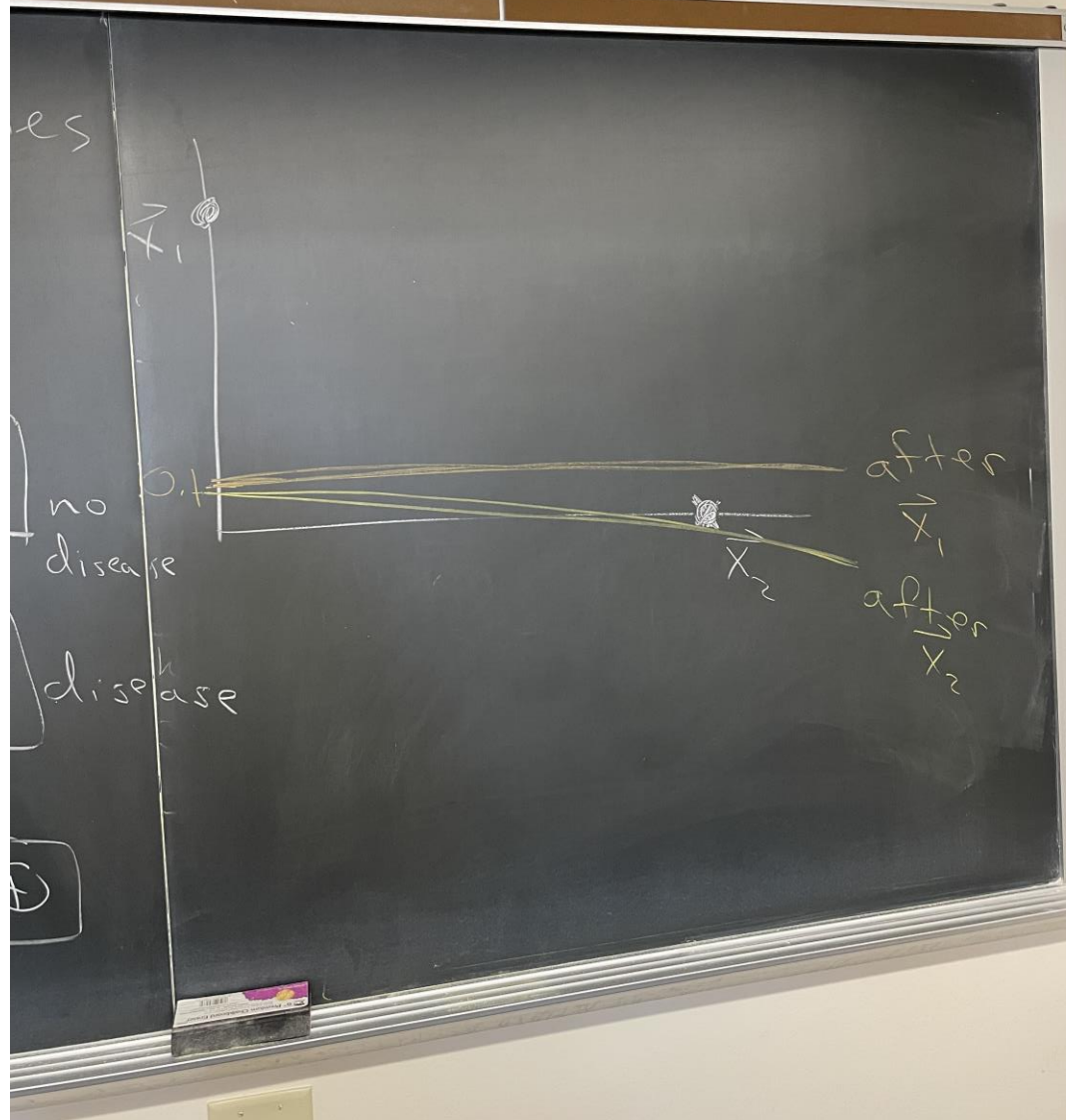
$$\hat{y} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 0.09 \\ -0.01 \end{bmatrix} = \begin{bmatrix} 0.09 \\ 0.08 \end{bmatrix}$$

$$J(\vec{w}) = \frac{1}{2} \begin{bmatrix} 0.09 \\ -0.08 \end{bmatrix} \cdot \begin{bmatrix} 0.09 \\ -0.08 \end{bmatrix}$$

$$\vec{y} - \hat{y} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} - \begin{bmatrix} 0.09 \\ 0.08 \end{bmatrix} = \begin{bmatrix} 0.91 \\ -0.08 \end{bmatrix}$$

$$J(\vec{w}) = 0.417$$

# Handout 6 (#4)



# Outline for today

- SGD (Stochastic Gradient Descent)
- Handout 6 (SGD solution example)
- **Analytic vs. SGD (pros and cons)**
- (if time) Polynomial regression

# Pros and Cons

(Analytic Solution)

## Gradient Descent

- requires multiple iterations
- need to choose  $\alpha$
- works well when  $p$  is large
- can support online learning

## Normal Equations

- non-iterative
- no need for  $\alpha$
- slow if  $p$  is large
  - matrix inversion is  $O(p^3)$

# Linear Regression Runtime

- $T$  = # iterations of SGD
- $n$  = # examples
- $p$  = # features

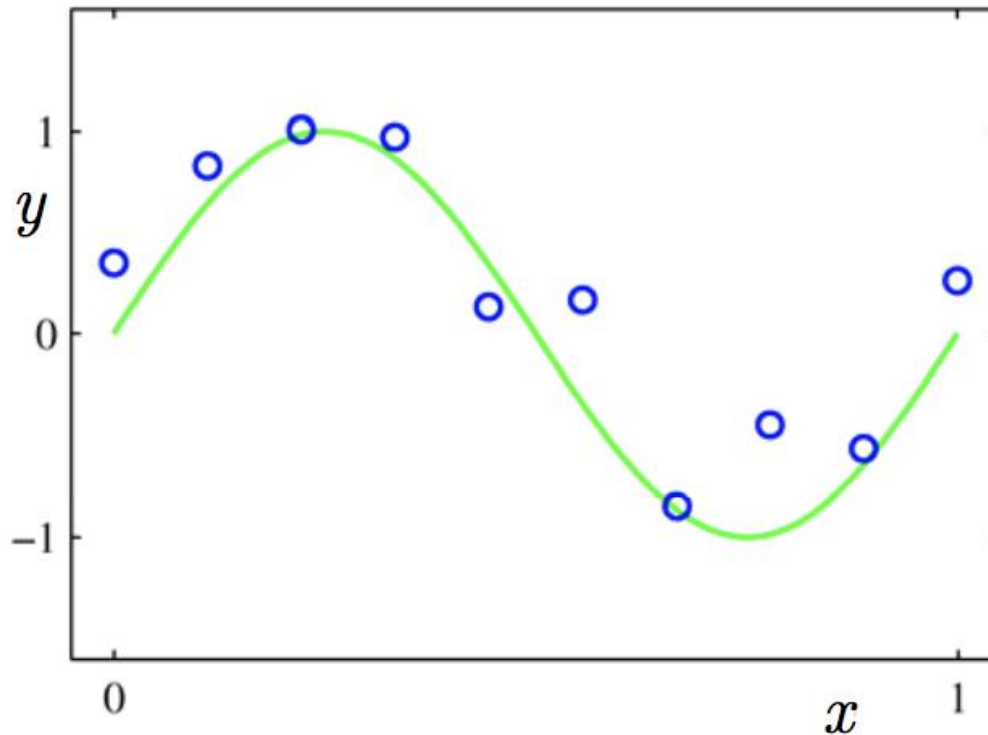
- 1) What is the runtime of SGD?
- 2) What is the runtime of the analytic solution?

# Outline for today

- SGD (Stochastic Gradient Descent)
- Handout 6 (SGD solution example)
- Analytic vs. SGD (pros and cons)
- (if time) Polynomial regression

# Polynomial Regression

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





# Polynomial Regression

$$\mathbf{X} = \begin{bmatrix} x_1^0 & x_1^1 & x_1^2 & \cdots & x_1^d \\ \vdots & & & & \\ x_n^0 & x_n^1 & x_n^2 & \cdots & x_n^d \end{bmatrix}$$