# CS 260: Foundations of Data Science

Prof. Sara Mathieson

Fall 2021

HAVERFORD
COLLEGE

# Admin

- **Lab 3** due Tuesday night
  - Pair programming option
- **Note-taker**: Matthew

# Outline for September 16

- Handout 5 (analytic solution example)

- SGD (Stochastic Gradient Descent)

- Handout 6 (SGD solution example)

- Analytic vs. SGD (pros and cons)

# Outline for September 16

- Handout 5 (analytic solution example)
- SGD (Stochastic Gradient Descent)
- Handout 6 (SGD solution example)
- Analytic vs. SGD (pros and cons)

PLEASE LEAVE COMPUTERS ON

Do not erase

$\boxed{\text{Handout 5}}$

- **transpose**: switch rows & cols

$$A = \begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix}_{2\times 3}, \quad A^T = \begin{bmatrix} a & d \\ b & e \\ c & f \end{bmatrix}_{3\times 2}$$

$$\left.\begin{array}{l} A^T A \Rightarrow 3\times 3 \\ A A^T \Rightarrow 2\times 2 \end{array}\right\} \text{square!}$$

B must be square

$$\boxed{BB^{-1} = B^{-1}B}$$

- **inverse**:

$$B^{-1}B = I \leftarrow \text{identity matrix ("1")}$$

2x2

$$B = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

$$B^{-1} = \frac{1}{ad-bc}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

$$B^{-1}B = \frac{1}{ad-bc}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix}\begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} ad-bc & \cancel{db-db} \to 0 \\ \cancel{-ac+ac} \to 0 & -cb+ad \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \star \text{ identity}$$

$$\frac{1}{ad-bc}$$

$$I_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

quare!

$$a\begin{bmatrix}3 & 5\\ -1 & 2\end{bmatrix} = \begin{bmatrix}3a & 5a\\ -a & 2a\end{bmatrix}$$

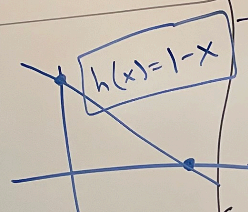
$$\cancel{(X^T)^{-1}X^T}\, X\,\vec{w} = \cancel{(X^T)^{-1}X^T\vec{y}}$$

solving for $\vec{w}$

$X^T$ is not square

$\Rightarrow$ not invertible!

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




$$(X^TX)^{-1} = \left(\begin{bmatrix}1 & 1\\ 1 & 0\end{bmatrix}\begin{bmatrix}1 & 1\\ 1 & 0\end{bmatrix}\right)^{-1}$$

$$= \begin{bmatrix}2 & 1\\ 1 & 1\end{bmatrix}^{-1} = \frac{1}{2-1}\begin{bmatrix}1 & -1\\ -1 & 2\end{bmatrix}$$

$$\cancel{(X^TX)^{-1}X^TX}\,\vec{w} = (X^TX)^{-1}X^T\vec{y}$$

$$\boxed{\underbrace{\hat{\vec{w}}}_{(p+1)\times 1} = \underbrace{(X^TX)^{-1}}_{\text{Variance of } X} \underbrace{X^T\vec{y}}_{\text{cov of } X \,\&\, \vec{y}}} \quad \star$$

$(p+1)\times n$

$(p+1)\times n$

$n\times(p+1)$

$n\times 1$

$$\vec{w} = \begin{bmatrix}1 & -1\\ -1 & 2\end{bmatrix}\underbrace{\begin{bmatrix}1 & 1\\ 1 & 0\end{bmatrix}\begin{bmatrix}0\\ 1\end{bmatrix}} = \begin{bmatrix}1 & -1\\ -1 & 2\end{bmatrix}\begin{bmatrix}1\\ 0\end{bmatrix} = \boxed{\begin{bmatrix}1\\ -1\end{bmatrix} = \hat{w}}$$

# Outline for September 16

- Handout 5 (analytic solution example)

- SGD (Stochastic Gradient Descent)

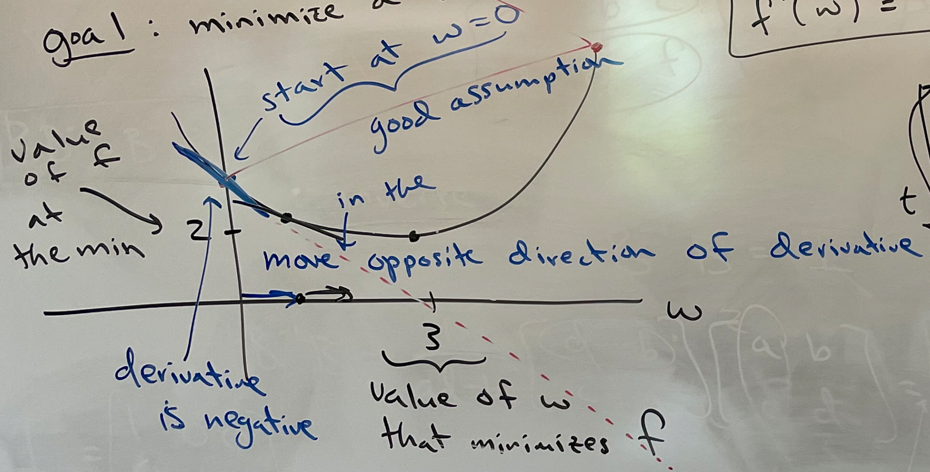
- Handout 6 (SGD solution example)

- Analytic vs. SGD (pros and cons)

PLEASE LEAVE COMPUTERS ON

# Gradient Descent

goal: minimize a function




$$f(w) = (w-3)^2 + 2$$

$$f(w) = w^2 - 6w + 11$$

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

$$\boxed{\underbrace{w}_{\text{update}} \leftarrow \underbrace{w}_{\text{curr } w} - \underbrace{\alpha}_{\text{step size}} f'(w)}$$

gradient descent

$\boxed{\alpha = 0.1}$

t (iteration)

① $w \leftarrow 0 - 0.1(2 \cdot 0 - 6)$

$\boxed{w \leftarrow 0.6}$

② $w \leftarrow 0.6 - 0.1(2(0.6) - 6)$

$\boxed{w \leftarrow 1.08}$

if $\left| f(w^t) - f(w^{t-1}) \right| < \varepsilon \longrightarrow 1\times10^{-8}$ # converged

$\Rightarrow$ stop

$$\boxed{\alpha = \frac{1}{t}}$$

Way of changing alpha adaptively

PLEASE LEAVE COMPUTERS ON

# Gradient Descent for Linear Regression

$$\vec{w}\cdot\vec{x}_i = w_0 + w_1 x_{i1} + \dots\ w_p x_{ip}$$

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

want to minimize

derivative of "inside"

wrt one weight

$$\frac{\partial J}{\partial w_j} = \sum_{i=1}^{n}\left(\vec{w}\cdot\vec{x}_i - y_i\right) x_{ij}$$

very slow when $n$ is large

$$\underbrace{\nabla J}_{\text{gradient}} = \begin{bmatrix} \frac{\partial J}{\partial w_0} \\ \frac{\partial J}{\partial w_1} \\ \vdots \\ \frac{\partial J}{\partial w_p} \end{bmatrix}$$

derivative

wrt one data point

$$\frac{\partial J_{x_i}}{\partial w_j} = \left(\vec{w}\cdot\vec{x}_i - y_i\right) x_{ij}$$

same for all $w_j$

$$\Rightarrow \boxed{\nabla J_{x_i} = \left(\vec{w}\cdot\vec{x}_i - y_i\right)\vec{x}_i}$$

$$\vec{x}_i = \begin{bmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{ij} \\ \vdots \\ x_{ip} \end{bmatrix}$$

# Stochastic Gradient Descent Algorithm

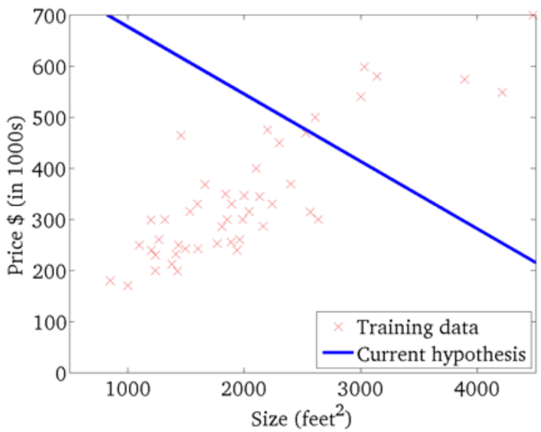
for $\underset{\text{iteration}}{\overset{\text{epoch}}{}}$ $t$ ... until convergence

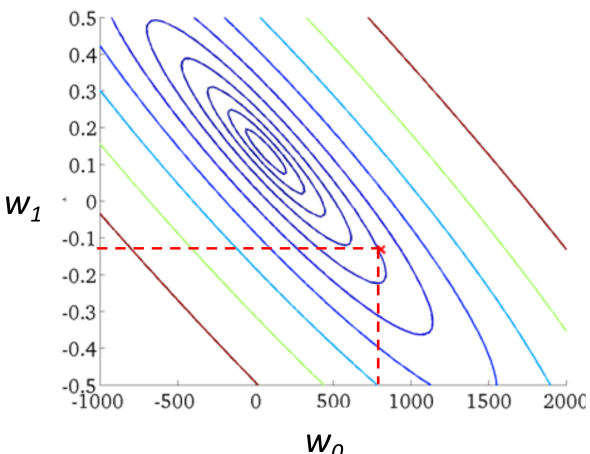
for $i = \overbrace{1, 2, 3 \ldots n}^{\text{usually shuffle}}$:

$$\vec{w} \leftarrow \vec{w} - \alpha \overbrace{\left( \vec{w} \cdot \vec{x}_i - y_i \right) \vec{x}_i}^{\text{gradient (of all weights) wrt one data point } \vec{x}_i}$$

check for convergence: $\left| J(\vec{w}^{t}) - J(\vec{w}^{t-1}) \right| < \varepsilon$

# Linear Model and Cost Function J

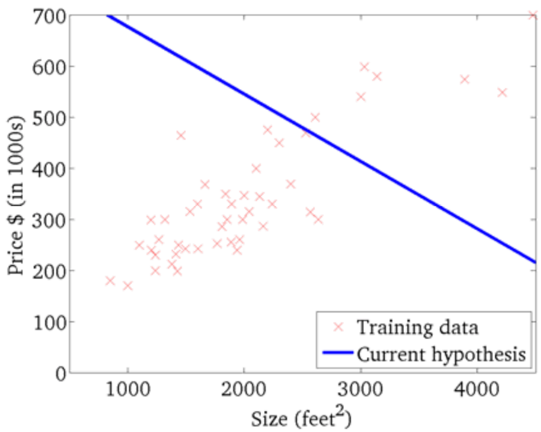


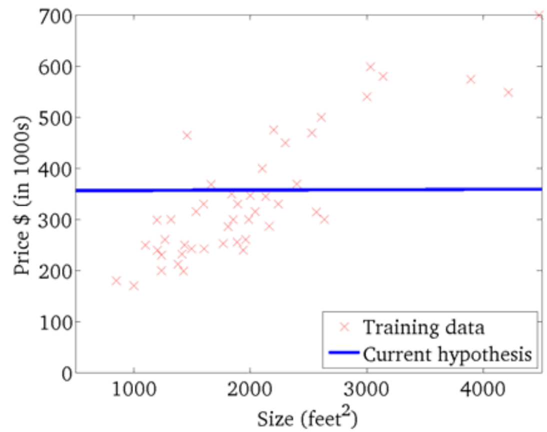



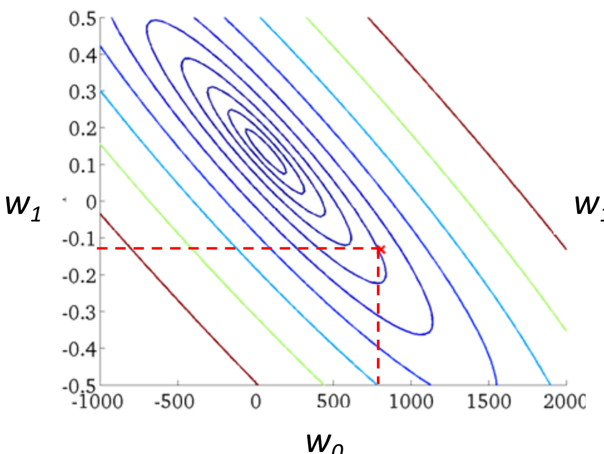

contour map

Slide: modified from Jessica Wu
Based on slide by Andrew Ng

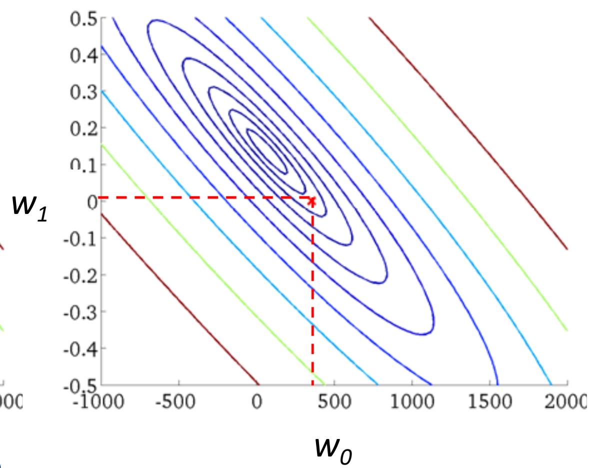
# Linear Model and Cost Function J



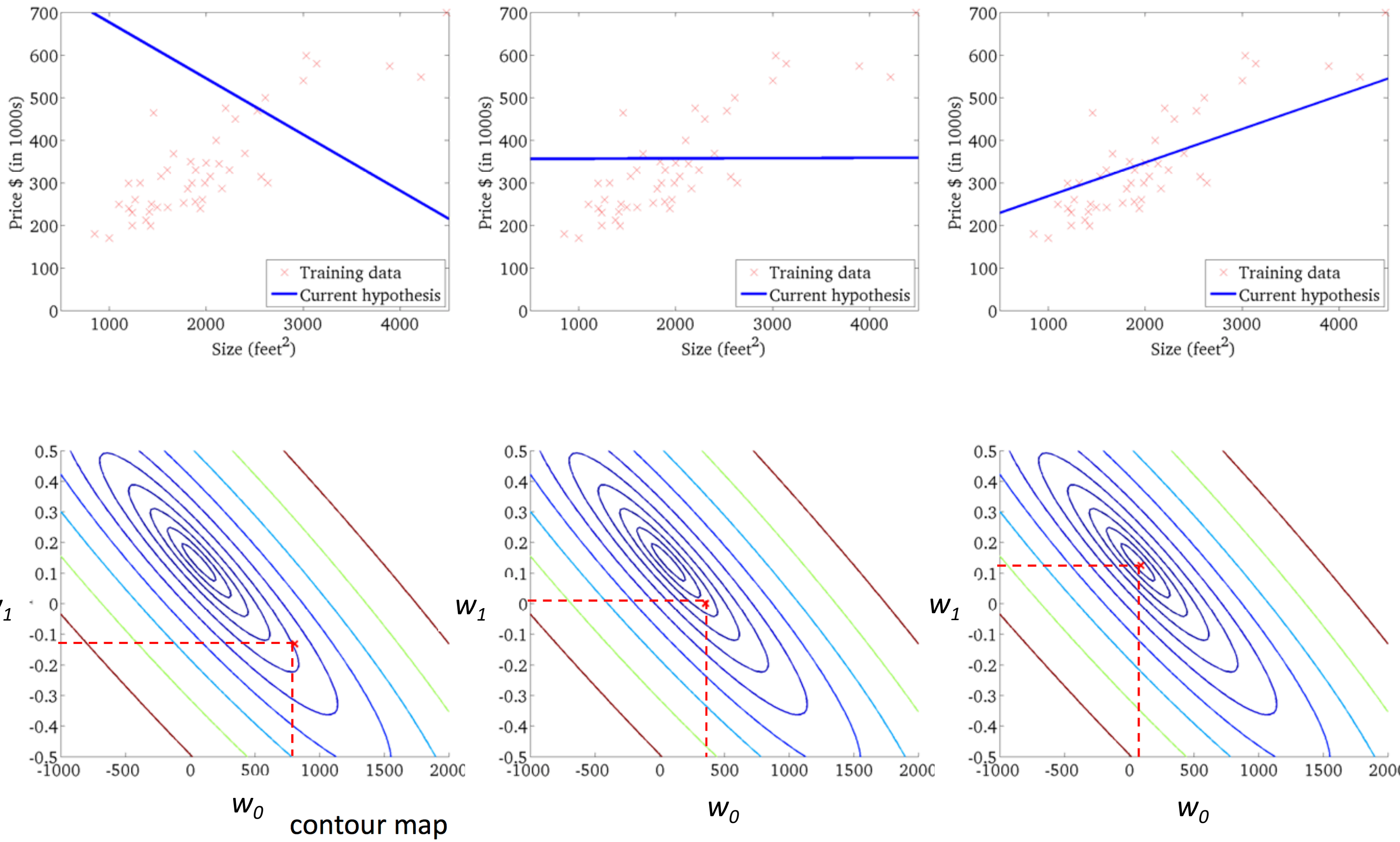




contour map



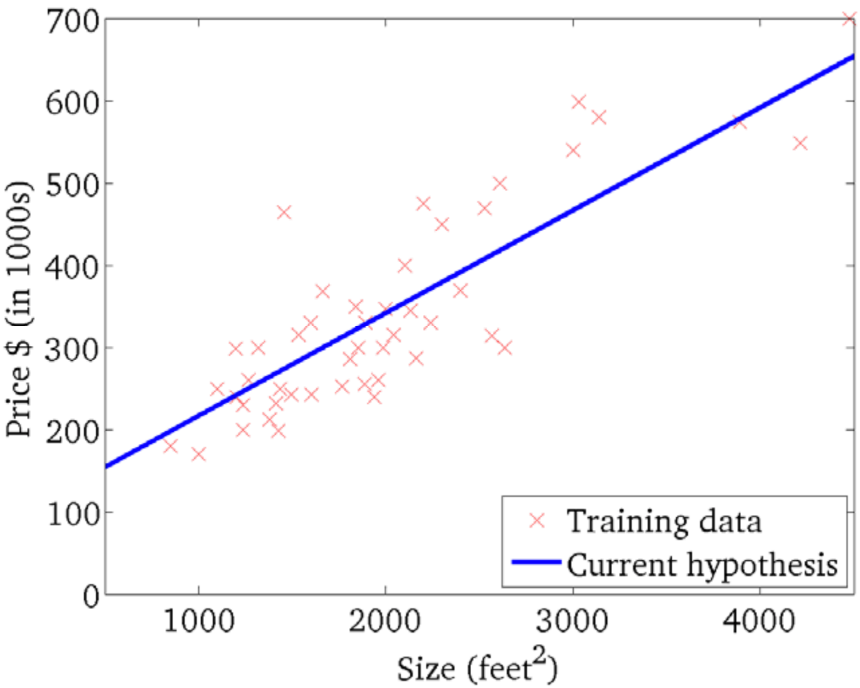

Slide: modified from Jessica Wu
Based on slide by Andrew Ng

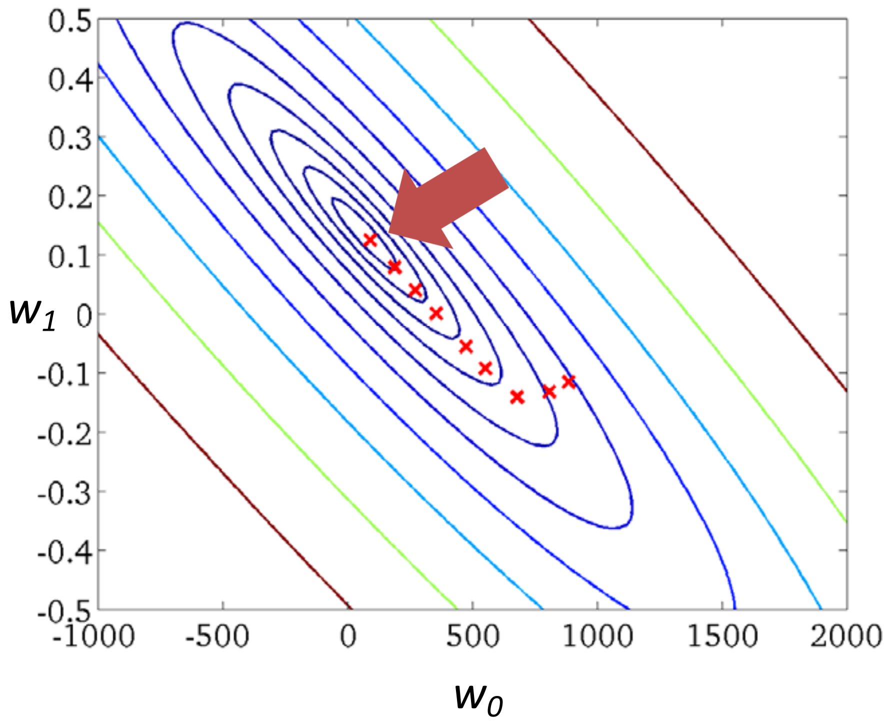
# Linear Model and Cost Function J




Slide: modified from Jessica Wu
Based on slide by Andrew Ng

# Gradient Descent: walking toward the minimum

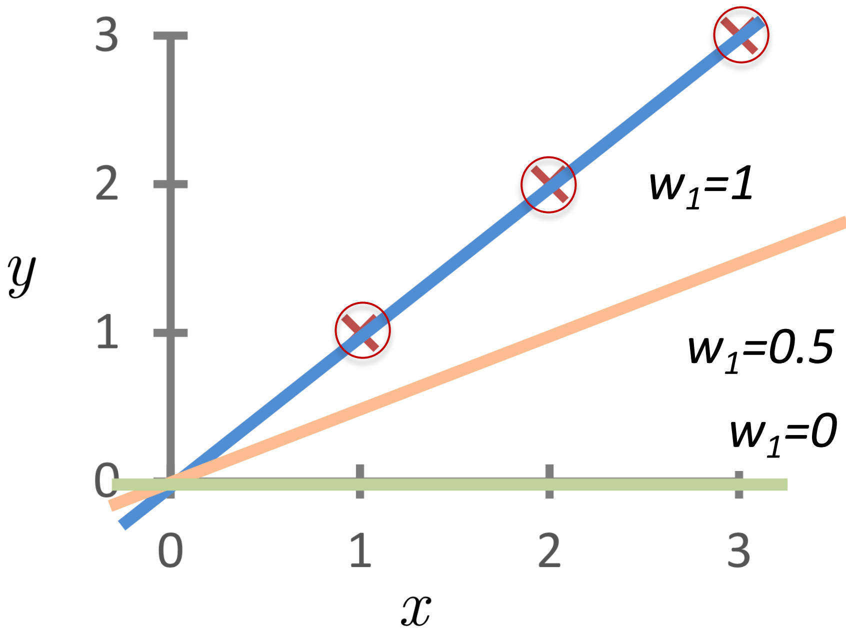


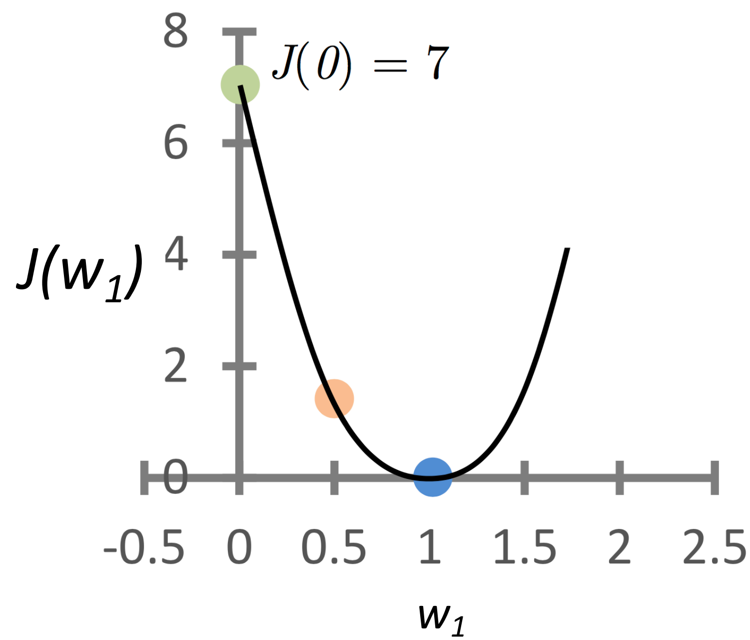
# Cost Function (extra practice)

$h_w(x) = w_1 x$

(assume $w_0 = 0$ for this example)




$J(w_1)$

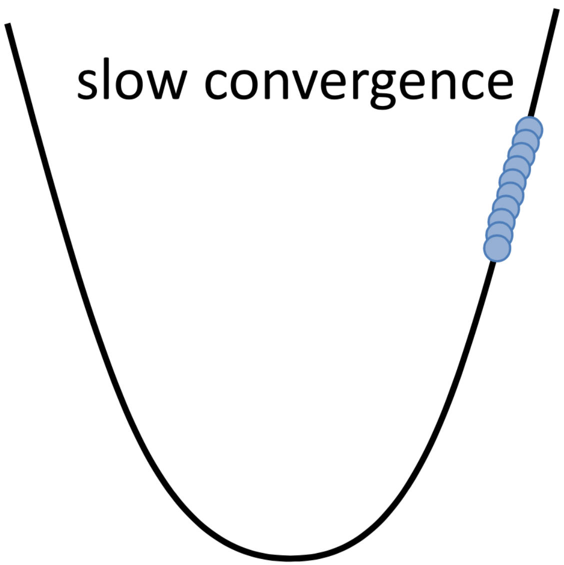



$$J(0.5) = \frac{1}{2}\left[(0.5-1)^2 + (1-2)^2 + (1.5-3)^2\right] = 1.75$$

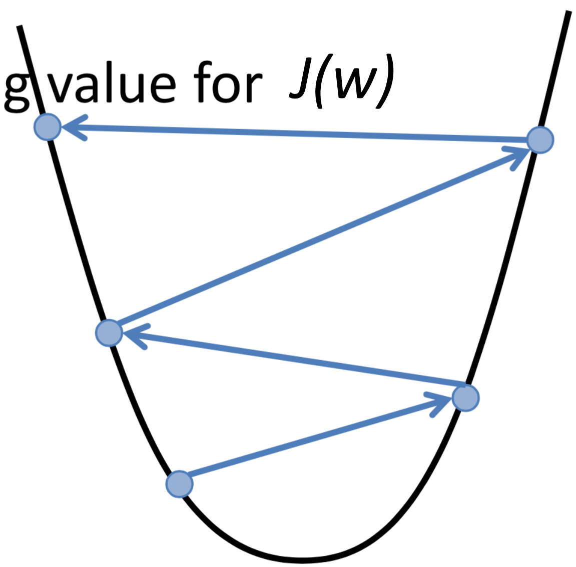
Slide: modified from Jessica Wu
Based on slide by Eric Eaton
[Example by Andrew Ng]

# Choosing the step size alpha

## $\alpha$ too small




## $\alpha$ too large



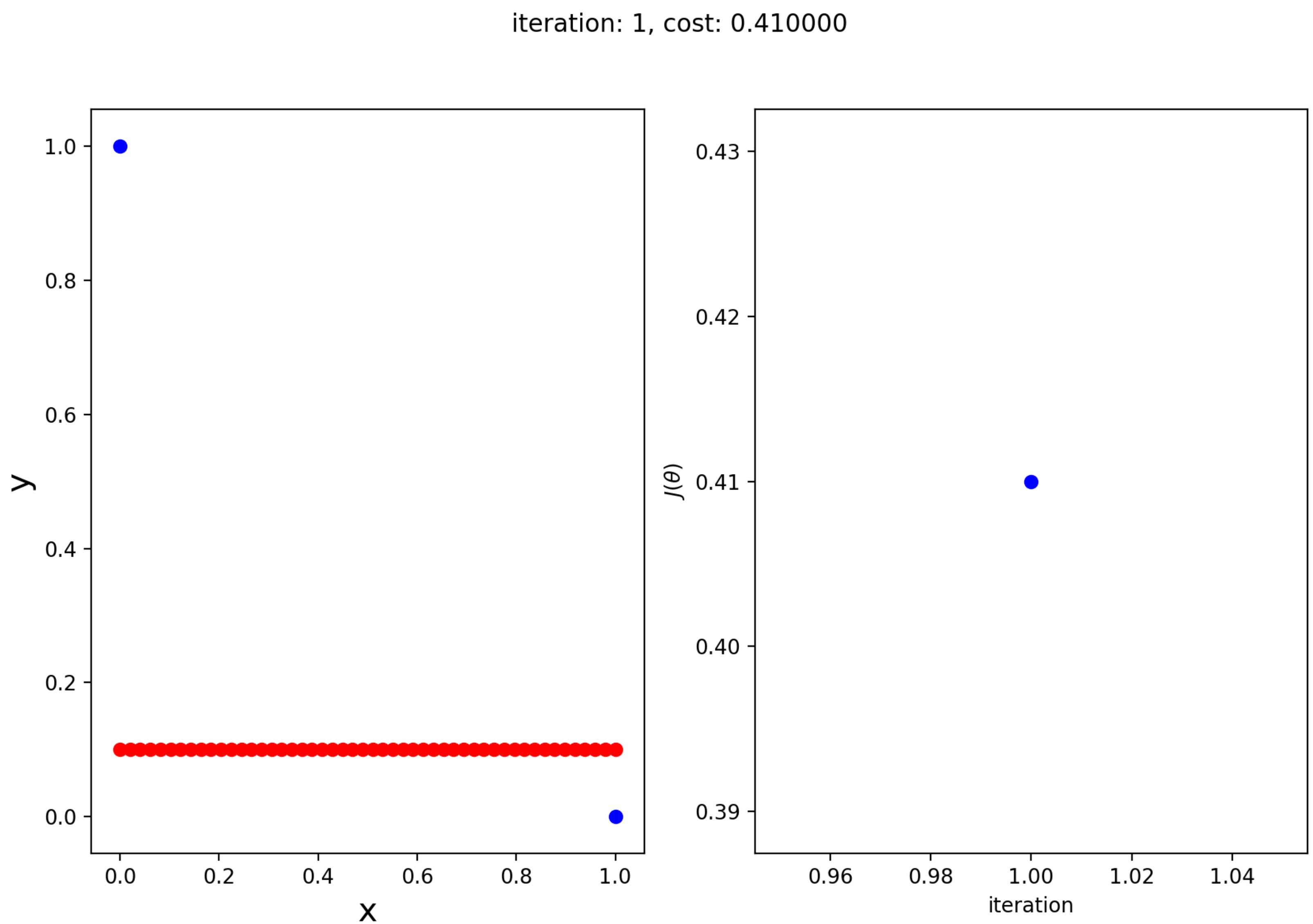

- may overshoot minimum
- may fail to converge (may even diverge)

Slide: modified from Jessica Wu
Based on slide by Eric Eaton
[Originally by Andrew Ng]

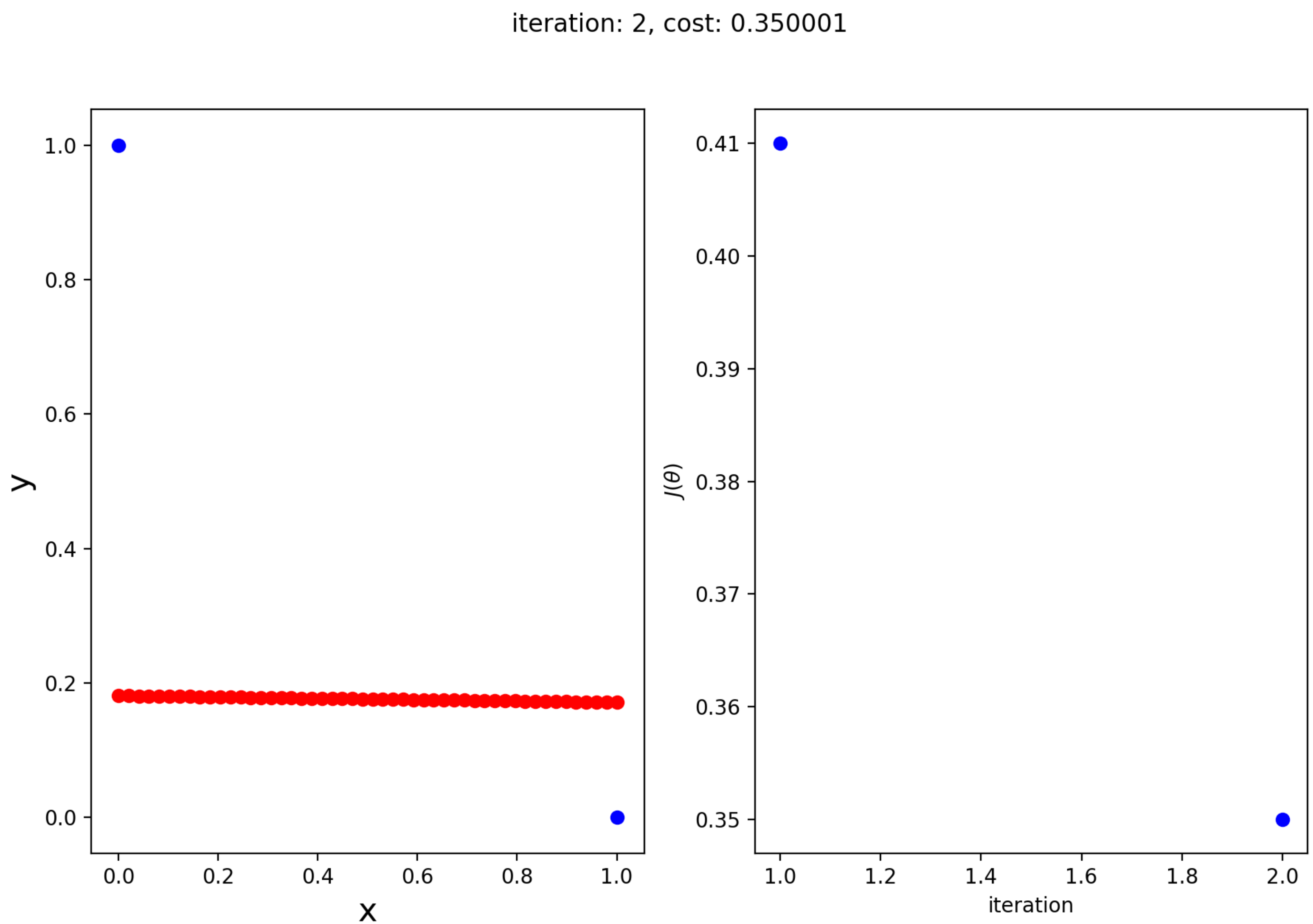
# SGD with our small dataset from the handouts

Note: this is with the original order of the points

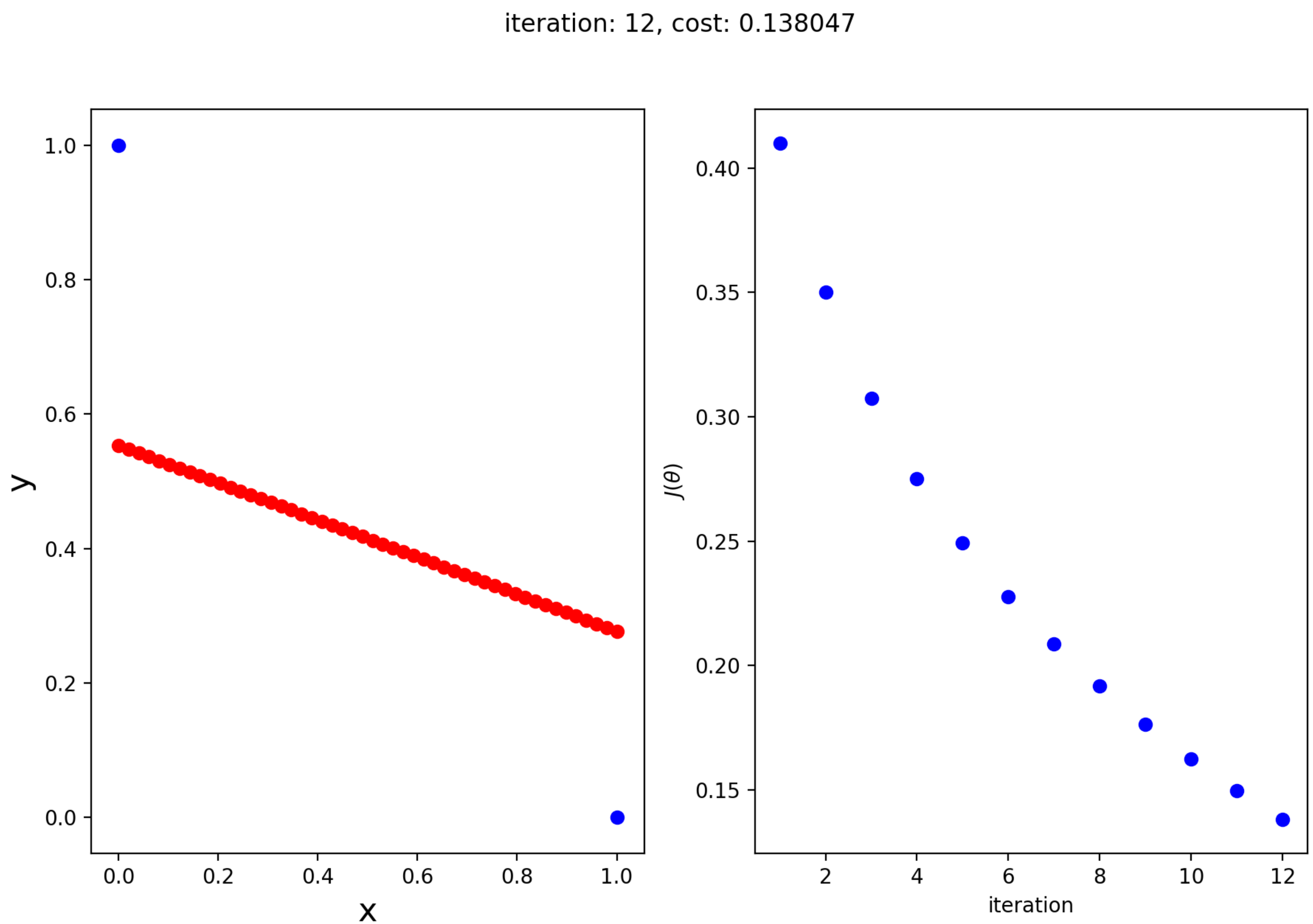
# Small example, iteration 1



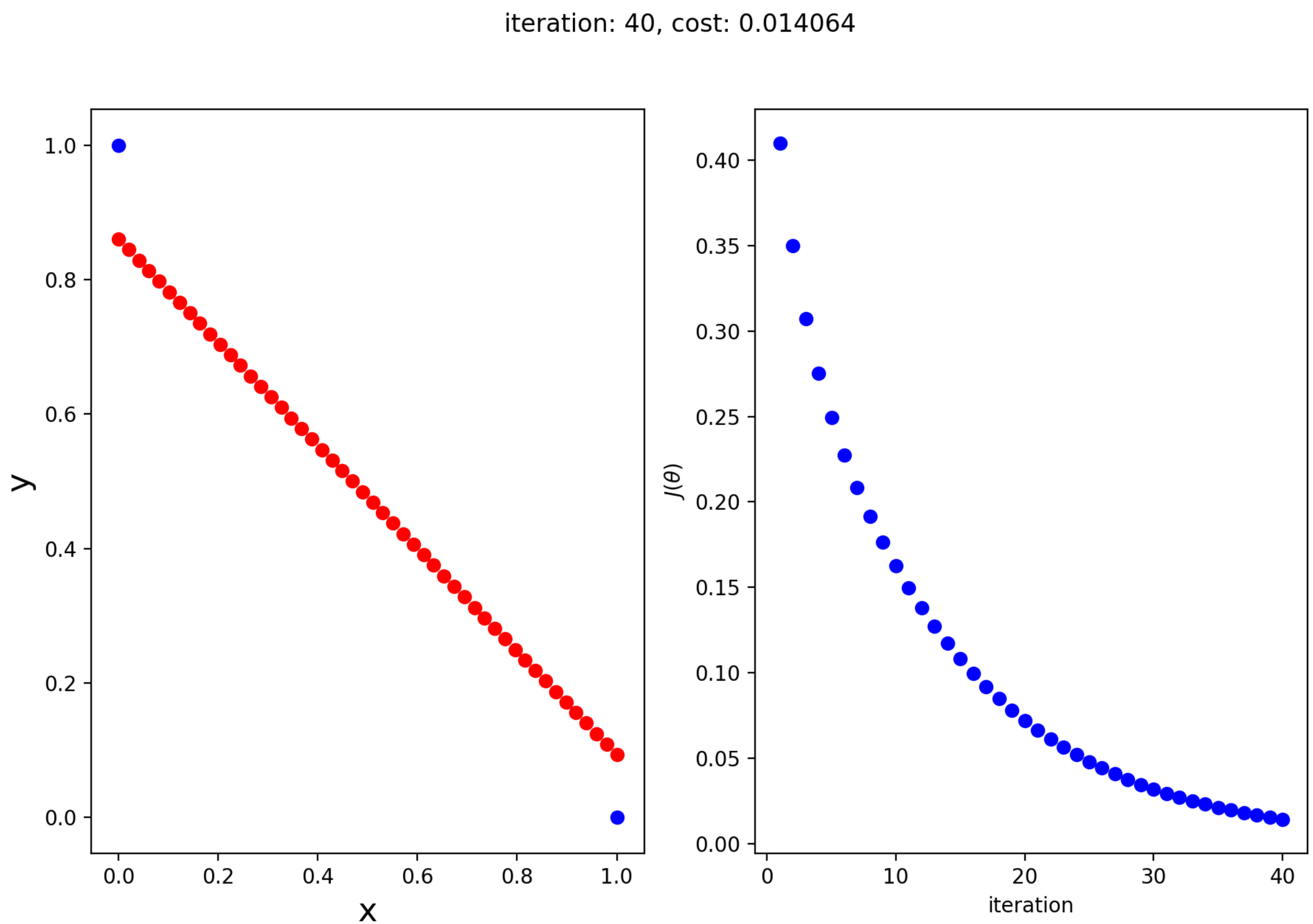
# Small example, iteration 2



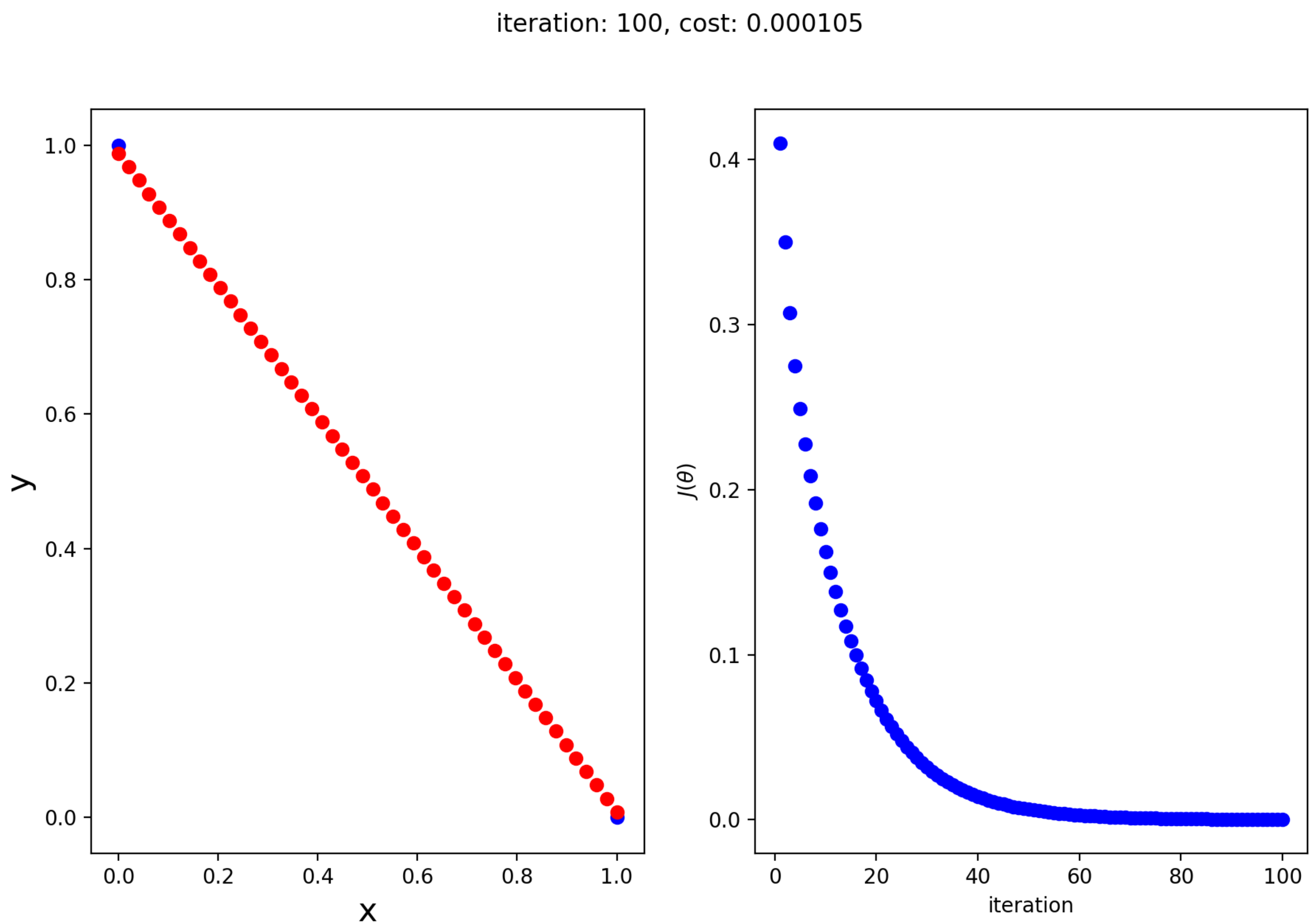
# Small example, iteration 12



# Small example, iteration 40



# Small example, iteration 100



# Outline for September 16

- Handout 5 (analytic solution example)
- SGD (Stochastic Gradient Descent)
- Handout 6 (SGD solution example)
- Analytic vs. SGD (pros and cons)

Computer Science

r optparse

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

1st point

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

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix} + 0.1\begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

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

2nd point

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

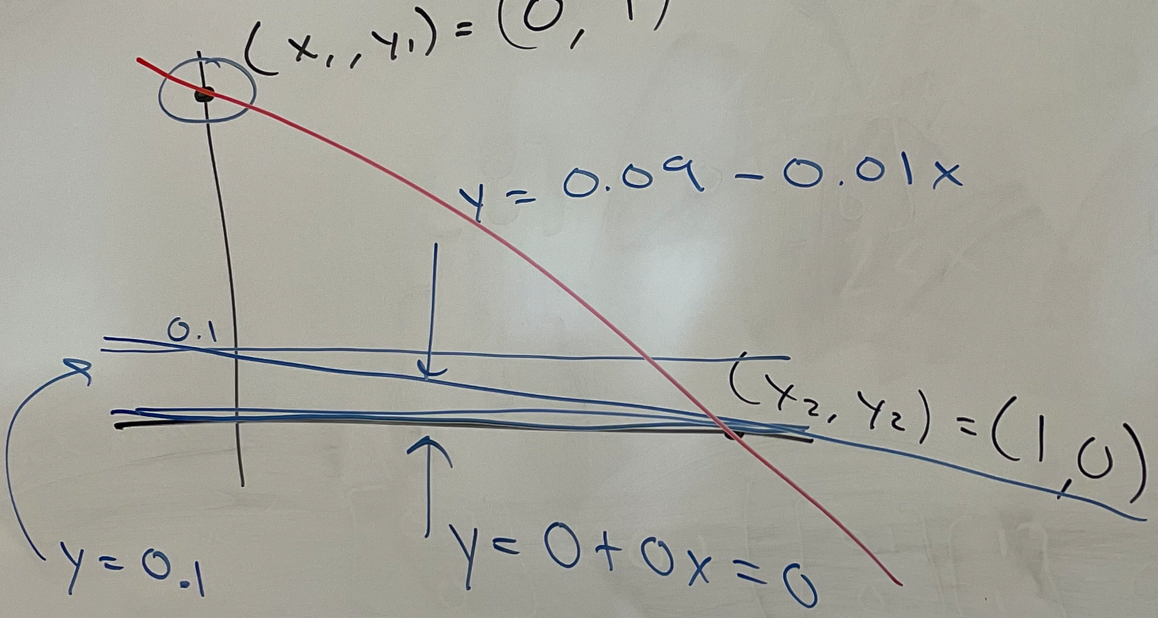
$$\vec{w} \leftarrow \begin{bmatrix} 0.09 \\ -0.01 \end{bmatrix}$$

fake ones

$$X = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 \end{bmatrix} \rightarrow \vec{x}_1^{\,T} \qquad y = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\vec{x}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\vec{x}_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$



# Outline for September 16

- Handout 5 (analytic solution example)
- SGD (Stochastic Gradient Descent)
- Handout 6 (SGD solution example)
- Analytic vs. SGD (pros and cons)

# Pros and Cons

**Gradient Descent**

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

(Analytic Solution)

**Normal Equations**

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

Slide: modified from Jessica Wu
Based on slide by Eric Eaton

# Linear Regression Runtime

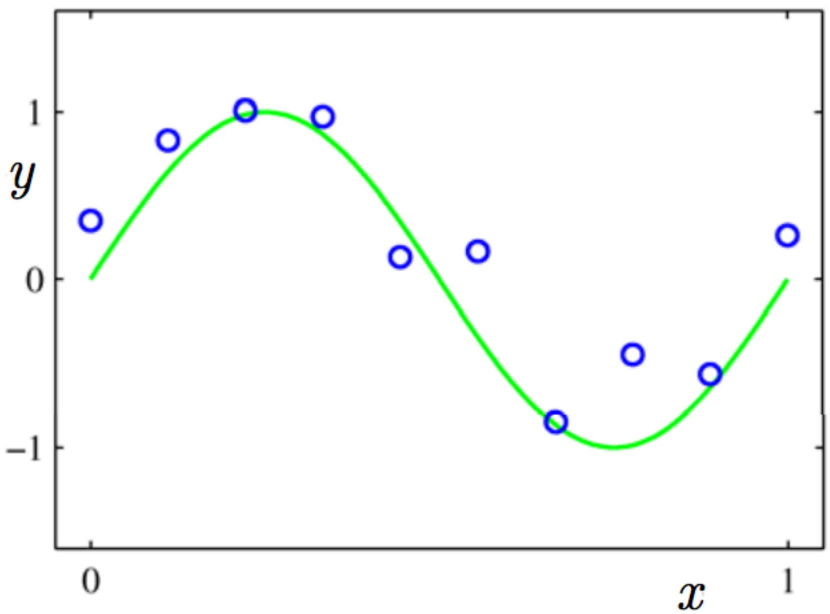
- T = # iterations of SGD
- n = # examples
- p = # features

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

Challenge outside of class!

# Extra topic: Polynomial Regression

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




Plot: Jessica Wu