


Copyright Notice

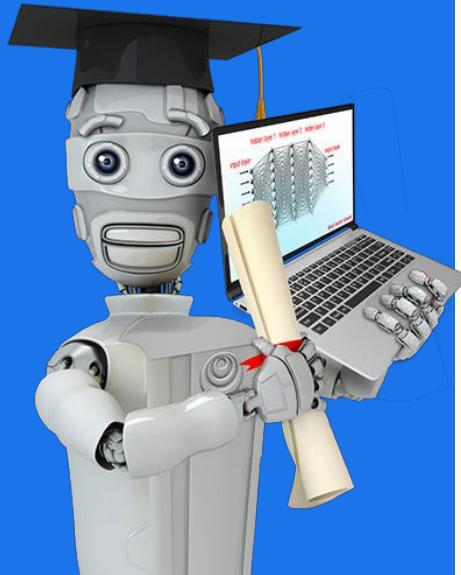
These slides are distributed under the Creative Commons License.

[DeepLearning.AI](#) makes these slides available for educational purposes. You may not use or distribute these slides for commercial purposes. You may make copies of these slides and use or distribute them for educational purposes as long as you cite [DeepLearning.AI](#) as the source of the slides.

For the rest of the details of the license, see <https://creativecommons.org/licenses/by-sa/2.0/legalcode>

Stanford
ONLINE

DeepLearning.AI



Machine Learning

Welcome!



eddy



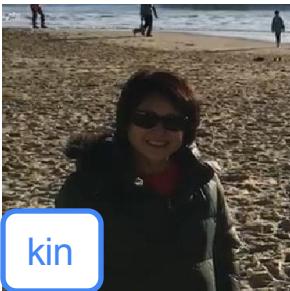
aarti



geoff



Ivy



kin



robert



andres



daniel

Re: Urgent Information :)

External

Spam ×

Congratulations!
You've won
a million dollars!



Compose

Mail

Inbox

Starred

Snoozed

Sent

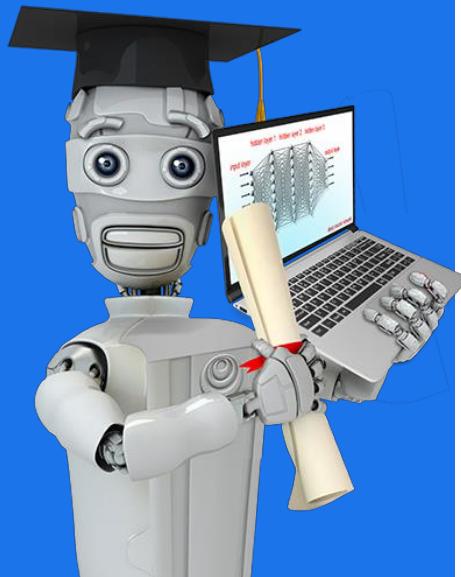
Drafts

42



Stanford
ONLINE

DeepLearning.AI



Machine Learning

Applications of
Machine Learning

Stanford
ONLINE

DeepLearning.AI



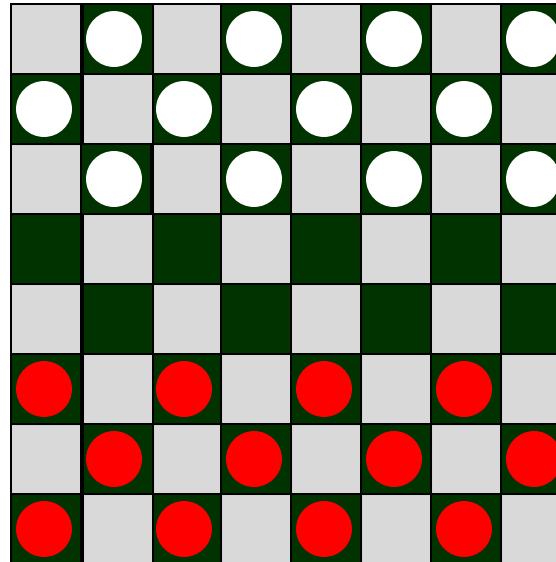
Machine Learning Overview

What is
Machine Learning?

Machine learning

“Field of study that gives computers the ability to learn without being explicitly programmed.”

Arthur Samuel (1959)



Question

If the checkers program had been allowed to play only ten games (instead of tens of thousands) against itself, a much smaller number of games, how would this have affected its performance?

- Would have made it better
 -  Would have made it worse
-

Machine learning algorithms

rapid advancements

used most in real-world applications

- Supervised learning ← course 1, 2
- Unsupervised learning ←
- Recommender systems
- Reinforcement learning

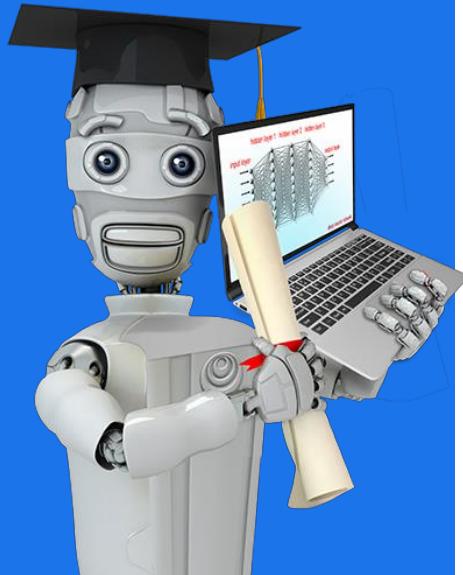
course 3

Practical advice for applying learning algorithms



Stanford
ONLINE

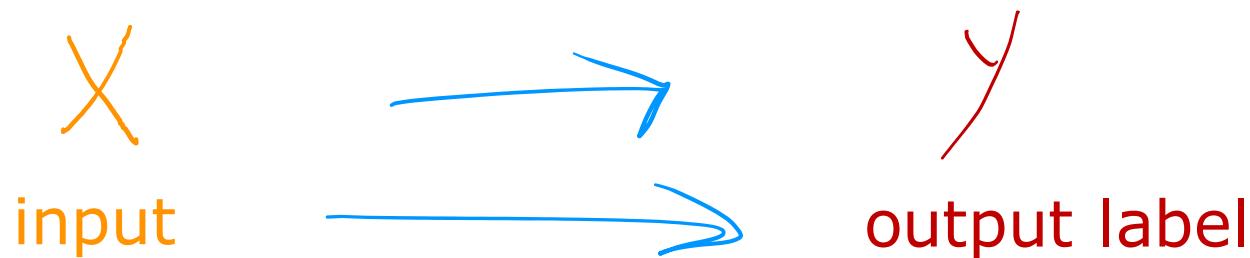
DeepLearning.AI



Machine Learning Overview

Supervised Learning Part 1

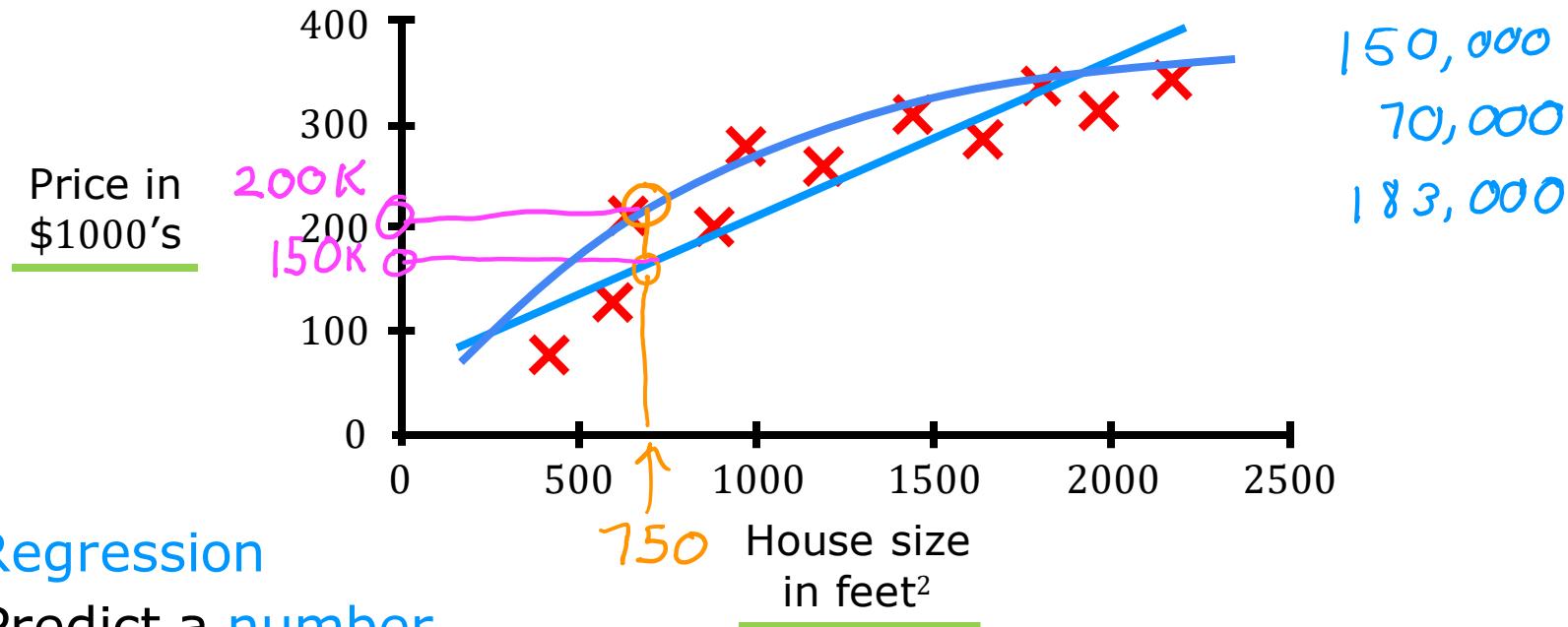
Supervised learning



Learns from being given “right answers”

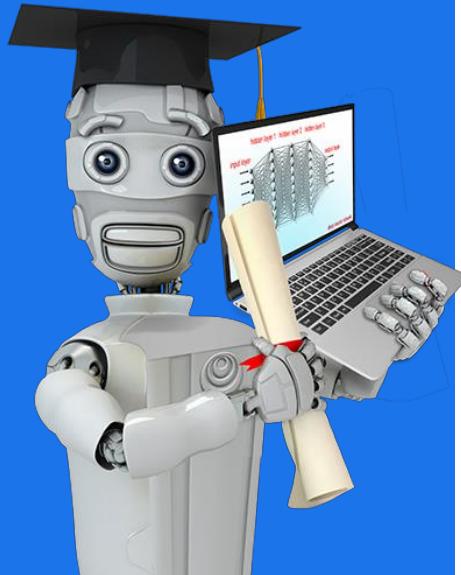
| Input (X) | Output (Y) | Application |
|-------------------|------------------------|---------------------|
| email | spam? (0/1) | spam filtering |
| audio | text transcripts | speech recognition |
| English | Spanish | machine translation |
| ad, user info | click? (0/1) | online advertising |
| image, radar info | position of other cars | self-driving car |
| image of phone | defect? (0/1) | visual inspection |

Regression: Housing price prediction



Stanford
ONLINE

DeepLearning.AI



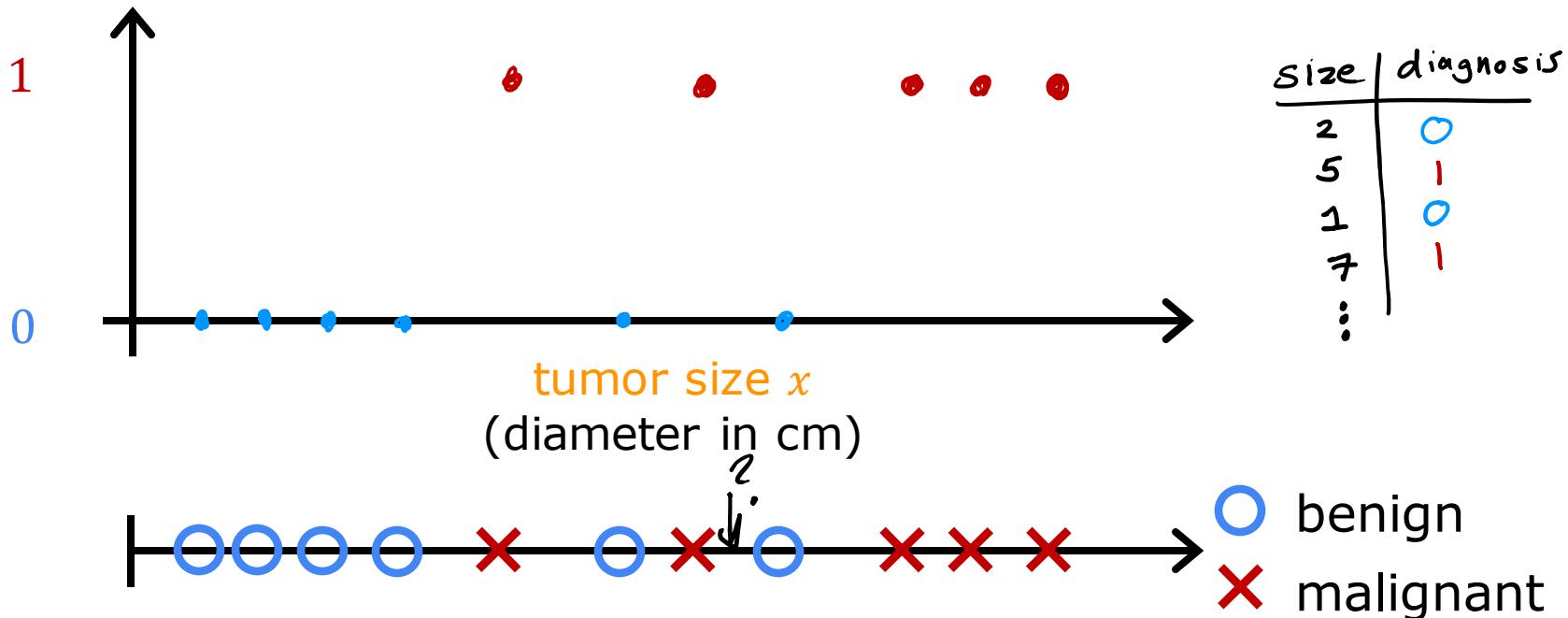
Machine Learning Overview

Supervised Learning Part 2

Classification: Breast cancer detection



malignant benign

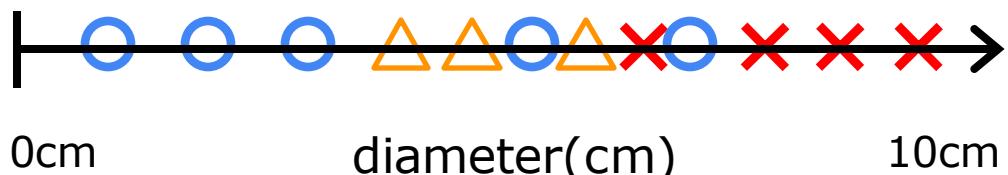


Classification: Breast cancer detection

○ benign

✗ malignant type 1

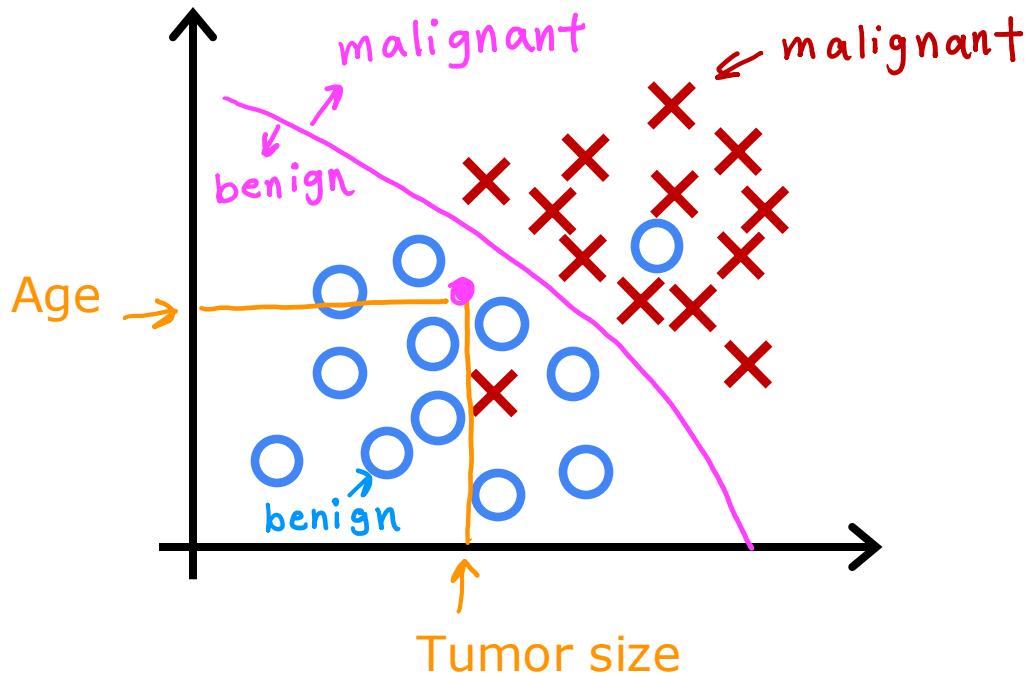
△ malignant type 2



Classification
predict categories cat dog benign malignant 0, 1, 2

small number of possible outputs

Two or more inputs



Supervised learning

Learns from being given “right answers”

Regression

Predict a number

infinitely many possible outputs

Classification

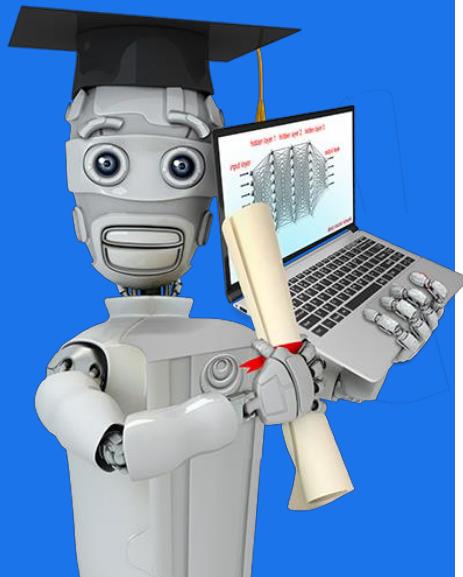
predict categories

small number of possible outputs



Stanford
ONLINE

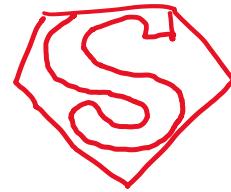
DeepLearning.AI



Machine Learning Overview

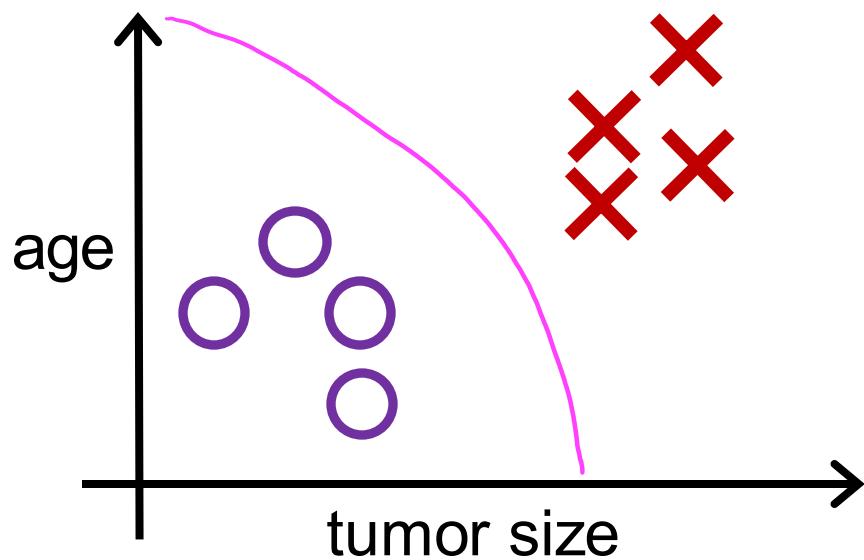
Unsupervised Learning Part 1

Previous: Supervised learning

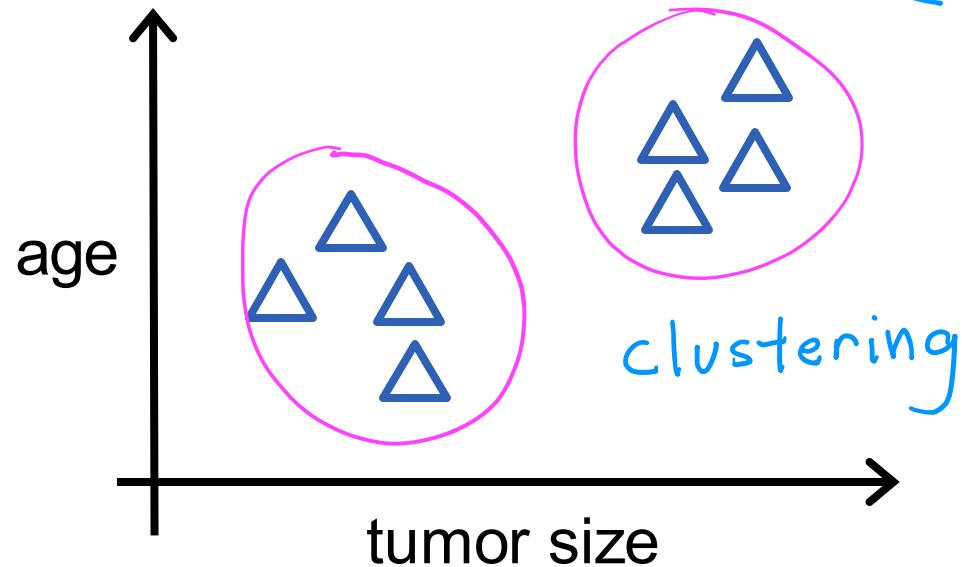


Now: Unsupervised learning

Supervised learning
Learn from data **labeled**
with the “**right answers**”



Unsupervised learning
Find something interesting
in **unlabeled** data.



Clustering: Google news



Giant **panda** gives birth to rare **twin** cubs at Japan's oldest **zoo**

USA TODAY · 6 hours ago



- Giant **panda** gives birth to **twin** cubs at Japan's oldest **zoo**

CBS News · 7 hours ago

- Giant **panda** gives birth to **twin** cubs at Tokyo's Ueno **Zoo**

WHBL News · 16 hours ago

- A Joyful Surprise at Japan's Oldest **Zoo**: The Birth of **Twin Pandas**

The New York Times · 1 hour ago

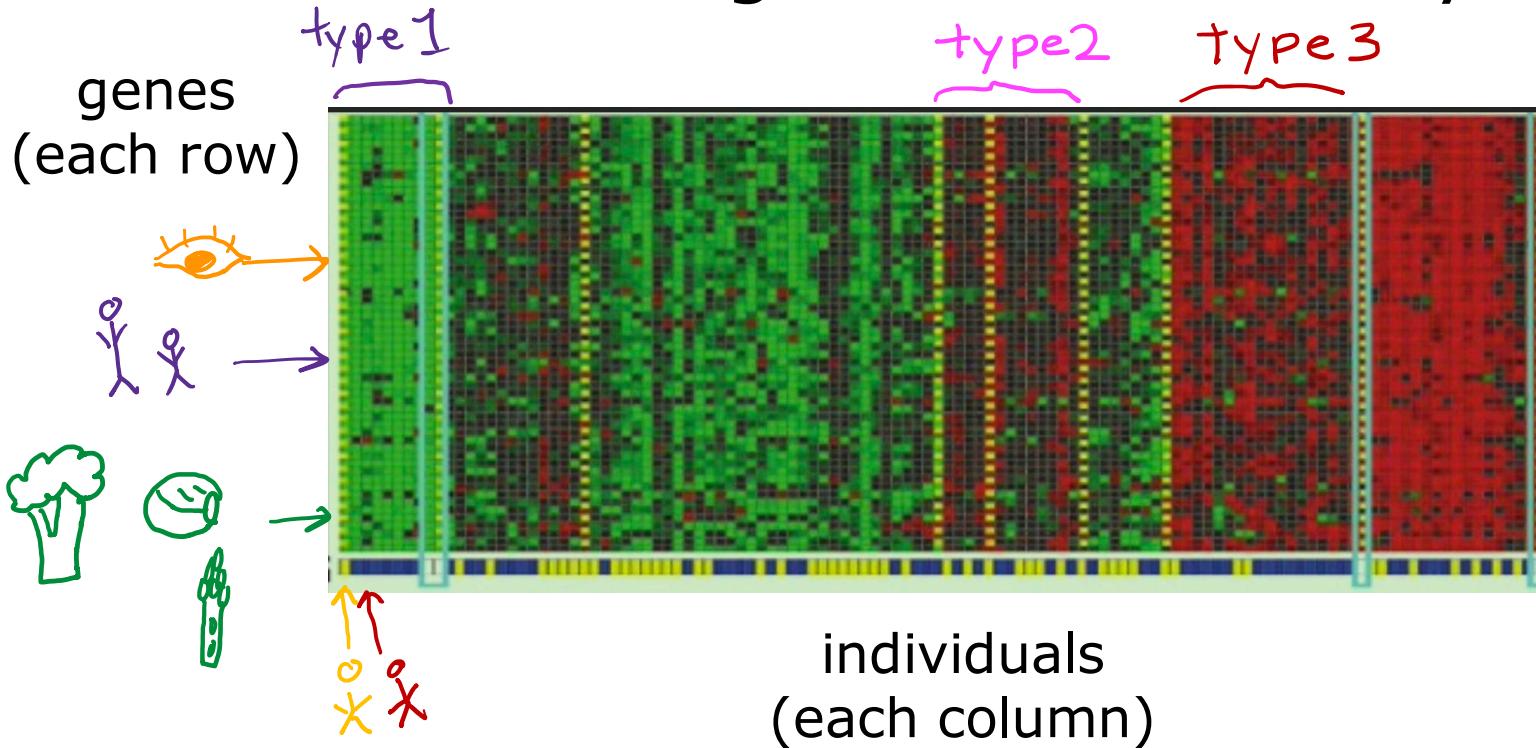
- **Twin** Panda **Cubs** Born at Tokyo's Ueno **Zoo**

PEOPLE · 6 hours ago

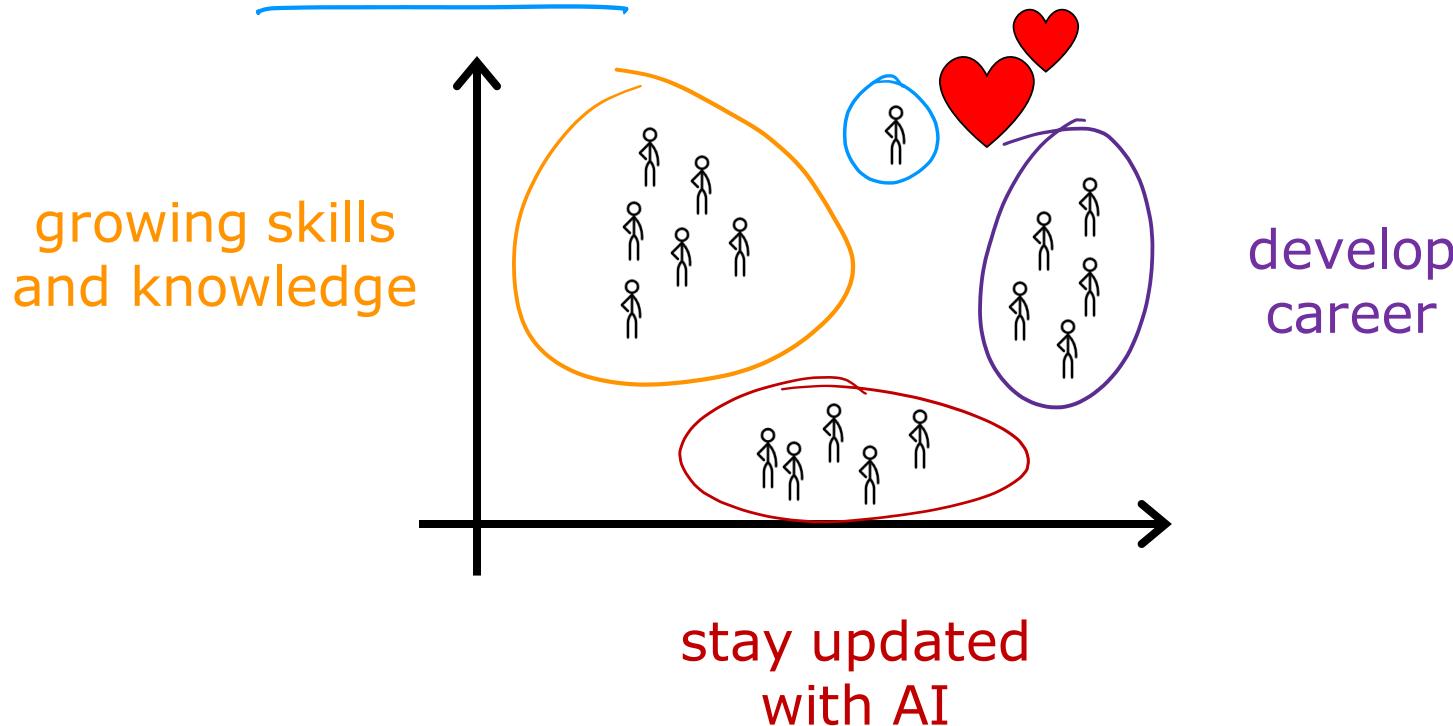
View Full Coverage



Clustering: DNA microarray



Clustering: Grouping customers



Stanford
ONLINE

DeepLearning.AI



Machine Learning Overview

Unsupervised Learning Part 2

Unsupervised learning

Data only comes with inputs x , but not output labels y .
Algorithm has to find **structure** in the data.

Clustering

Group similar data points together.

Dimensionality reduction

Compress data using fewer numbers.

Anomaly detection

Find unusual data points.



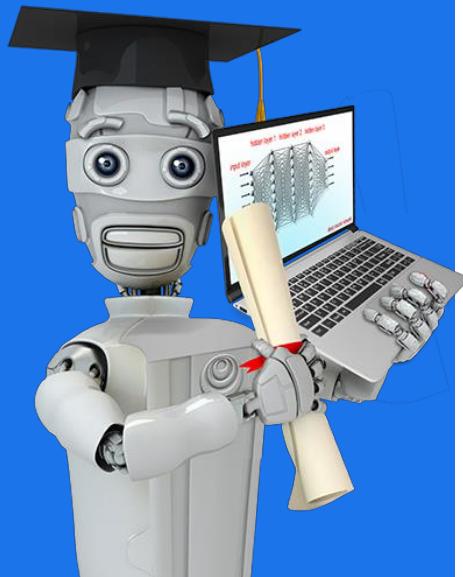
Question

Of the following examples, which would you address using an **unsupervised** learning algorithm?

-  Given email labeled as spam/not spam, learn a spam filter.
-  Given a set of news articles found on the web, group them into sets of articles about the same story.
-  Given a database of customer data, automatically discover market segments and group customers into different market segments.
-  Given a dataset of patients diagnosed as either having diabetes or not, learn to classify new patients as having diabetes or not

Stanford
ONLINE

DeepLearning.AI

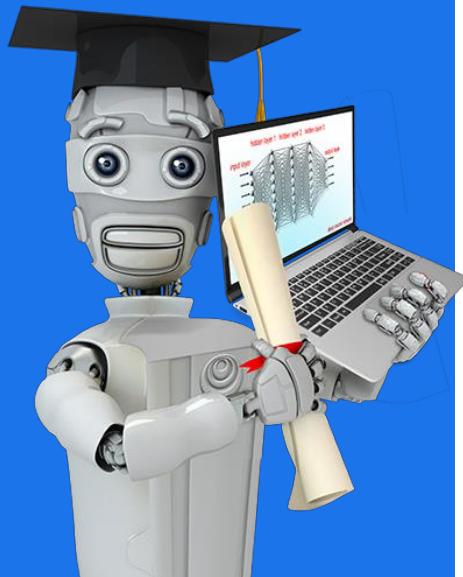


Machine Learning Overview

Jupyter Notebooks

Stanford
ONLINE

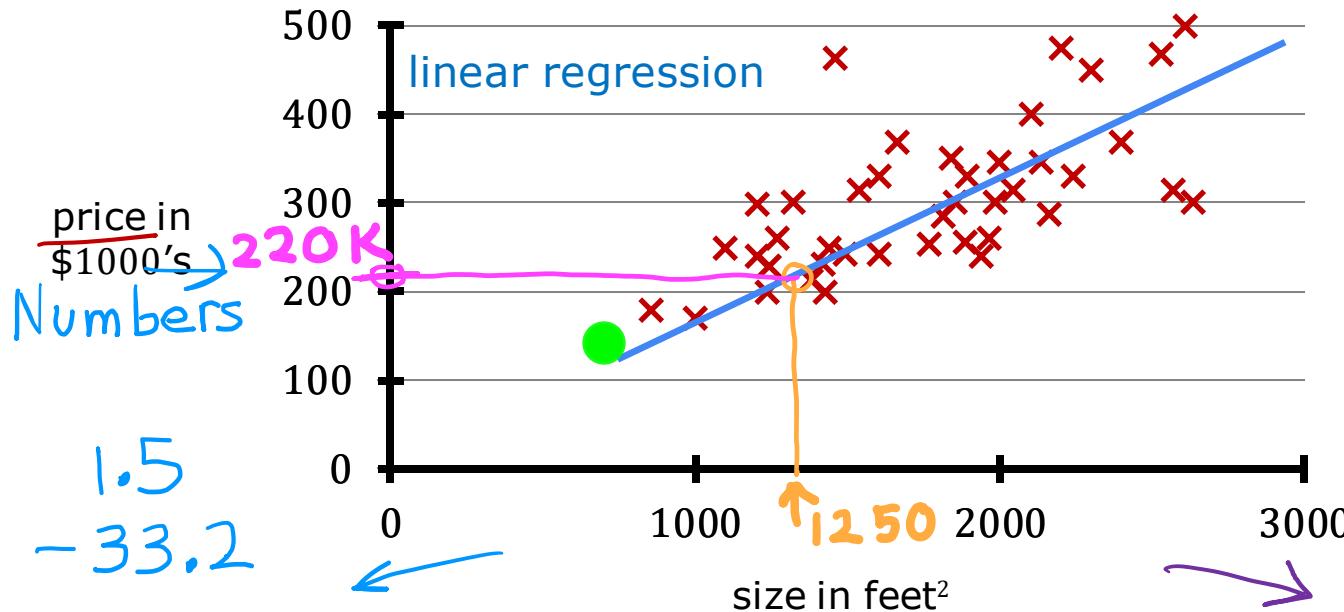
DeepLearning.AI



Linear Regression with One Variable

Linear Regression Model Part 1

House sizes and prices



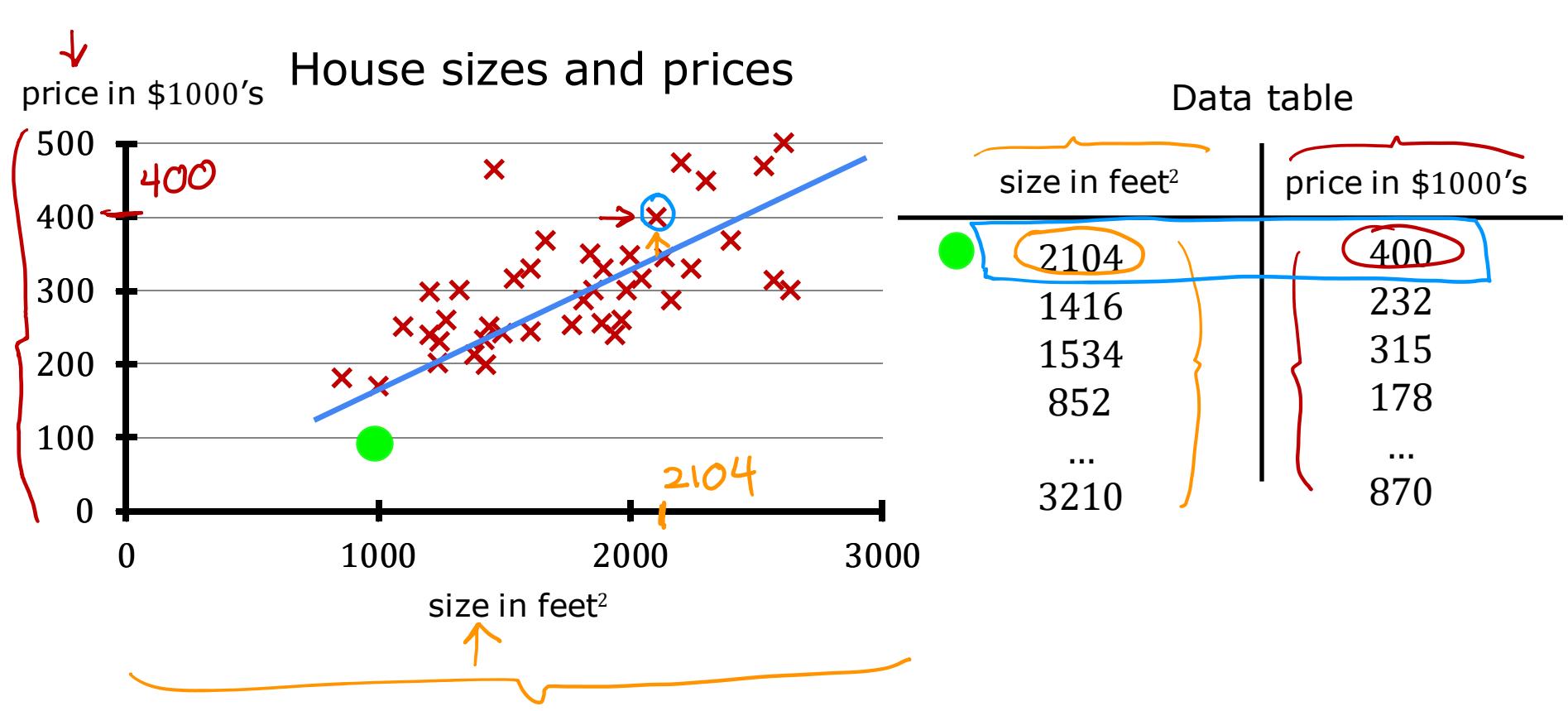
Regression model
Predicts numbers
Infinitely many possible outputs

Supervised learning model
Data has "right answers"

Classification model
Predicts categories
Small number of possible outputs

categories
cat } 2
dog }

disease ↗ 10



Terminology

Training set: Data used to train the model

x

size in feet²

(1)

2104

(2)

1416

(3)

1534

(4)

852

...

(47)

...

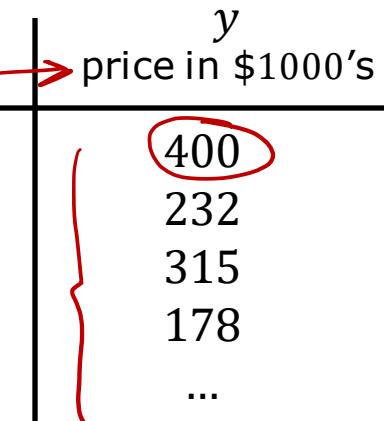
3210

$$x^{(1)} = 2104$$

$$(x^{(1)}, y^{(1)}) = (2104, 400)$$

$$x^{(2)} = 1416$$

$$X^{(2)} \neq X^2 \text{ not exponent}$$



Notation:

x = "input" variable
feature

y = "output" variable
"target" variable

m = number of training examples

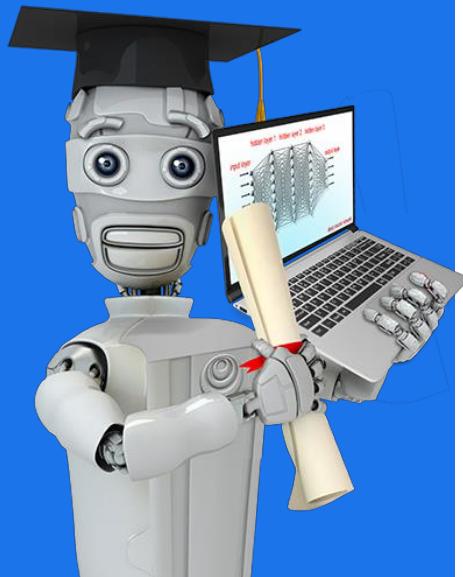
(x, y) = single training example

$$(x^{(i)}, y^{(i)})$$

$(x^{(i)}, y^{(i)})$ = i^{th} training example
index $(1^{\text{st}}, 2^{\text{nd}}, 3^{\text{rd}} \dots)$

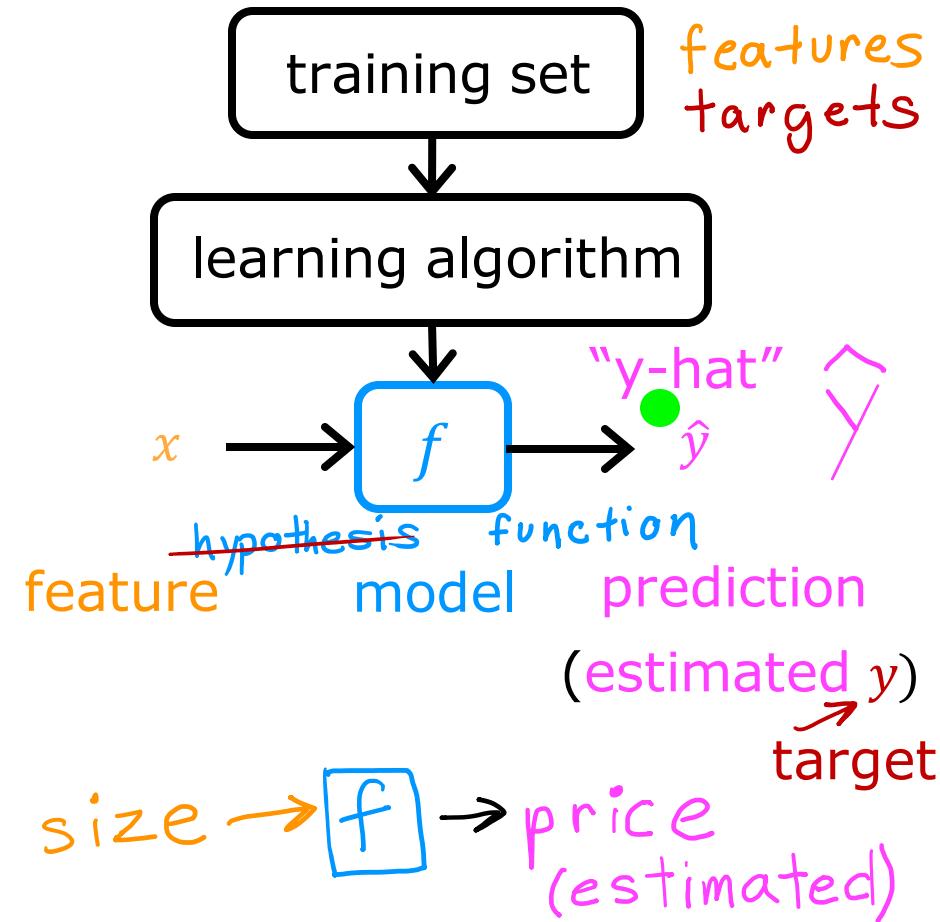
Stanford
ONLINE

DeepLearning.AI



Linear Regression with One Variable

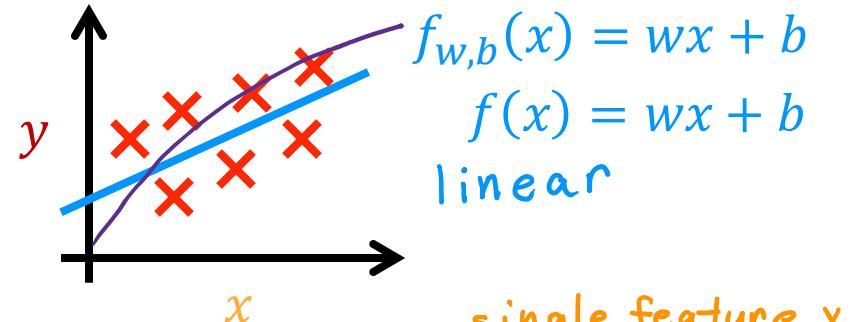
Linear Regression Model Part 2



How to represent f ?

$$f_{w,b}(x) = wx + b$$

$$f(x)$$

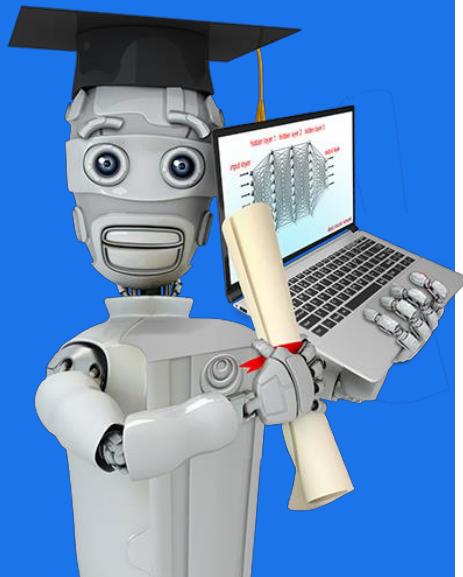


Linear regression with one variable.
size

Univariate linear regression.
one variable

Stanford
ONLINE

DeepLearning.AI



Linear Regression with One Variable

Cost Function

Training set

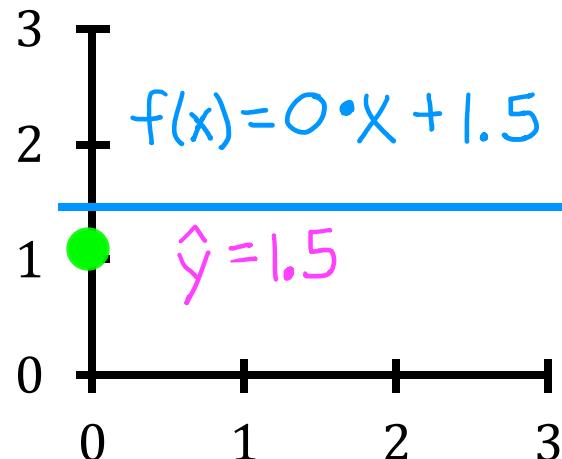
| features | targets |
|-----------------------------------|------------------------|
| size in feet ² (x) | price \$1000's (y) |
| 2104 | 460 |
| 1416 | 232 |
| 1534 | 315 |
| 852 | 178 |
| ... | ... |

$$\text{Model: } f_{w,b}(x) = wx + b$$

w, b : parameters
coefficients
weights

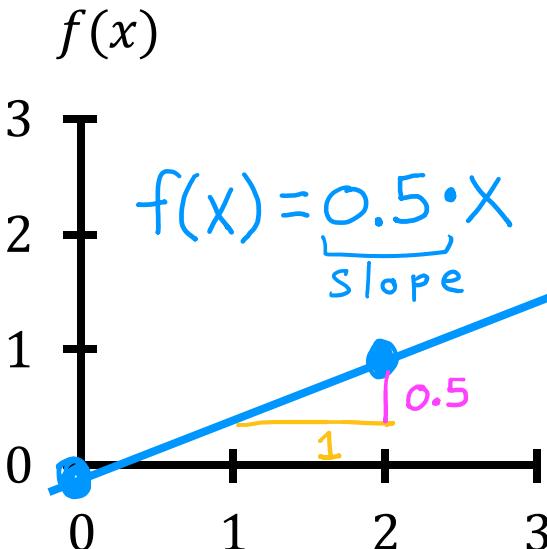
What do w, b do?

$$f_{w,b}(x) = wx + b$$

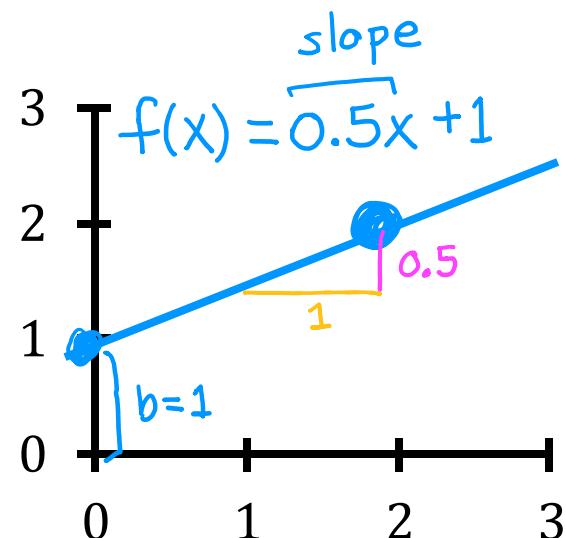


$$\rightarrow w = 0$$
$$\rightarrow b = 1.5$$

(y-intercept)

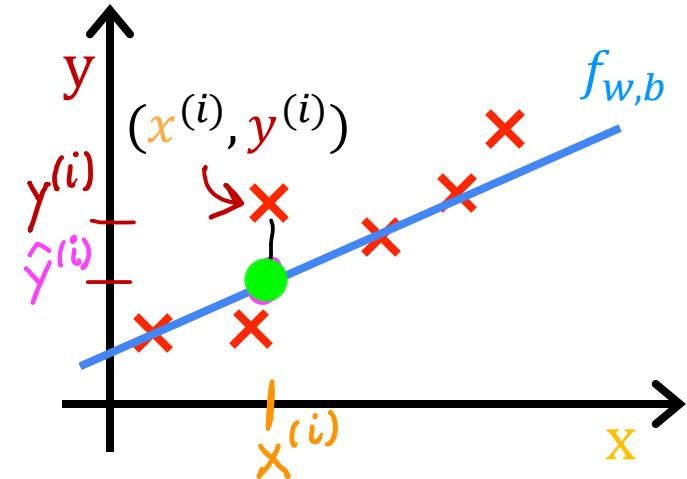


$$\rightarrow w = 0.5$$
$$\rightarrow b = 0$$



$$\rightarrow w = 0.5$$
$$\rightarrow b = 1$$

Cost function: Squared error cost function



$$\hat{y}^{(i)} = f_{w,b}(x^{(i)})$$

$$f_{w,b}(x^{(i)}) = w x^{(i)} + b$$

$$\bar{J}(w, b) = \frac{1}{2m} \sum_{i=1}^m (\hat{y}^{(i)} - y^{(i)})^2$$

m = number of training examples

$$J(w, b) = \frac{1}{2m} \sum_{i=1}^m (f_{w,b}(x^{(i)}) - y^{(i)})^2$$

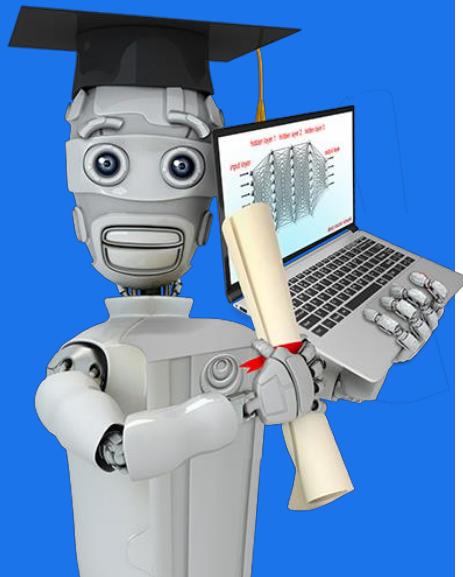
intuition (next!)

Find w, b :

$\hat{y}^{(i)}$ is close to $y^{(i)}$ for all $(x^{(i)}, y^{(i)})$.

Stanford
ONLINE

DeepLearning.AI



Linear Regression with One Variable

Cost Function
Intuition

model:

$$\underline{f_{w,b}(x) = wx + b}$$

parameters:

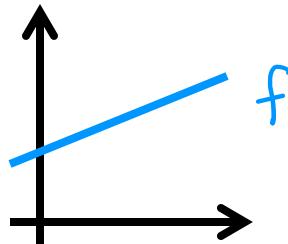
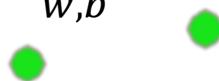
$$\underline{w, b}$$

cost function:

$$J(w, b) = \frac{1}{2m} \sum_{i=1}^m (f_{w,b}(x^{(i)}) - y^{(i)})^2$$

goal:

$$\underset{w,b}{\text{minimize}} J(w, b)$$



simplified

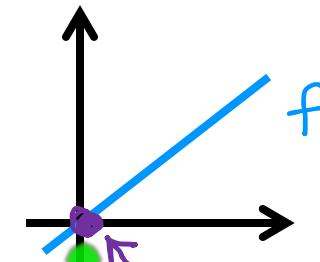
$$f_w(x) = \underline{wx}$$

$$b = \emptyset$$

$$w$$

$$\underline{J(w)} = \frac{1}{2m} \sum_{i=1}^m (\underline{f_w(x^{(i)})} - y^{(i)})^2$$

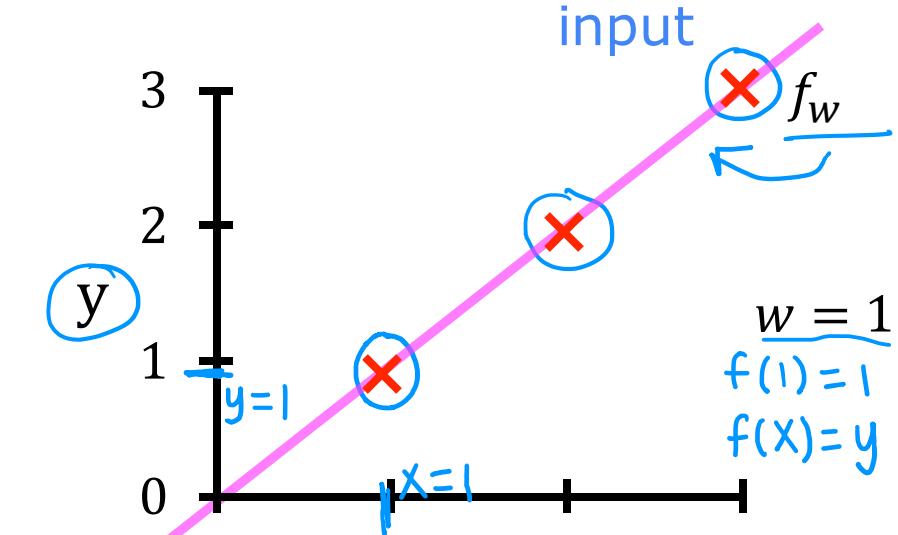
$$\underset{w}{\text{minimize}} \underline{J(w)}$$



$$\omega x^{(i)}$$

$\rightarrow f_w(x)$

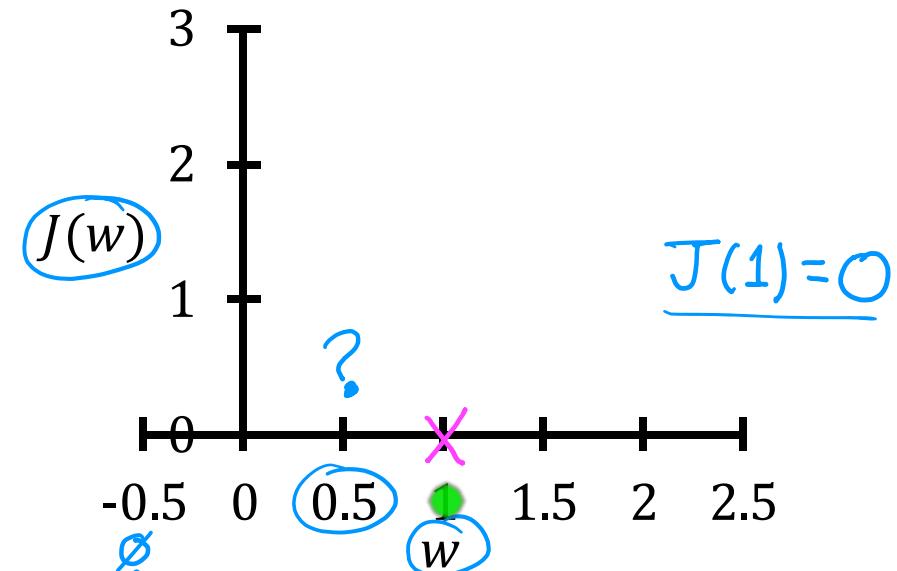
(for fixed w , function of x)



$$J(w) = \frac{1}{2m} \sum_{i=1}^m (f_w(x^{(i)}) - y^{(i)})^2 = \frac{1}{2m} \sum_{i=1}^m (wx^{(i)} - y^{(i)})^2$$

$J(w)$

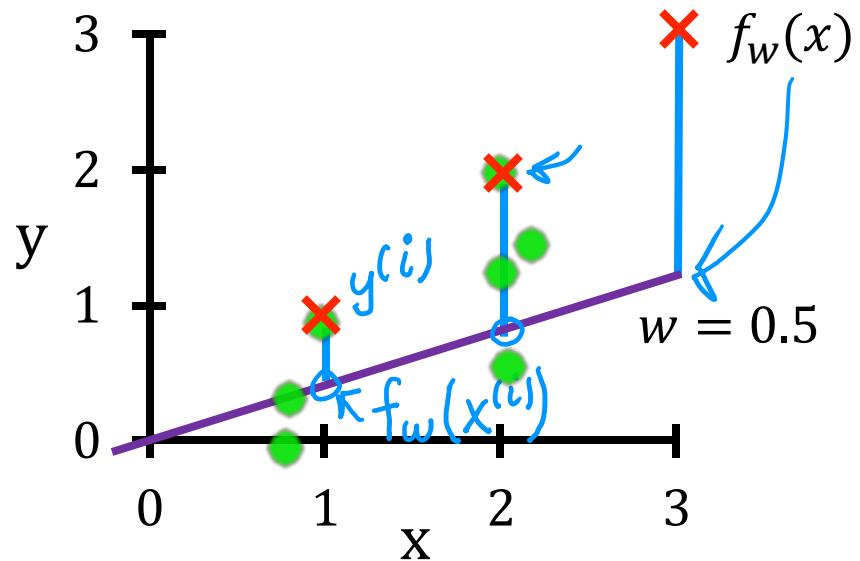
(function of w)
parameter



$$= \frac{1}{2m} (0^2 + 0^2 + 0^2) = 0$$

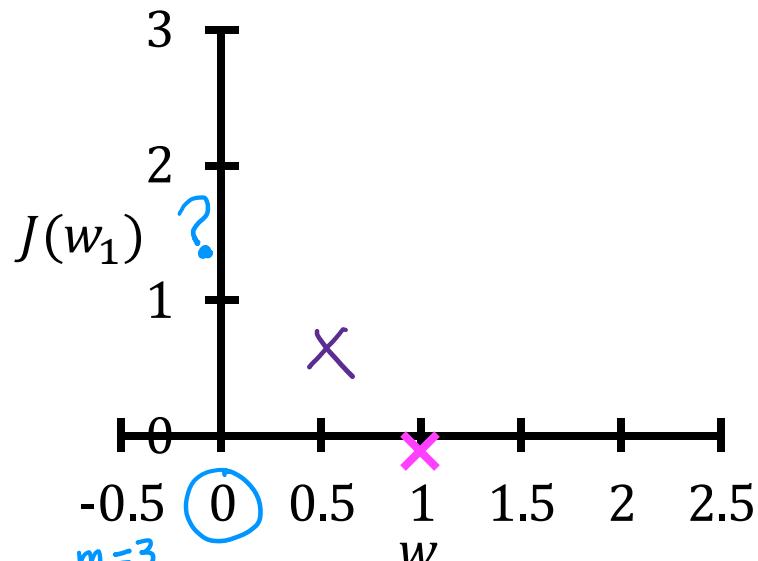
$f_w(x)$

(function of x)

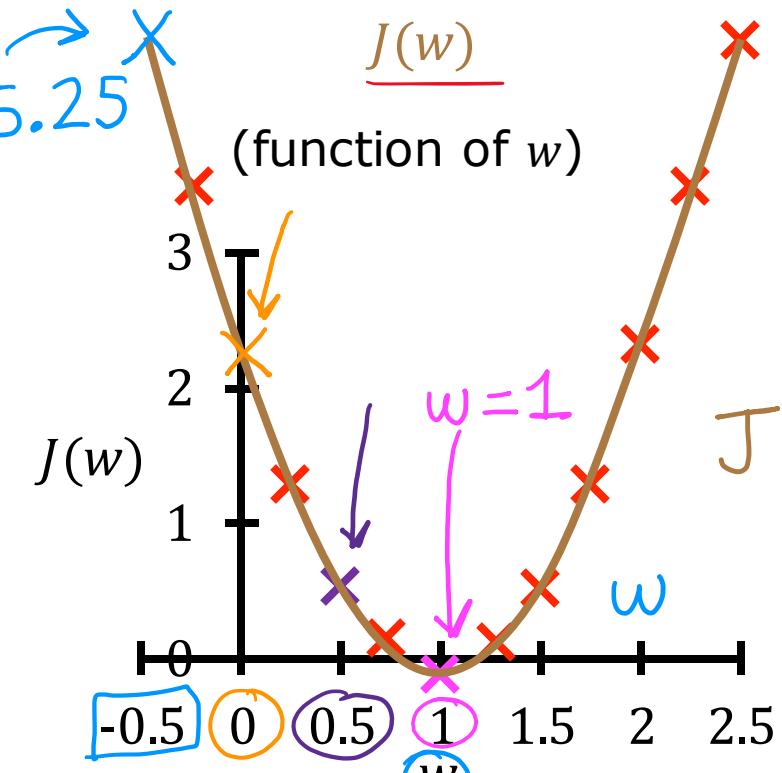
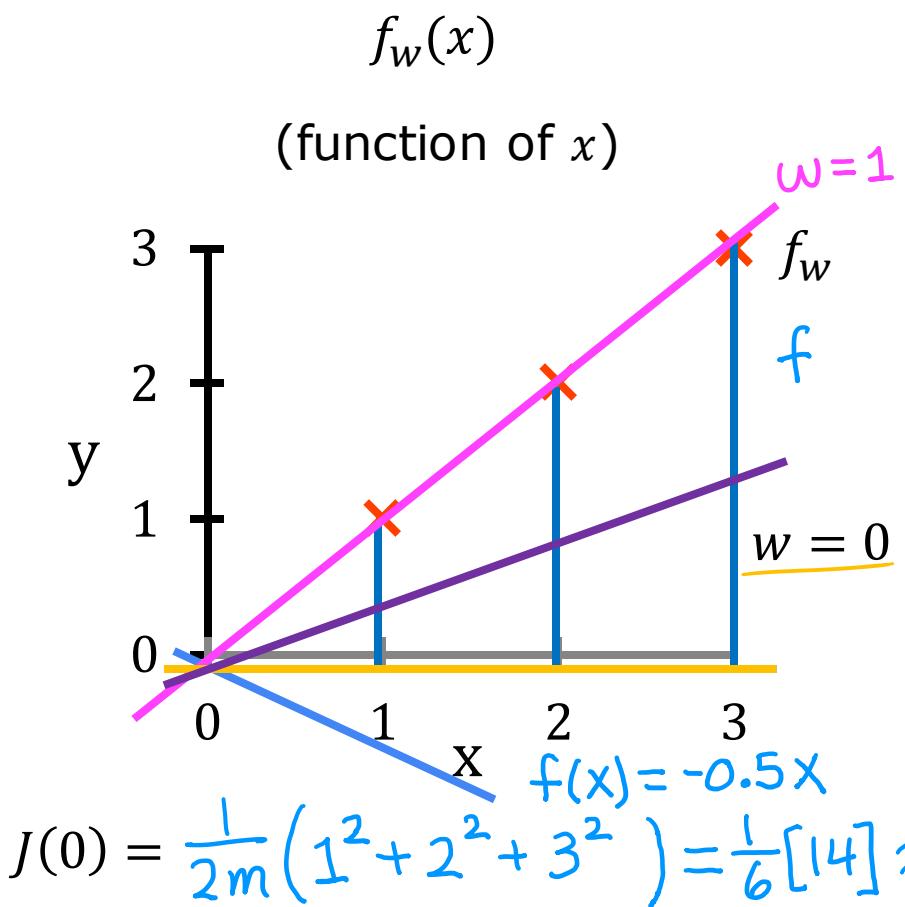


$J(w)$

(function of w)



$$J(0.5) = \frac{1}{2m} \left[(0.5-1)^2 + (1-2)^2 + (1.5-3)^2 \right] = \frac{1}{2 \times 3} [3.5] = \frac{3.5}{6} \approx 0.58$$



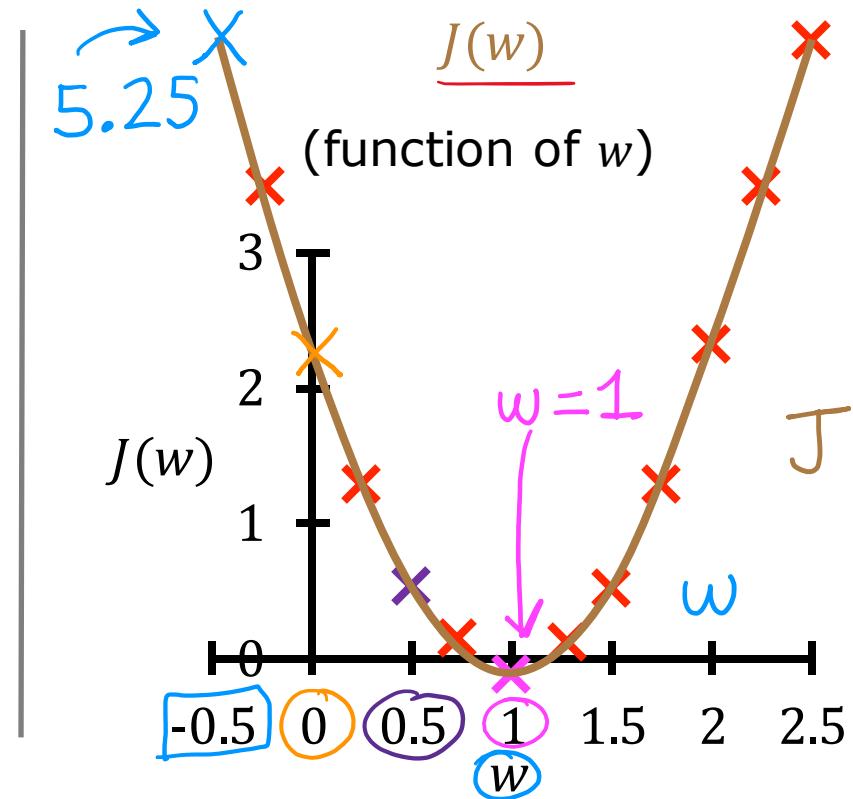
how to choose w ?

goal of linear regression:

$$\underset{w}{\text{minimize}} J(w)$$

general case:

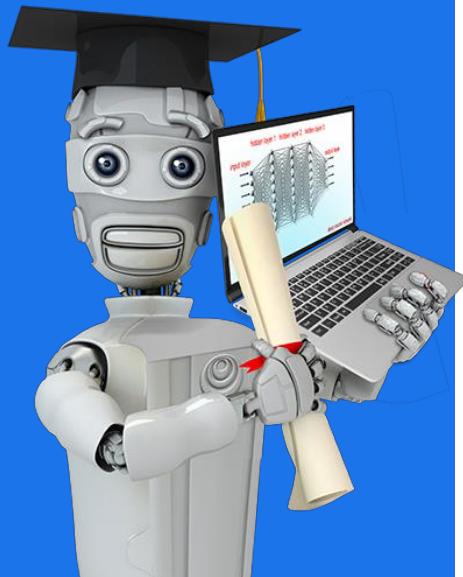
$$\underset{w,b}{\text{minimize}} J(w, b)$$



choose w to minimize $J(w)$

Stanford
ONLINE

DeepLearning.AI



Linear Regression with One Variable

Visualizing
the Cost Function

Model

$$f_{w,b}(x) = wx + b$$

Parameters

w, b

~~before: $b=0$~~

Cost Function

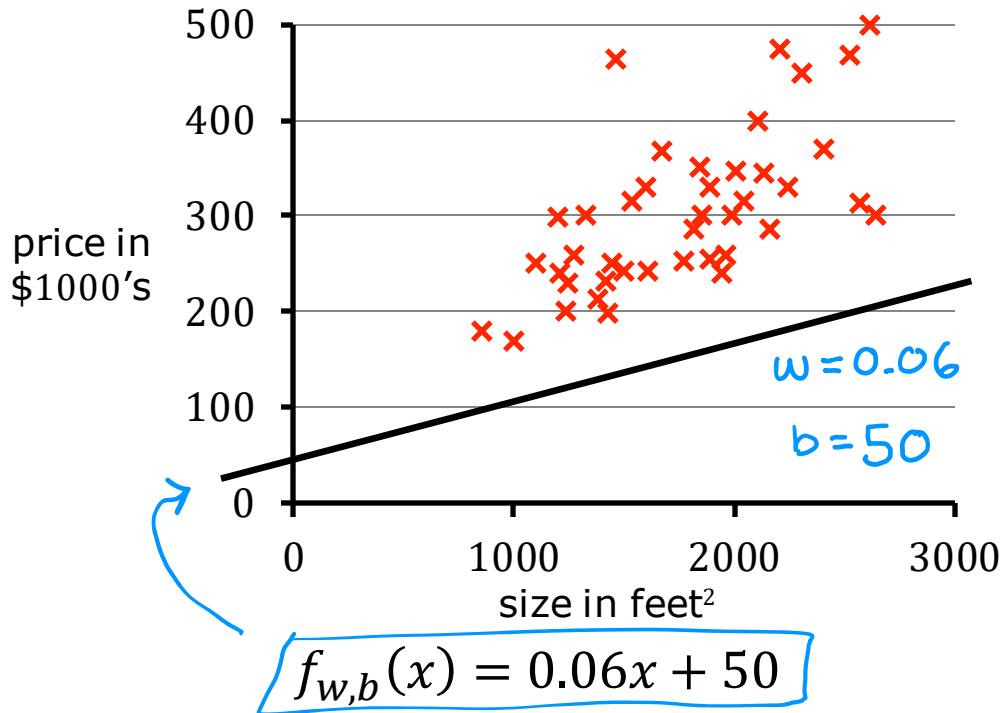
$$J(w, b) = \frac{1}{2m} \sum_{i=1}^m (f_{w,b}(x^{(i)}) - y^{(i)})^2$$

Objective

$$\underset{w,b}{\text{minimize}} J(w, b)$$

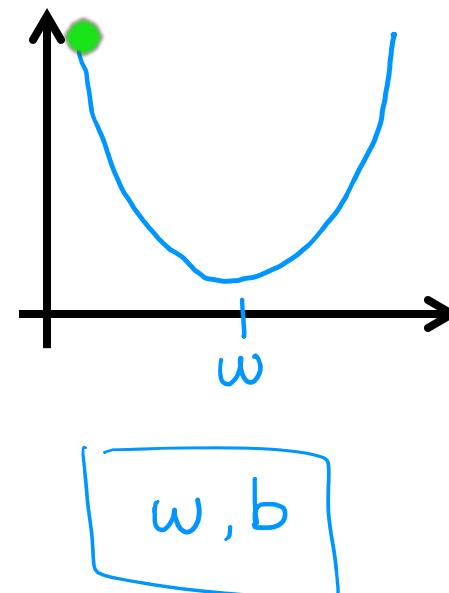
$$\underline{f_{w,b}}$$

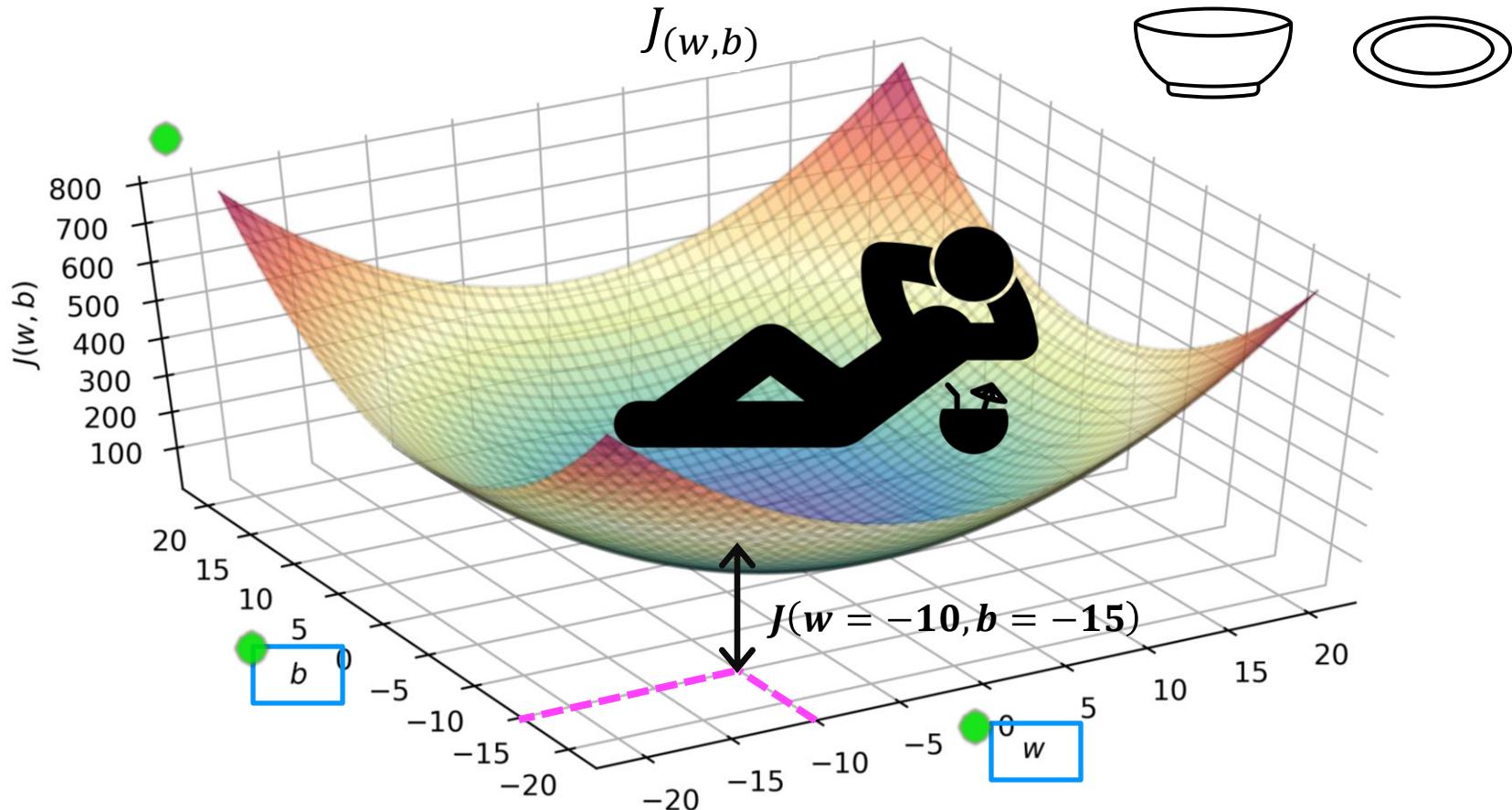
(function of x)



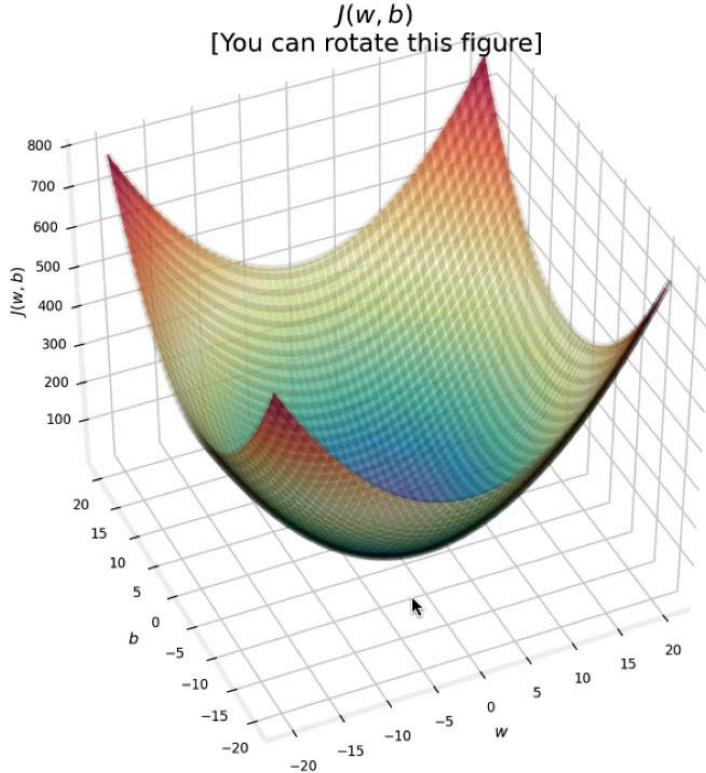
$$\underline{J}$$

(function of w, b)





3D surface plot



Alternative
contour plot

Mount Fuji

Legend

- Contour
- Mount Fuji

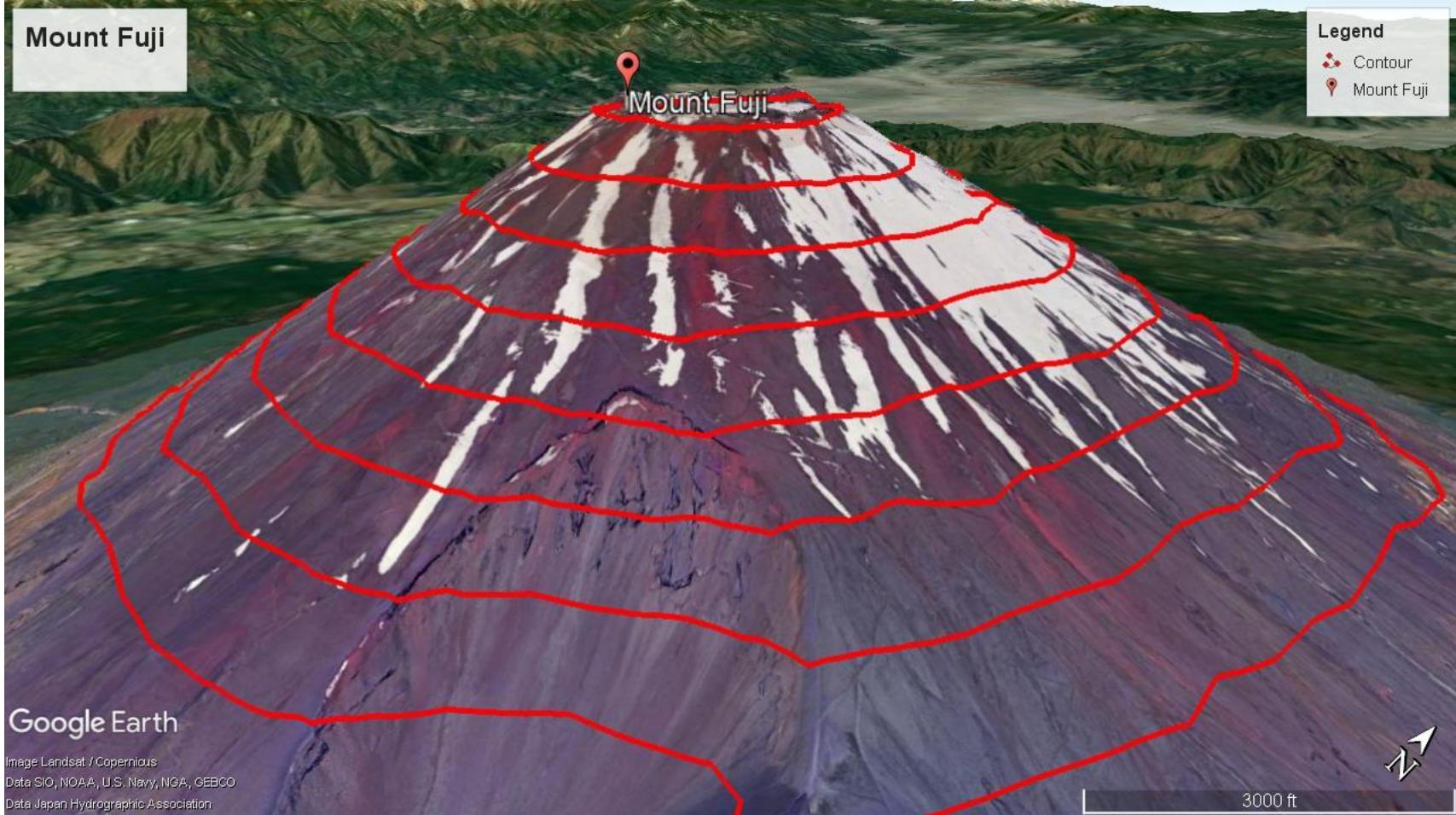
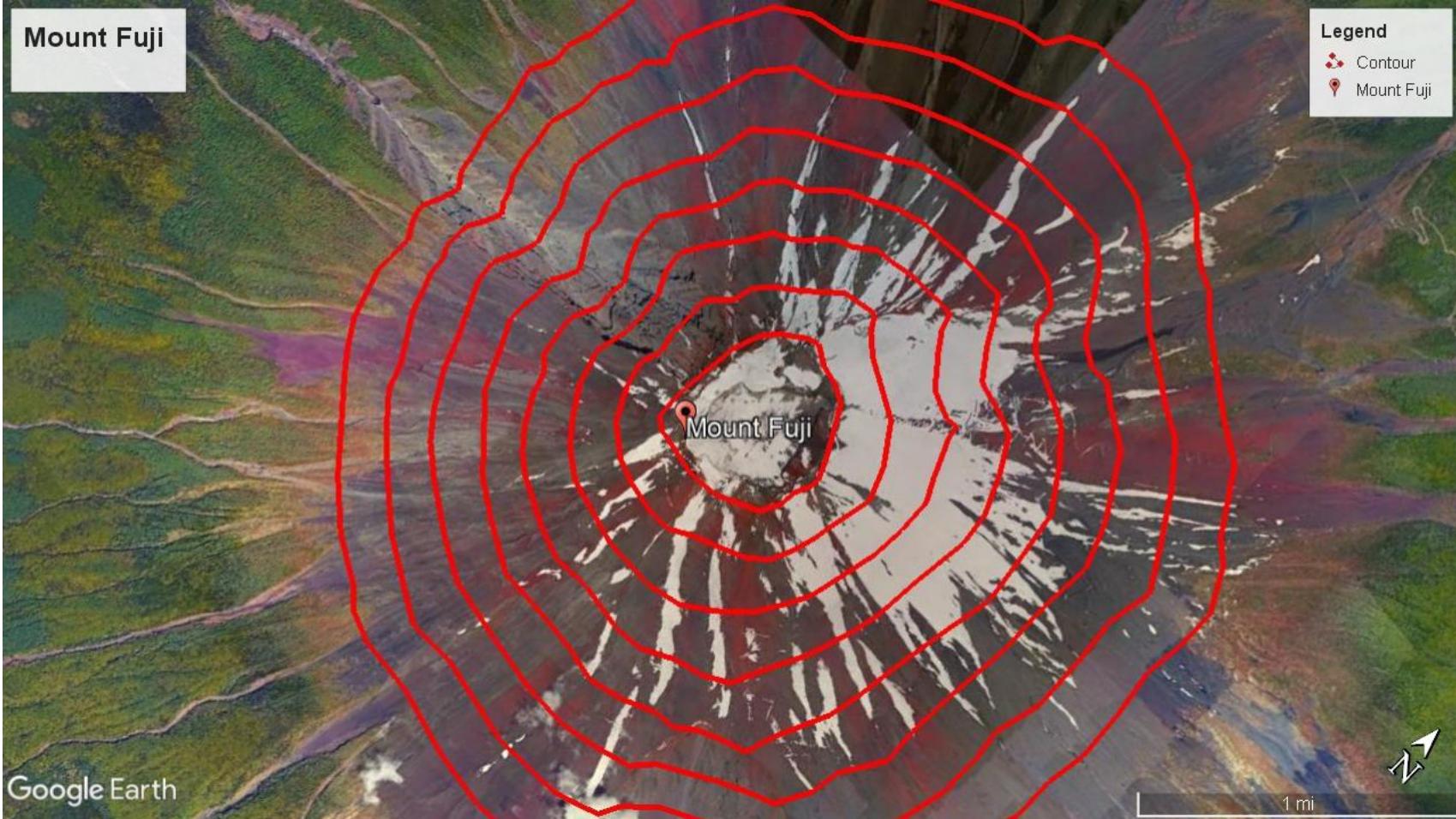


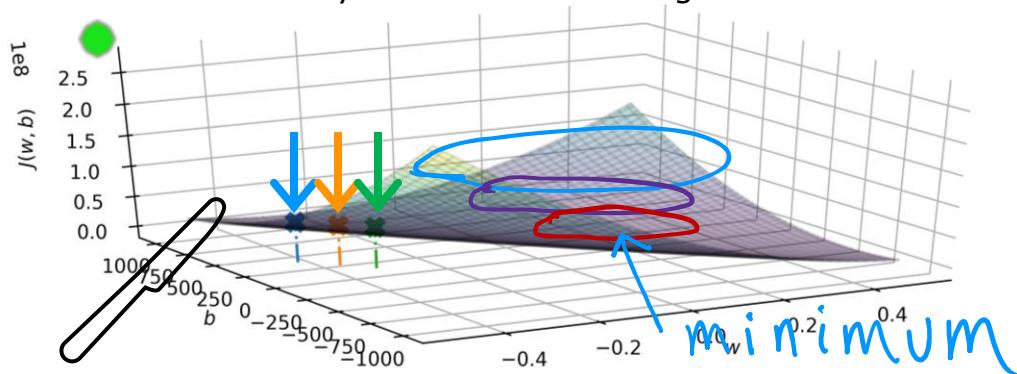
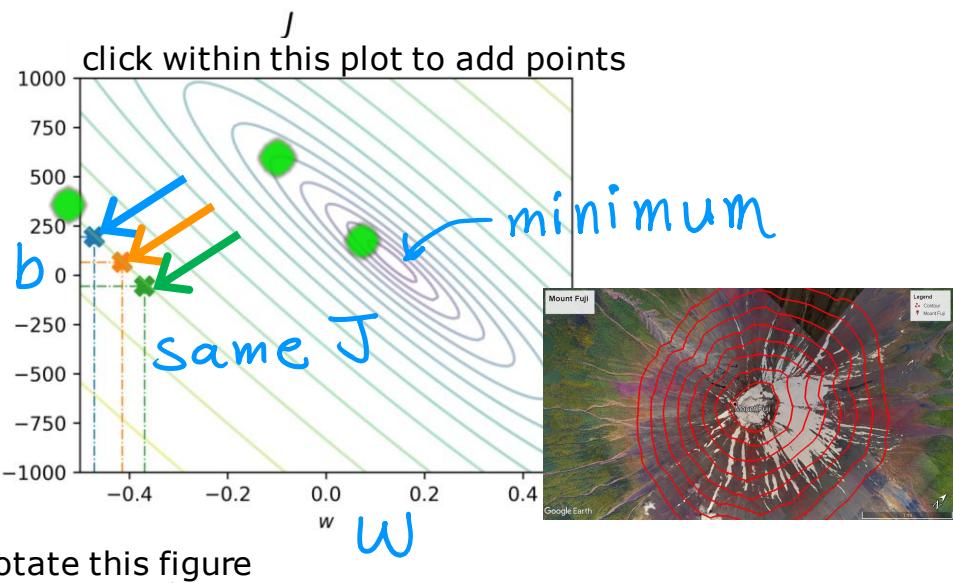
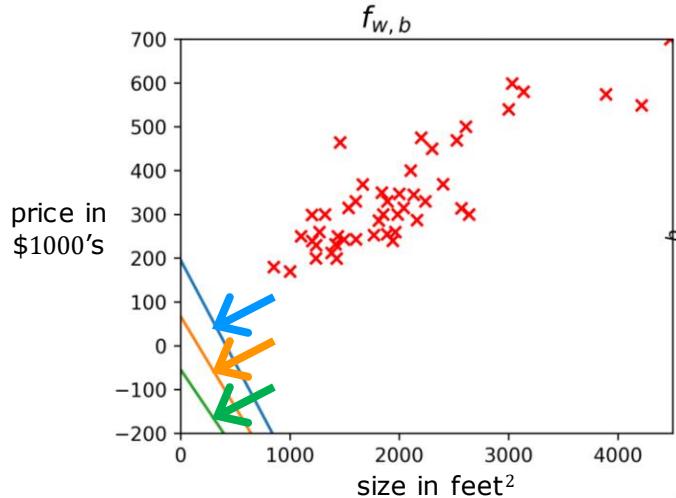
Image Landset / Copernicus
Data SIO, NOAA, U.S. Navy, NGA, GEBCO
Data Japan Hydrographic Association

Mount Fuji

Legend

- Contour
- Mount Fuji





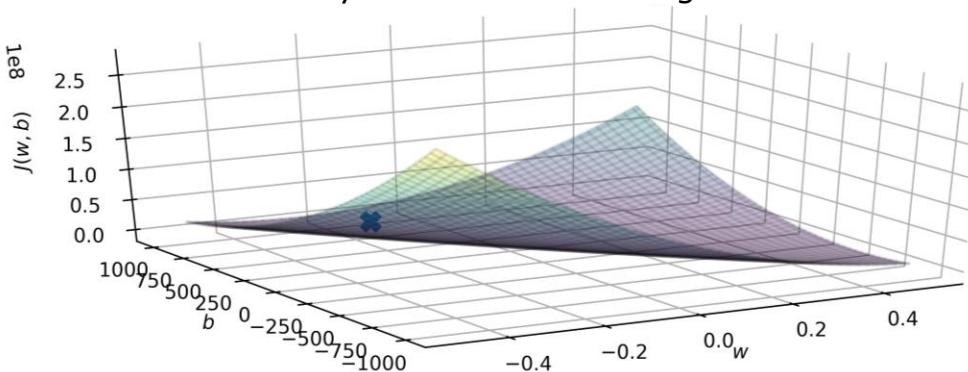
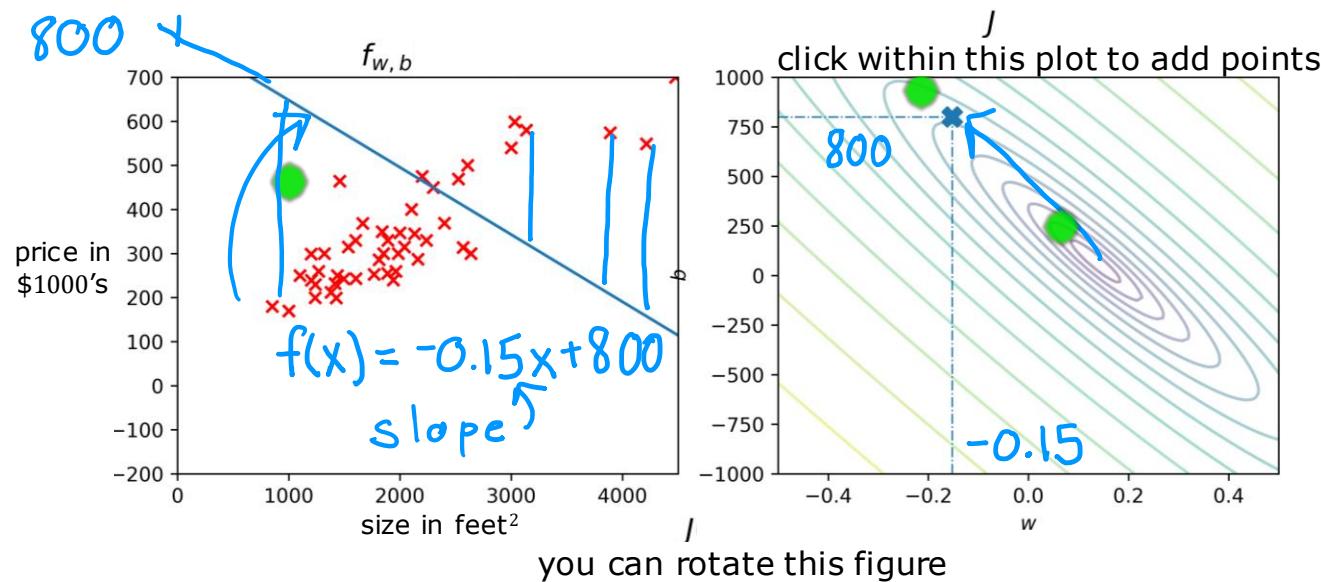
Stanford
ONLINE

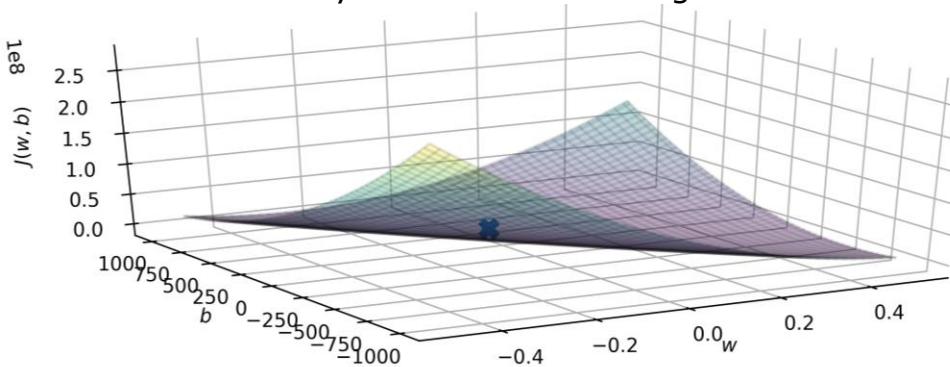
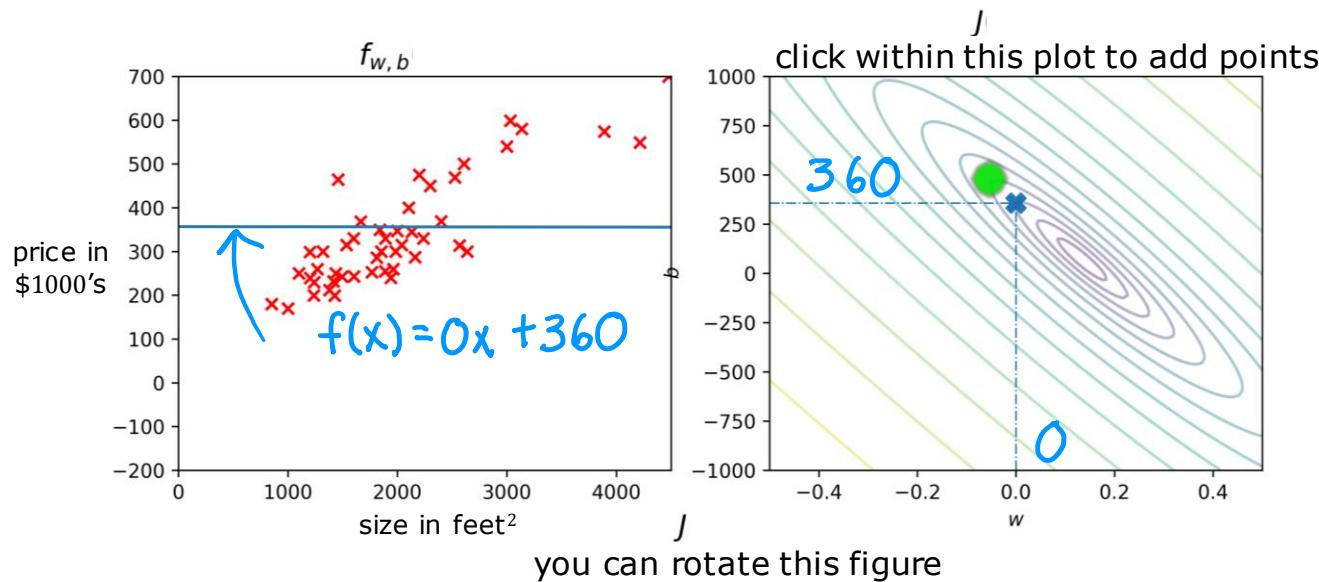
DeepLearning.AI

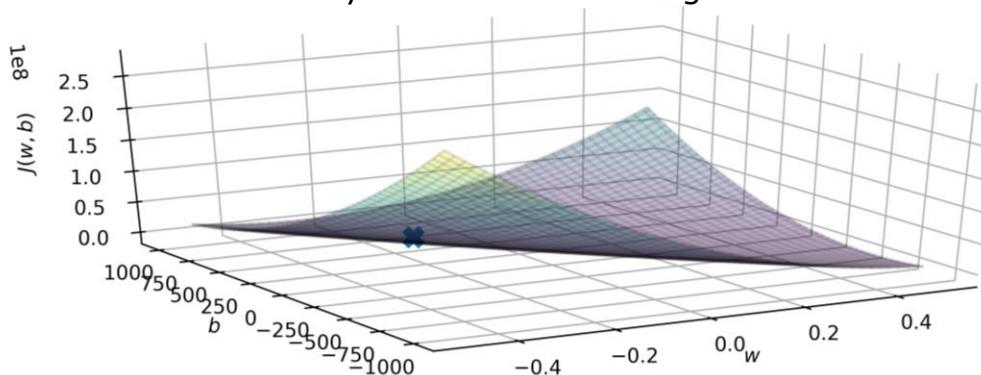
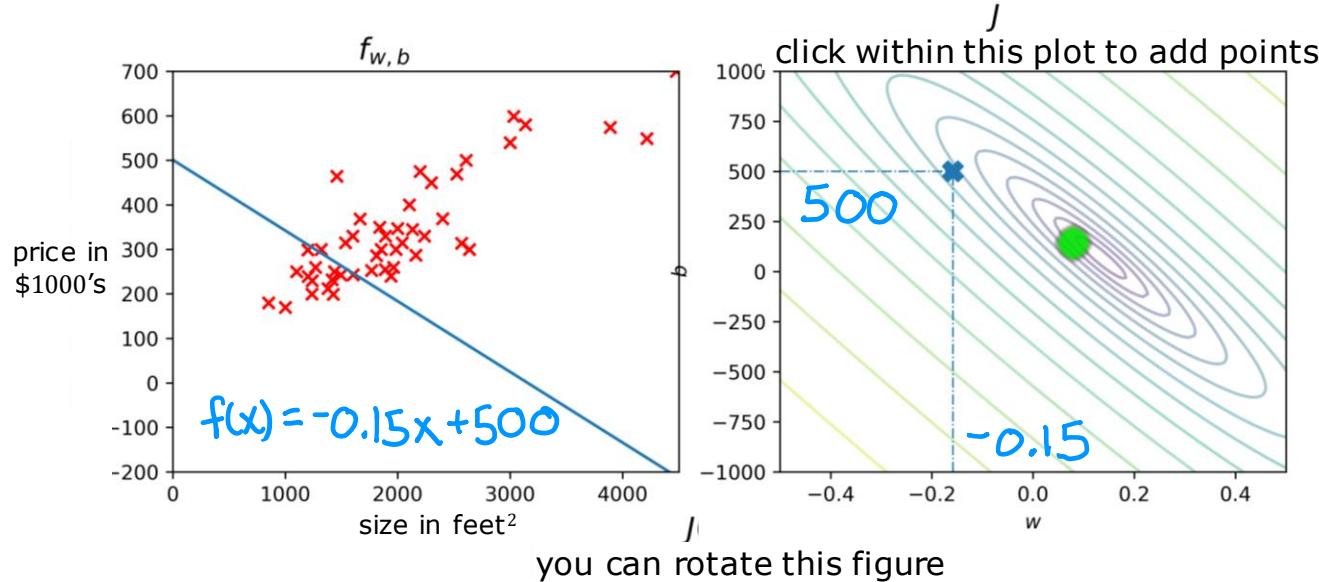


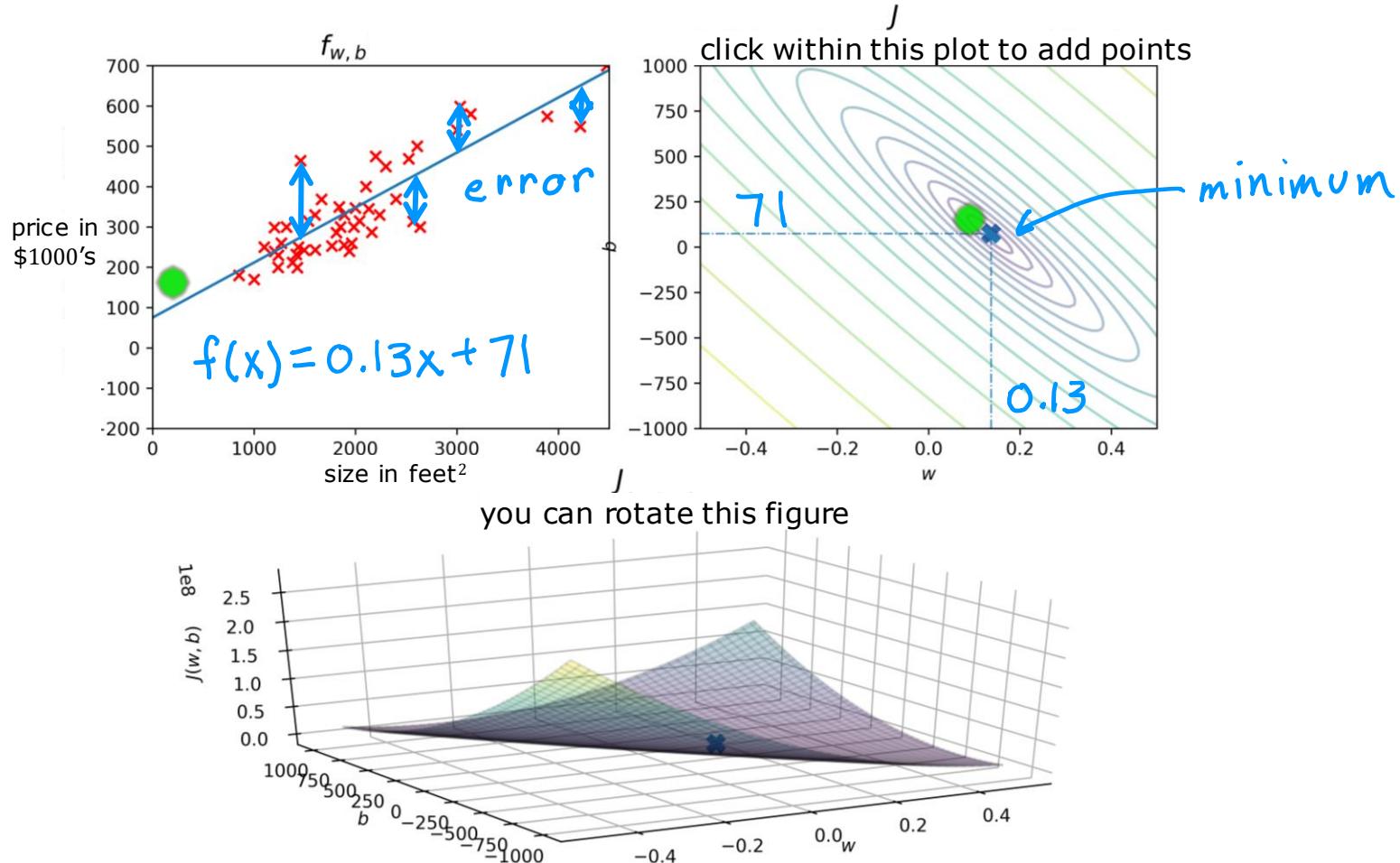
Linear Regression with One Variable

Visualization examples



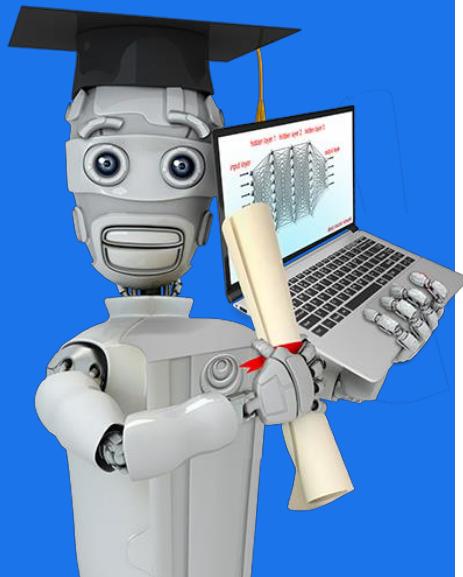






Stanford
ONLINE

DeepLearning.AI



Training Linear Regression

Gradient Descent

Have some function $\underline{J(w, b)}$ for linear regression
or any function

Want $\min_{w, b} \underline{J(w, b)}$ $\min_{w_1, \dots, w_n, b} \underline{J(w_1, w_2, \dots, w_n, b)}$

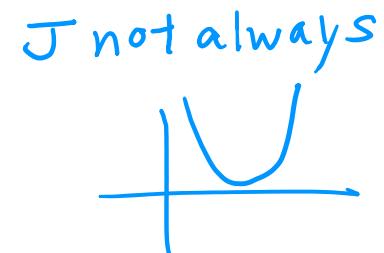
Outline:

Start with some $\underline{w, b}$ (set $w=0, b=0$)

Keep changing w, b to reduce $J(w, b)$

Until we settle at or near a minimum

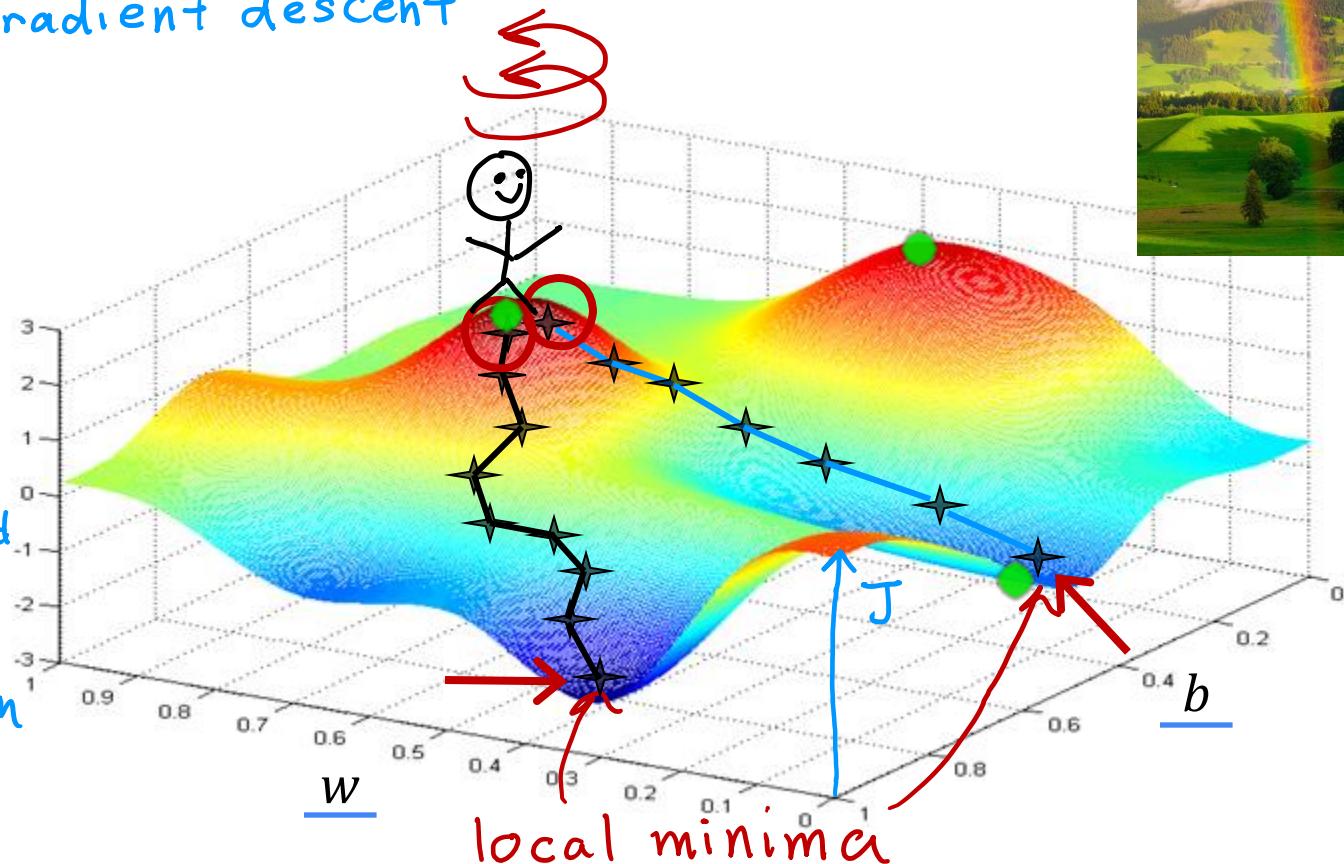
may have >1 minimum



gradient descent

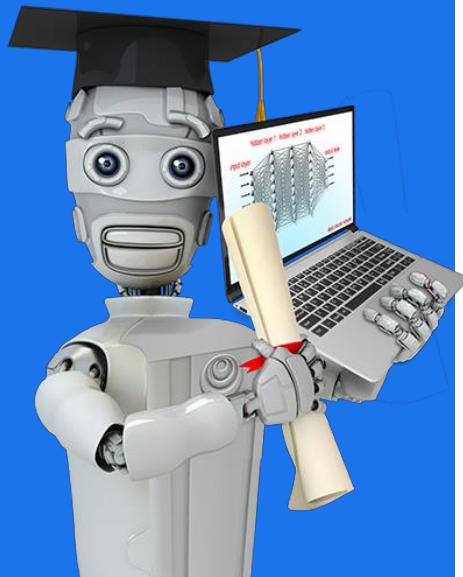
$$J(w, b)$$

not squared
error cost
not linear
regression



Stanford
ONLINE

DeepLearning.AI



Training Linear Regression

Implementing Gradient Descent

Gradient descent algorithm

Repeat until convergence

$$\left\{ \begin{array}{l} \underline{w} = w - \alpha \boxed{\frac{\partial}{\partial w} J(w, b)} \\ \underline{b} = b - \alpha \frac{\partial}{\partial b} J(w, b) \end{array} \right.$$

Learning rate
Derivative

Simultaneously
update w and b

Assignment

$$a = c$$

$$a = a + 1$$

Code

Truth assertion

$$a = c$$

$$a = a + 1$$

Math

$$a == c$$

Correct: Simultaneous update

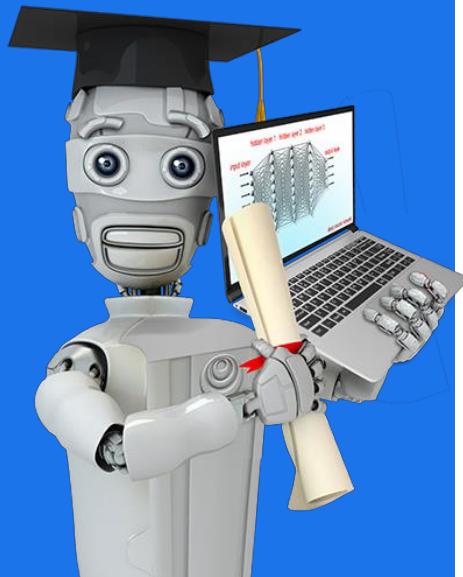
$$\left. \begin{array}{l} \text{tmp_w} = w - \alpha \frac{\partial}{\partial w} J(w, b) \\ \text{tmp_b} = b - \alpha \frac{\partial}{\partial b} J(w, b) \\ w = \text{tmp_w} \\ b = \text{tmp_b} \end{array} \right\}$$

Incorrect

$$\left. \begin{array}{l} \text{tmp_w} = w - \alpha \frac{\partial}{\partial w} J(w, b) \\ \text{tmp_w} = w \\ \text{tmp_b} = b - \alpha \frac{\partial}{\partial b} J(\text{tmp_w}, b) \\ b = \text{tmp_b} \end{array} \right\}$$

Stanford
ONLINE

DeepLearning.AI



Training Linear Regression

Gradient Descent Intuition

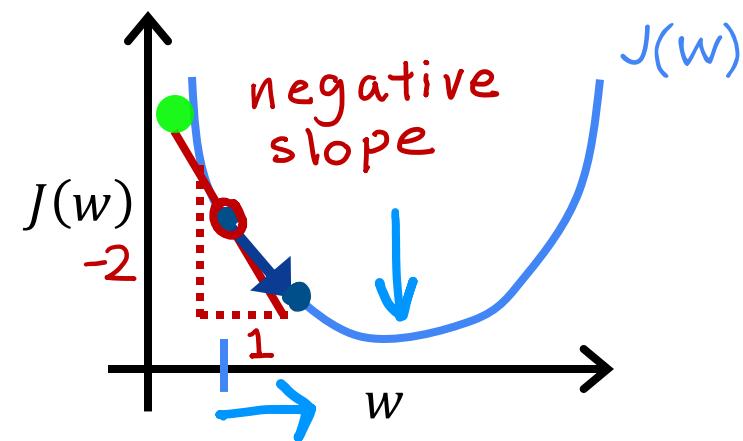
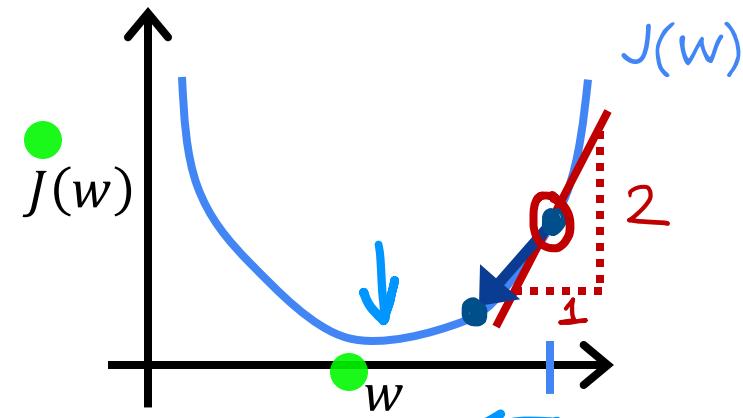
Gradient descent algorithm

- repeat until convergence {
 learning rate α
 $w = w - \alpha \frac{\partial}{\partial w} J(w, b)$ *derivative*
 $b = b - \alpha \frac{\partial}{\partial b} J(w, b)$

$$J(w)$$

$$w = w - \alpha \frac{\partial}{\partial w} J(w)$$

$$\min_w J(w)$$



$$w = w - \alpha \frac{\frac{d}{dw} J(w)}{> 0}$$

$w = w - \underline{\alpha} \cdot (\text{positive number})$

$$\frac{d}{dw} J(w) < 0$$

$w = w - \alpha \cdot (\text{negative number})$

Stanford
ONLINE

DeepLearning.AI



Training Linear Regression

Learning Rate

$$w = w - \alpha \frac{d}{dw} J(w)$$

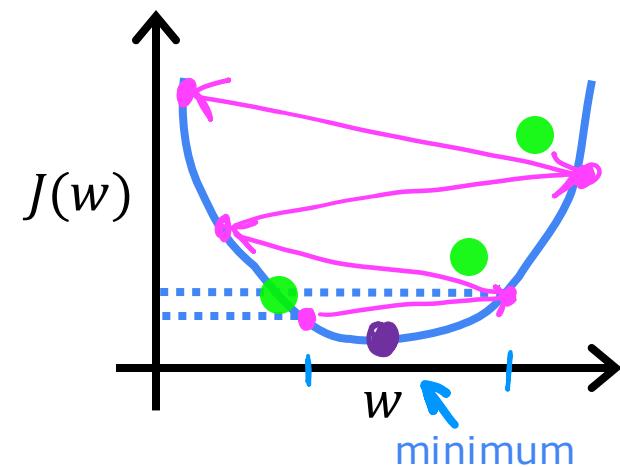
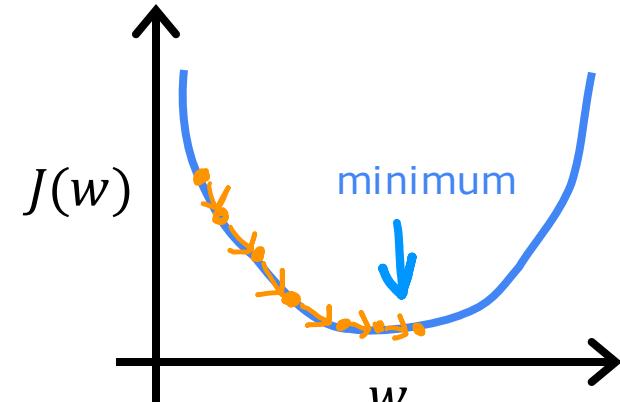
If α is too small...

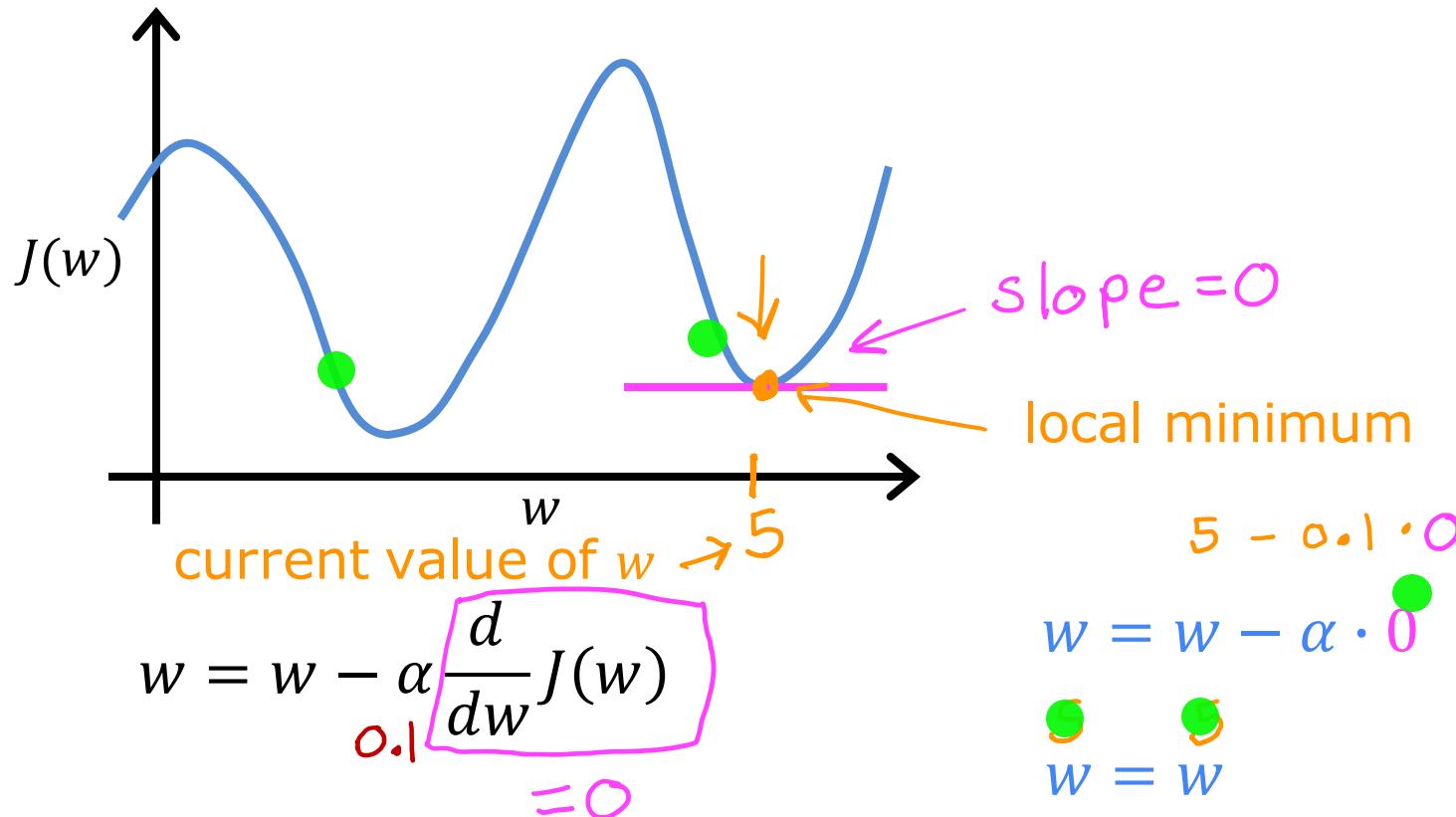
Gradient descent may be slow.

If α is too large...

Gradient descent may:

- Overshoot, never reach minimum
- Fail to converge, diverge





Can reach local minimum with fixed learning rate α

$$w = w - \alpha \frac{d}{dw} J(w)$$

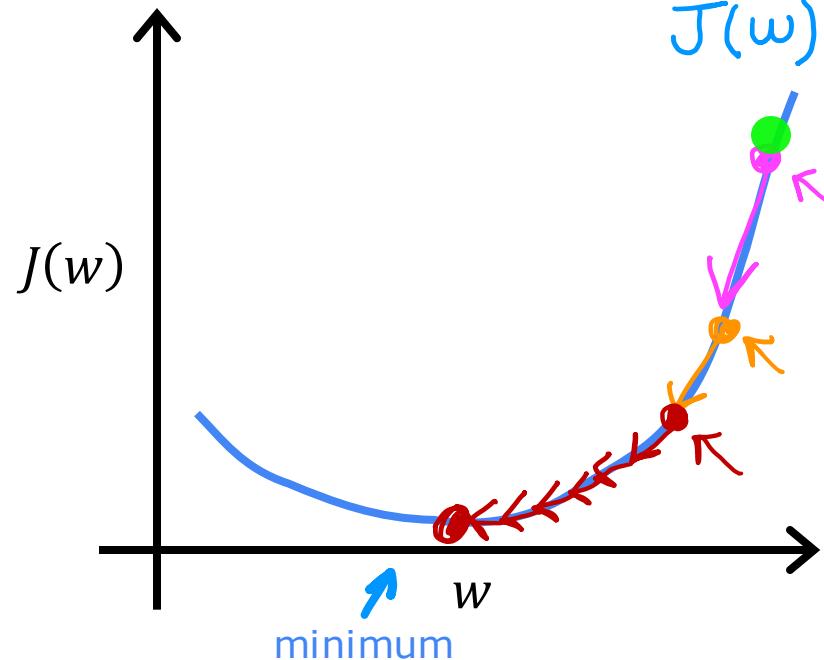
Diagram illustrating the effect of different learning rates α on the update step:

- smaller**: A small blue step.
- not as large**: An orange step.
- large**: A large red step.

Near a local minimum,

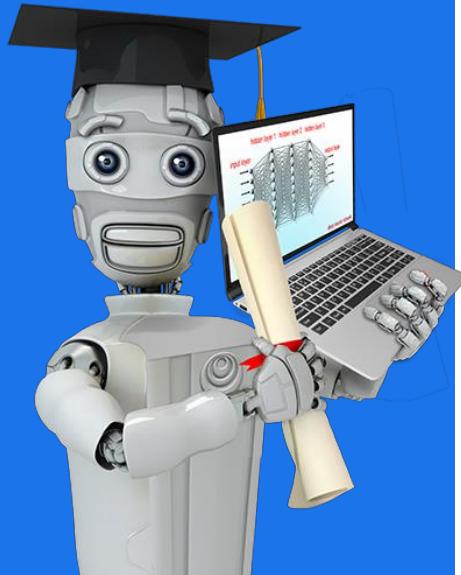
- Derivative becomes smaller
- Update steps become smaller

Can reach minimum without decreasing learning rate α



Stanford
ONLINE

DeepLearning.AI



Training Linear Regression

Gradient Descent
for Linear Regression

Linear regression model

$$f_{w,b}(x) = wx + b$$

Cost function

$$J(w, b) = \frac{1}{2m} \sum_{i=1}^m (f_{w,b}(x^{(i)}) - y^{(i)})^2$$

Gradient descent algorithm

repeat until convergence {

$$w = w - \alpha \frac{\partial}{\partial w} J(w, b)$$

$$b = b - \alpha \frac{\partial}{\partial b} J(w, b)$$

}

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

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

next slide
is optional!

(Optional)

$$\frac{\partial}{\partial w} J(w, b) = \frac{\partial}{\partial w} \frac{1}{2m} \sum_{i=1}^m (f_{w,b}(x^{(i)}) - y^{(i)})^2 = \frac{\partial}{\partial w} \frac{1}{2m} \sum_{i=1}^m (\underline{wx^{(i)} + b} - y^{(i)})^2$$

$$= \cancel{\frac{1}{2m}} \sum_{i=1}^m (\underline{wx^{(i)} + b} - y^{(i)}) \cancel{2x^{(i)}} = \boxed{\frac{1}{m} \sum_{i=1}^m (f_{w,b}(x^{(i)}) - y^{(i)})x^{(i)}}$$

$$\frac{\partial}{\partial b} J(w, b) = \frac{\partial}{\partial b} \frac{1}{2m} \sum_{i=1}^m (f_{w,b}(x^{(i)}) - y^{(i)})^2 = \frac{\partial}{\partial b} \frac{1}{2m} \sum_{i=1}^m (\underline{wx^{(i)} + b} - y^{(i)})^2$$

$$= \cancel{\frac{1}{2m}} \sum_{i=1}^m (\underline{wx^{(i)} + b} - y^{(i)}) \cancel{2} = \boxed{\frac{1}{m} \sum_{i=1}^m (f_{w,b}(x^{(i)}) - y^{(i)})}$$

no $x^{(i)}$

Gradient descent algorithm

repeat until convergence {

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

}

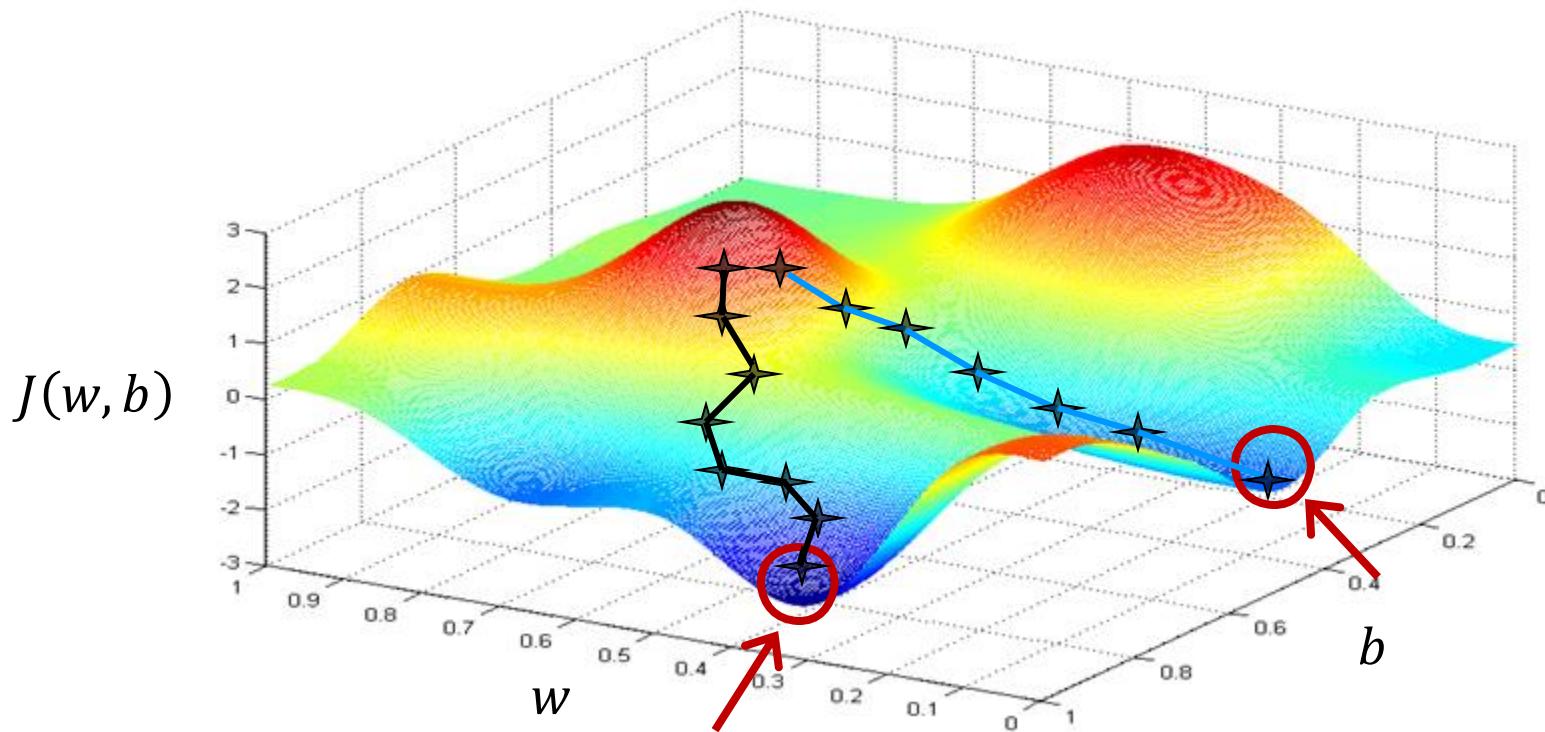
Update w and b simultaneously

$\frac{\partial}{\partial w} J(w, b)$

$\frac{\partial}{\partial b} J(w, b)$

$f_{w,b}(x^{(i)}) = wx^{(i)} + b$

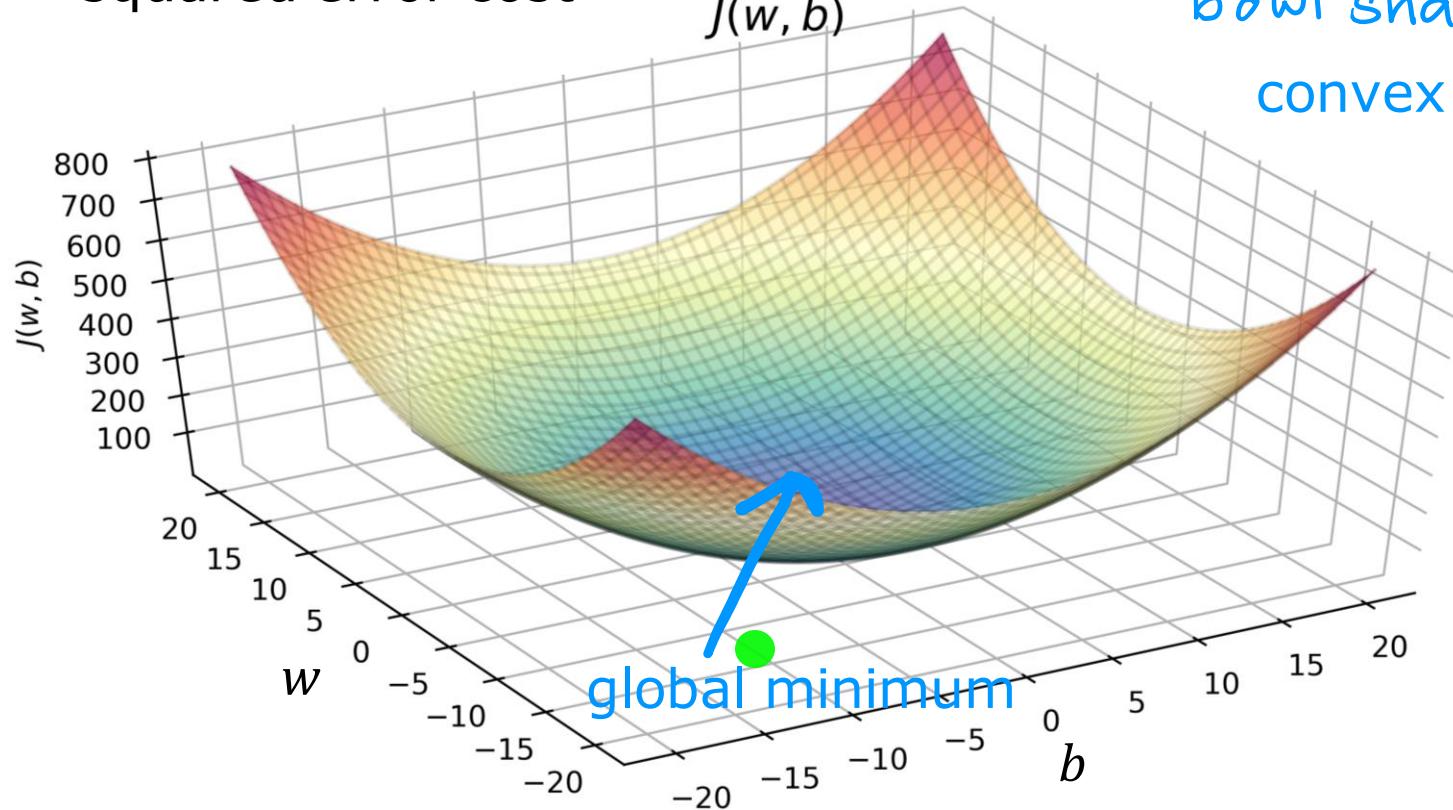
More than one local minimum



squared error cost

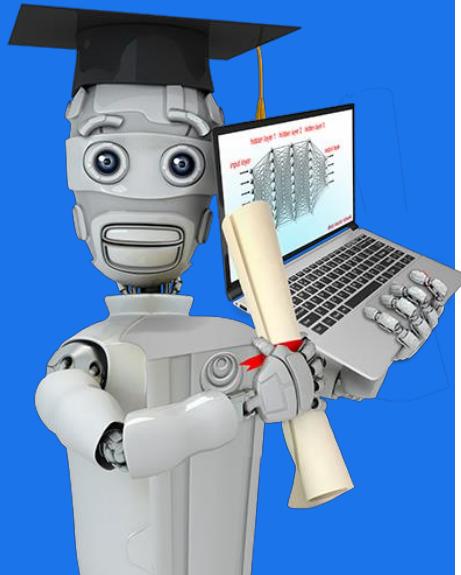
$$J(w, b)$$

bowl shape
convex function



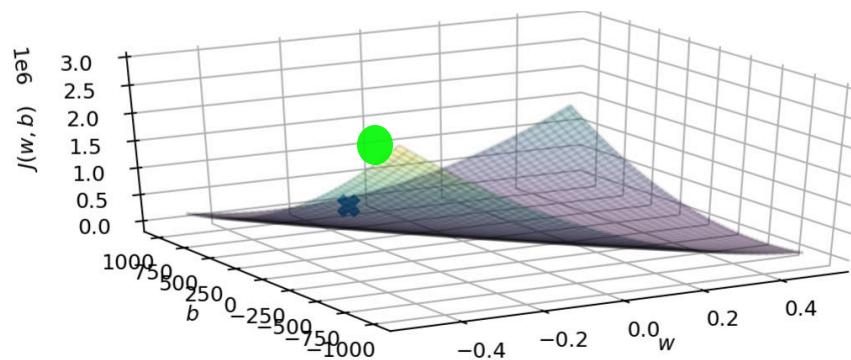
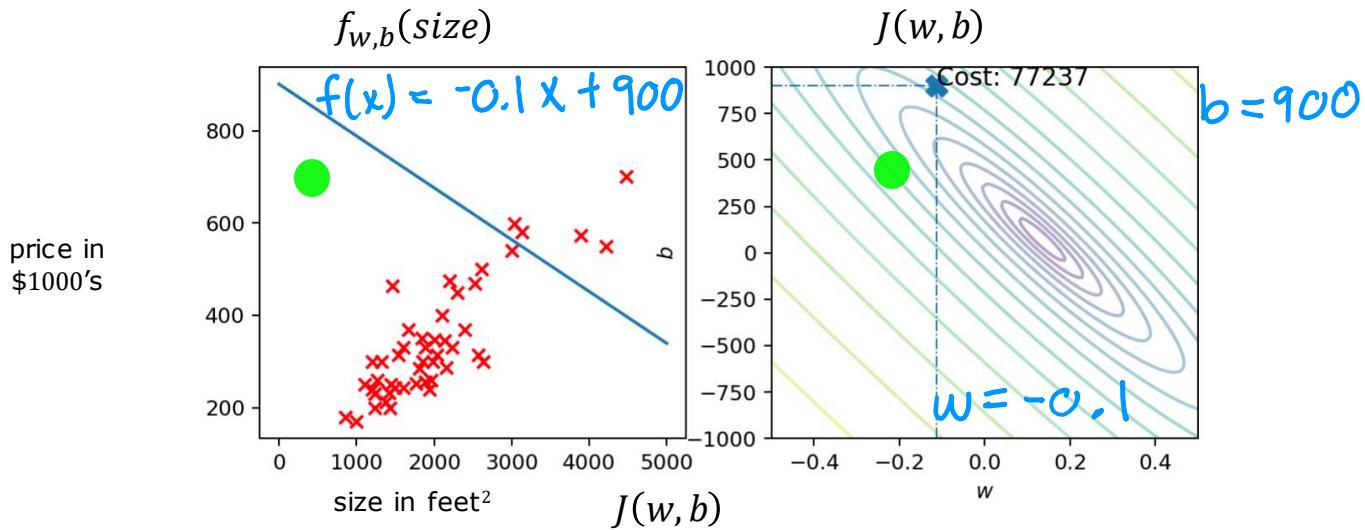
Stanford
ONLINE

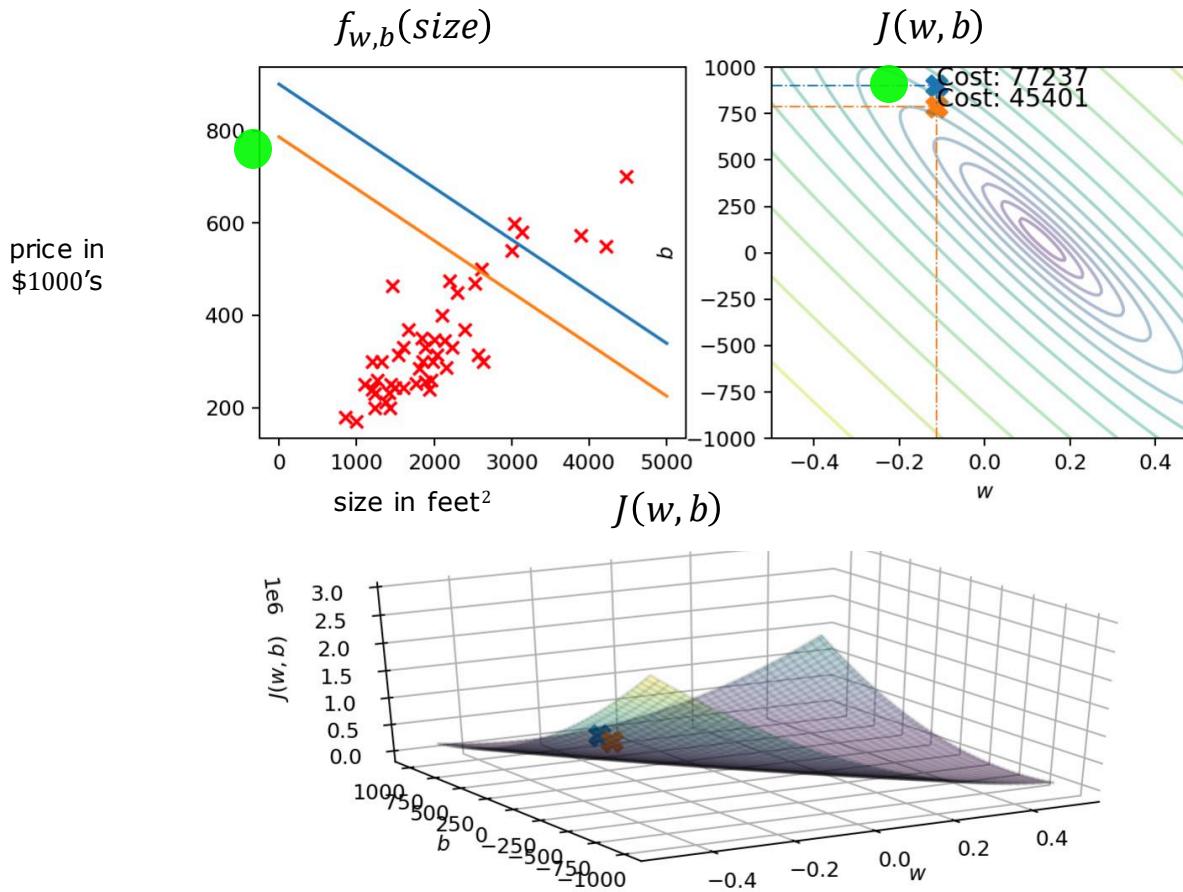
DeepLearning.AI



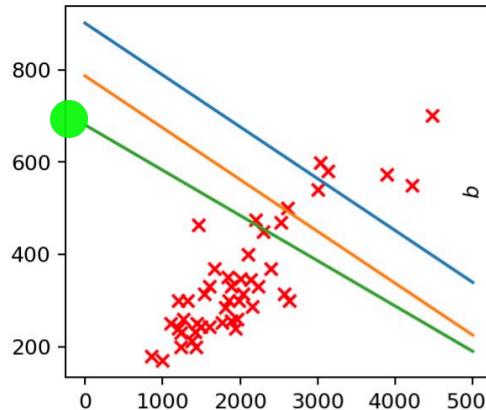
Training Linear Regression

Running Gradient Descent

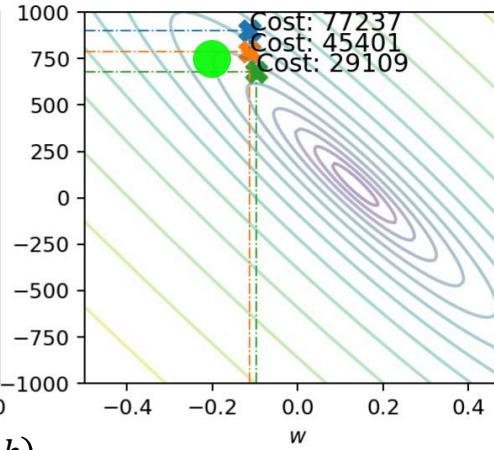




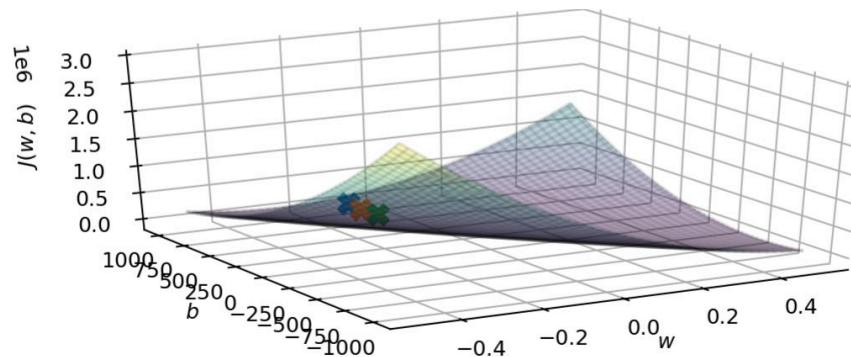
$f_{w,b}(\text{size})$

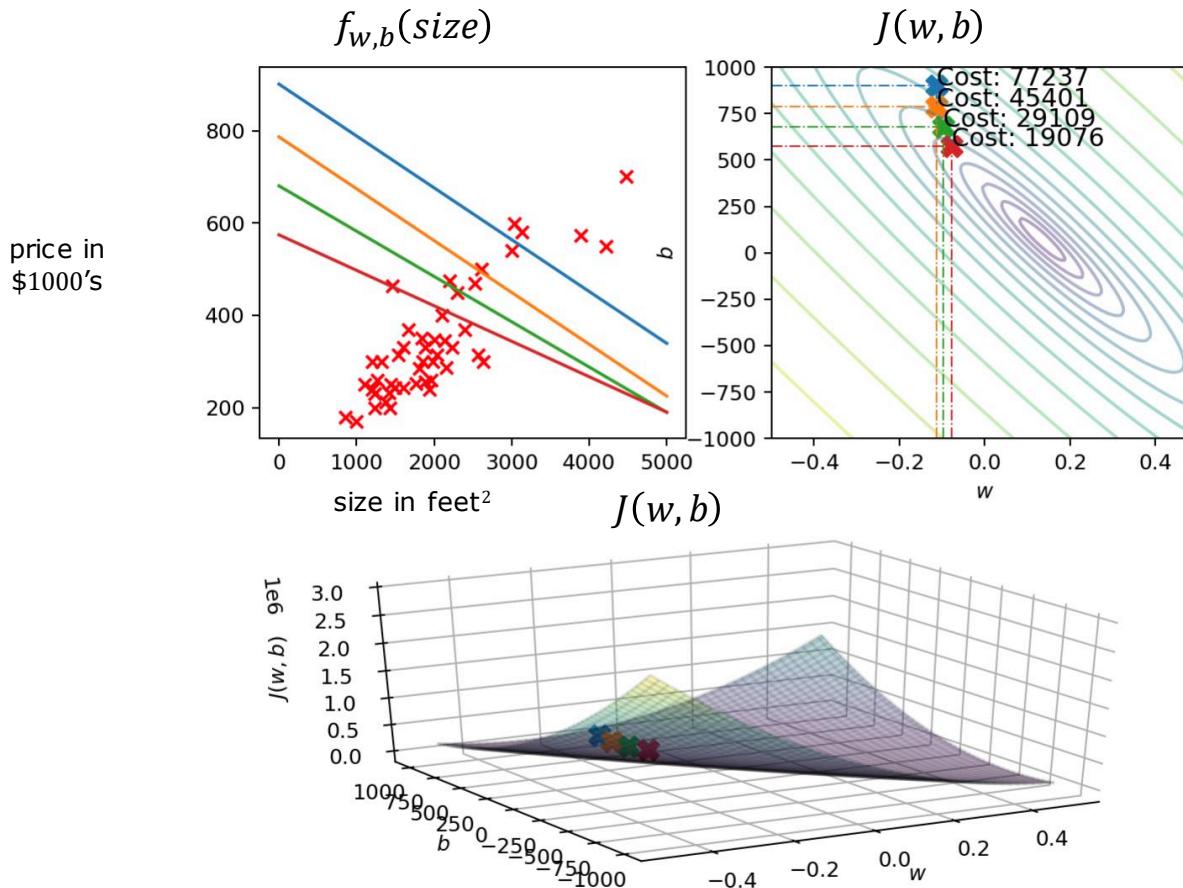


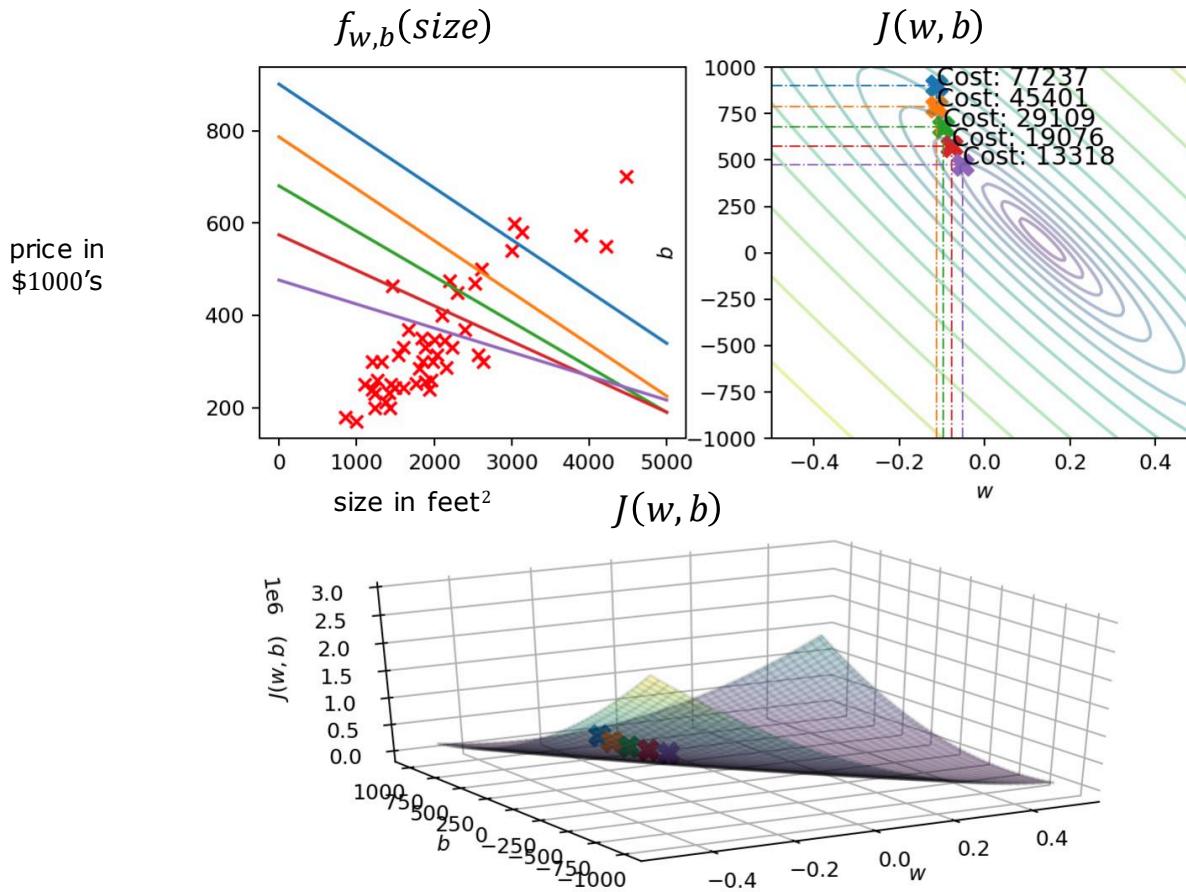
$J(w, b)$

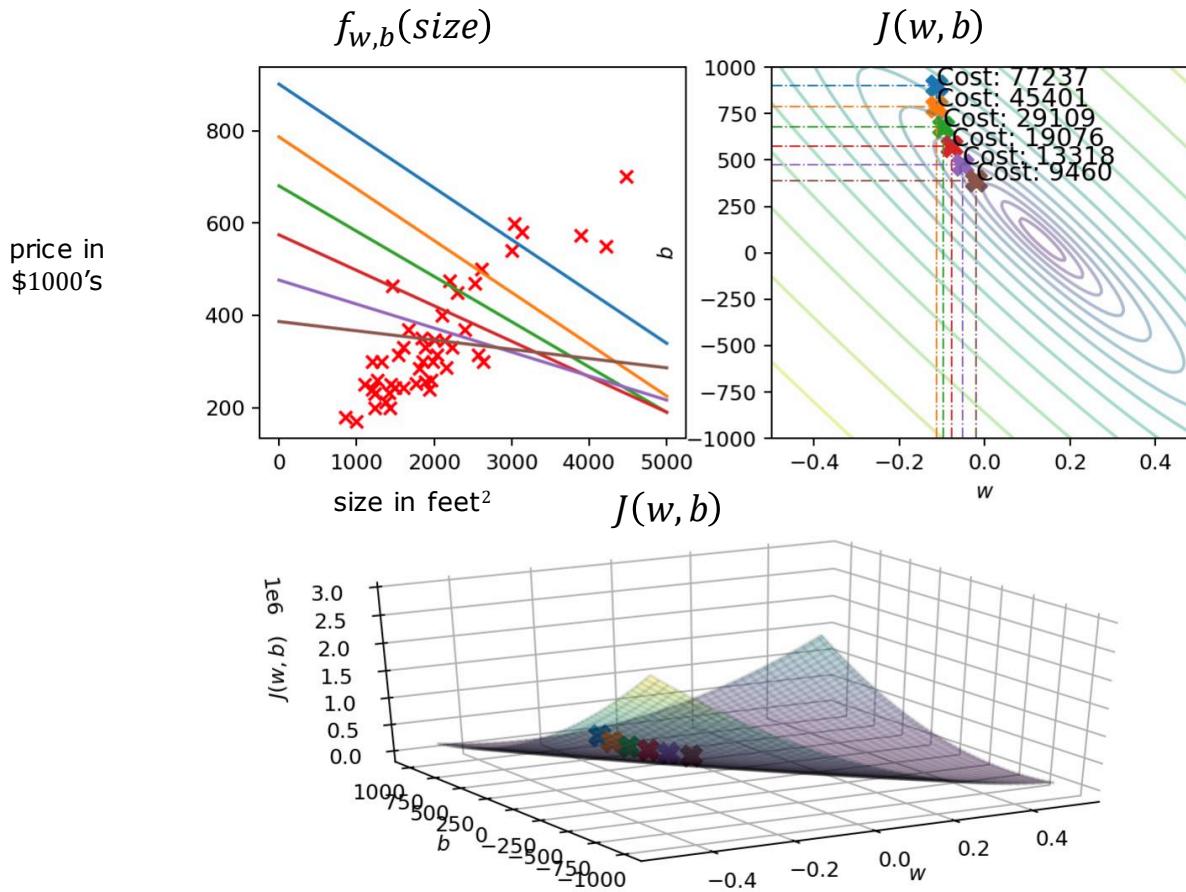


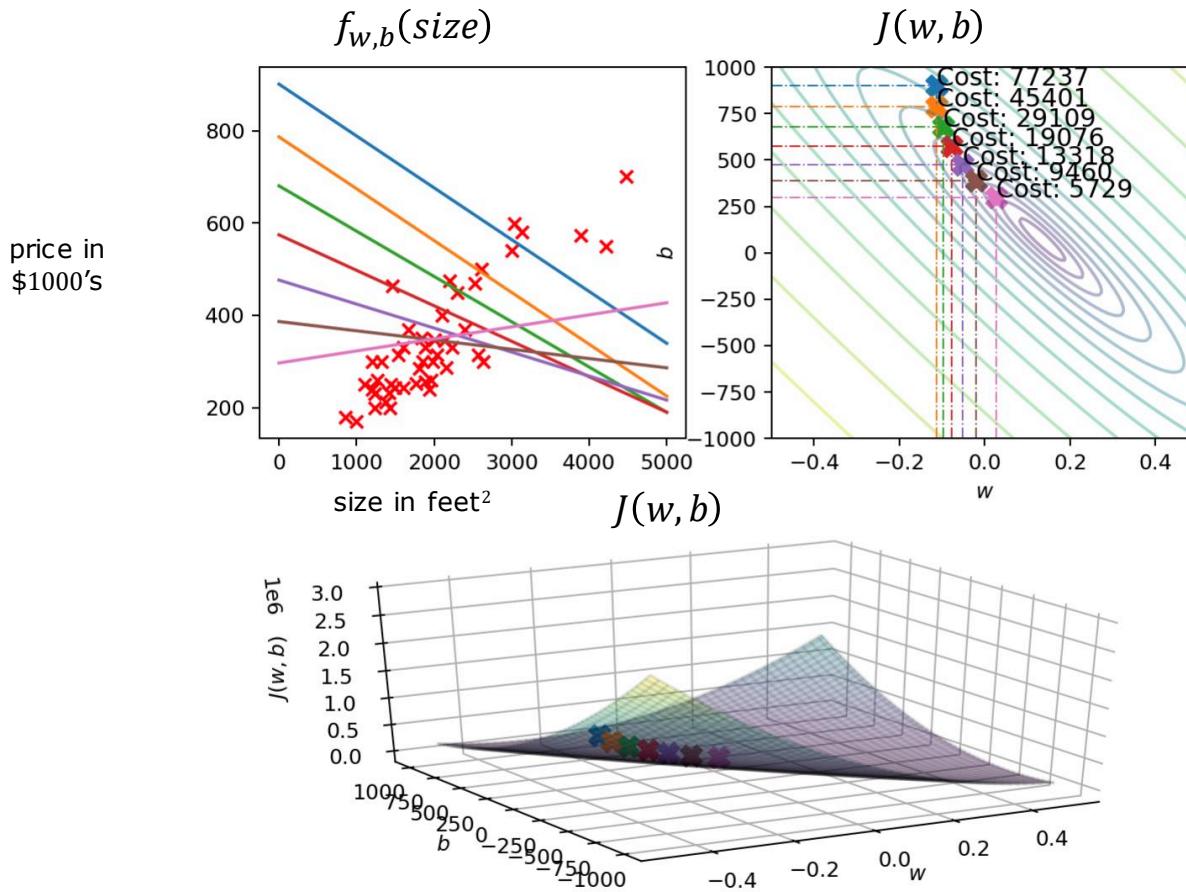
$J(w, b)$

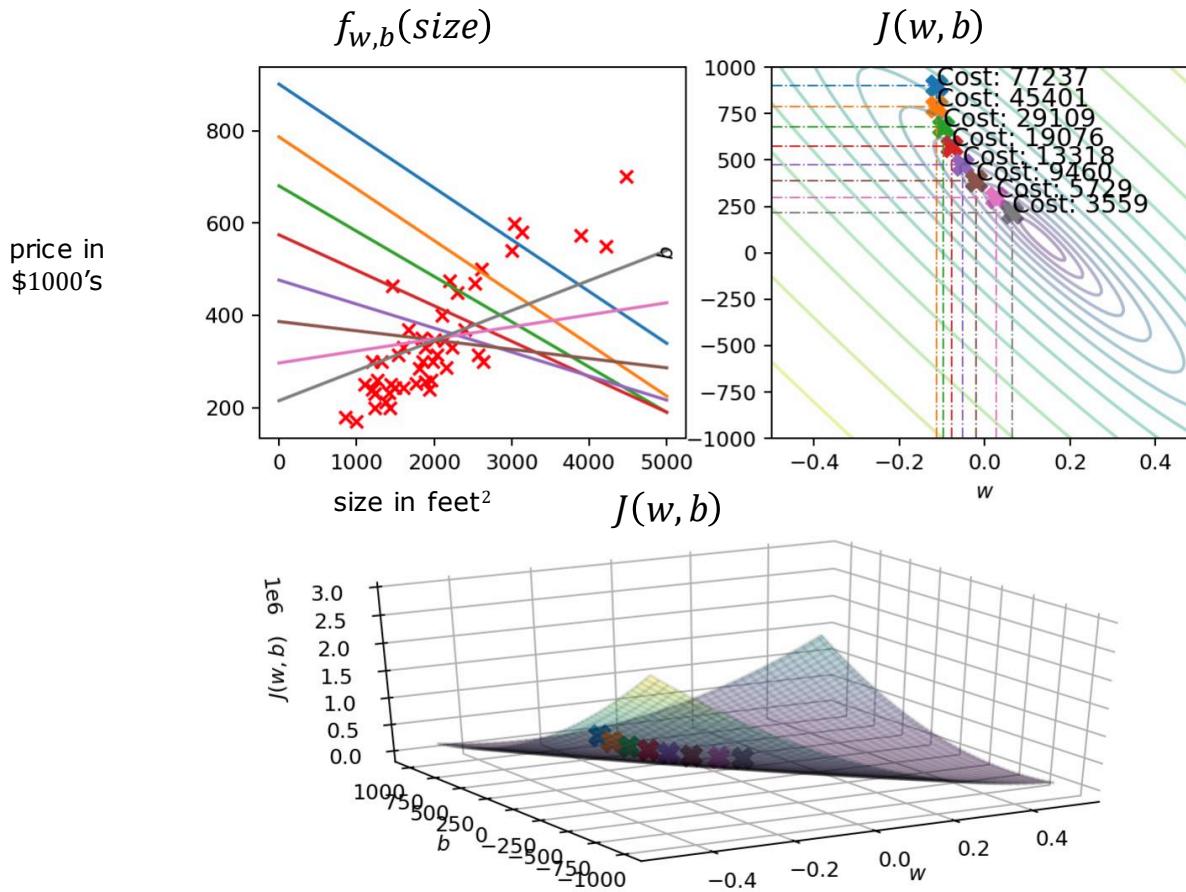


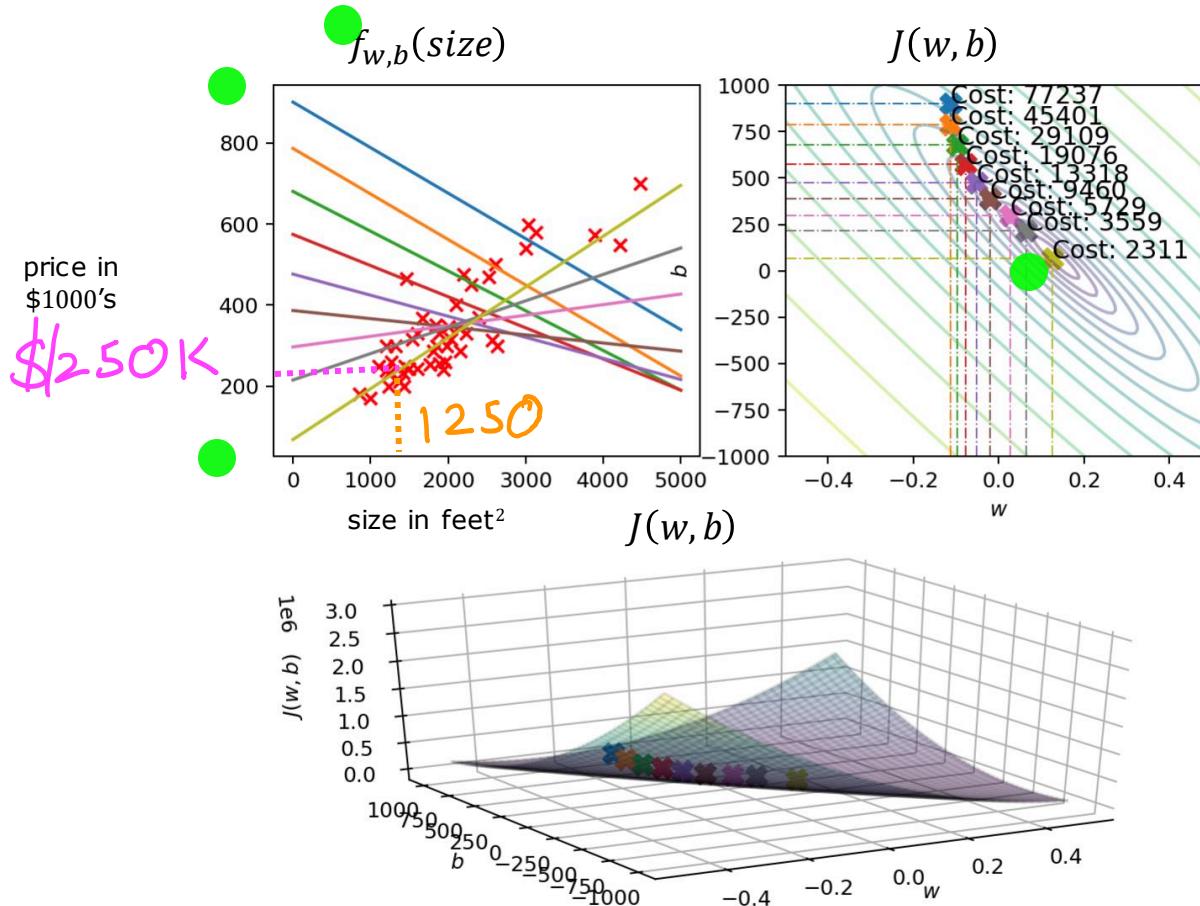












“Batch” gradient descent

“Batch”: Each step of gradient descent uses all the training examples.

other gradient descent: subsets

