Lecture 1

Machine Learning Basics

Sung Kim <hunkim+ml@gmail.com>

Basic concepts

- What is ML?
- What is learning?
  - supervised
  - unsupervised
- What is regression?
- What is classification?

Machine Learning

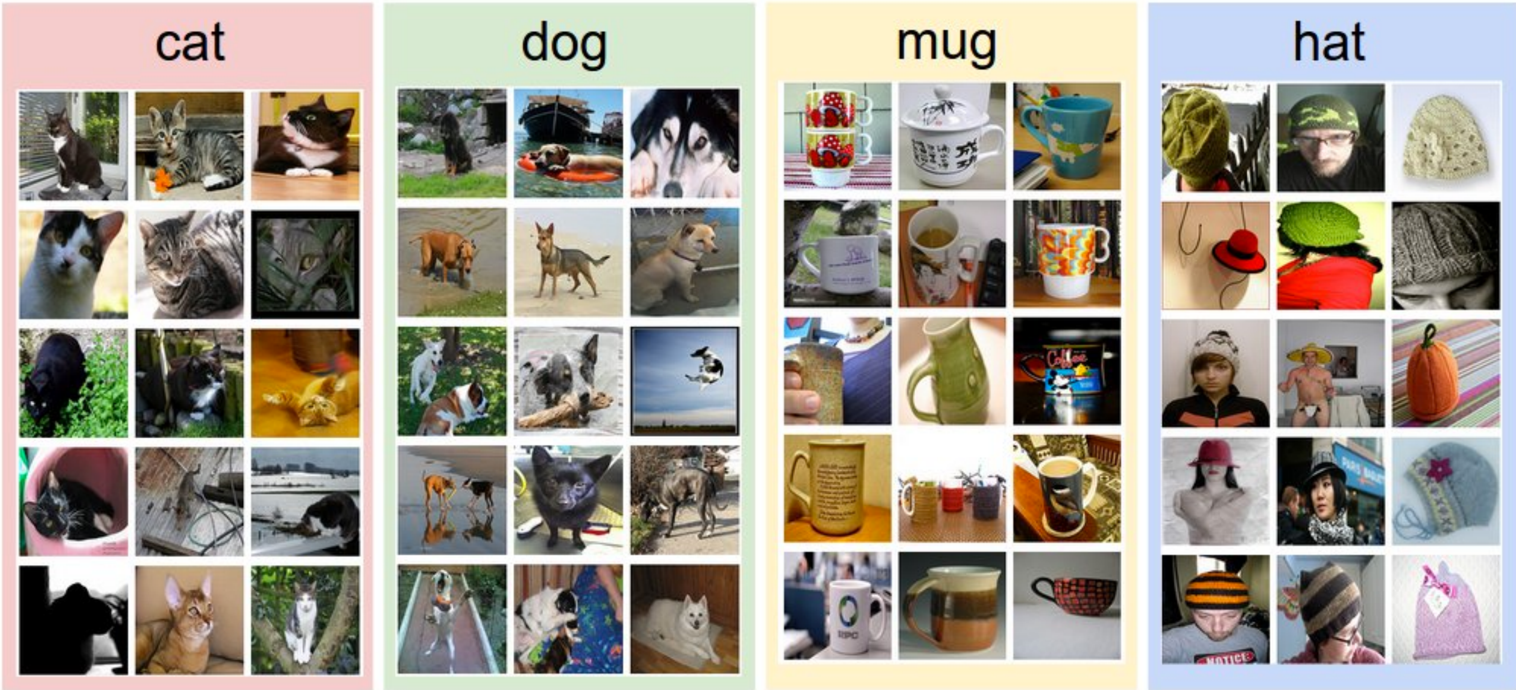
- Limitations of explicit programming
  - Spam filter: many rules
  - Automatic driving: too many rules
- Machine learning: "Field of study that gives computers the ability to learn without being explicitly programmed” Arthur Samuel (1959)

Supervised/Unsupervised learning

- Supervised learning:
  - learning with labeled examples - training set

Supervised learning

An example training set for four visual categories.



http://cs231n.github.io/classification/

Supervised/Unsupervised learning

- Supervised learning:
  - learning with labeled examples
- Unsupervised learning: un-labeled data
  - Google news grouping
  - Word clustering

Supervised learning

- Most common problem type in ML
  - **Image labeling:** learning from tagged images
  - **Email spam filter**: learning from labeled (spam or ham) email
  - **Predicting exam score**: learning from previous exam score and time spent

Training data set

AlphaGo

Types of supervised learning

- Predicting final exam score based on time spent
  - regression
- Pass/non-pass based on time spent
  - binary classification
- Letter grade (A, B, C, E and F) based on time spent
  - multi-label classification

Predicting final exam score based on time spent

x (hours)	y (score)
10	90
9	80
3	50
2	30

Pass/non-pass based on time spent

x (hours)	y (pass/fail)
10	P
9	P
3	F
2	F

Letter grade (A, B, …) based on time spent

x (hours)	y (grade)
10	A
9	B
3	D
2	F

Next
Linear regression

Lecture 2

Linear Regression

Sung Kim <hunkim+ml@gmail.com>

Acknowledgement

- Andrew Ng's ML class
  - https://class.coursera.org/ml-003/lecture
  - http://www.holehouse.org/mlclass/ (note)
- Convolutional Neural Networks for Visual Recognition.
  - http://cs231n.github.io/
- Tensorflow
  - https://www.tensorflow.org
  - https://github.com/aymericdamien/TensorFlow-Examples

Predicting exam score: regression

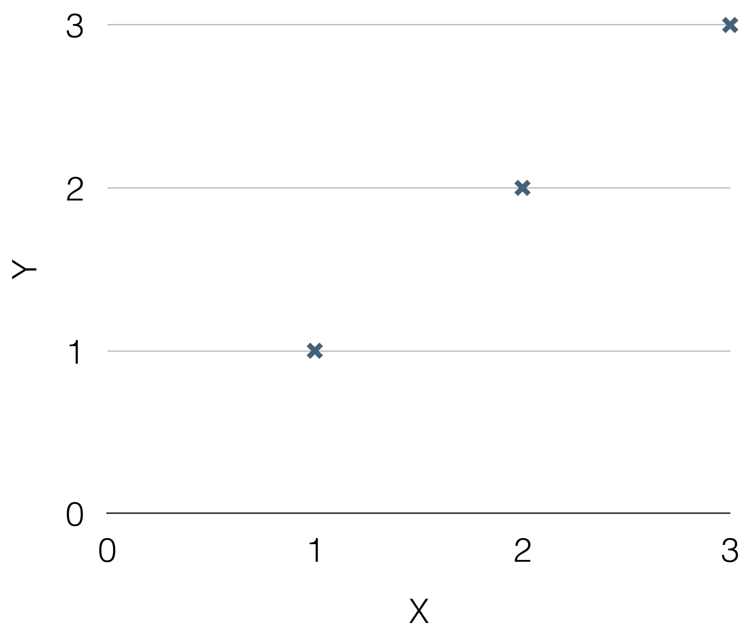
x (hours)	y (score)
10	90
9	80
3	50
2	30

Regression (data)

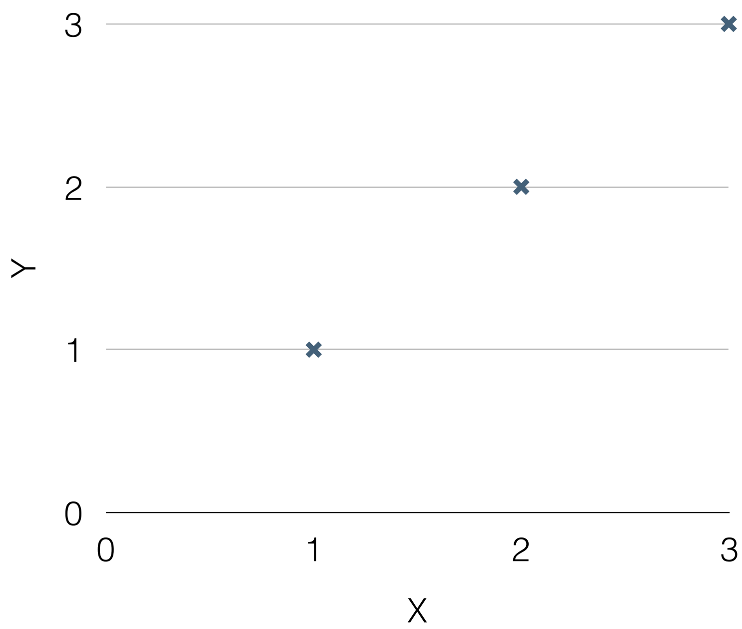
x	y
1	1
2	2
3	3

Regression (presentation)

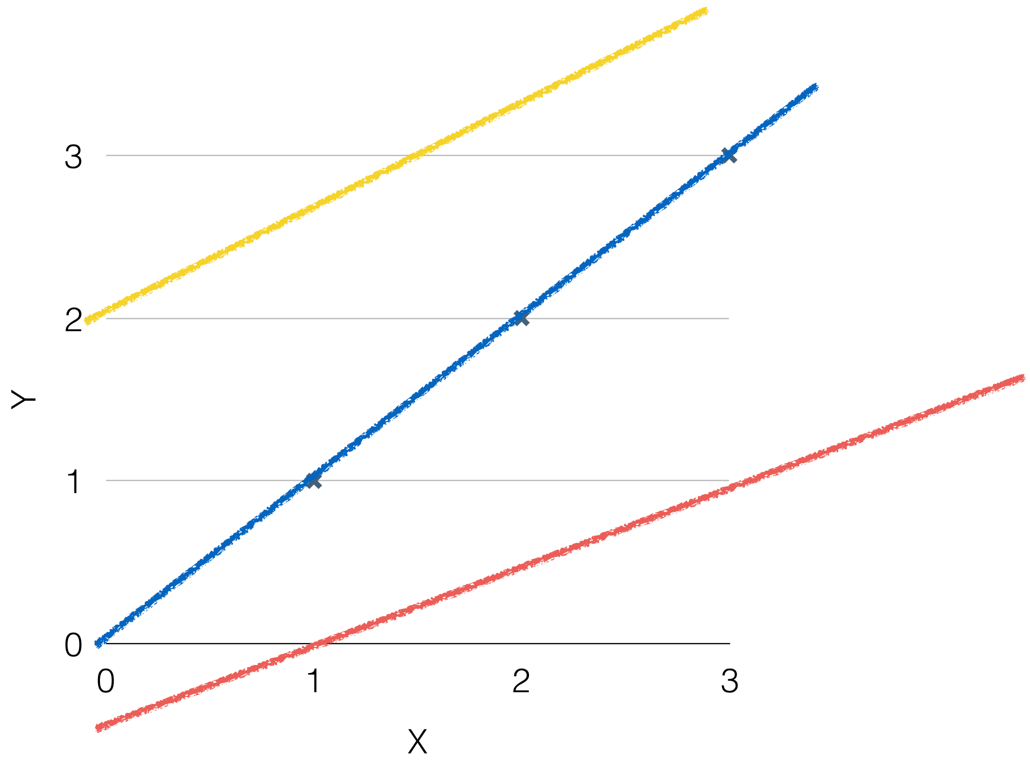
X	Y
1	1
2	2
3	3



(Linear) Hypothesis

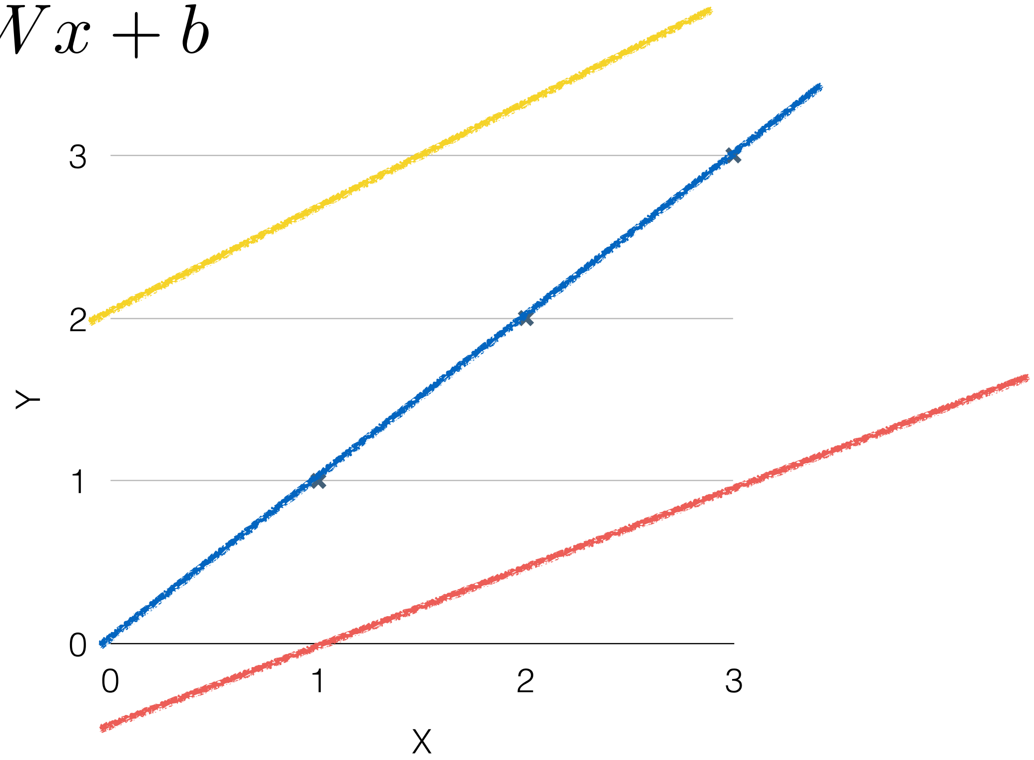


(Linear) Hypothesis

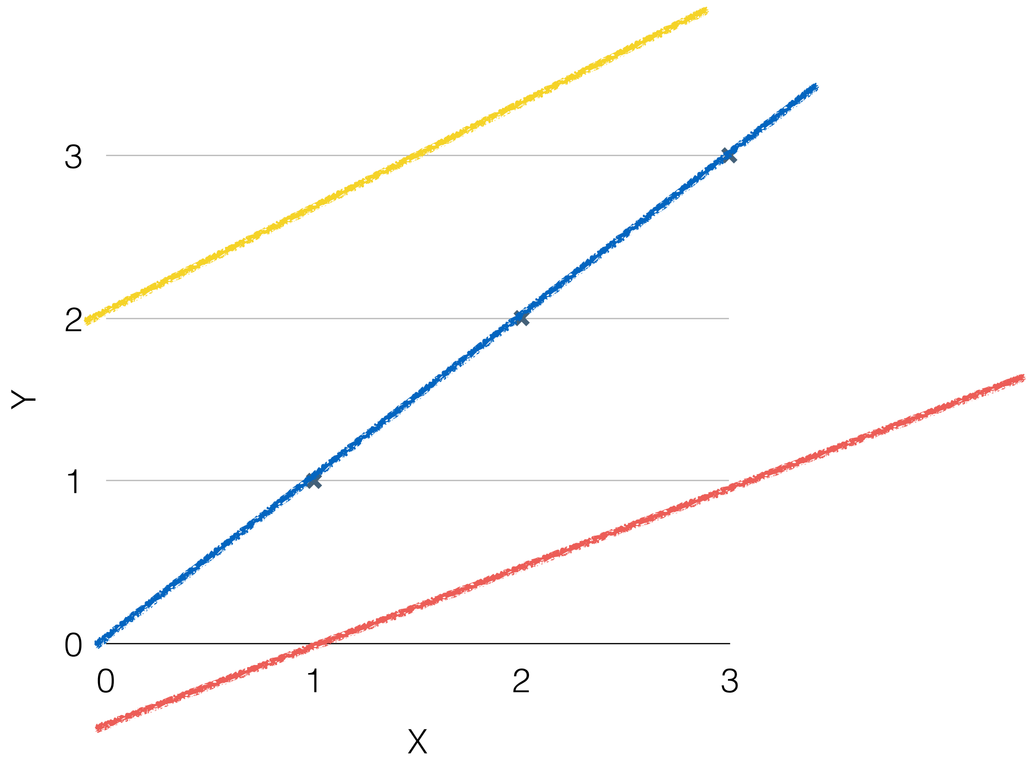


(Linear) Hypothesis

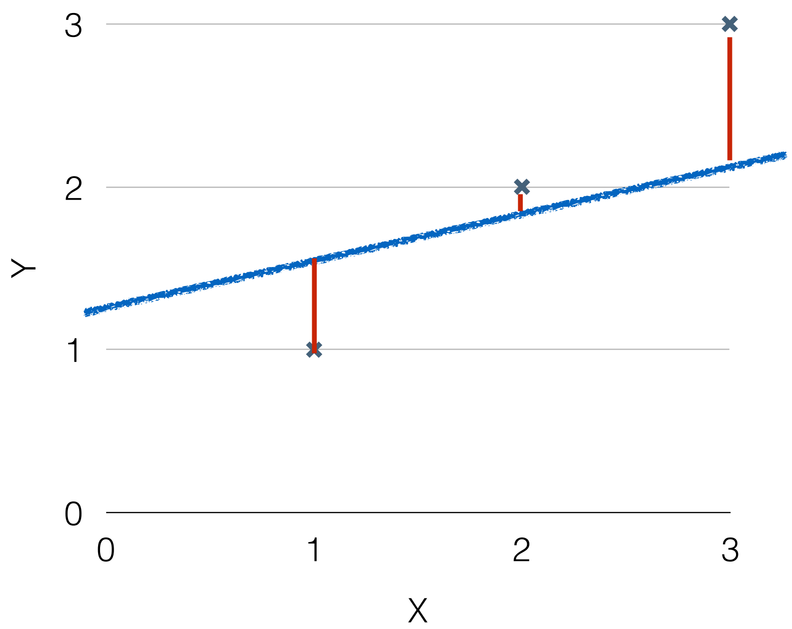
$$H(x) = Wx + b$$



Which hypothesis is better?



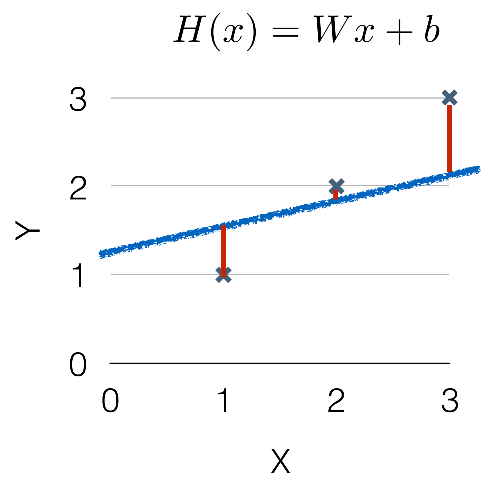
Which hypothesis is better?



Cost function

- How fit the line to our (training) data

$$H(x) - y$$



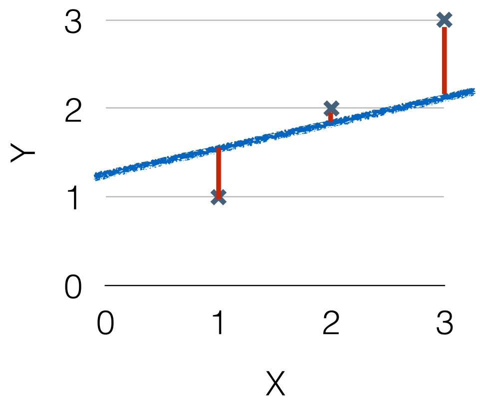
Cost function

- How fit the line to our (training) data

$$\frac{(H(x^{(1)}) - y^{(1)})^2 + (H(x^{(2)}) - y^{(2)})^2 + (H(x^{(3)}) - y^{(3)})^2}{3}$$

$$cost = \frac{1}{m} \sum_{i=1}^{m} (H(x^{(i)}) - y^{(i)})^2$$

$$H(x) = Wx + b$$



Cost function

$$cost = \frac{1}{m} \sum_{i=1}^{m} (H(x^{(i)}) - y^{(i)})^2$$

$$H(x) = Wx + b$$

$$cost(W, b) = \frac{1}{m} \sum_{i=1}^{m} (H(x^{(i)}) - y^{(i)})^2$$

Goal: Minimize cost

$$\underset{W,b}{\text{minimize}}\, cost(W, b)$$

Lecture 3

How to minimize cost

Sung Kim <hunkim+mr@gmail.com>

Acknowledgement

- Andrew Ng's ML class
  - https://class.coursera.org/ml-003/lecture
  - http://www.holehouse.org/mlclass/ (note)
- Convolutional Neural Networks for Visual Recognition.
  - http://cs231n.github.io/
- Tensorflow
  - https://www.tensorflow.org
  - https://github.com/aymericdamien/TensorFlow-Examples

Hypothesis and Cost

$$H(x) = Wx + b$$

$$cost(W, b) = \frac{1}{m} \sum_{i=1}^{m} (H(x^{(i)}) - y^{(i)})^2$$

Simplified hypothesis

$$H(x) = Wx$$

$$cost(W) = \frac{1}{m} \sum_{i=1}^{m} (Wx^{(i)} - y^{(i)})^2$$

What *cost(W)* looks like?

$$cost(W) = \frac{1}{m}\sum_{i=1}^{m}(Wx^{(i)} - y^{(i)})^2$$

X	Y
1	1
2	2
3	3

- *W*=1, *cost(W)*=?

What *cost(W)* looks like?

$$cost(W) = \frac{1}{m} \sum_{i=1}^{m} (Wx^{(i)} - y^{(i)})^2$$

X	Y
1	1
2	2
3	3

- W=1, cost(W)=0

$$\frac{1}{3}((1 * 1 - 1)^2 + (1 * 2 - 2)^2 + (1 * 3 - 3)^2)$$

- W=0, cost(W)=4.67

$$\frac{1}{3}((0 * 1 - 1)^2 + (0 * 2 - 2)^2 + (0 * 3 - 3)^2)$$

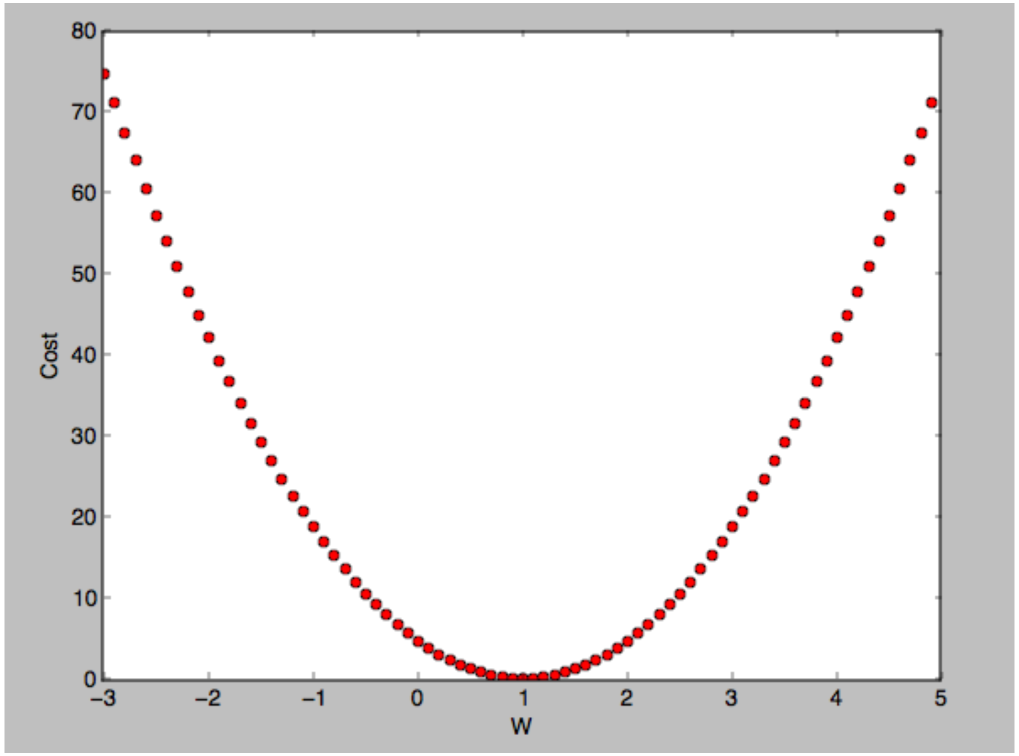
- W=2, cost(W)=?

What *cost(W)* looks like?

- W=1, cost(W)=0
- W=0, cost(W)=4.67
- W=2, cost(W)=4.67

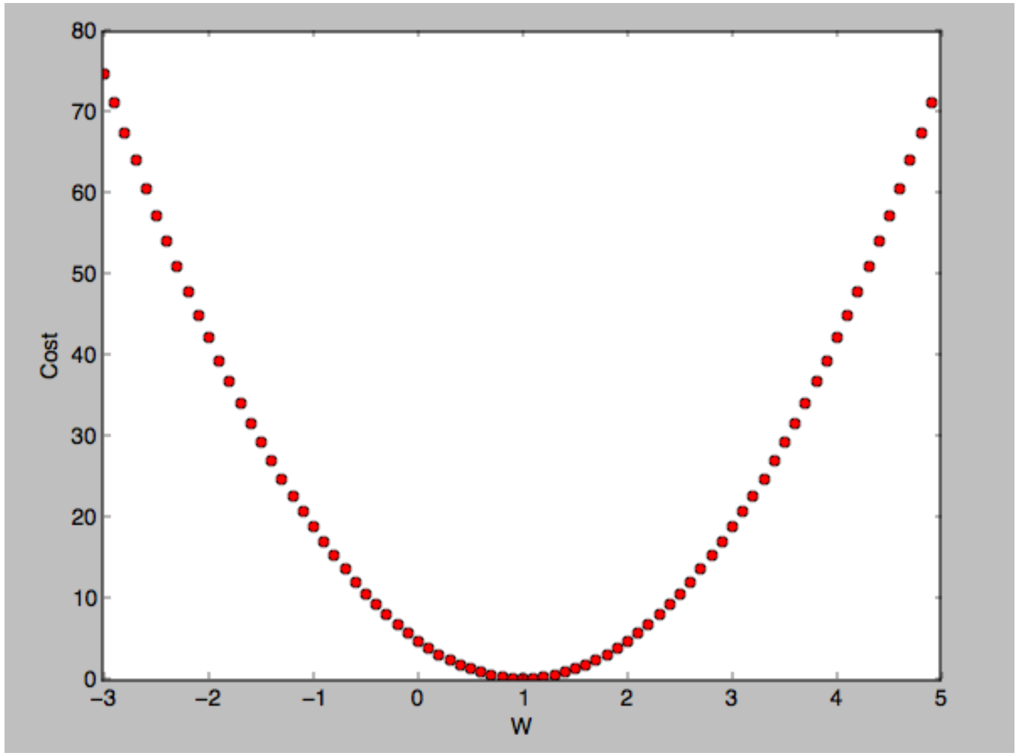
What *cost(W)* looks like?

$$cost(W) = \frac{1}{m}\sum_{i=1}^{m}(Wx^{(i)} - y^{(i)})^2$$



How to minimize cost?

$$cost(W) = \frac{1}{m}\sum_{i=1}^{m}(Wx^{(i)} - y^{(i)})^2$$

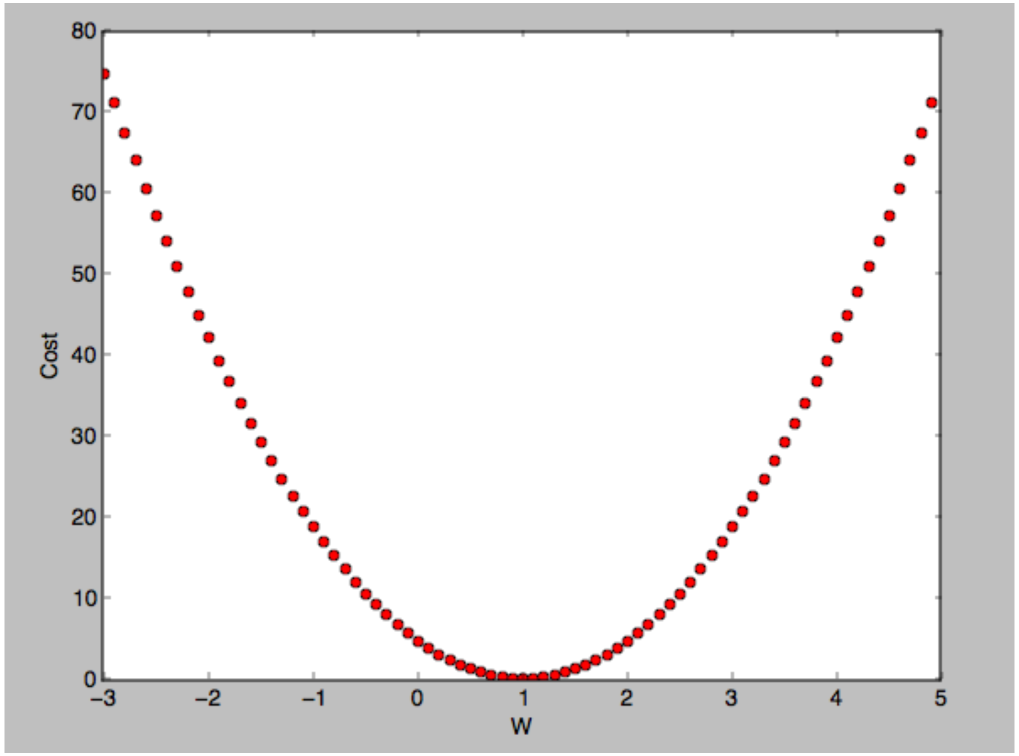


Gradient descent algorithm

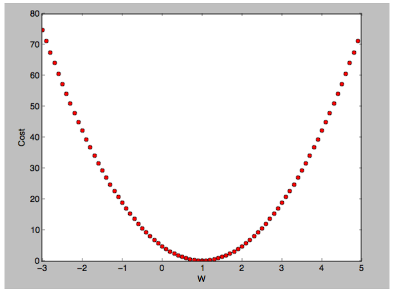
- Minimize cost function
- Gradient descent is used many minimization problems
- For a given cost function, *cost (W, b)*, it will find *W, b* to minimize cost
- It can be applied to more general function: *cost (w1, w2, …)*

How it works?

How would you find the lowest point?



How it works?



- Start with initial guesses
  - Start at 0,0 (or any other value)
  - Keeping changing W and b a little bit to try and reduce cost(W, b)
- Each time you change the parameters, you select the gradient which reduces cost(W, b) the most possible
- Repeat
- Do so until you converge to a local minimum
- Has an interesting property
  - Where you start can determine which minimum you end up

http://www.holehouse.org/mlclass/01_02_Introduction_regression_analysis_and_gr.html

Formal definition

$$cost(W) = \frac{1}{m}\sum_{i=1}^{m}(Wx^{(i)} - y^{(i)})^2$$

$$cost(W) = \frac{1}{2m}\sum_{i=1}^{m}(Wx^{(i)} - y^{(i)})^2$$

Formal definition

$$cost(W) = \frac{1}{2m} \sum_{i=1}^{m} (Wx^{(i)} - y^{(i)})^2$$

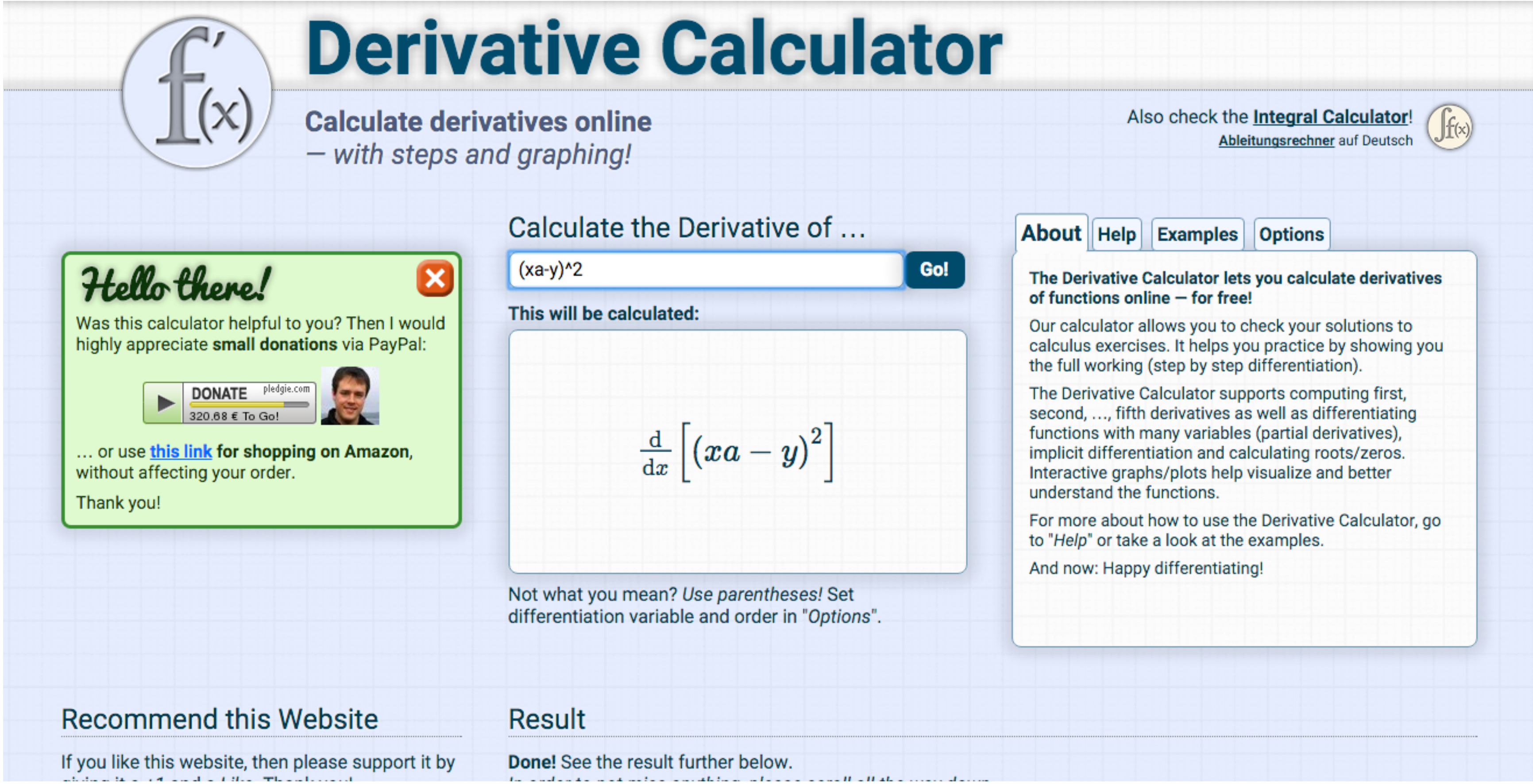
$$W := W - \alpha \frac{\partial}{\partial W} cost(W)$$

Formal definition

$$W := W - \alpha \frac{\partial}{\partial W} \frac{1}{2m} \sum_{i=1}^{m} (Wx^{(i)} - y^{(i)})^2$$

$$W := W - \alpha \frac{1}{2m} \sum_{i=1}^{m} 2(Wx^{(i)} - y^{(i)})x^{(i)}$$

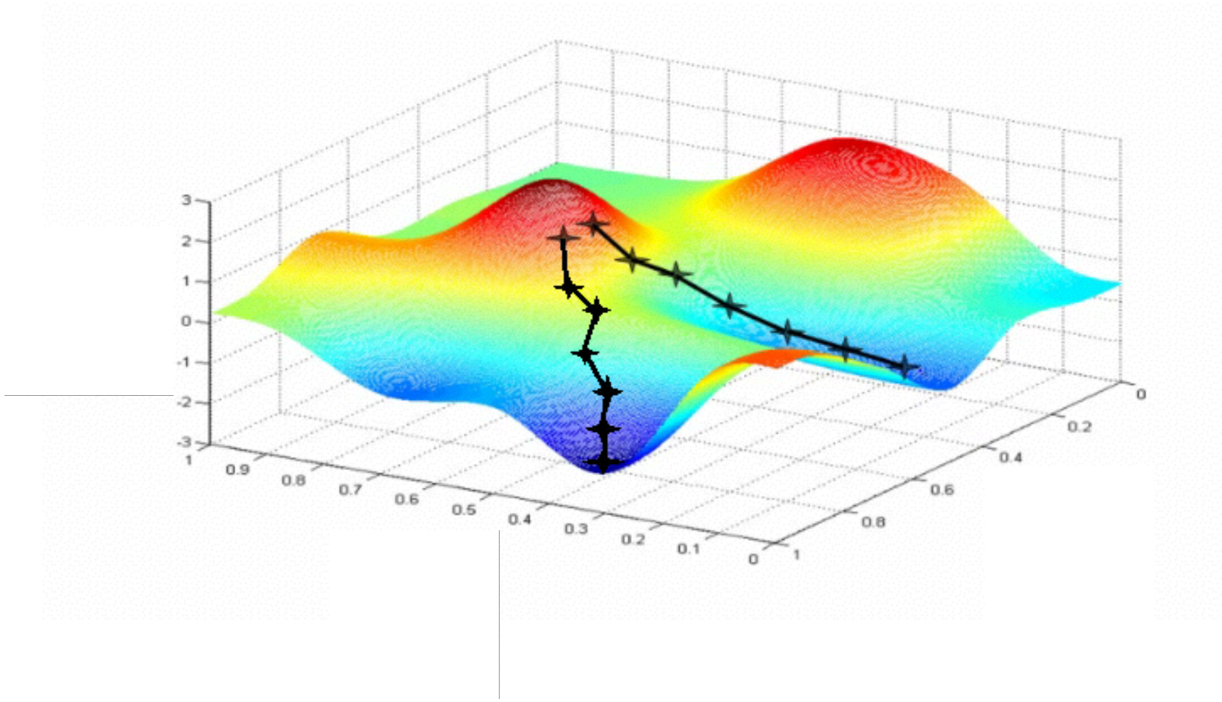
$$W := W - \alpha \frac{1}{m} \sum_{i=1}^{m} (Wx^{(i)} - y^{(i)})x^{(i)}$$


Derivative Calculator
Calculate derivatives online
— with steps and graphing!
Also check the Integral Calculator!
Ableitungsrechner auf Deutsch
Hello there!
Was this calculator helpful to you? Then I would highly appreciate small donations via PayPal:
DONATE pledgie.com
320.68 € To Go!
… or use this link for shopping on Amazon, without affecting your order.
Thank you!
Calculate the Derivative of …
(xa-y)^2
Go!
This will be calculated:
$\frac{d}{dx}\left[(xa-y)^2\right]$
Not what you mean? Use parentheses! Set differentiation variable and order in "Options".
About
Help
Examples
Options
The Derivative Calculator lets you calculate derivatives of functions online — for free!
Our calculator allows you to check your solutions to calculus exercises. It helps you practice by showing you the full working (step by step differentiation).
The Derivative Calculator supports computing first, second, …, fifth derivatives as well as differentiating functions with many variables (partial derivatives), implicit differentiation and calculating roots/zeros. Interactive graphs/plots help visualize and better understand the functions.
For more about how to use the Derivative Calculator, go to "Help" or take a look at the examples.
And now: Happy differentiating!
Recommend this Website
If you like this website, then please support it by
Result
Done! See the result further below.

Gradient descent algorithm

$$W := W - \alpha \frac{1}{m} \sum_{i=1}^{m} (Wx^{(i)} - y^{(i)}) x^{(i)}$$

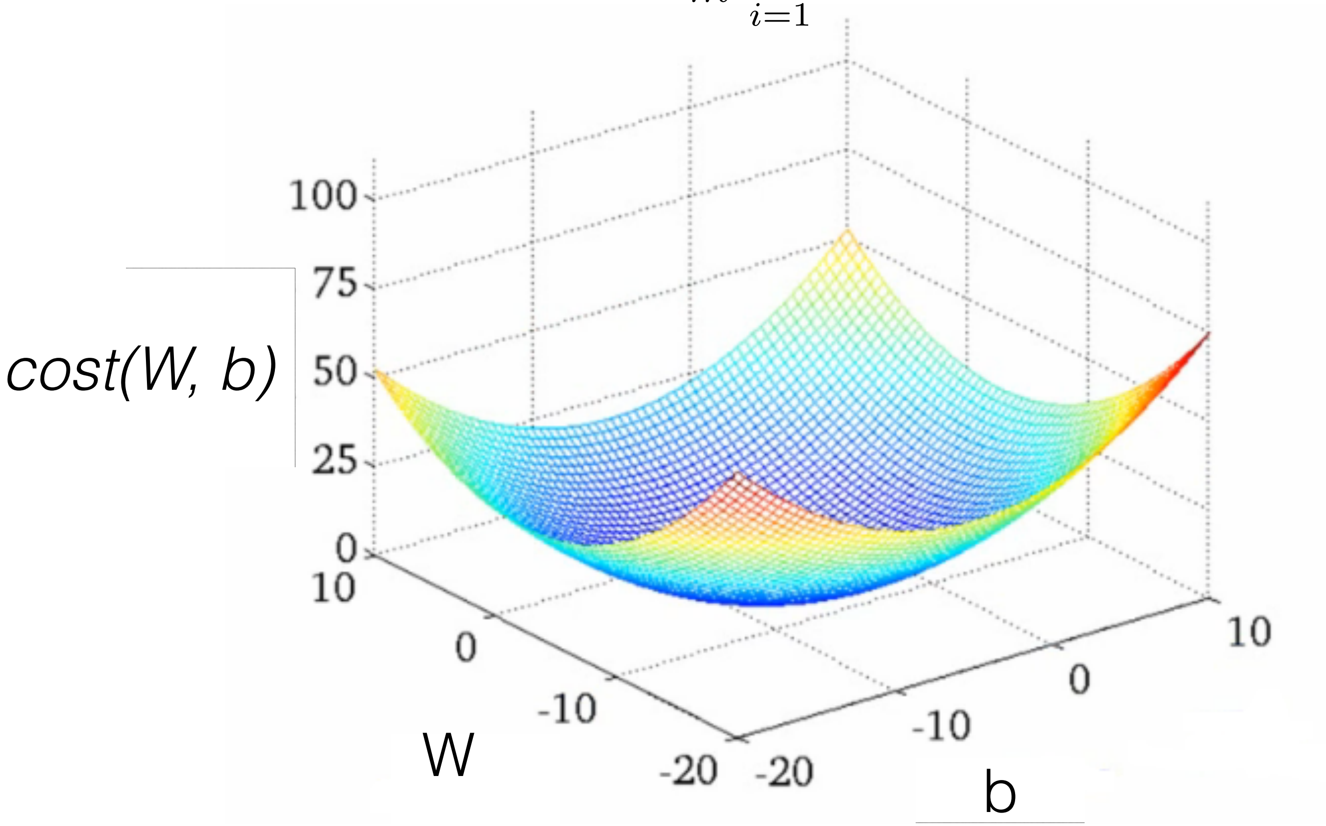
Convex function



www.holehouse.org/mlclass/

Convex function

$$cost(W, b) = \frac{1}{m} \sum_{i=1}^{m} (H(x^{(i)}) - y^{(i)})^2$$



www.holehouse.org/mlclass/


Next
Multivariable logistic
regression