Machine Learning

Under the Guidance of **Andrew Ng**

**Stanford** **University**

My Learning Journal

Name: Prakash Dahal

Date: 4/21/2019

Table of Contents

[1. Introduction: 3](#_Toc6999082)

[a. Machine Learning: 3](#_Toc6999083)

[i. Supervised Learning: 5](#_Toc6999084)

[ii. Un-Supervised Learning: 9](#_Toc6999085)

[2. Model and Cost Function: 12](#_Toc6999086)

[2.1. Model Representation 12](#_Toc6999087)

[2.2. Cost Function 15](#_Toc6999088)

[2.3. Cost Function Intuition I 18](#_Toc6999089)

[2.4. Cost Function Intuition II 21](#_Toc6999090)

[3. Parameter Learning: 23](#_Toc6999091)

[3.1. Gradient Descent 23](#_Toc6999092)

[3.2. Gradient Descent Intuition 25](#_Toc6999093)

[3.3. Gradient Descent for Linear Regression 25](#_Toc6999094)

[4. Linear Algebra: 26](#_Toc6999095)

# Introduction:

## Machine Learning:

We are using a learning algorithm dozen of time a day. Example; Google or Bing has learned to rank web pages. Facebook recognizes the picture of our friends, spam mail checking and so on.

AI researchers believe that the best way is the learning algorithms which tries to mimic the learning of human brain. Only math and algorithm are not enough unless we know how it works and where it is applicable. Machine learning is a field/part of AI.

Machine Learning

* Grew out of work in AI
* New capability for computers

Examples;

* Database mining (E.g.: Web click data, medical records, biology, engineering)
* Applications can’t program by hand (E.g.: Autonomous helicopter, handwritten recognition, most of Natural Language Processing (NLP), Computer vision)
* Self-customizing programs (E.g.: Amazon Netflix production recommendation)
* Understanding human learning (brain, real AI)

What is machine learning and when to use:

Though the definition of what is machine learning or what machine learning is not is not well accepted. Some of them have tried to define it.

* Arthur Samuel (1959) - Older informal definition

Machine Learning: “The field of study that gives computers the ability to learn without being explicitly programmed.”

He wrote checkers playing program though he was not good at it. Over the time, when he programmed maybe tens of thousands of games against himself, what sort of board position tended to lead to win or lose was learned. It eventually learned to play better than Arthur himself.

* Tom Mitchell (1998) -Modern Definiiton

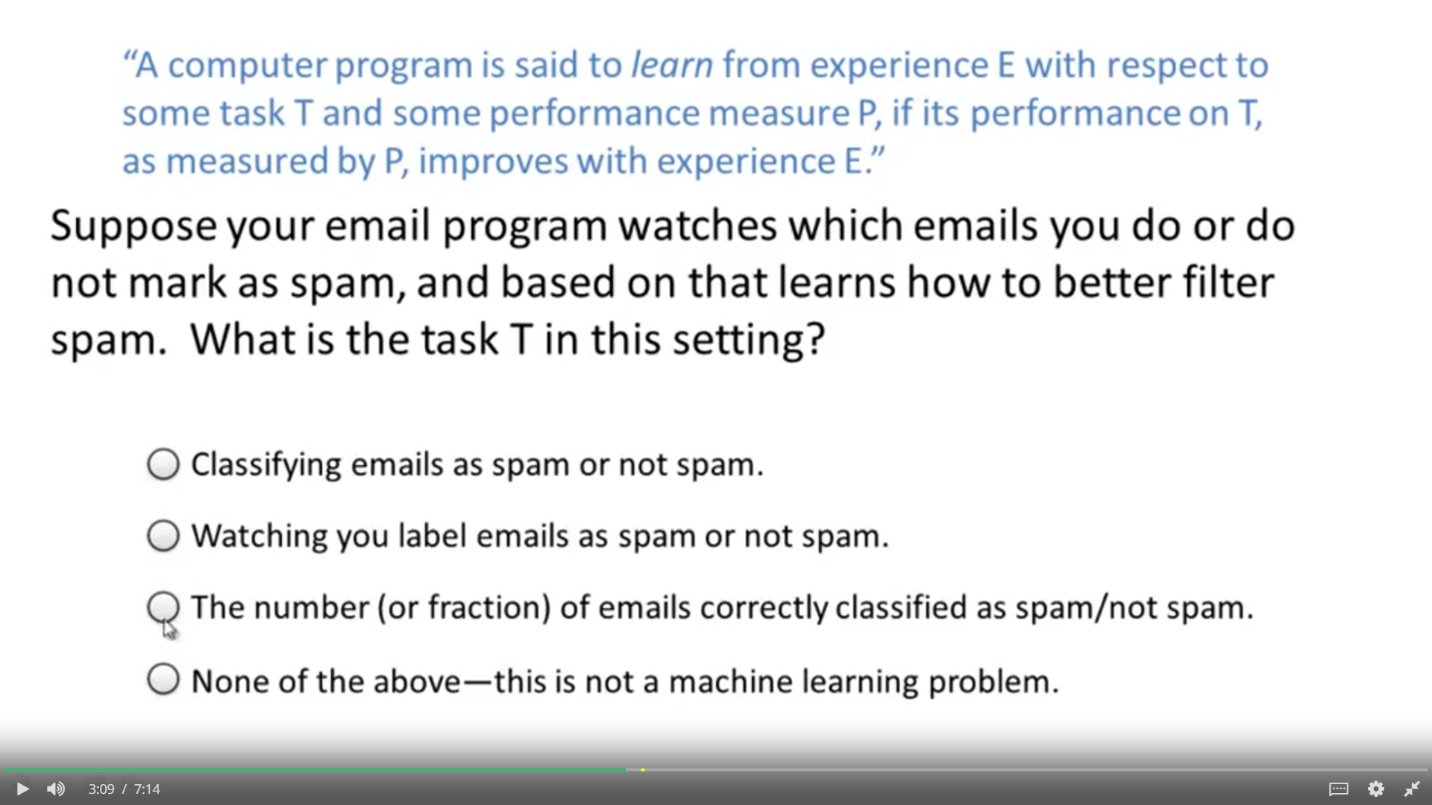
Well-posed Learning Problem: “A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E”

Example: Playing Checkers:

Experience (**E**) = the experience of playing many games of checkers.

Task (**T**) = task of playing checkers

Performance (**P**) = Probability of wining next game.

Question & Answer:

*Answer*:

*Classifying emails as spam or not spam* is the *task (T)* that need to be performed.

*Watching you label emails as spam or not* is *Experience (E)*since that is the result given with the experience.

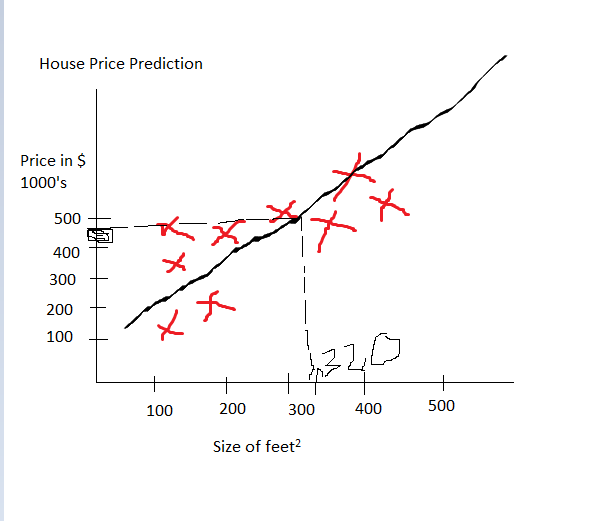
*The number (or fraction) of emails correctly classified as spam/not spam* is *Performance (P)*because it gives the performance of the result whether it is correct or not.

Machine Learning Algorithms:

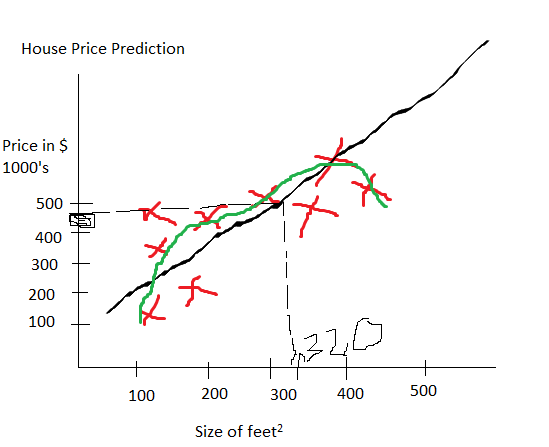
* Supervised Learning
* Unsupervised Learning
* Others: Reinforcement learning, recommender systems.

### Supervised Learning:

Supervised learning provides right answer along with its features. In the data set correct output is also given. Supervised learning problems are categorized into “Regression” and “Classification” problem. Regression predicts the result as per the given input while Classification classify the result into classes.

Example:

While analyzing just one factor for price of house. When size of feet increases, the price of house also increases. Here the straight line passes from the optimum values so that the prediction is accurate.



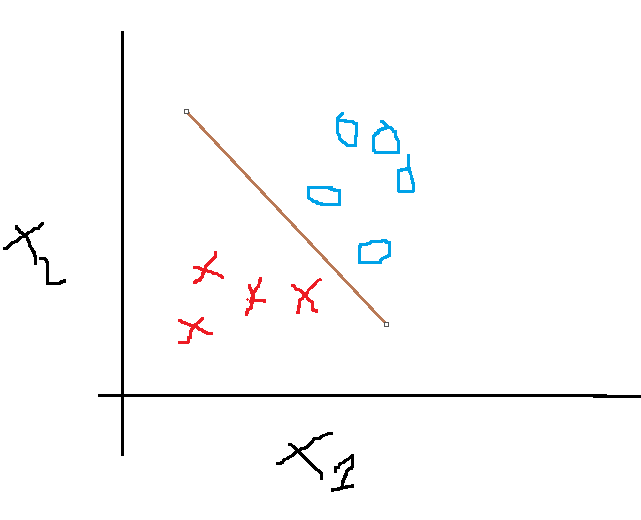
The optimum price is given but the when the line is in quadratic form.

Example: Prediction of age from the given image

This is an example of Regression.

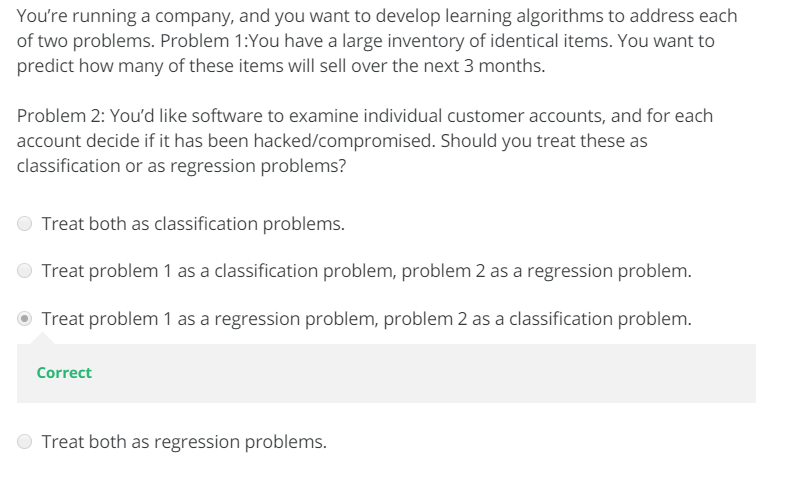
Classification is another type of supervised learning like with the size of tumor separating whether the tumor is malignant or benign. This is discrete valued classification or binary classification (0 or 1). Classification can be on more than 2 for example classification of cat, dog, wolf and leopard.

Classification representation:



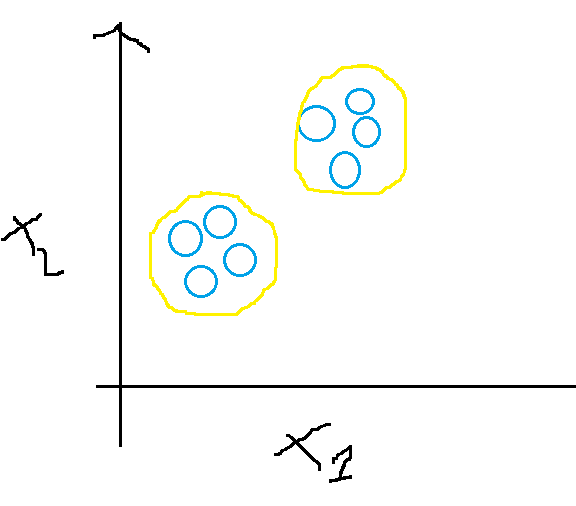
Example: Classify boy or a girl from the given image.

Question Answer:



### Un-Supervised Learning:

Unsupervised learning is provided with dataset, but label is not provided. Here, program is asked if it can find any structure or pattern in the dataset.

Program can make different cluster and give such output. This is called as clustering algorithm.

Unsupervised can be categorized into clustering data and non-clustering:

Examples of unsupervised learning:

* Organize computing clusters:

Machines tending to work together, if they are clustered the can perform well.

* Social network analysis

Which friend you emailed the most. Facebook friends or google+ circles, makes cohesive groups of friends.

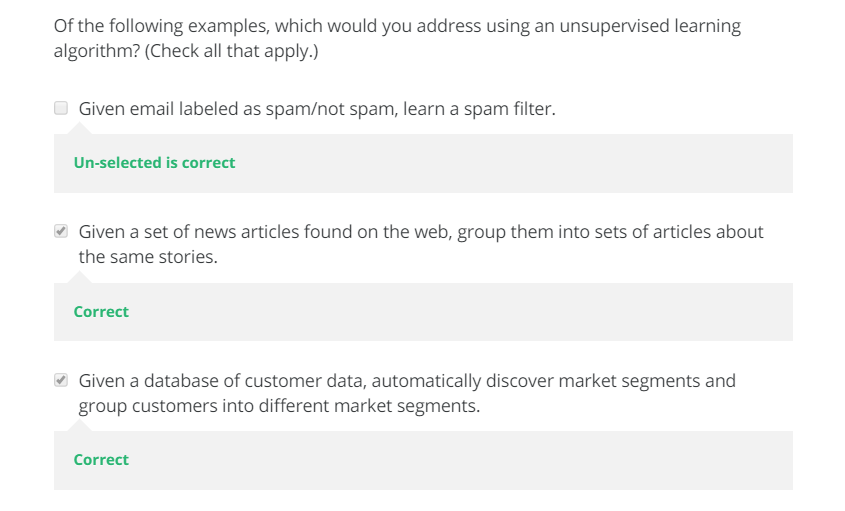
* Market segmentation
* Astronomical data analysis

Cocktail party problem: (Non - clustering)

If two people are speaking and there are two microphones A and B. First person is closer to microphone A and Second person is closer to microphone B. But the voice of single speaker is captured by both microphone. This can be separated using the below mentioned formula which is called as cocktail party problem.

[W,s,v] = svd((repmat(sum(x.\*x,1),size(x,1),size(x,1),1).\*x)\*x’);

Question:



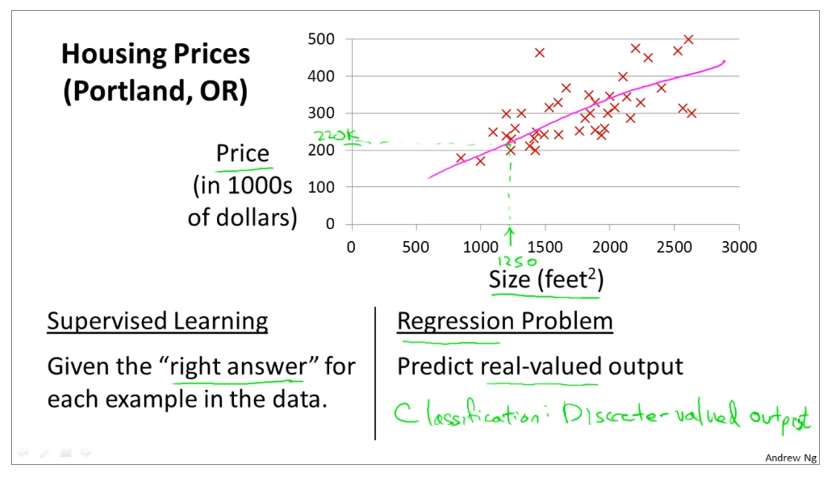
# Model and Cost Function:

* 1. Model Representation:

How model looks like and overall process of supervised learning.

Example:

Looking at the regression problem of house price prediction.



Supervised Learning: “Right answer” for data.

* Regression Problem (Predict real-valued output)

In supervised learning, there must be training set and machine will learn from the given data.

Training dataset of house (x) and its output as (y)

|  |  |
| --- | --- |
| **Size in feet2 (x)** | **Price ($) in 1000’s (y)** |
| 2104  1416  1534 … | 460  232  315 … |

Notation:

**m** = Number of training examples

**x**’s = input variable / features

**y**’s = output variables / “target” variable

(x,y) 🡪 one training example

(x(i), y(i)) 🡪 ith training example

*Note: the i of x and y represents for index not exponential*.

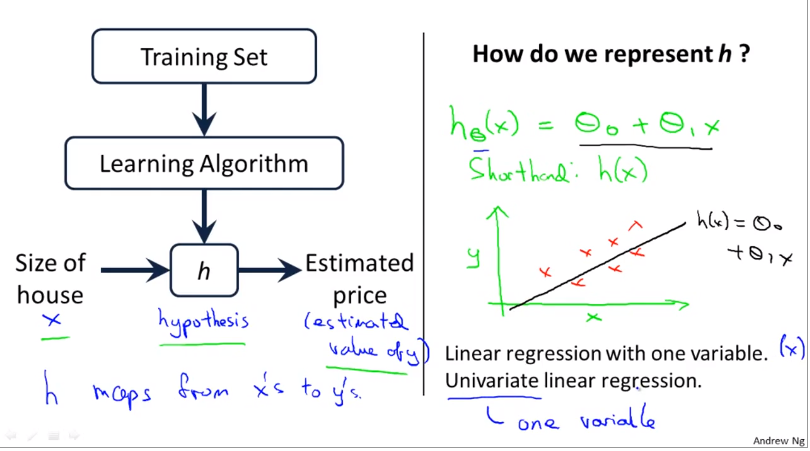
Example:

x1 🡪 2104, x2 🡪 1416, …

y1 🡪 460, y2 🡪 232, …

m = number of training data (maybe 1033 rows of dataset)

Here X represents space of input values, Y represents space of output values.



Training set is feed to learning algorithm which gives a function denoted as h (hypothesis). This hypothesis takes the feature that is the size of the house (x) and gives the result that is estimate price (y)

h is a function which matches x’s to y’s.

When designing a learning algorithm, representation of hypothesis is the key factor.

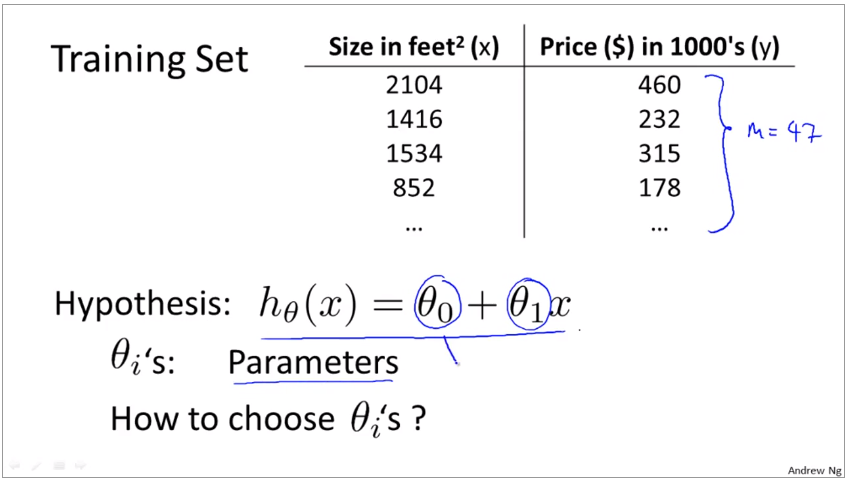
Training function represented as h: X🡪Y. so h(x) is a “good predictor” for corresponding value of y.

hθ​(x) = θ​0 + θ​1X

Short hand of *hθ*​(*x*) = h(x)

* 1. Cost Function:

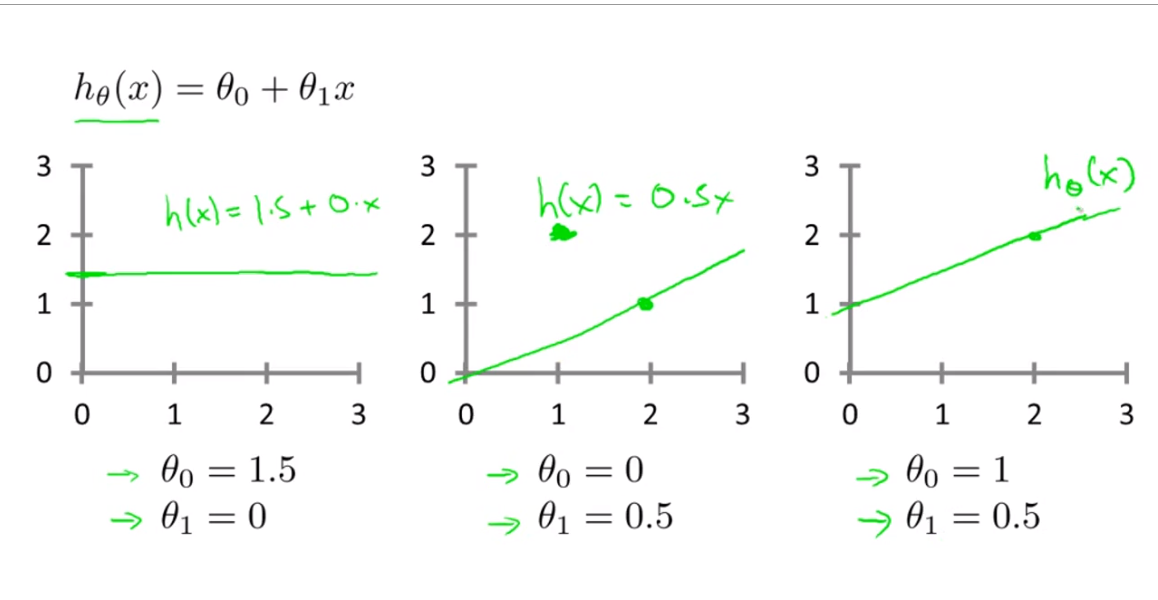
Cost function is to make the best line which includes all the dataset



How to choose theta value?

The function here is in the form of y = mx + c where θ​0 is c and θ​1 is m

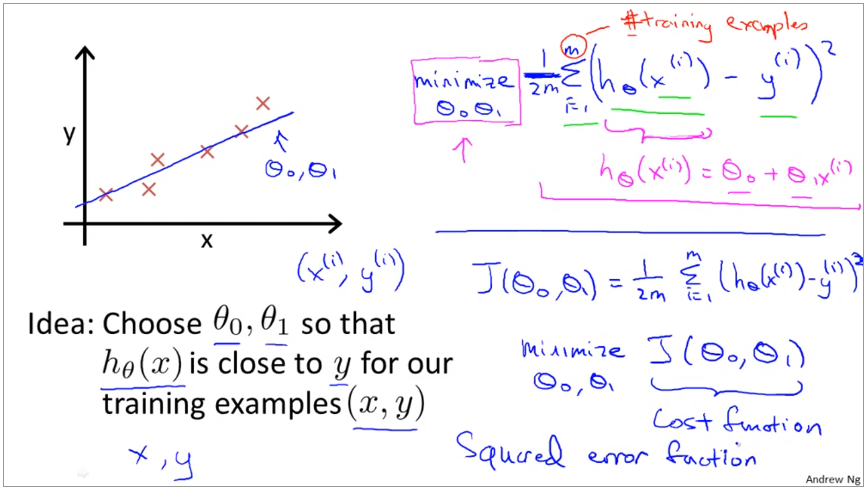
While plotting assuming some values of θ​0 and θ​1, following result is obtained.



Here when θ​0 has value 1.5 but θ​1 is 0 it forms straight line from the 1.5. and so on.

But while choosing θ​0 and θ​1 it should be near to ‘y’ so that the error is minimum.

Graphical representation is given below:



This is the formula to calculate the cost function. It is called as “*Mean squared error*” mean is halved (1/2) so that the computation is easy.

Cost Function Formula:

* 1. Cost Function Intuition I:

What cost function is doing and why to use this cost function.

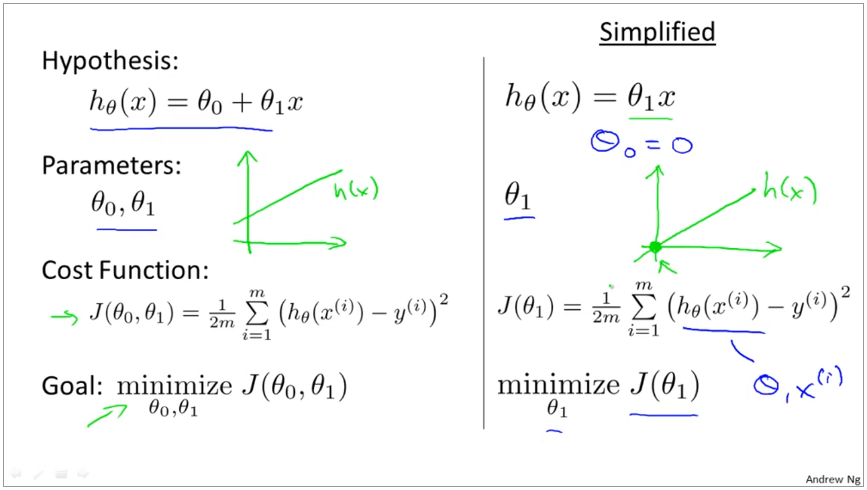
**Key things:**

Hypothesis: hθ​(x) = θ​0 + θ​1X

Parameters: θ​0, θ​1

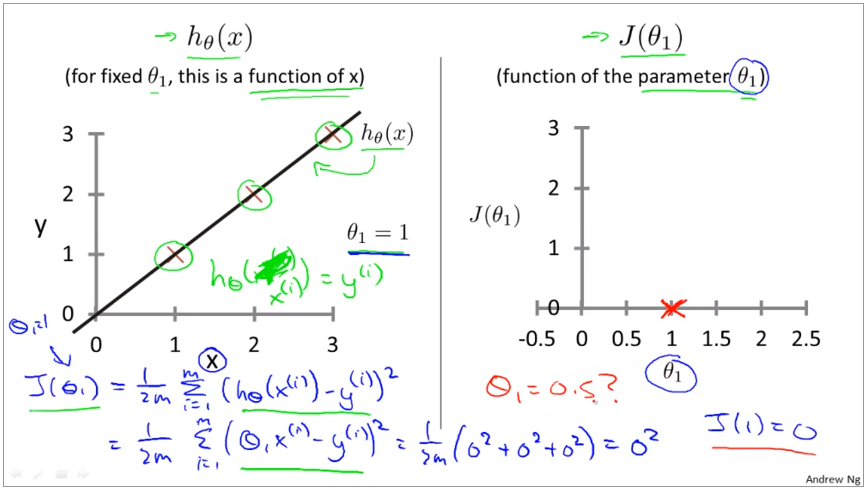
Cost Function:

Goal: minimize



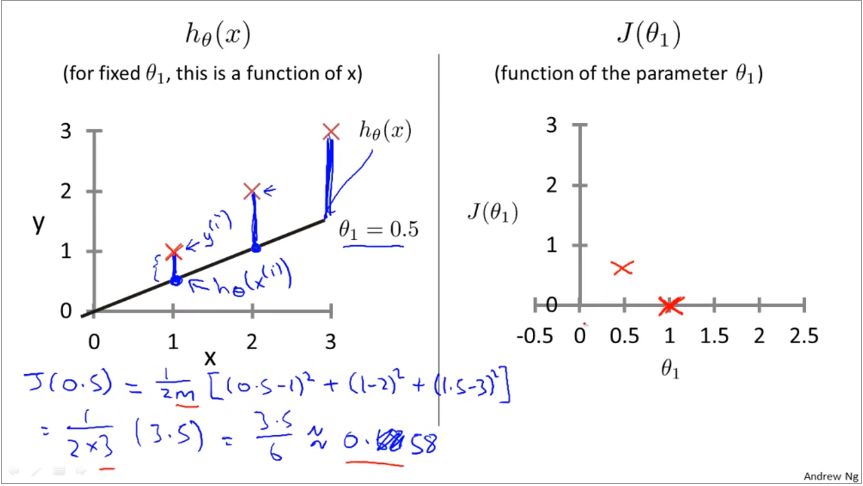
Difference between hypothesis hθ​(x) and Cost J(θ1) function:

Hypothesis gives the hypothesis line and deducting the actual value with the hypothesis gives the cost function which is J(θ1).

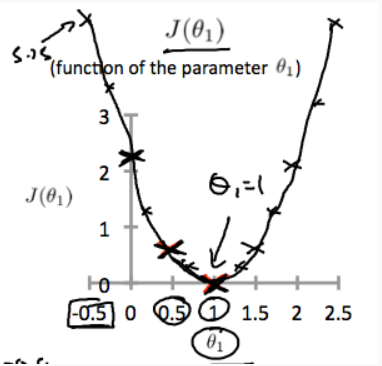


If θ1 is 1 then the hypothesis line is passing between points (1,1), (2,2) ... where actual values also belong therefore the cost function is 0.

Let’s see when the θ1 = 0.5, the line is slanted and when x is 1 y is 0.5, when x = 2, y is 1



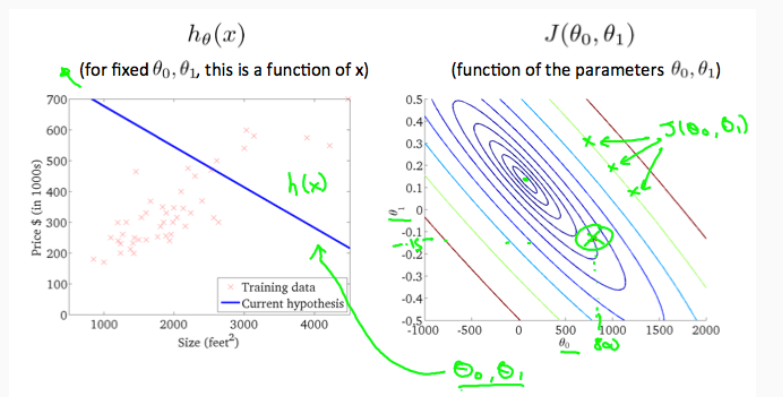
Therefore, here the hypothesis is not exact with the actual value. There comes the gap in the cost function. When hypothesis is not correct the cost function does not give correct value. Number of costs can be obtained but the actual cost is 1. So, our objective is to obtain that value.



Optimization objective for our learning algorithm is to minimize the J(θ1)

* 1. Cost Function Intuition II:

Representation of graph through contour plot. If the parameters are not single, then the representation will be on 3D format.

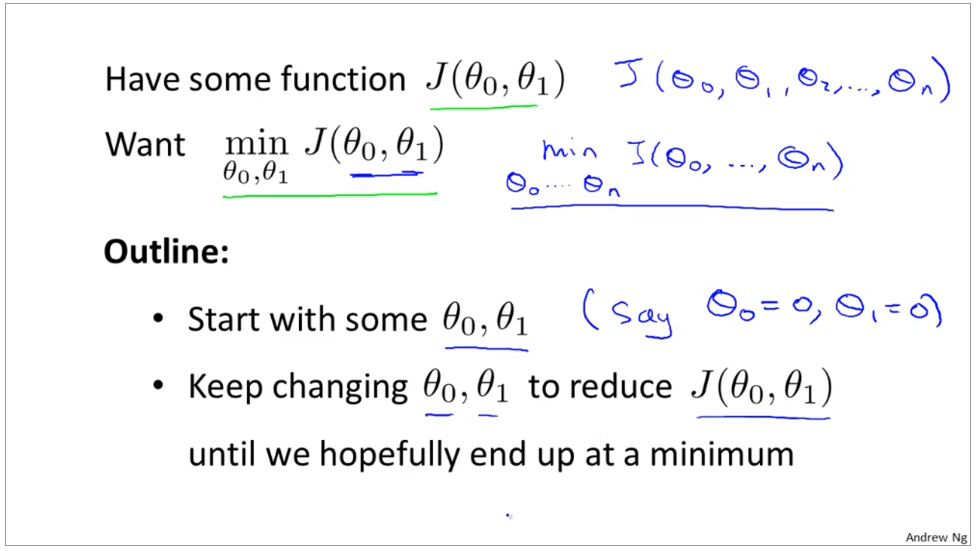


# Parameter Learning:

* 1. Gradient Descent:

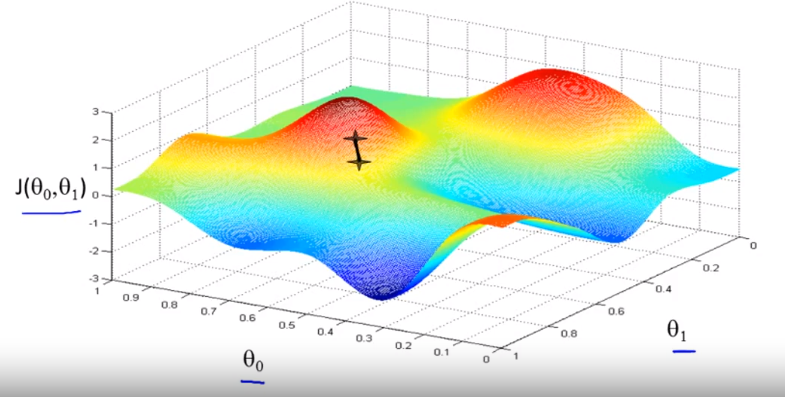
Gradient Descent is an algorithm which is used to minimize the cost function J (θ0 θ1). Gradient Descent is not used only in linear regression but all over the machine learning program.

Gradient Descent to minimize the function J.

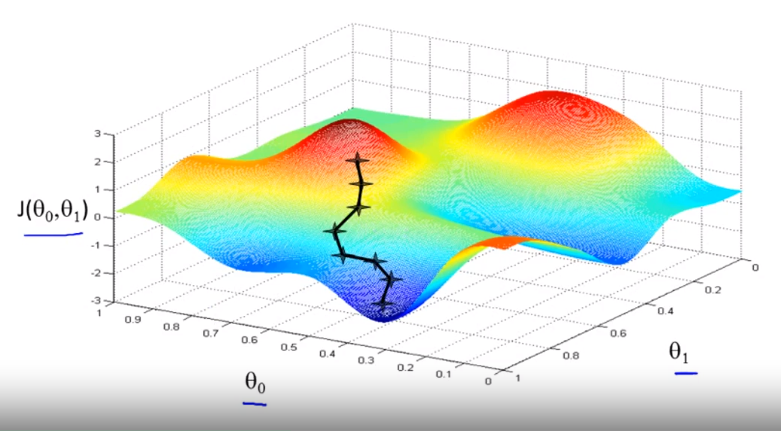


Here in gradient descent, we have cost function J (θ0 , θ1 , . . . θn) and we want to minimize this cost function. Here we keep changing the values of θ little bit so that we get the minimum error.

While looking graphically,

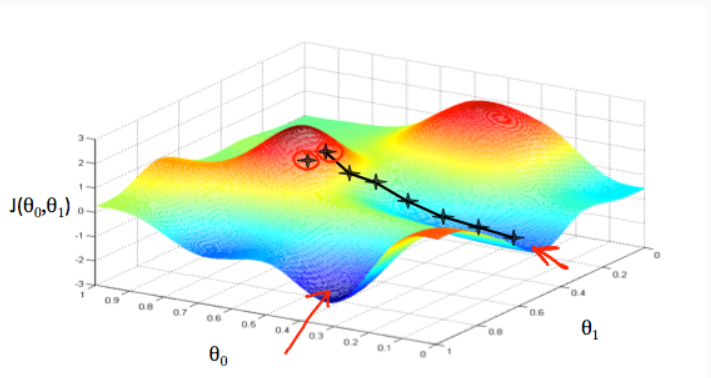


If a person is standing there at the first point. Now Gradient descent takes some small steps to find the minimum point.



Here taking different small steps, we reached at the minimum point.

Now taking derivative (the tangential line to a function)



Here two red arrows are showing the minimum point.

The direction in which the step is taken is determined by the partial derivative of *J*(*θ*0​ , *θ*1​)

Gradient Descent Algorithm:

Repeat until convergence

{

(for j = 0 and j = 1)

}

Note:

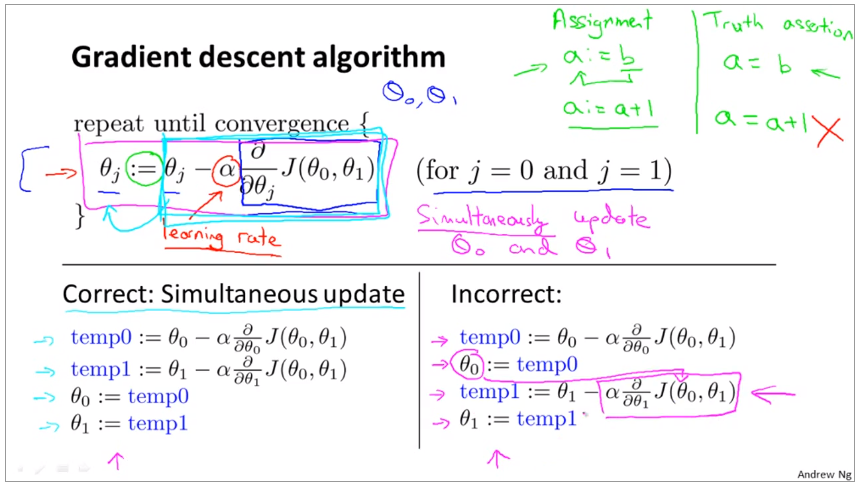
|  |  |
| --- | --- |
| Assignment | Truth assertion |
| a: = b  a: = a+1 | a = b  a = a+1 [Wrong] |

= learning rate

Now let’s implementing this formula:

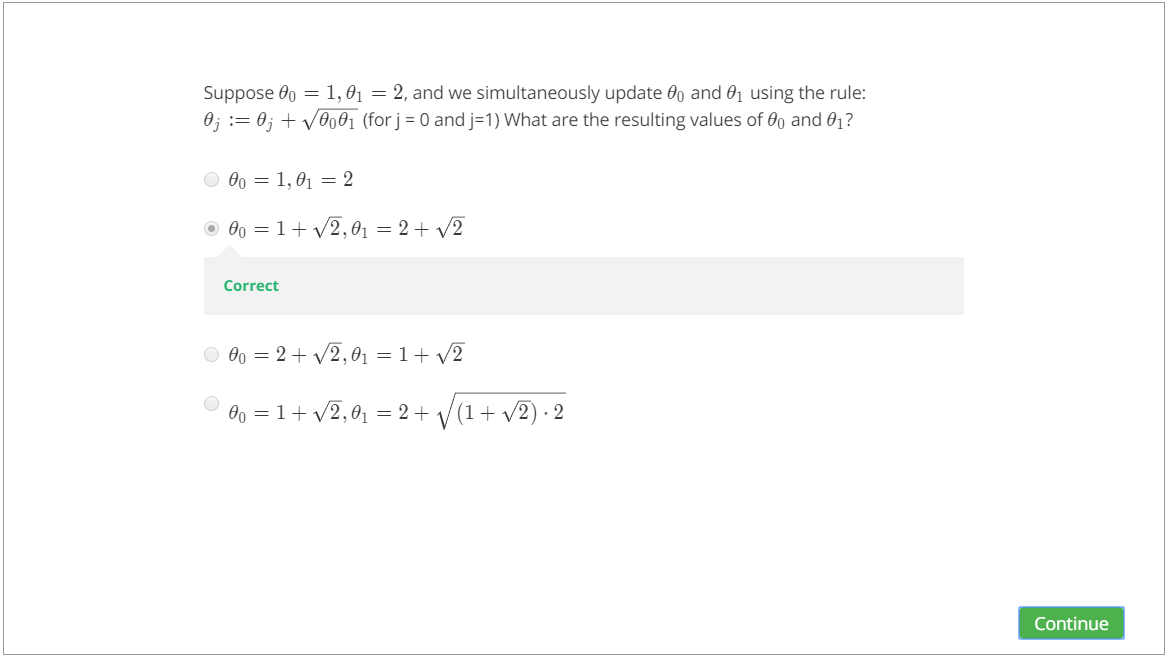
An is a number called as learning rate which controls the steps. If learning rate is large, then means very aggressive or large gradient descent procedure will take place.

Here in gradient descent will take place. Simultaneously data will be updated.



Some time incorrect update take place which does not seems like mistake, but it can make a great impact.

Here on the right side, updated at first and then again is updated. Here after updating , same updated is used to update which is incorrect. It leads for non-simultaneous update.



* 1. Gradient Descent Intuition:
  2. Gradient Descent for Linear Regression:

# Linear Algebra: