

Abdelrahman Rezk

Web Developer

NLP & ML student

AOU University

What is machine learning
Supervised learning
Unsupervised learning
Cost function
Gradient Descent
Linear Algebra

Machine Learning

ال machine learning هو ازای تخلی الاله الى عندك تتعلم من نفسها من غير ما تديها برامح او حاجات محددة يعني بمعنى اصح انت بتدربها على بعض الحاجات وهي المفروض تطور نفسها بنفسها من خلال التجارب التي بتمر عليها وتعلم من الماضي زى الانسان كده معنا مبتتعلم بس قشطه.

By Tom Mitchell:

He says, 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

انى الكومبيتر بتعلم من خلال التجارب التي مر عليها زى مثلا لو بنعمل clarify emails as spam or not ال الاول الكومبيتر بيتعلم انتا ازاي بتصنيف ال emails اما كده او كده بعدها بتتدى تجرب بعض ال tasks الجديدة الى هو مشفهاش ومن خلال ال performance ببناعه الى هو الميلات الى صنفها صح واللى مصنفهاش صح.

Example: playing checkers.

E = the experience of playing many games of checkers

T = the task of playing checkers.

P = the probability that the program will win the next game.

In general, any machine learning problem can be assigned to one of two broad classifications:

Supervised learning and Unsupervised learning.

ال machine learning بيتقسم ل تلت حاجات مهمه
1- ال supervised learning وده ببساطة انى بديله داتا وبديله بعض ال labels الخاصة بالبيانات ديه عشان بتتدى انه يتعلم منها ويقدر ياخد اكشن بعد كده وبرضه انا بتتابع الكلام ده مع ال model

● Linear Regression

- The term Supervised Learning refers to the fact that we gave the algorithm a data set in which the, called, "right answers" were given.

عشان فى الاول انتا بتديله داتا بقيم حقيقية زى اسعار الشقق output مقابل مميزات فى الشقق وهو بعدها بيتدى يتعلم مع نفسه من الحاجات ديه عشان لما تديله داتا جديده يقدر يفیدك ويقولك سعر شقة ما كام.

او لا كده بالنسبة لل machine learning الجزء الاول فيه الا وهو ال supervised learning يعتمد في التوقع بناء على جزئين مهمين الا وهم ال input and output ويبحصل فى ال supervised حاجتين مهمين اما انى يحتاج اعمل classification or regression

ال regression الا وهو التوقع وده ببساطة يشتغل على فكره انه بياخد بيانات ليها علاقه ببعض البيانات ديه عباره عن features وهو بيتدى يشتغل على الكلام ده كويس جدا وذاكره ويتعلم منه على سبيل المثال مثلا اسعار الشقق وعندى حوالي 10 الاف raw data الخاصه بالشقق متسجل فيها مكان الشقه والمساحه وغير بقا من الميزات وفي المقابل ال input بتاع ال features ديه بيكون فيه model output لـ prediction برضاه الى هو اسعار كل شقه منهم اد ايه فالمودل بيشتغل على الكلام ده وفي الآخر بيحصل عملية test انك بتشوف بقا شقه جديده بمواصفات جديدة وتدبها للموديل فيبدأ انه يعمل ليها انى سعرها اد كذا.

ال regression بيتوقع النتائج ديه بطرق مختلفه ممكن تكون linear او curved وغيره
By regression problem, I mean we're trying to predict a continuous valued output.

بعض التطبيقات المختلفة لل regression حاجه زى اسعار المنازل والأرصاد الجوية والمشتريات بالنسبة لعميل جديد
بناء على مواصفات العملاء الى من النوع ده.

بينما بقا ال classification هو انه بيحاول يقسم الحاجة لجروبات مختلفة زى الصور مثلا وانه بحاول يعمل تصنيف لصور القطط او الكلاب فلما تيجي تديله صوره جديده يقولك لا ديه صوره كلب او صوره قطة وهكذا.

The term classification refers to the fact, that here, we're trying to predict a discrete value output zero or one, malignant or benign.

يعنى في الآخر انا بحاول اتوقع قيم منفصلة عن بعضها لكن ال regression هو عباره عن انى بحاول اتوقع سعر حاجه ممكن تتغير مع تغير ال data الى داخله ليا اىاما ال classification هو توقع حاجه معينه مثلا من set او مثلا التوقع ده بيكون binary اما يحصل او لا.

So this is an example of a supervised learning algorithm. And it's supervised learning because we're given the, quotes, "right answer" for each of our examples. Namely we're told what was the actual house, what was the actual price of each of the houses in our data set were sold for and moreover, this is an example of a regression problem where the term regression refers to the fact that we are predicting a real-valued output namely the price.

2- ال unsupervised learning وده انا بدليه داتا وهو المفروض يتعلم منها من غير ما اديله اى label او حاجه يتعلم منها هو بيتدى يقسم الداتا ويتعلم مع نفسه.

انتا بتديله الداتا وهو بيحاول يلاقي structure مختلفة في الداتا ديه بمعنى بيحاول انه يقسم الداتا لجموعات بناء على الحاجات الى ليها علاقه ببعض.

So this is Unsupervised Learning because we're not telling the algorithm in advance that these are type 1 people, those are type 2 persons, those are type 3 persons and so on and instead what were saying is yeah here's a bunch of data. I don't know what's in this data. I don't know who's and what type. I don't even know what the different types of people are, but can you automatically find structure in the data from the you automatically cluster the individuals into these types that I don't know in advance? Because we're not giving the algorithm the right answer for the examples in my data set, this is Unsupervised Learning.

Unsupervised Learning

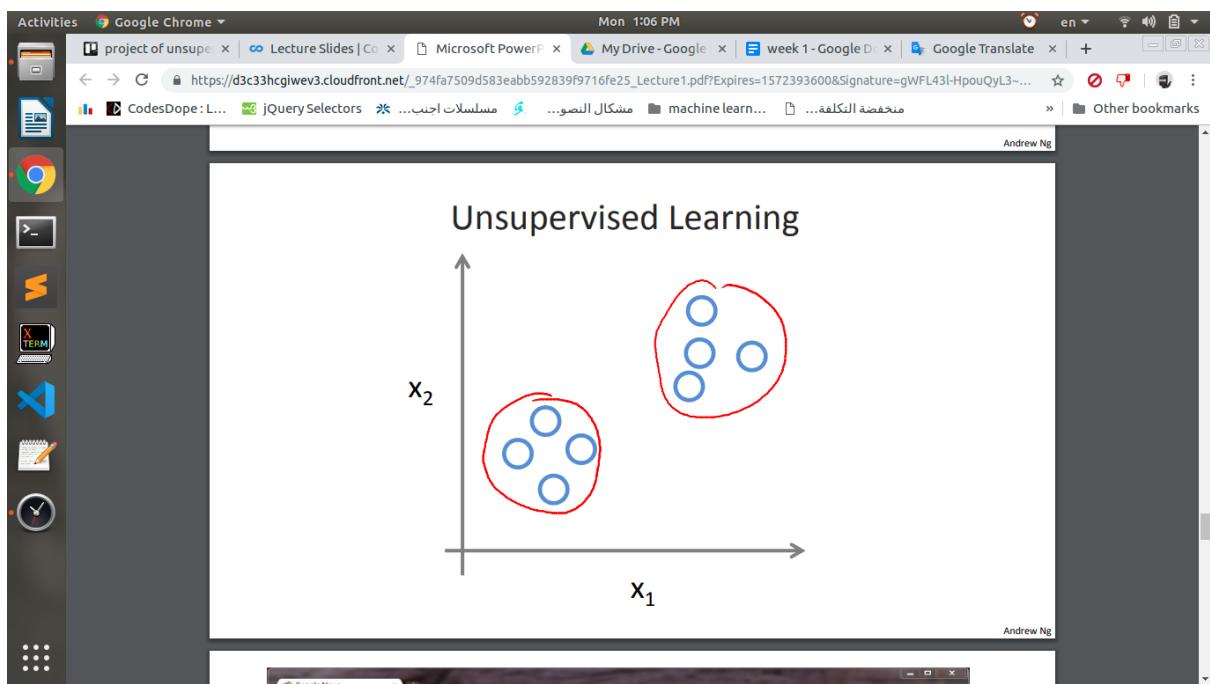
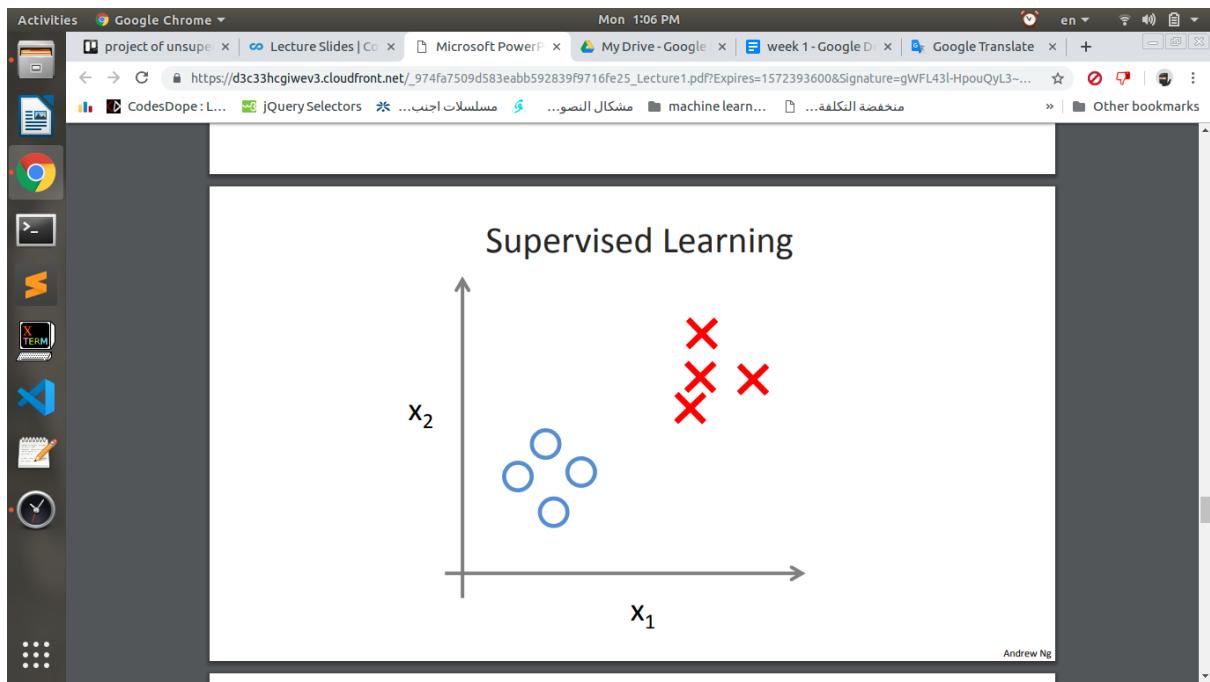
Unsupervised learning allows us to approach problems with little or no idea what our results should look like. We can derive structure from data where we don't necessarily know the effect of the variables.

We can derive this structure by clustering the data based on relationships among the variables in the data.

With unsupervised learning there is no feedback based on the prediction results.

Clustering: Take a collection of 1,000,000 different genes, and find a way to automatically group these genes into groups that are somehow similar or related by different variables, such as lifespan, location, roles, and so on.

Non-clustering: The "Cocktail Party Algorithm", allows you to find structure in a chaotic environment. (i.e. identifying individual voices and music from a mesh of sounds at a [cocktail party](#)).



Training set of housing prices (Portland, OR)

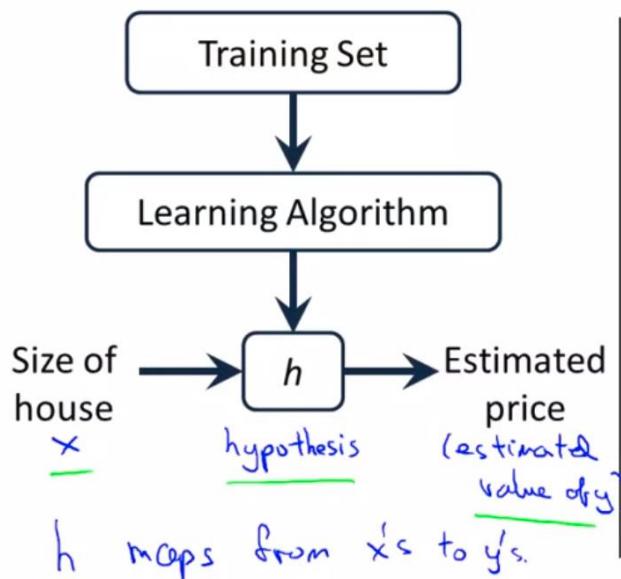
Size in feet ² (x)	Price (\$) in 1000's(y)
→ 2104	460
1416	232
→ 1534	315
852	178
...	...

Notation:

- m = Number of training examples
- x 's = "input" variable / features
- y 's = "output" variable / "target" variable
- (x, y) - one training example
- $(x^{(i)}, y^{(i)})$ - ith training example

$$\begin{cases} x^{(1)} = 2104 \\ x^{(2)} = 1416 \\ \vdots \\ y^{(1)} = 460 \end{cases}$$

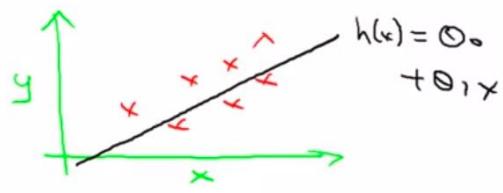
Andrew Ng



How do we represent h ?

$$h_{\theta}(x) = \underline{\theta_0 + \theta_1 x}$$

Shorthand: $h(x)$



Linear regression with one variable. (x)
Univariate linear regression.
one variable

Andrew Ng

Activities Google Chrome Mon 8:33 PM

Machine.Lec General Task Microsoft My Drive week 1-Go Google Tra Lecture Sli _ec21cea31...

https://d3c33hcgiwev3.cloudfront.net/_ec21cea314b2ac7de627706501b5baa_Lecture2.pdf?Expires=1572393600&Signature=dkMaCwz1kiPIS-6cz...

CodesDope: L... jQuery Selectors مسلسلات اجنبی... مسلسلات انجمنی... machine learn... منحصنة الكلفة... Other bookmarks

Hypothesis:

$$h_{\theta}(x) = \theta_0 + \theta_1 x$$

Parameters:
 θ_0, θ_1

Cost Function:

$$\rightarrow J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

Goal: minimize $J(\theta_0, \theta_1)$

Simplified

$$h_{\theta}(x) = \theta_1 x$$

 $\theta_0 = 0$

$$J(\theta_1) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$
minimize $J(\theta_1)$ $\theta_1, x^{(i)}$

Andrew Ng

Lecture2.pdf Show all

Idea: Choose θ_0, θ_1 so that $h_{\theta}(x)$ is close to y for our training examples (x, y)

x, y

#Training examples

$$\min_{\theta_0, \theta_1} \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

$$h_{\theta}(x^{(i)}) = \theta_0 + \theta_1 x^{(i)}$$

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

$$\min_{\theta_0, \theta_1} J(\theta_0, \theta_1)$$

Cost function

Squared error function

Andrew Ng

ازاي بقا بقدر احسب ال cost function الا وهى نسبة ال error الى عندي والى هى بتكون الفرق بين ال line الى انا برسمه والنقط ال actual بمعنى الفرق بين ال actual and predictive الى هو ال y وال $h(x)$ s

وده بيخليني اعرف نسبة ال error عندي اد ايه ومن خلال ده بحاول انى قللها على قدر ما قدر.

$h(x)$ is consider about theta 0 + theta1*x(i) and i is th-ieth raw in our data.

طبعا فيه فرق بين حاجتين مهمين الا وهمما ال d (theta) الى هو الرقم الى انا بختاره لل theta وبعدها بحسب بقا الحاجه الثانية وهى $J(\theta)$ الى هى ال cost function من اختياري لل theta بالرقم ده طلع بكتذا.

Activities Google Chrome ▾ Mon 4:18 PM

project of unsupe x | Microsoft Power x | My Drive - Google x | week 1 - Google D x | Google Translate x | Cost Function - In x +

https://www.coursera.org/learn/machine-learning/supplement/u3qF5/cost-function-intuition-i

CodesDope:... jQuery Selectors مسلسلات اح... مسکال المصو... machine learn... منحصنة الكلفة... Other bookmarks

coursera Explore What do you want to learn? Search Abdelrahman Rezk

Machine Learning > Week 1 > Cost Function - Intuition I

Cost Function - Intuition I

If we try to think of it in visual terms, our training data set is scattered on the x-y plane. We are trying to make a straight line (defined by $h_{\theta}(x)$) which passes through these scattered data points.

Our objective is to get the best possible line. The best possible line will be such so that the average squared vertical distances of the scattered points from the line will be the least. Ideally, the line should pass through all the points of our training data set. In such a case, the value of $J(\theta_0, \theta_1)$ will be 0. The following example shows the ideal situation where we have a cost function of 0.

$\rightarrow h_{\theta}(x)$
(for fixed θ_0 , this is a function of x)

$\rightarrow J(\theta_1)$
(function of the parameter θ_1)

$$J(\theta_0) = \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

$$= \frac{1}{m} \sum_{i=1}^m (\theta_0 x^{(i)} + \theta_1 - y^{(i)})^2 = \frac{1}{m} (0^2 + 0^2 + 0^2) = 0^2 = 0$$

Activities Google Chrome Mon 4:19 PM

project of unsupe x Microsoft Power x My Drive - Google x week 1 - Google D x Google Translate x Cost Function - Int x +

https://www.coursera.org/learn/machine-learning/supplement/u3qF5/cost-function-intuition-i

CodesDope: L... jQuery Selectors مسلسلات أحب... مسلسلات الحصو... machine learn... متحفصة الكلفة... Other bookmarks

Abdelrahman Rezk

coursera Explore What do you want to learn?

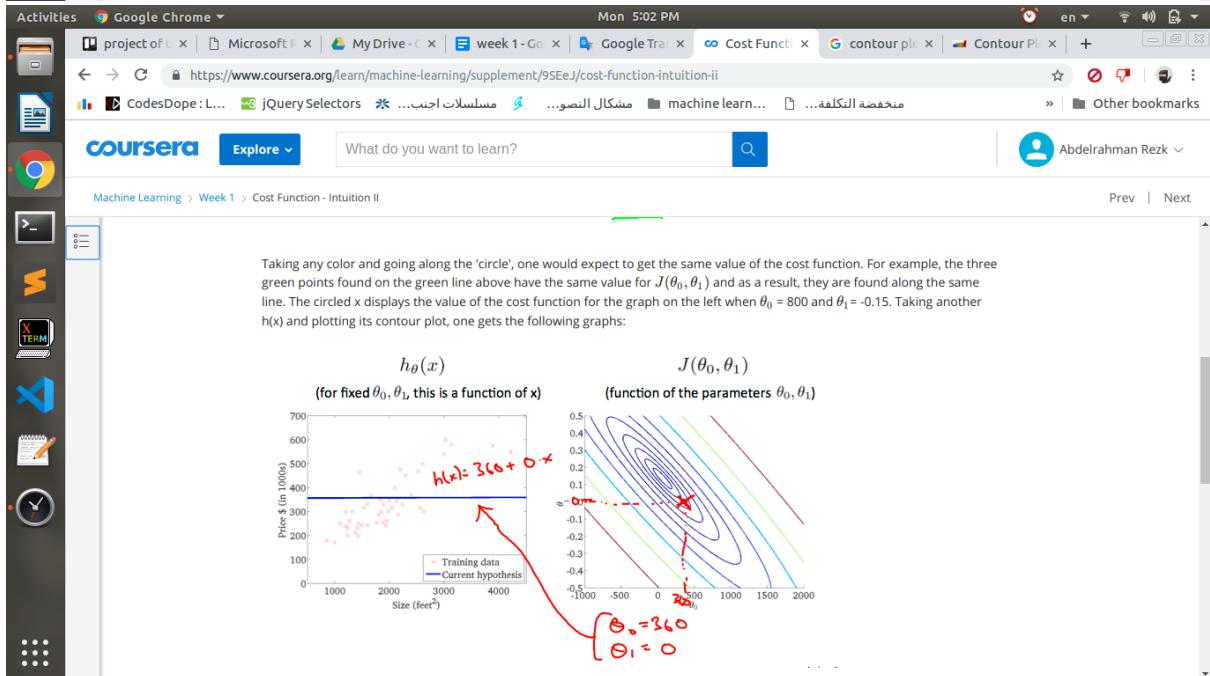
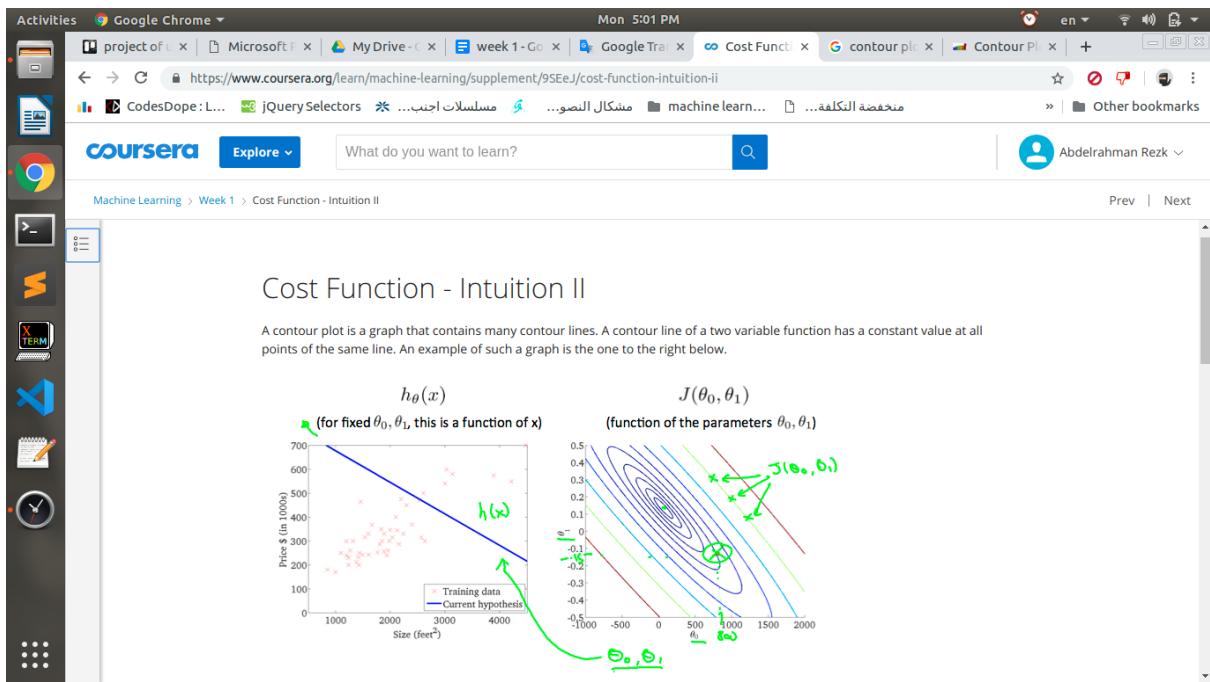
Machine Learning > Week 1 > Cost Function - Intuition I

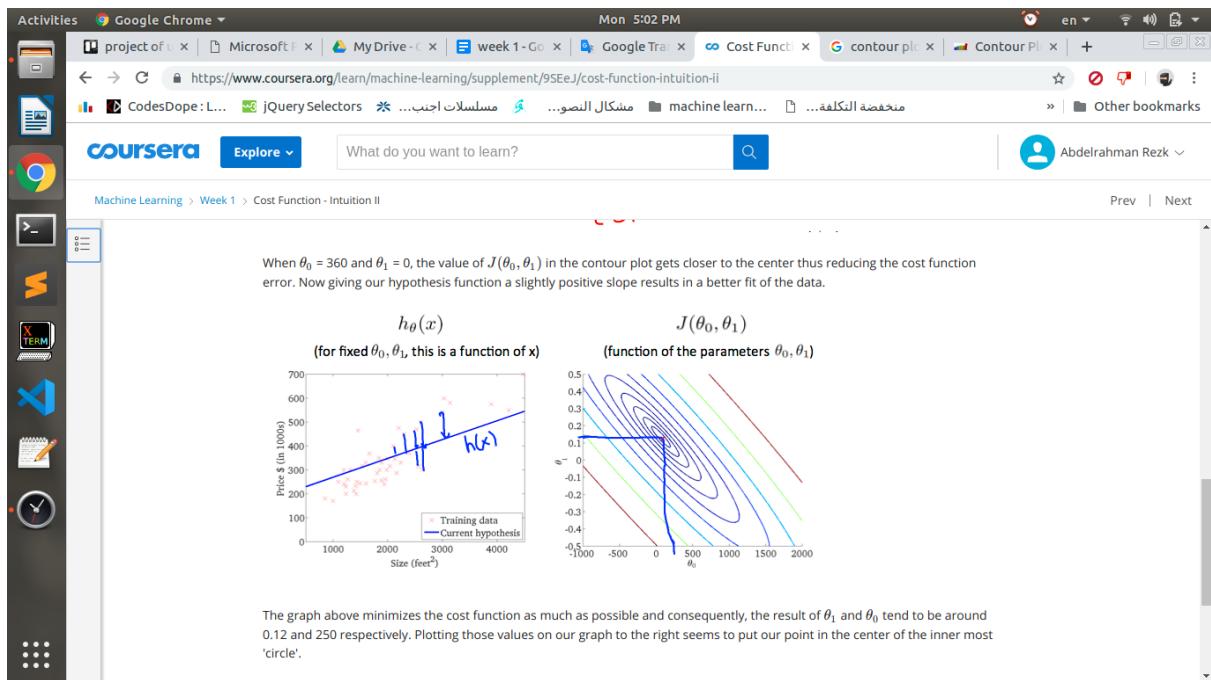
When $\theta_1 = 1$, we get a slope of 1 which goes through every single data point in our model. Conversely, when $\theta_1 = 0.5$, we see the vertical distance from our fit to the data points increase.

$$\begin{aligned} h_{\theta}(x) &\text{ (for fixed } \theta_1, \text{ this is a function of } x) \\ J(\theta_1) &\text{ (function of the parameter } \theta_1) \\ \text{for } \theta_1 = 0.5: \\ J(0.5) &= \frac{1}{2} \sum (h_{\theta}(x_i) - y_i)^2 \\ &= \frac{1}{2} \sum [(0.5 \cdot 1 + 0.5 \cdot 2 + 0.5 \cdot 3) - (1, 2, 3)]^2 \\ &= \frac{1}{2} \sum (3.5 - 1, 2, 3)^2 \approx 0.5858 \\ &J(0) = ? \end{aligned}$$

This increases our cost function to 0.58. Plotting several other points yields to the following graph:

هنا انا مخترش الا theta 1 فقط وخليل 0 بـ 0

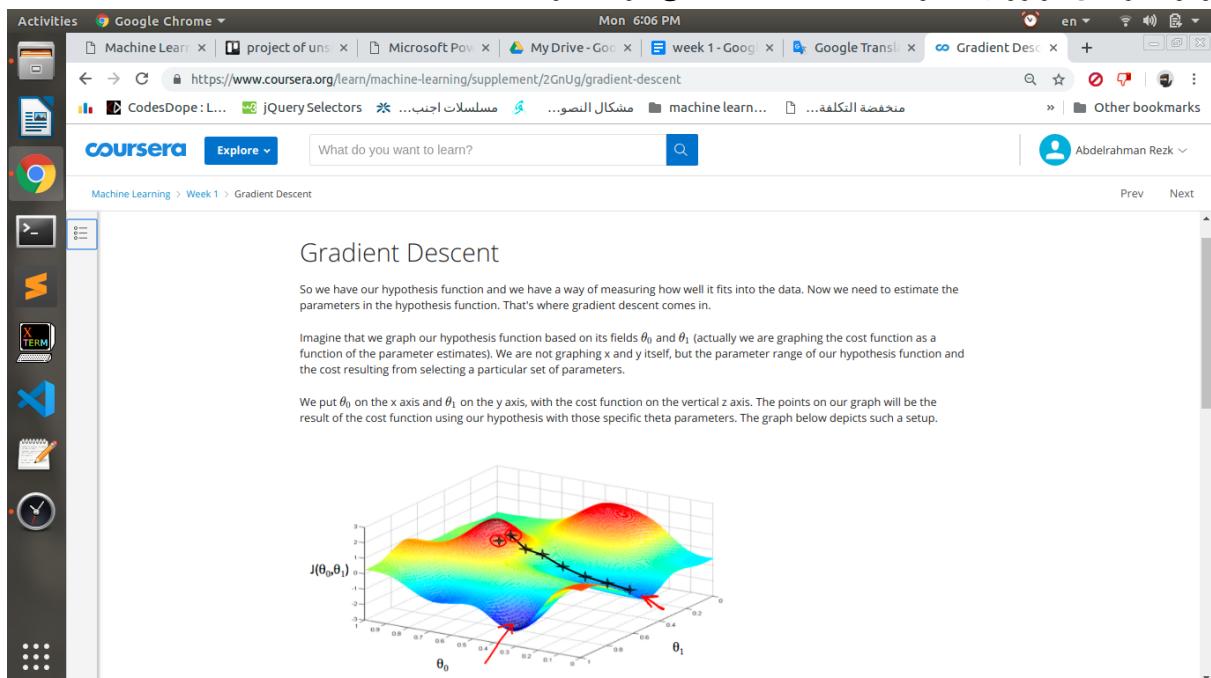




طیب دلوقتی انا عرفت ازای بحسب ال cost function بعد اما اختار قیم theta 1 and theta 0 و اشتغل علی المعادله بتاعت ال $J(\theta_0, \theta_1)$ ولکن اانا لو عندي داتا كتيره مش هقد اشتغل manually انى اجرب کل نقطه و اشوفها بتعمل ايه معايا بناء علی قیم ال theta إلی اختارتاه و هنا بيجي دور

Algorithm called gradient descent

وهو عبارة عن خوارزمية بتحاول انها تقال نسبة الخطأ على قدر ما تقدر.



Mon 6:11 PM

Activities Google Chrome ▾

Machine Learn... project of uns... Microsoft Pow... My Drive - Goo... week 1-Goo... Google Transl... Gradient Desc... +

https://www.coursera.org/learn/machine-learning/supplement/2GnUg/gradient-descent

Coursera Explore What do you want to learn? Search

Machine Learning > Week 1 > Gradient Descent

We will know that we have succeeded when our cost function is at the very bottom of the pits in our graph, i.e. when its value is the minimum. The red arrows show the minimum points in the graph.

The way we do this is by taking the derivative (the tangential line to a function) of our cost function. The slope of the tangent is the derivative at that point and it will give us a direction to move towards. We make steps down the cost function in the direction with the steepest descent. The size of each step is determined by the parameter α , which is called the learning rate.

For example, the distance between each 'star' in the graph above represents a step determined by our parameter α . A smaller α would result in a smaller step and a larger α results in a larger step. The direction in which the step is taken is determined by the partial derivative of $J(\theta_0, \theta_1)$. Depending on where one starts on the graph, one could end up at different points. The image above shows us two different starting points that end up in two different places.

The gradient descent algorithm is:

repeat until convergence:

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$$

where

j=0,1 represents the feature index number.

repeat until convergence:

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$$

where

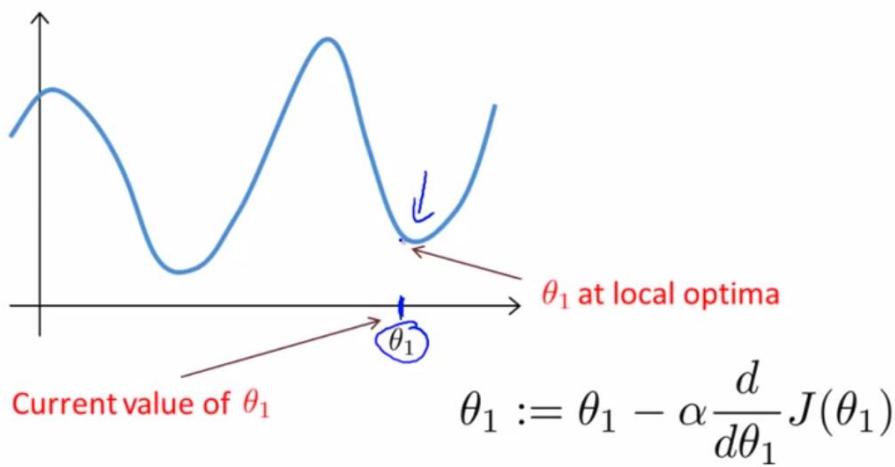
j=0,1 represents the feature index number.

At each iteration j, one should simultaneously update the parameters $\theta_1, \theta_2, \dots, \theta_n$. Updating a specific parameter prior to calculating another one on the $j^{(th)}$ iteration would yield to a wrong implementation.

<u>Correct: Simultaneous update</u>	<u>Incorrect:</u>
$\rightarrow \text{temp0} := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$ $\rightarrow \text{temp1} := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$ $\rightarrow \theta_0 := \text{temp0}$ $\rightarrow \theta_1 := \text{temp1}$	$\rightarrow \text{temp0} := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$ $\rightarrow \theta_0 := \text{temp0}$ $\rightarrow \text{temp1} := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$ $\rightarrow \theta_1 := \text{temp1}$

الى فى المعادلة ده هو اللي يساعدنى ويبخلينى اعرف انا هامشى ازاي عشان اوصل لل minimum error وده من خلال انى لما بختار نقطة ما بروح اشوف هى بتتس الخط ازاي واشوف ال slope بتاعها اما negative ومن خلاله بعرف انا هامشى ازاي.

لما ال gradient descent و ال theta بتاعتها بتوصلى لل minupmum optimal bottom الى هو بتقلل ال error لأكثر حاجه ممكنه ساعتها ال slope الناتج من ال derivative بيكون ب 0 وبعد كده قيمه ال theta مش هتتغير. فى ال gradient descent انتا كل اما بتاخ خطوة فى اتجاه انك تقلل ال error الخطوه نفسها بتقل لانك فى كل مرر بتضرب ال learning rate فى ال derivative الناتج عن ال slope الجديد وهكذا.



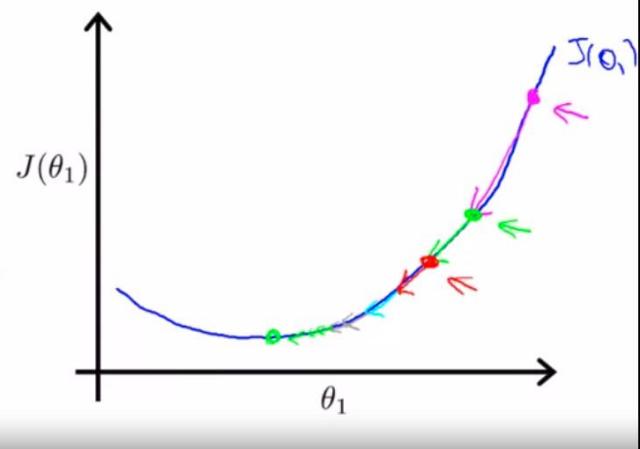
It turns out the local optimum,
your derivative will be equal to zero.

Andrew Ng

Gradient descent can converge to a local minimum, even with the learning rate α fixed.

$\theta_1 := \theta_1 - \alpha \frac{d}{d\theta_1} J(\theta_1)$

As we approach a local minimum, gradient descent will automatically take smaller steps. So, no need to decrease α over time.



انتا هنا مش تحتاج انك تقلل ال learning rate لأنك كده في كل مره الخطوه نفسها بتقل بسبب اني قيمة ال learning rate can be fixed بنقل لذلك ال

Gradient descent algorithm

```
repeat until convergence {  
     $\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$   
    (for  $j = 1$  and  $j = 0$ )  
}
```

Linear Regression Model

$$h_{\theta}(x) = \theta_0 + \theta_1 x$$
$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

Activities Google Chrome Mon 7:23 PM

Machine Learni project of uns Microsoft Pow My Drive - Goog week 1 - Goog Google Transl Gradient Desc +

https://www.coursera.org/learn/machine-learning/supplement/QKEdR/gradient-descent-intuition

CodesDope:... jQuery Selectors مسلسلات احباب... مسلسلات المصو... machine learn... متحفظة التكلفة... Other bookmarks

Coursera Explore What do you want to learn? Search Abdelrahman Rezk

Machine Learning > Week 1 > Gradient Descent Intuition

Prev | Next

- Video: Gradient Descent 11 min
- Reading: Gradient Descent 3 min
- Video: Gradient Descent Intuition** 11 min
- Reading: Gradient Descent Intuition 3 min
- Video: Gradient Descent For Linear Regression 10 min
- Reading: Gradient Descent For Linear Regression 6 min

Review

Regardless of the slope's sign for $\frac{d}{d\theta_1} J(\theta_1)$, θ_1 eventually converges to its minimum value. The following graph shows that when the slope is negative, the value of θ_1 increases and when it is positive, the value of θ_1 decreases.

$J(\theta_1)$ ($\theta_1 \in \mathbb{R}$)

$\theta_1 := \theta_1 - \alpha \cdot \frac{\partial}{\partial \theta_1} J(\theta_1) \geq 0$

$\theta_1 := \theta_1 - \alpha \cdot (\text{positive number})$

negative slope

$\frac{\partial}{\partial \theta_1} J(\theta_1) \leq 0$

$\theta_1 := \theta_1 - \alpha \cdot (\text{negative number})$

Machine Learning > Week 1 > Gradient Descent Intuition

On a side note, we should adjust our parameter α to ensure that the gradient descent algorithm converges in a reasonable time. Failure to converge or too much time to obtain the minimum value imply that our step size is wrong.

$\theta_1 := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_1)$

If α is too small, gradient descent can be slow.

If α is too large, gradient descent can overshoot the minimum. It may fail to converge, or even diverge.

Activities Google Chrome Mon 7:24 PM

Machine Learni project of uns Microsoft Pow My Drive - Goog week 1 - Goog Google Transl Gradient Desc +

CodesDope:... jQuery Selectors مسلسلات احباب... مسلسلات احباب... machine learn... متحفظة التكلفة... Other bookmarks

coursera Explore What do you want to learn? Abdelrahman Rezk

Machine Learning > Week 1 > Gradient Descent Intuition

Parameter Learning

- Video: Gradient Descent 11 min
- Reading: Gradient Descent 3 min
- Video: Gradient Descent Intuition 11 min
- Reading: Gradient Descent Intuition 3 min
- Video: Gradient Descent For Linear Regression 10 min
- Reading: Gradient Descent For Linear Regression 6 min

Review

- Reading: Lecture Slides 20 min
- Quiz: Linear Regression with One Variable

How does gradient descent converge with a fixed step size?

The intuition behind the convergence is that $\frac{d}{d\theta_1} J(\theta_1)$ approaches 0 as we approach the bottom of our convex function. At the minimum, the derivative will always be 0 and thus we get:

$$\theta_1 := \theta_1 - \alpha * 0$$

Gradient descent can converge to a local minimum, even with the learning rate α fixed.

$\theta_1 := \theta_1 - \alpha \frac{d}{d\theta_1} J(\theta_1)$

As we approach a local minimum, gradient descent will automatically take smaller steps. So, no need to decrease α over time.

Andrew Ng

از ای بقا اینی احظ ای gradient descent with cost function to fit best line in your data

When specifically applied to the case of linear regression, a new form of the gradient descent equation can be derived. We can substitute our actual cost function and our actual hypothesis function and modify the equation to :

```

repeat until convergence: {
     $\theta_0 := \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^m (h_\theta(x_i) - y_i)$ 
     $\theta_1 := \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^m ((h_\theta(x_i) - y_i)x_i)$ 
}

```

where m is the size of the training set, θ_0 a constant that will be changing simultaneously with θ_1 and x_i, y_i are values of the given training set (data).

Note that we have separated out the two cases for θ_j into separate equations for θ_0 and θ_1 ; and that for θ_1 we are multiplying x_i at the end due to the derivative. The following is a derivation of $\frac{\partial}{\partial \theta_j} J(\theta)$ for a single example :

$$\begin{aligned}
\frac{\partial}{\partial \theta_j} J(\theta) &= \frac{\partial}{\partial \theta_j} \frac{1}{2} (h_\theta(x) - y)^2 \\
&= 2 \cdot \frac{1}{2} (h_\theta(x) - y) \cdot \frac{\partial}{\partial \theta_j} (h_\theta(x) - y) \\
&= (h_\theta(x) - y) \cdot \frac{\partial}{\partial \theta_j} \left(\sum_{i=0}^n \theta_i x_i - y \right) \\
&= (h_\theta(x) - y) x_j
\end{aligned}$$

الجزء الكبير في ال linear regression هو الرياضه او المعادله الخطيه و معناها هو العلاقة بين متغيريين حين يكون كلا من هذان المتغيران لهما اوس = 1

استخدام الدوال في الرياضه مع الماشين ليرننج ي يكون على حسب انا شغال على ايه فمثلا ممكن استخدم المعادله الخطيه لو هي دايمما في زياده لكن لو قدام مثلا الحاجه لما بتزيد بتقل بعد شويه فممكن تكون معادله من الدرجه الثانيه وهكذا.

The multivariable form of the hypothesis function accommodating these multiple features is as follows:

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 + \dots + \theta_n x_n$$

In order to develop intuition about this function, we can think about θ_0 as the basic price of a house, θ_1 as the price per square meter, θ_2 as the price per floor, etc. x_1 will be the number of square meters in the house, x_2 the number of floors, etc.

Using the definition of matrix multiplication, our multivariable hypothesis function can be concisely represented as:

$$h_\theta(x) = [\theta_0 \quad \theta_1 \quad \dots \quad \theta_n] \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_n \end{bmatrix} = \theta^T x$$

This is a vectorization of our hypothesis function for one training example; see the lessons on vectorization to learn more.

Remark: Note that for convenience reasons in this course we assume $x_0^{(i)} = 1$ for $(i \in 1, \dots, m)$. This allows us to do matrix operations with theta and x. Hence making the two vectors ' θ ' and ' $x^{(i)}$ ' match each other element-wise (that is, have the same number of elements: n+1).

Data Rescaling

هو انى احاول اخل جميع الداتا الى عندي وفيها ارقام ما بين ال -1 و 1 وده بيساعدنى انى الجراف نفسه بيكون معقول وليه علاقه بعضه على عكس اما يكون عندي قيم صغيره جدا وقيم كبيره جدا والصورتين الى تحت دول بيوضحوا قبل وبعد ال scaling

The screenshot shows a Google Chrome browser window with the Coursera website open. The URL is https://www.coursera.org/learn/machine-learning/supplement/CTA0D/gradient-descent-in-practice-i-feature-scaling. The page title is "Machine Learning > Week 2 > Gradient Descent in Practice I - Feature Scaling". On the left, there's a sidebar with various course activities like reading and video lessons. The main content area describes two techniques: feature scaling and mean normalization. It provides a formula for feature scaling: $x_i := \frac{x_i - \mu_i}{s_i}$, where μ_i is the average of all values for feature i , and s_i is the range of values (max - min). It also notes that dividing by the range or standard deviation gives different results. A note mentions that quizzes use standard deviation while programming exercises use range. An example is given for housing prices ranging from 100 to 2000 with a mean of 1000: $x_i := \frac{price - 1000}{1900}$. A "Mark as completed" button is visible at the bottom right.

Normal Equation

ديه بتحل مشكله انى افترض قيم للشيتات ومن خلالها بقدر اوصل لقيم الشيتا الى انا عاييزها عشان اشتغل على ال gradient descent واجبيه فى خطوه واحدة بدل المشكله الى بتواجهنى فى تحديده ال learning rate ولكن المشكله هنا بتكون فى ال inverse العمليه بتاعتنه نفسها بتأخذ وقت كبير جدا كل اما الماتركس تكبر على عكس انى احاول مع ال learning rate رغم انى معرفش عدد الخطوات الى هاخدتها قد ايه ولكن ممكن فى range ال 10 الاف وده الى هو عدد ال features .normal question

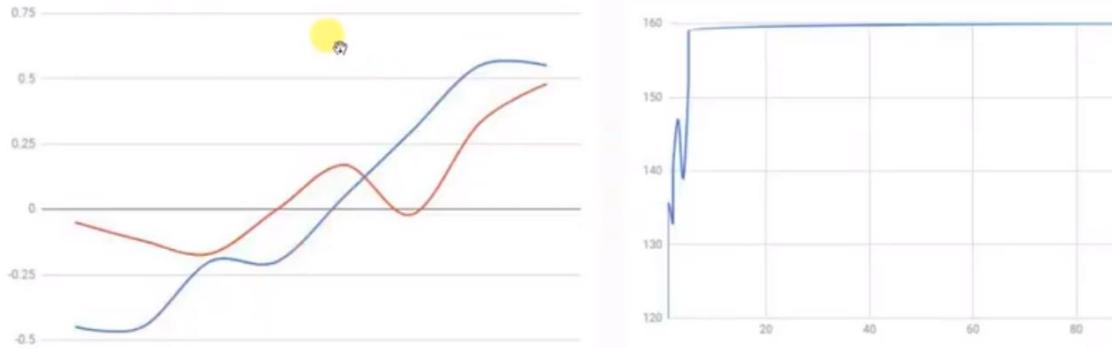
ولازم اخلى بالى لما اجي اجيب ال normal question ممكن لما اجي اجيب ال inverse الاقى الماتركس نفسها الى هو ملهاش inverse

Handwritten notes at the top left show the formula for feature scaling: $x_i \leftarrow \frac{x_i - \mu_i}{s_i}$. To its right, a bullet point states: "فالحل انتا نخلي سكيل جميع القيم على 1 بس بحيث تكون القيمة القصوى ليها واحد". Below this are two tables. The first table shows house price data with columns for X (عدد الغرف), Y (السعر), and their respective means and standard deviations (المتوسط, المدى). The second table shows the scaled version of the same data.

									المتوسط	المدى	
X	عدد الغرف	1	1	2	2	3	4	5	2.8	4	
Y	السعر	120	135	133	140	147	139	153	159	140	39

X	عدد الغرف	-0.45	-0.45	-0.2	-0.2	0.05	0.3	0.55	0.55
Y	السعر	-0.05	-0.12	-0.17	0	0.17	-0.02	0.33	0.48

تدرج البيانات Data Rescaling



14

Gradient Descent in Practice II - Learning Rate

Google Chrome Fri 8:12 AM Activities

coursera Explore What do you want to learn? Prev | Next

Machine Learning > Week 2 > Gradient Descent in Practice II - Learning Rate

Debugging gradient descent. Make a plot with *number of iterations* on the x-axis. Now plot the cost function, $J(\theta)$ over the number of iterations of gradient descent. If $J(\theta)$ ever increases, then you probably need to decrease α .

Automatic convergence test. Declare convergence if $J(\theta)$ decreases by less than E in one iteration, where E is some small value such as 10^{-3} . However in practice it's difficult to choose this threshold value.

Making sure gradient descent is working correctly.

$\min_{\theta} J(\theta)$

$J(\theta)$ should decrease after every iteration

Example automatic convergence test:

Declare convergence if $J(\theta)$ decreases by less than 10^{-3} in one iteration.

Andrew Ng

It has been proven that if learning rate α is sufficiently small, then $J(\theta)$ will decrease on every iteration.

Fri 8:13 AM

Activities Google Chrome ▾

T227 MTA Review x | Google Translate x | Gradient Descent x | Features and Poly x | My Drive - Google x | week 1 & 2 - Goog x | + | - | x

coursera | Other bookmarks

Machine Learning > Week 2 > Gradient Descent In Practice II - Learning Rate

What do you want to learn? | Search

Abdelrahman Rezk

It has been proven that if learning rate α is sufficiently small, then $J(\theta)$ will decrease on every iteration.

Making sure gradient descent is working correctly.

- For sufficiently small α , $J(\theta)$ should decrease on every iteration.
- But if α is too small, gradient descent can be slow to converge.

To summarize:

- If α is too small: slow convergence.
- If α is too large: may not decrease on every iteration and thus may not converge.

إنتا ممكن وانتا شغال لما تشووف ال features تكرر انك تعمل features جديده ممكن تكون تجميع ل لاكثر من feature زى مثلا الطول والعرض بتاع المنزل تخليهم الاتنين فى features واحد وتسميه المساحة وممكن انك ت create feature مش موجود بس يكون ليه علاقه بالحاجة ومؤثر فيها.

Fri 8:31 AM

Activities Google Chrome ▾

T227 MTA Review Sum x | Google Translate x | Features and Polynomial Regression x | My Drive - Google Drive x | week 1 & 2 - Google Doc x | + | - | x

coursera | Other bookmarks

Machine Learning > Week 2 > Features and Polynomial Regression

Descent In Practice I - Feature Scaling 3 min

Video: Gradient Descent In Practice II - Learning Rate 8 min

Reading: Gradient Descent In Practice II - Learning Rate 4 min

Video: Features and Polynomial Regression 7 min

Reading: Features and Polynomial Regression 3 min

Computing Parameters Analytically

Submitting Programming Assignments

Our hypothesis function need not be linear (a straight line) if that does not fit the data well.

We can **change the behavior or curve** of our hypothesis function by making it a quadratic, cubic or square root function (or any other form).

For example, if our hypothesis function is $h_{\theta}(x) = \theta_0 + \theta_1 x_1$ then we can create additional features based on x_1 , to get the quadratic function $h_{\theta}(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2$ or the cubic function $h_{\theta}(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \theta_3 x_1^3$

In the cubic version, we have created new features x_2 and x_3 where $x_2 = x_1^2$ and $x_3 = x_1^3$.

To make it a square root function, we could do: $h_{\theta}(x) = \theta_0 + \theta_1 x_1 + \theta_2 \sqrt{x_1}$

One important thing to keep in mind is, if you choose your features this way then feature scaling becomes very important.

e.g. if x_1 has range 1 - 1000 then range of x_1^2 becomes 1 - 1000000 and that of x_1^3 becomes 1 - 1000000000

Mark as completed

ال features الى بتكرر معها بطريقة متناسبة مع بعضها زى لما قسم حاجه على حاجه او الاتنين يكونوا بيتقسما على رقم معين غالبا ال features ديه ممكن امسحها كلها واحلى واحد منهم فقط لأن يعتبر مفيش غير feature واحد منهم هو الى هياشر معايا لأنهم فى تناسب مع بعضهم وده مش هاي ساعدى بحاجه غير وقت وتكلفة على الفاضي يعني محتاج اخلى بالى وانا شغال من كل حاجه