

Abdelrahman Rezk

AOU University

NLP & ML student

# بسم الله الرحمن الرحيم

الملخص ده بإذن الله لازم تكون شوفت قبله اول اسبوعين من Andrew Ng

<https://www.coursera.org/learn/machine-learning>

وفيه مقابل ليه بالعربى مهندس هشام وده بصراحة ربنا يباركله ويجازيه خير على العلم الى بيقدمه والمساعدته الجميله وده برده اول اسبوعين فى الكورس فى البلاى ليست بتاعته.

[https://www.youtube.com/watch?v=5\\_GKIhoXKyl&list=PL6-3IRz2XF5UJE2PbY7UU4SHi7UpV1mXo](https://www.youtube.com/watch?v=5_GKIhoXKyl&list=PL6-3IRz2XF5UJE2PbY7UU4SHi7UpV1mXo)

المذكور هنا هايكون جهد شخصى ومحاولة إنى اساعد فى تبسيط المصطلحات والمعادلات مع حل امثله عملية تربط جميع الاجزاء المختلفه خلال اول اسبوعين بإذن الله فالمفروض إنك تكون شوفتهم.

"وهاكون شاكر لو كان هناك دعوة بظهر الغيب"

احنا فى الآخر بنحاول اننا نقلل الفرق بين  $h(x)$  الى هى القيمة المتوقعة وال  $y$  الى هى القيمة الحقيقية.

### بعض المصطلحات الهامة

input x	output y
0	4
1	7
2	7
3	8

$m$  = training examples

عدد الامثلة الى هاقدر استخدمها عشان اعمل training للموديل بتاعى فى المثال الى فوق 4

$n$  = number of features

الحاجات المميزه الى بتساعدنى اعمل predict كويس وهنا معنديش الا 1 بس الى هو  $x$

$y$  = actual value or output value

انا فى الآخر بتعامل مع supervised learning فيحاول انى اعمل mapping from input to output معين وده الى انا بخلى الموديل بتاعى يدرب عليه ويحاول يقلل الفرق قدر الامكان بين القيم المتوقعة والقيم الحقيقية.

$y^{\wedge} = h(x)$  = predicted value

ديه القيمة المتوقعة

$\Sigma$  summation sign

$\Theta$  thetas parameters

قيم ال thetas الاخير هى الى بتستخدم عشان احاول اعمل predict لحاجه جديدة مشفتهاش قبل كده عن طريق قيم ال thetas الى انا طلعتها من ال training examples.

$\alpha$  alpha called learning rate

ديه الى بتاثر فى تغير تقايل نسبة الخطأ بصورة كبيره او صغيره على حسب قيمتها.

## المعادلات المستخدمة

### أولا ال Cost function

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

vectorized Version of cost function

$$J(\theta) = \frac{1}{2m} * (X\theta - \vec{y})^T * (X\theta - \vec{y})$$

دلوقت نقدر نطبق على اول حاجه ونعرف انى نفس المعادلتين هايطلعوا نفس الناتج بدل ما اعمل loop ممكن عطلول استخدم ال vectorize version ولكن هانطبق على الاتنين ونشوف الدنيا هاتمشى ازاي.

خلينا نفترض انى قيمه  $\theta_0 = 1$  and  $\theta_1 = 3$  وبما ان معادلة ال  $h(x) = 1 + 3x$

لازم تكون عارف انى  $x^1$  معناها اول مثال فى الداتا يعنى اول row لكن  $x_1$  معناها اول feature

فيكده  $x_1^1$  معناها اول feature فى اول مثال وهنا معنديش الا feature واحد لكن 8 training examples.

$x_1$	$y$	$H(x)$	$(h_{\theta}(x^i) - y^i)$	$(h_{\theta}(x^i) - y^i)^2$
100	300	$h(x) = 1 + 3x$ $1+3*100=301$	$301-300=$ 1	<b>12 =</b> <b>1</b>
95	285	286	1	<b>1</b>
90	270	271	1	<b>1</b>
80	240	241	1	<b>1</b>
80	235	241	6	<b>36</b>
70	200	211	11	<b>121</b>
70	205	211	6	<b>36</b>
60	180	181	1	<b>1</b>
				$\sum_{i=1}^m (h_{\theta}(x^i) - y^i)^2 =$ <b><math>1+1+1+1+36+121+36+1 = 198</math></b>

Then

$$J(\theta) =$$

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

M= 8 training examples so

$$J(\theta) =$$

$$\frac{1}{2 * 8} * 198 = 12.375$$

### vectorize version solution of cost function

$$J(\theta) = \frac{1}{2m} * (X\theta - \vec{y})^T * (X\theta - \vec{y})$$

Assume  $X_0^i = 1$  for 8 training examples

حول كل حازه من المعادله بتاعت ال vectorize ل matrix or vector عشان نقدر نتعامل معاها

لذلك هاتكون بالشكل ده بعد اما افترضنا اني  $X_0$  هاتشاي 1

هاتكون بالشكل ده

$$\begin{matrix} 300 & 1 & 100 \\ 285 & 1 & 95 \\ 270 & 1 & 90 \\ 240 & 1 & 80 \\ 235 & 1 & 80 \\ 200 & 1 & 70 \\ 205 & 1 & 70 \\ 180 & 1 & 60 \end{matrix} \quad \begin{matrix} \Theta \\ \text{ال } x \\ \text{وال } y \end{matrix}$$

بما ان  $\Theta = 2 \times 1$  وال  $X = 8 \times 2$  بيقا  $X\Theta$  ينفع يضربوا لان  $2 \times 1$  and  $8 \times 2$  والناتج هايكون matrix  $8 \times 1$  والي هي تنفع تطرح من  $y$  بعدها بيحصلها transpose فتبقا  $1 \times 8$  والجزء الثاني هايكون  $8 \times 1$  فساعتها ققدر اعمل multiplication بين جزئين المعادله والناتج الي هو هايكون رقم واحد يضرب في  $1/2m$  وده بيقا ال cost function.

من المفترض انك تكون عارف matrix operations بتتعمل ازاى.

$$J(\theta) = \frac{1}{2m} * (X\theta - \vec{y})^T * (X\theta - \vec{y})$$

$$(X\theta - \vec{y})^T = \begin{bmatrix} 1 & 100 & 300 & 301 & 300 & 1 \\ 1 & 95 & 285 & 286 & 285 & 1 \\ 1 & 90 & 270 & 271 & 270 & 1 \\ 1 & 80 & 240 & 241 & 240 & 1 \\ 1 & 80 & 235 & 241 & 235 & 1 \\ 1 & 70 & 200 & 211 & 200 & 11 \\ 1 & 70 & 205 & 211 & 205 & 6 \\ 1 & 60 & 180 & 181 & 180 & 1 \end{bmatrix} * \begin{bmatrix} 1 \\ 3 \end{bmatrix} - \begin{bmatrix} 240 \\ 235 \end{bmatrix} = \begin{bmatrix} 241 \\ 240 \\ 241 \\ 240 \\ 235 \end{bmatrix} = \begin{bmatrix} 1 \\ 6 \end{bmatrix} = \text{transpose it} =$$

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 6 & 11 & 6 & 1 \end{bmatrix}$$

Then the same for  $(X\theta - \vec{y})$

هايدينا نفس الناتج بس من غير transpose فهايكون الناتج

$$\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 6 \\ 11 \\ 6 \\ 1 \end{bmatrix}$$

$$\text{so } (X\theta - \vec{y})^T * (X\theta - \vec{y}) =$$

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 6 & 11 & 6 & 1 \end{bmatrix} * \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 6 \\ 11 \\ 6 \\ 1 \end{bmatrix} = 1 + 1 + 1 + 1 + 36 + 111 + 36 + 1 = 198$$

$$\text{then } * \frac{1}{2m} = \frac{1}{2 * 8} * 198 = 12.375$$

وده ببساعدنى جدا فى انى معملش اى loops مجرد بس matrixes operations.

**ثانيا ال Gradient Descent**

$$\theta := \theta - \alpha * \frac{1}{m} * \sum_{i=1}^m ((h_{\theta}(x^i) - y^i) * X^i))$$

نفس الكلام افترض  $\theta_0 = 1$  and  $\theta_1 = 3$  وقيمة الفا  $\alpha = 0.002$

$X_1$	$y$	$H(x)$	$(h_{\theta}(x^i) - y^i)$	$(h_{\theta}(x^i) - y^i) * X^i$
100	300	$h(x) = 1 + 3 X$ $1+3*100=301$	$301-300=$ 1	<b><math>1*100=100</math></b>
95	285	286	1	<b>95</b>
90	270	271	1	<b>90</b>
80	240	241	1	<b>80</b>
80	235	241	6	<b>480</b>
70	200	211	11	<b>770</b>
70	205	211	6	<b>420</b>
60	180	181	1	<b>60</b>

$$\sum_{i=1}^m ((h_{\theta}(x^i) - y^i) * X^i))$$

=

$$100+95+90+80+480+770+420+60=2095$$

Then

$$\alpha * \frac{1}{m} * \sum_{i=1}^m ((h_{\theta}(x^i) - y^i) * X^i))$$

$$\alpha * \frac{1}{m} * 2095$$

$$0.002 * \frac{1}{8} * 2095$$

$$=.52$$

Then

$$\theta_1 := \theta_1 - \alpha * \frac{1}{m} * \sum_{i=1}^m ((h_{\theta}(x^i) - y^i) * X^i))$$

هنا المفروض اني 0 theta بتطرح قبل ما اضرب في  $X^i$  لآخره يعني المعادلة بتكون كالتالي

$$\theta_0 := \theta_0 - \alpha * \frac{1}{m} * \sum_{i=1}^m ((h_{\theta}(x^i) - y^i))$$

فالمجموع بدل ما كان 2095 بعد الضرب هايبقا قبل الضرب 28 لما يضرب في 0.002/8. الناتج هايكون 0.007 لما نطرح 0 theta الى هي 1 هايكون الناتج

$$\text{Theta } 0 = 1 - 0.007 = 0.993$$

لكن

$$\text{Theta } 1 = 3 - ((0.002 / 8) * (2095)) = 2.48$$

وبعد كده بقا عن طريق قيم ال thetas الجديدة ابتدى اني اعمل iterations لحد اما اوصل لل global minimum.

## ال vectorize version of gradient descent

$$\theta := \theta - \frac{\alpha}{m} * X^T * (X\theta - \bar{y})$$

Assume  $X_0^i = 1$  for 8 training examples

حول كل حاجه من المعادله بتاعت ال vectorize ل matrix or vector عشان نقدر نتعامل معاها

لذلك هاتكون بالشكل ده بعد اما افترضنا اني  $X_0$  هاتشاي 1 لما اجي اضربها في theta فقط لكن في العادي هي  $8*1$

هاتكون بالشكل ده

300	1	100
285	1	95
270	1	90
240	1	80
235	1	80
200	1	70
205	1	70
180	1	60

ال  $\theta$  ال 3  $\frac{1}{3}$  x وال y

بما ان  $\theta = 2*1$  وال  $X = 8*2$  بيقا  $X\theta$  ينفع يضربوا لان  $2*1$  and  $8*2$  والناتج هايكون matrix  $8*1$  والي هي تنفع تطرح من y والناتج هايكون  $8*1$  vector of الجزء الاول هايكون  $8*1$  بعد ما عمل transpose ل  $X^T$  فساعتها



تقدر اعمل multiplication بين جزئين المعادله والناتج الى هو هايكون رقم واحد يضرب في  $\frac{\alpha}{m}$  وبعدها نطرح من قيم ال thetas الى اكبر من 0 لان theta 0

$$\theta := \theta - \frac{\alpha}{m} * X^T * (X\theta - \vec{y})$$

$$\frac{1}{3} - \frac{.002}{8} * \begin{matrix} 100 & 95 & 90 & 80 & 80 & 70 & 70 & 60 \end{matrix} * \begin{pmatrix} 1 & 80 \\ 1 & 80 \\ 1 & 70 \\ 1 & 70 \\ 1 & 60 \end{pmatrix} * \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} - \begin{pmatrix} 240 \\ 235 \\ 200 \\ 205 \\ 180 \end{pmatrix} =$$

$$\frac{1}{3} - \frac{.002}{8} * \begin{matrix} 100 & 95 & 90 & 80 & 80 & 70 & 70 & 60 \end{matrix} * \begin{pmatrix} 1 & 80 \\ 1 & 80 \\ 1 & 70 \\ 1 & 70 \\ 1 & 60 \end{pmatrix} * \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} - \begin{pmatrix} 301 \\ 286 \\ 271 \\ 241 \\ 241 \\ 211 \\ 211 \\ 181 \end{pmatrix} = \begin{pmatrix} 300 \\ 285 \\ 270 \\ 240 \\ 235 \\ 200 \\ 205 \\ 180 \end{pmatrix}$$

$$\frac{1}{3} - \frac{.002}{8} * \begin{matrix} 100 & 95 & 90 & 80 & 80 & 70 & 70 & 60 \end{matrix} * \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 11 \\ 6 \\ 1 \end{pmatrix} = \frac{1}{3} - \frac{.002}{8} * 2095 = \frac{1}{3} - .52$$

بعد كده بقا بقدر اني اجرب عدد من الخطوات لحد اما اوصل للـ *global minimum*.

الى فات ده كده كله كان تطبيق لما يكون عدد ال *features* واحد فقط.

تطبيق عملي على *Multivariable Regression*.

لما يكون عندي اكثر من *feature* فهنا عندي *3 features* بييمثلوا مميزات السيارة وال *y* وهى سعر السيارة بالنسبه لل *3 feature* الخاصين بيها فهنا مثلا اول مثال هو

$$X1 = 5 \text{ and } X2 = 20 \text{ and } X3 = 6$$

بمعنى اني اول سياره الى هي *X1* على حسب ال *features* الى المذكوره فوق سعرها بيكون هو *y=12*.

فهنا عندي *5 training example and 3 features* لذلك *m=5 and n=3*.

$X1$	$X2$	$X3$	$Y$
العمر	القدرة	الاسطوانات	السعر
5	20	6	114
5	35	6	120
6	38	8	123
7	40	8	121
7	46	10	135

بعد كده هافترض قيم  $thetas$  مقابل كل  $feature$  ولكن هايكون مقابل  $theta_0$  هايكون  $X_0 = 1$  لكل مثال.

فلنفترض قيم  $thetas$

$$\theta_0 = 5 \ \& \ \theta_1 = 2 \ \& \ \theta_2 = 3 \ \& \ \theta_3 = 6$$

اولا ال  $Cost function for Multivariable Regression$  هاتكون

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

$$h_{\theta}(x^i) = \theta^T X$$

كل  $training example$  بيتمثل عندى  $vector of 4*1$  الى هو بعد افترض انى  $X_0 = 1$  + ال  $feature 3$  هايكون 4 بالنسبة لمثال واحد هايكون  $1 * 4$  لكن لو بتكلم عن  $X$  كلها هنا يتمثل  $4*5$  الى هو  $4 features$  فى عدد ال  $training data$  الى هما 5 وبما انى  $theta = 4*1$  فلما اعملها  $transpose$  هاتبقا  $1*4$  وهنا فقدر اضربها فى ال  $X$  الى ههى  $4*5$  والنتائج هايكون  $5*1 vector$  هاتمطرحه من  $y$  الى هى  $5*1$  وبعدها نربع كل رقم من ال  $vector$  ونجمع ثم نضرب فى  $1/2m$ .

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

اولا

$$h_{\theta}(x^i) = \theta^T X = \begin{matrix} & & & & & 1 & 1 & 1 & 1 & 1 \\ & & & & & 5 & 5 & 6 & 7 & 7 \\ 5 & 2 & 3 & 6 & * & 20 & 35 & 38 & 40 & 46 \\ & & & & & 6 & 6 & 8 & 8 & 10 \end{matrix} = \begin{matrix} 111 & 156 & 179 & 187 & 217 \end{matrix}$$

هانعمل  $transpose$  عشان نقدر نطرح الارقام ديه من  $y$  فهايكون الى هى المعادلة ديه  $(h_{\theta}(x^i) - y^i)$

111	114	-3	9
156	120	36	1296
179 - 123 = 56	then to power of 2 =		3136 =
187	121	66	4356
217	135	82	6724

$$\sum_{i=1}^m (h_{\theta}(x^i) - y^i)^2 = 9 + 1296 + 3136 + 4356 + 6724 = 15521 * \frac{1}{2m} = 15521 * \frac{1}{2*5} = 1552.1$$

**vectorize version solution of cost function with multivariable regression**

بداًم كده كده بضرب فى الاخر فى matrix of X فى vector of theta فسواء عملت transpose لاي فيهم وضربته فى التانى الناتجين هايكونوا زى بعض الاختلاف فقط هايكون فى deamination

$$J(\theta) = \frac{1}{2m} * (X\theta - \vec{y})^T * (X\theta - \vec{y})$$

$$(X\theta - \vec{y}) = \begin{matrix} 1 & 5 & 20 & 6 & 5 & 114 & 111 & 114 & -3 \\ 1 & 5 & 35 & 6 & 2 & 120 & 156 & 120 & 36 \\ 1 & 6 & 38 & 6 & 3 & 121 & 187 & 121 & 66 \\ 1 & 7 & 40 & 8 & 6 & 121 & 187 & 121 & 66 \\ 1 & 7 & 46 & 10 & 6 & 135 & 217 & 135 & 82 \end{matrix} - 123 = 179 - 123 = 56 \text{ then } (X\theta - \vec{y})^T = \begin{matrix} -3 & 36 & 56 & 66 & 82 \end{matrix}$$

$$(X\theta - \vec{y})^T * (X\theta - \vec{y}) = \begin{matrix} -3 & 36 & 56 & 66 & 82 \end{matrix} * \begin{matrix} -3 \\ 36 \\ 56 \\ 66 \\ 82 \end{matrix} = 15521 * \frac{1}{2*5} = 1552.1$$

**ثانياً ال Gradient Descent** هايتم التطبيق فى المعادلات الخاصة بيه سواء كان vectorize or not

$$\theta := \theta - \alpha * \frac{1}{m} * \sum_{i=1}^m ((h_{\theta}(x^i) - y^i) * X^i)$$

**Vectorize version**

$$\theta := \theta - \frac{\alpha}{m} * X^T * (X\theta - \vec{y})$$

واخلى بالى انى فى حاله  $\theta_0$  هى الواحديه الى بجيها لوحدها بالمعادله نفسها عدا الضرب فى  $X^i$ .

بإذن الله الجاى هايكون نفس التطبيق على Logistic regression واتمنى من  
الله اكون قدرت اساعد بحاجه بسيطة وإن شاء الله فقدر اكمل باقى الكورس على  
خير.