التاريخ: / /　　　　الموضوع: L(1)

Introduction to Machine learning:

* programming : input + equations = output　Ex: $\overset{\text{output}}{y} = 3\underset{\text{input}}{x^2}$

* Machin learning: input + Output = Equation

Ex:
مثال لو أنا معايا عدد من الشقق بالمساحة
بتاعتها بالسعر حسب المساحة والمطلوب
أعرف لو ظهرت شقة جديدة أحدد سعرها
بناءً على الـ dataset دي

60 Final
25 mid
15 Lab
غياب 3 مرات "محاضرة" سحب
غياب 3 مرات "سكشن" خصم في أعمال السنة

input	output
Size	Price
90	✓
100	✓
105	✓
⋮	⋮

input →
بيتعلم منهم　ML Algorithms → Equation (Model)
output →
بحيث لو معايا input أقدر أطلع الـ output

Data Set الجدول

- Row ≡ training sample (example)
- m = Number of rows (training samples)

التاريخ: / /　　　　　　　　　　　　　　　　　　　　الموضوع:

ML Algorithms:

1) Supervised learning
2) Unsupervised learning } Content ✓
3) Reinforcement learning

Supervised learning

• Dataset: input + output
→ Regression
→ Classification

Unsupervised learning

• Dataset: input only
→ Clustering

* Regression: تنبؤ

* output (Continous Values)

Ex:

Size	price
90	1M
100	1.2M

1000,000 : 2000,000
Range

* Classification: تصنيف

* output (Discrete Values)

Ex: 0, 1, 2, ...

x	y
	1
	1
	1
	0

1 (yes)
0 (No)

* Clustering: Divide dataset to groups

✓ → group A
• → group B

x
✓
•
✓
✓
•
•

فأما أدخله حاجة مش موجودة قبل كدة هيحطها في group على أساس الصفات أقرب لأنهى جروب

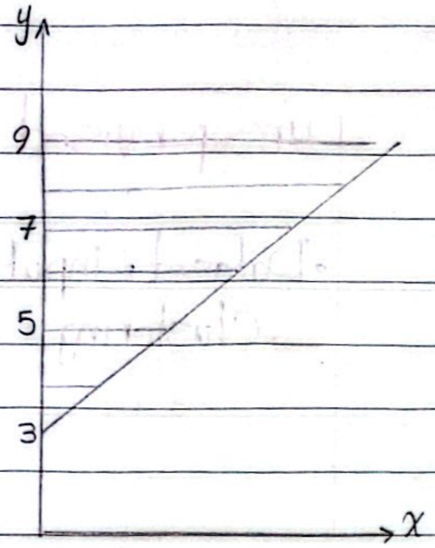
* linear Regression with one Variable:

Regression → Dataset (i/p + o/p) → Continous

one → One i/p (Feature)

linear → Straight line Equation

i/p: x	actual output: y
0	3
1	5
2	7
3	9



$y = mx + b$

Model Representation:

Dataset

↓

ML Algorithm
[liner regression with one Variable]

↓

$x \xrightarrow{i/P}$ Equation → predicted output h (hypothesis Function)

$h = \theta_0 + \theta_1 x$

unknowns: θ_0, θ_1

التاريخ: / /　　　　　　　　　　　　　　الموضوع: L(2)

1 linear regression with one Variable:

linear regression → Supervised: input + output → Continous

one Variable → one input, one Feature

input	output

output → actual output

learning Algorithm
linear regression
with one variable

x → Equation (Model) → h

h → predicted output
h (hypothesis)

From straght line equation

$$y = mx + b$$

$$y = b + mx$$

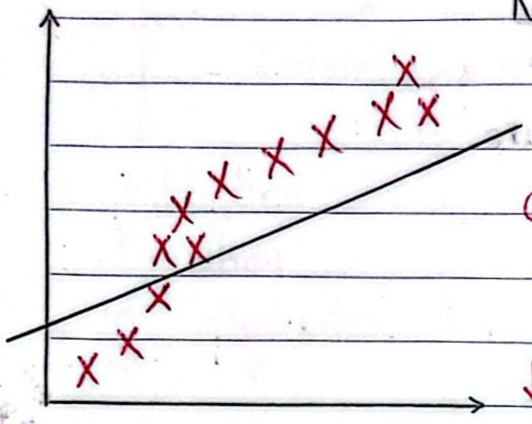
$$\boxed{h = \theta_0 + \theta_1 x}$$

θ_0, θ_1 unknowns

التاريخ: / / الموضوع:

$h = \theta_0 + \theta_1 x$

أول حاجة هشوف قيم لـ θ_0, θ_1 ← Random



• أول محاولة نسبة جداً ليه؟ القيم random، الـ straight line مش معدي على نقط كتير.

• كل محاولة الخط بيتحسن ← Error ↓

Error Equation (Cost function) MSE

$$J = \frac{1}{2m} \sum (\underbrace{h}_{\text{predicted}} - \underbrace{y}_{\text{actual}})^2$$

→ Target : Minimize J

minimize ↳ $h = \theta_0 + \theta_1 x$

↳ minimize θ_0, θ_1

Gradient descent algorithm:

$$\theta_{new} = \theta_{old} - \alpha \frac{\partial J}{\partial \theta}$$

→ هظبطها مرتين عشان عندي θ_0, θ_1

• $\theta_{0\,new} = \theta_{0\,old} - \alpha \dfrac{\partial J}{\partial \theta_0}$

القوس power

$$\frac{\partial J}{\partial \theta_0} = \frac{\cancel{2}}{\cancel{2}m} \sum \underbrace{(h-y)'}_{} (1)$$

تفاضل القوس ← $(\theta_0 + \theta_1 x - y)$

$$\frac{\partial J}{\partial \theta_0} = \frac{1}{m} \sum (h-y)$$

$$\boxed{\theta_{0\,new} = \theta_{0\,old} - \frac{\alpha}{m} \sum (h-y)}$$

• $\theta_{1\,new} = \theta_{1\,old} - \alpha \dfrac{\partial J}{\partial \theta_1}$

$$\frac{\partial J}{\partial \theta_1} = \frac{2}{2m} \sum \underbrace{(h-y)}_{} (x)$$

تفاضل القوس ← $(\theta_0 + \theta_1 x - y)$

$$\frac{\partial J}{\partial \theta_1} = \frac{1}{m} \sum (h-y) x$$

$$\boxed{\theta_{1\,new} = \theta_{1\,old} - \frac{\alpha}{m} \sum (h-y) x}$$

Example:

Consider the following training data for linear regression you're using linear regression and assume

$\theta_0 = 2$, $\theta_1 = 2$ and $\alpha = 0.1$

x	y
2	3
3	4
4	5

• Compute the total error and update the values of θ_0, θ_1 after the first iteration.

Solution

• $J = \frac{1}{2m} \sum (h-y)^2$

• $\theta_0 = \theta_0 - \frac{\alpha}{m} \sum (h-y)$

• $\theta_1 = \theta_1 - \frac{\alpha}{m} \sum (h-y)x$

• $h = \theta_0 + \theta_1 x$

$h = 2 + 2x$

first iteration:

$\theta_0 = 2, \theta_1 = 2$

x	y	h	$(h-y)$	$(h-y)x$	$(h-y)^2$
2	3	6	3	6	9
3	4	8	4	12	16
4	5	10	5	20	25
		$\sum$	12	38	50

$J = \frac{1}{2m} \sum (h-y)^2 = \frac{1}{6}(50) = 8.3$ $J = 8.3$ #

• $\theta_0 = \theta_0 - \frac{\alpha}{m} \sum (h-y)$

$\theta_0 = 2 - \frac{0.1}{3}(12) = 1.6$ $\theta_0 = 1.6$ #

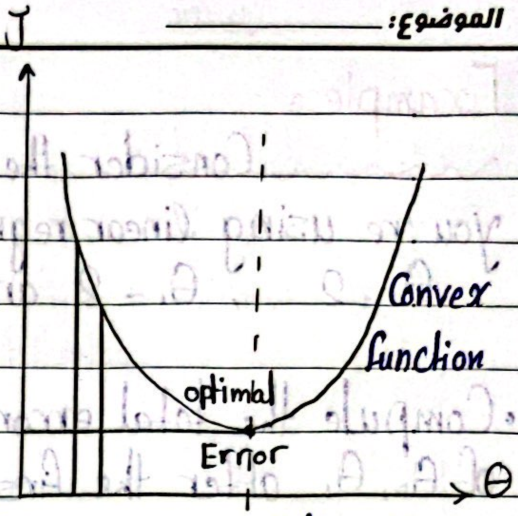
• $\theta_1 = \theta_1 - \frac{\alpha}{m} \sum (h-y)x$

$\theta_1 = 2 - \frac{0.1}{3}(38) = 0.73$ $\theta_1 = 0.73$ #

الموضوع: ________ التاريخ: / /

- $J = \frac{1}{2m} \Sigma (h-y)^2$
- $\theta_{new} = \theta_{old} - \alpha \frac{\partial J}{\partial \theta}$

ليه بنستخدم MSE ؟ لإن زي الرسمة
في الحالتين بقدر أقرب من optimal Error



$\theta \uparrow \rightarrow J \downarrow$	$\theta \uparrow \rightarrow J \uparrow$
$\frac{\partial J}{\partial \theta}$ (-ve)	$\frac{\partial J}{\partial \theta}$ (+ve)
$\theta_{new} = \theta_{old} - \alpha \frac{\partial J}{\partial \theta}$	$\theta_{new} = \theta_{old} - \alpha \frac{\partial J}{\partial \theta}$
$\theta_{old} \underbrace{- \; -ve}_{+}$	$\theta_{old} \underbrace{- \; +ve}_{-}$
بالتالي θ_{new} هتزيد	بالتالي θ_{new} هتقل

* linear Regression with multiple Variables

اللي أخدناه في اللي فات كان 1variable ← 1 feature

Regression : Contains input & output

Multiple Variable : Multiple Features ($x_1, x_2, \ldots$

linear : linear function → High processing function

Data set with multiple features

x_1	x_2	x_3	x_4	$x_5 \ldots$	x_n	y

يعتبر نفس معادلات ال 1variable
بس هنزود θ حسب عدد ال features
وكدة كدة θ_0 ثابتة

$\theta_0 \longrightarrow \theta_n$

↳ Features number

with one Variable	with multiple Variables
• $h = \theta_0 + \theta_1 x$	• $h = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_n x_n$
• $j = \frac{1}{2m} \sum (h-y)^2$	• $j = \frac{1}{2m} \sum (h-y)^2$
• $\theta_0 = \theta_0 - \frac{\alpha}{m} \sum (h-y)$	• $\theta_0 = \theta_0 - \frac{\alpha}{m} \sum (h-y)$
• $\theta_1 = \theta_1 - \frac{\alpha}{m} \sum (h-y) x$	• $\theta_1 = \theta_1 - \frac{\alpha}{m} \sum (h-y) x_1$
ال θ لينا كـ 1variable أو multiple ↳ Gredient Decent	• $\theta_2 = \theta_2 - \frac{\alpha}{m} \sum (h-y) x_2$
	⋮
	• $\theta_n = \theta_n - \frac{\alpha}{m} \sum (h-y) x_n$

الموضوع: ______ التاريخ: / /

*feature scalinig:

① Mean - normalization
② Min-Max scaling

x_1	x_2	y
⋮	⋮	
⋮	⋮	
⋮	⋮	
⋮	⋮	

Multiple
لو عندى مجموعة من ال features ، x_1, x_2 لو قيمهم مختلفة
يعنى x_1 قيمها صغيرة أوى و x_2 كبيرة أوى
لو عطيتها زى ماهى لـ linear Regression
الموديل هياخد وقت طويل أوى
وممكن ميوصلش أصلا بسبب إنهم بُعاد أو عن بعض

$x_1 : 0 \rightarrow 5$
$x_2 : 100 \rightarrow 30000$

1 Mean - normalization:

$$x' = \frac{x - \mu}{max - min}$$

; $x \longrightarrow$ الى عايزة أحولها
; min, max $\longrightarrow$ أعلى وأقل قيمة من عمود x_1 مثلا هطلع

$x' \downarrow$ new $x \longrightarrow$ after scaling ; $\mu \longrightarrow$ average $\longrightarrow \dfrac{\text{مجموع القيم}}{\text{عددهم}}$

2 Min-Max scalinig:

$$x' = \frac{x - min}{max - min}$$

Ex: Given dataset, Apply feature scaling using → Mean-normalization
→ Min-Max scaling

x_1	x_2	y
200	2.2	✓
160	1.9	✓
180	2.1	✓
240	2	✓

Solution

• Mean-normalization

→ after

$$x' = \frac{x - \mu}{max - min}$$

x_1'	x_2'	y
0.0625	0.5	
0.4375	-0.5	
-0.1875	0.17	
0.5625	-0.17	

$$\mu_1 = \frac{200+160+180+240}{4} = 195$$

$$x_1' = \frac{x_1 - 195}{240-160} \rightsquigarrow x_1' = \frac{x_1 - 195}{80}$$ ← ① هحول بيها كل x_1

$$\mu_2 = \frac{2.2+1.9+2.1+2}{4} = 2.05$$

$$x_2' = \frac{x - 2.05}{2.2-1.9} \rightsquigarrow x_2' = \frac{x_2 - 2.05}{0.3}$$ ← ② هحول بيها كل x_2

• Min-Max scalinig

→ after

$$x' = \frac{x - min}{max - min}$$

x_1'	x_2'	y
0.5	1	
0	0	
0.25	0.66	
1	0.33	

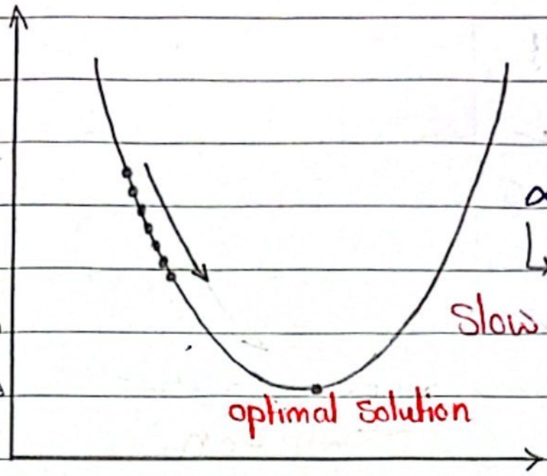
$$x_1' = \frac{x_1 - 160}{240-160} \rightsquigarrow x_1' = \frac{x_1 - 160}{80}$$

$$x_2' = \frac{x_2 - 1.9}{2.2-1.9} \rightsquigarrow x_2' = \frac{x_2 - 1.9}{0.3}$$

* learning rate: (α) → given

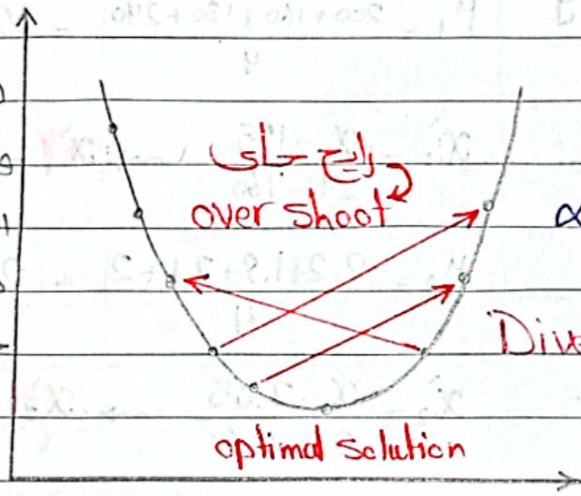
ايه تأثير α؟ هي معدل التعلم
يعني الـ model بتاعي هيتعلم بسرعة
ولا ببطء (الـ range بتاعها من 0: 1 zero)

هنا لو α ← too small هيتحرك ببطء
فالوصول لنقطة الحل ← Convergence
هيكون بطيء جداً slow جداً
ودي حاجة مش حلوة XXX



$$\theta_{new} - \theta_{old} - \alpha \frac{\partial J}{\partial \theta}$$

هنا زي الرسمة الأسهم رايحة جاية
ومش عارفين نوصل لنقطة الحل
الـ تباعد ← Divergence جداً
طب ليه بيحصل؟ بسبب إني باخد
خطوات كبيرة فقيمة الـ θ بتعلى



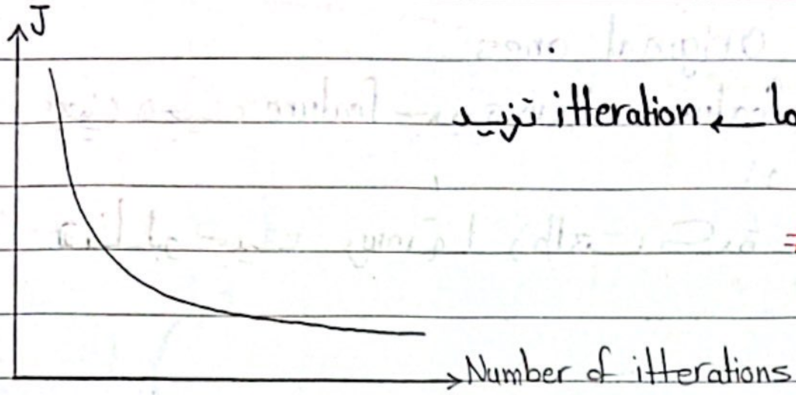
$$\theta_{new} = \theta_{old} - \frac{\alpha}{m} \sum h - y$$

الحل لا تبقى too small, too large ← Solution: α small

أغلب القيم اللي بتظبط (0.001, 0.01, 0.1)
بناء على projects سابقة

الموضوع: ______ التاريخ: / /

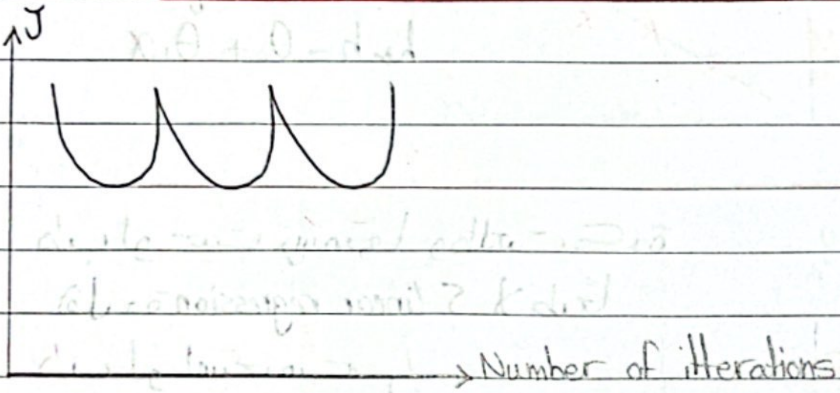
• Make sure gradient descent is working Correctly :



لو أنا ماشي صح المفروض

الـ model بتاعي ← J تقل كل ما ← itteration تزيد

Correct model



هنا ايه بيحصل ؟ رايح جاي

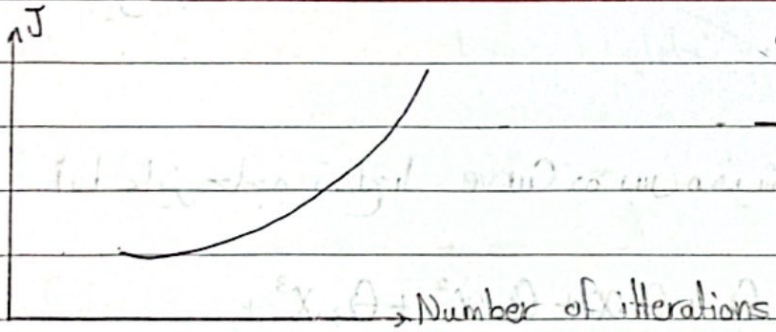
↳ over shooting

↓

α too large

↓

نرجع نظبطها شوية



هنا غلط خالص طب ايه السبب ؟

→ Code Error or Rule Error

يعني الطبيعي $\theta_0 = \theta_0 - \frac{\alpha}{m}\sum(h-y)$

اتلغبطت وحطيت ⊕ مثلاً

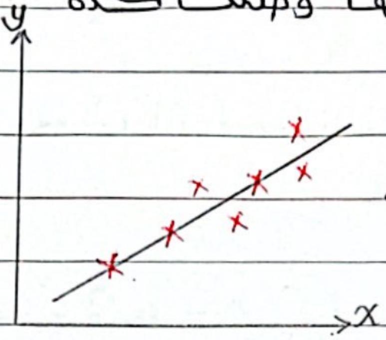
*Feature Engineering:

Extract new features from Original ones

يعني هجيب feature جديدة من ال features القديمة الي عندي

Size x	Price y

هنا لو جيت رسمتها وطلعت كدة



$\therefore$ linear Regression

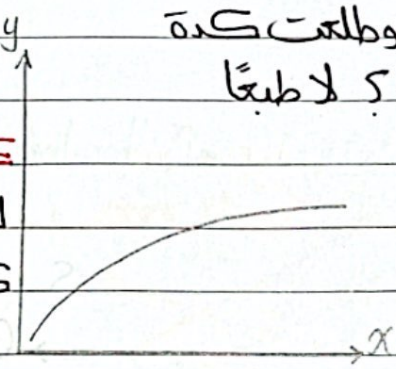
$\Rightarrow h = \theta_0 + \theta_1 x$

طب لو جيت رسمتها وطلعت كدة

هل دة linear regression ؟ لا طبعًا

طب لو استخدمته

كبير جدًا Error $\Rightarrow$



Polynomial regression

كثيرات الحدود $\Leftarrow$

لو هطلع من ال feature الي عندي x

كذا feature جديدة $\rightarrow$ $x^2, x^3, \dots$

أنا عايز حاجة تطلع ال Curve دة بس مفيش غير x واحدة بس !!

$x_1 \quad x_2$

x	x^2	$x^3 \dots$	y

$\therefore h = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \dots$

طب افرض المعادلة دي مظبطتش بردو ؟!

هروح أزود features تانية وأشوف وهكذا

أنا برسم ال Data وأشوف على حسب الي طلعلي هشتغل بايه ؟

الموضوع: ______ التاريخ: / /

Classification

① Binary Classification

Binary → 2 Classes ثنائي

- person
  - → 1 → مريض
  - → 0 → مش مريض

- Tumor (ورم)
  - → Malignant (1) خبيث
  - → Bengin (0) حميد

$y \in \{0, 1\}$

② Multi-Class Classification problem

لو عندي مجموعة صور (قطط، كلاب، سمك)

مثلا 3 Classes لـ

يعني مش 1,0 بس

$y \in \{0, 1, 2\}$

logistic Regression: Binary بيتعامل مع الـ classification ← أول Algorithm في الـ Classification

Ex: Tumor → mali (1) خبيث / → Ben (0) حميد

الـ output → 0 or 1 → Classification

لو جيت طبقت linear reg. → $h = \theta_0 + \theta_1 x$

قيم الـ h → $h < 0$ or $h > 1$

ينفع أقول أنا عيان 0.4؟ لا أكيد

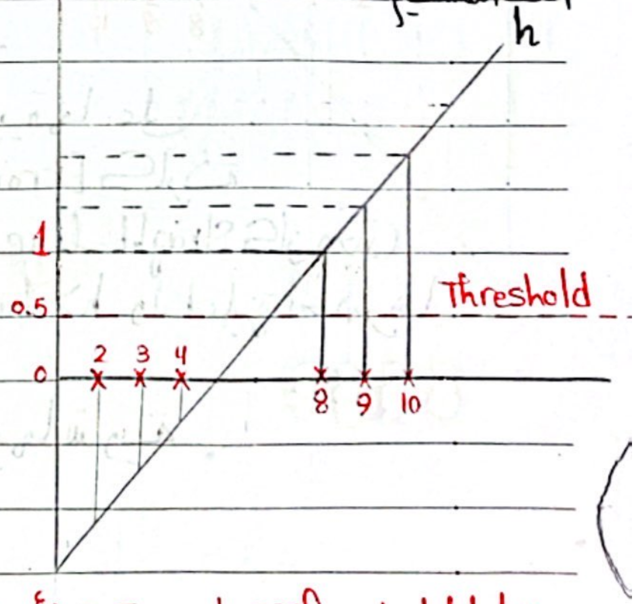
لازم 1 or 0 ↓ Threshold

$h \geq$ Threshold → 1

$h <$ Threshold → 0

x	y
2	0
3	0
4	0
8	1
9	1
10	1

الخط المستقيم



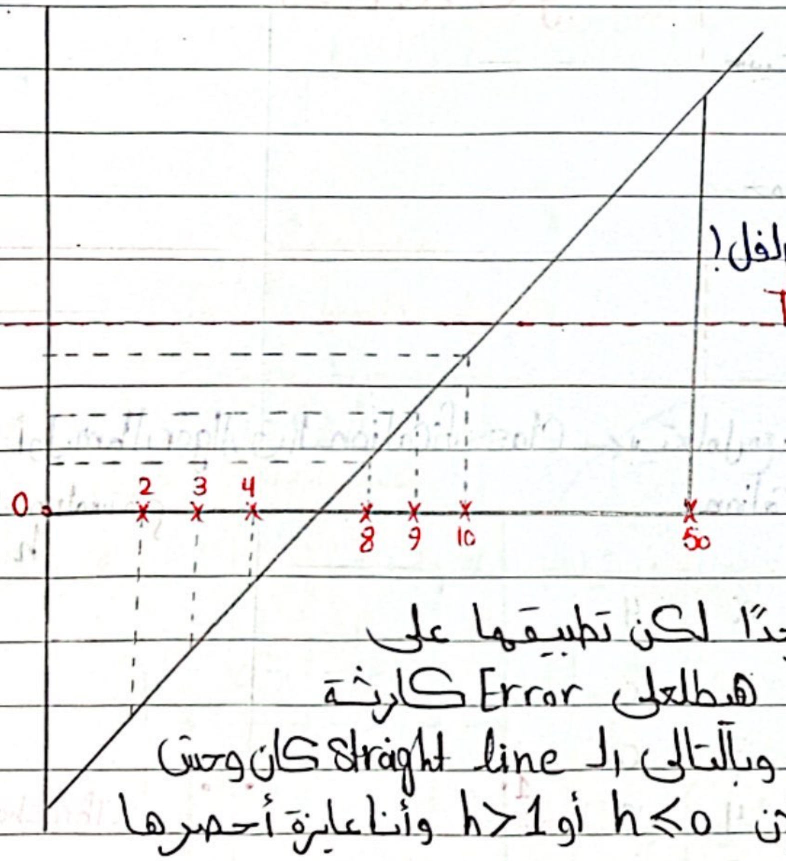
هنا الـ Threshold بتساوي 0.5 فأي قيمة أصغر من 0.5 هديها Zero ← Ben حميد

أكبر من أو تساوي 0.5 هديها 1 ← Mali خبيث

الموضوع: التاريخ: / /

تاني ايه فكرة الـ Threshold ← حد فاصل, أنا عندي قيم مختلفة بأطلع منها إن الورم خبيث Mali ← 1 أو حميد Ben ← 0
فقولنا الـ Threshold مثلا بـ 0.5 أصبح أي قيمة أصغر من 0.5 هتكون بـ Zero وأي قيمة أكبر من أو تساوي 0.5 هتكون بـ 1.
حل جميل وكل حاجة بس غلطة صغيرة هتبوظ الدنيا

⇐ المشكلة في المعادلة بتاعة الـ linear regression ← $h = \theta_0 + \theta_1 x$
ليه بقا؟ لإن لما أزود قيم الـ x بشكل مختلف ← هيأثر على الـ straight line فبالتالي النتائج هتطلع غلط بالنسبة للـ Threshold زي الرسمة دي.



هنا النقطة:
2,3,4 → 0
8,9,10 → 0
< Threshold
يعني الراجل جي بيموت وبتقوله لا زي الفل!
→ Error XXX

فكرة الـ Threshold كويسة جدًا لكن تطبيقها على المعادلة دي $h = \theta_0 + \theta_1 x$ هيطلعلي Error كارثة
لو حصل تغيير في قيم x وبالتالي الـ straight line كان وحش فحصل تشوه في الـ Data لإن $h \leq 0$ أو $h > 1$ وأنا عايزة أحصرها بين الـ 0,1
فهاخد المعادلة دي أغير فيها شوية

Binary Segmoid:

$$g(z) = \frac{1}{1+e^{-z}} \quad ; \; -\infty \leq z \leq \infty \quad ; \; 0 \leq g(z) \leq 1$$

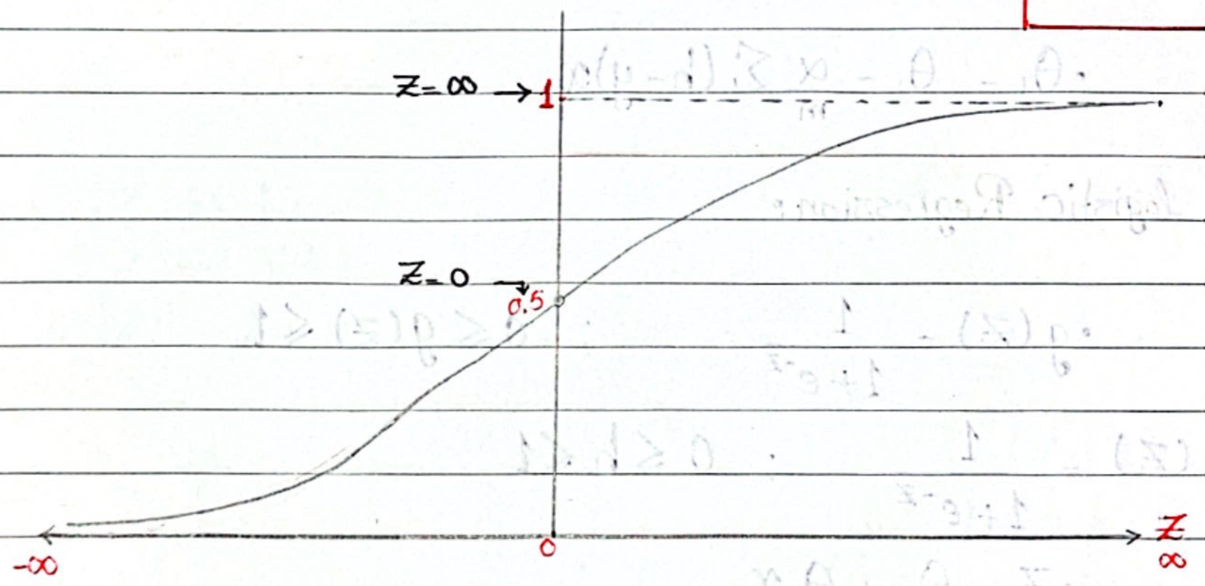
→ Assume:

① $z = 0 \; , \; g(z) = \frac{1}{1+e^{0}} = \frac{1}{1+1} = 0.5$

② $z = \infty \; , \; g(z) = \frac{1}{1+e^{-\infty}} = \frac{1}{1} = 1$

③ $z = -\infty \; , \; g(z) = \frac{1}{1+e^{-(-\infty)}} = \frac{1}{1+\infty} = \frac{1}{\infty} = 0$

$\frac{1}{\infty} = 0$
$e^{0} = 1$
$e^{\infty} = \infty$
$e^{-\infty} = 0$



كدة ايه الى ال function دى عملته؟ إنها خلت القيم محصورة بين 0,1
بدل ما كان بيطلع Error بسبب إن ال h ممكن تطلع أكبر من 1 أو أصغر من 0

* logistic Regression:

يستخدم مجال Binary classification يعني عندي 2 Classes

→ Remember: linear Regression

- $h = \theta_0 + \theta_1 x$
- $J = \frac{1}{2m} \sum (h-y)^2$
- $\theta_0 = \theta_0 - \frac{\alpha}{m} \sum (h-y)$
- $\theta_1 = \theta_1 - \frac{\alpha}{m} \sum (h-y)x$

→ logistic Regression:

- $g(z) = \frac{1}{1+e^{-z}}$; $0 \leq g(z) \leq 1$
- $h = g(z) = \frac{1}{1+e^{-z}}$; $0 \leq h \leq 1$
- $z = \theta_0 + \theta_1 x$

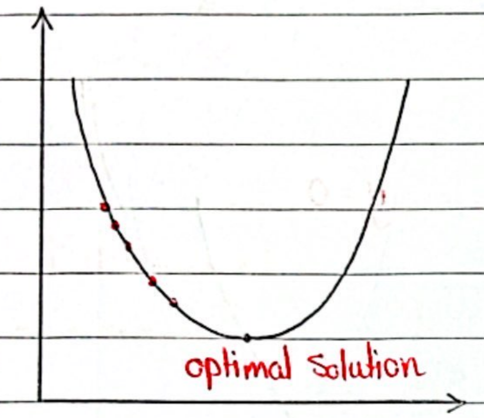
* Threshold:

$h \geq$ threshold → ① ← قيمة الـ h

$h <$ threshold → ⓪ ← قيمة الـ h

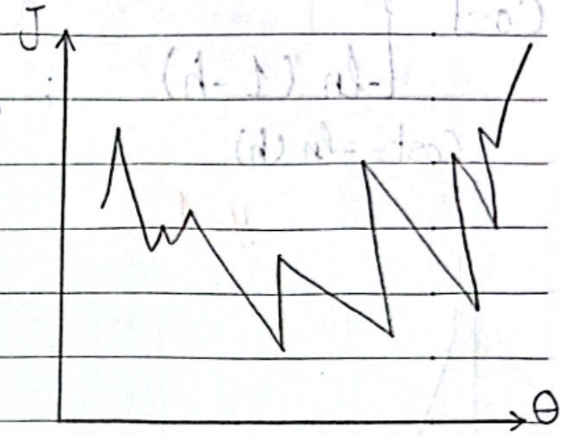
$$J = \frac{1}{2m}\sum (h-y)^2$$

linear Regression



هنا القانون كويس لإني بقرب من الـ optimal solution

logistic Regression



هنا حسب الـ initial هيكون في optimal
يعني الـ optimal solution مش ثابت
XXX Non Convex

القانون بتاع $h = \frac{1}{1+e^{(\theta_0+\theta_1 x)}}$

هنا ممكن موصلش للـ optimal أصلا
لإن θ كل شوية تتغير فمش هينفع
استخدم قانون الـ $J=\frac{1}{2m}\sum(h-y)^2$ X XX

* سؤال امتحان ليه $J=\frac{1}{2m}\sum(h-y)^2$ مينفعش في الـ logistic Regression؟

لإنه بيطلع كذا optimal حسب الـ initial اللي هيبدأ من عندها

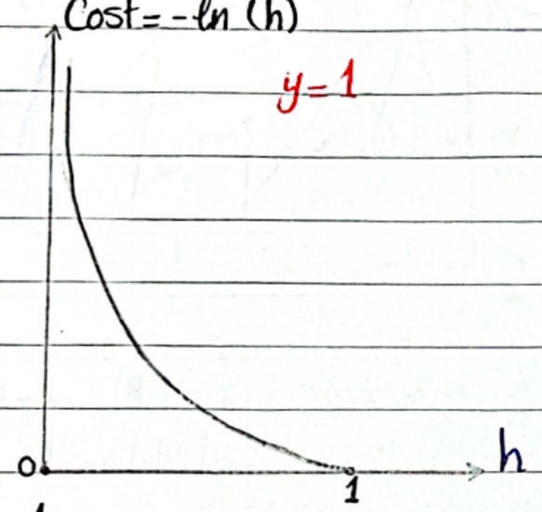
التاريخ: / / الموضوع:

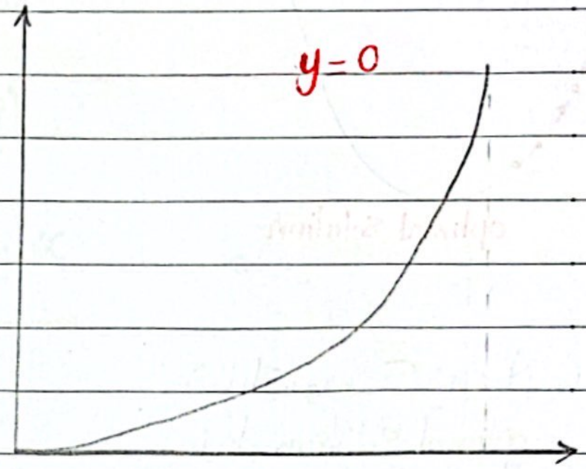
logistic Regression:

$$J = \frac{\sum Cost}{m}$$

$$\ln 0 = -\infty$$
$$\ln 1 = 0$$

$$Cost = \begin{cases} -\ln(h) & ; y = 1 \\ -\ln(1-h) & ; y = 0 \end{cases}$$





$y=1$

$Cost = -\ln(h)$

- $h = 0 \rightarrow Cost = \infty$
- $h = 1 \rightarrow Cost = 0$

$y=0$

$Cost = -\ln(1-h)$

- $h = 0 \rightarrow Cost = 0$
- $h = 1 \rightarrow Cost = \infty$

كلما h تزيد هبعد عن الحقيقة ال Error ↑

كلما h تقل يقرب من الحقيقة ال Error ↓

Rule ← $\boxed{Cost = -y \ln(h) - (1-y)\ln(1-h)}$

الموضوع: ______ التاريخ: / /

* Minimize $J \longrightarrow$ Target

$\hookrightarrow \theta_0, \theta_1$ لو كويسين هيخلوا h كويسة هيبقى الـ Cost كويس $\rightarrow J$

$$J = \frac{\Sigma Cost}{m}$$

$\longrightarrow$ Gradient descent Algorithm : $\theta_{new} = \theta_{old} - \alpha \frac{\partial J}{\partial \theta}$

$\hookrightarrow \theta_{0_{new}} = \theta_{0_{old}} - \frac{\alpha}{m} \Sigma (h-y)$

$\hookrightarrow \theta_{1_{new}} = \theta_{1_{old}} - \frac{\alpha}{m} \Sigma (h-y)x$

* Summary:

- $Z = \theta_0 + \theta_1 x$
- $h = \frac{1}{1+e^{-Z}}$
- $J = \frac{\Sigma Cost}{m}$
- $Cost = -\ln(h) \quad ; y=1$
- $Cost = -\ln(1-h) \quad ; y=0$
- $\theta_0 = \theta_0 - \frac{\alpha}{m} \Sigma (h-y)$
- $\theta_1 = \theta_1 - \frac{\alpha}{m} \Sigma (h-y)x$

Example: Consider the following training data for logistic regression:

x	y
3	1
5	1
10	0
12	0

You are using logistic regression and values of thetas came out to be as: $\theta_0 = 2$, $\theta_1 = 1$

- Draw the decision boundry
- Compute the total error.
- Find θ_0, and θ_1 for the next iteration, assume $\alpha = 0.1$

Solution

x	y	z	h	$(h-y)$	$(h-y)x$	Cost
3	1	5	0.99331	-0.00669	-0.02007	$6.71 * 10^{-3}$
5	1	7	0.99909	-0.00091	-0.00455	$9.1 * 10^{-4}$
10	0	12	0.99999	0.99999	9.9999	11.51
12	0	14	0.999999	0.999999	11.999988	13.82
			Σ	1.992	21.98	25.34

- $J = \frac{\Sigma \text{Cost}}{m} = \frac{25.34}{4} = 6.33$ #

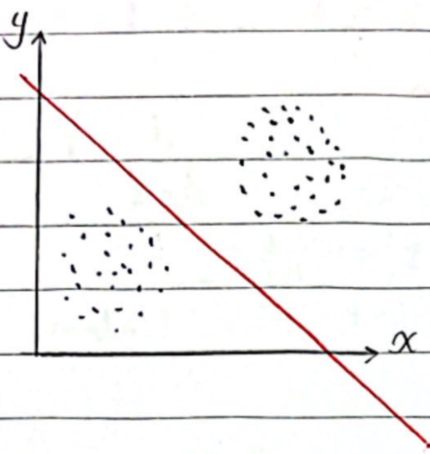
- $\theta_0 = \theta_0 - \frac{\alpha}{m} \Sigma (h-y)$

$\theta_0 = 2 - \frac{0.1}{4} [1.992] = 1.95$ #

- $\theta_1 = \theta_1 - \frac{\alpha}{m} \Sigma (h-y)x$

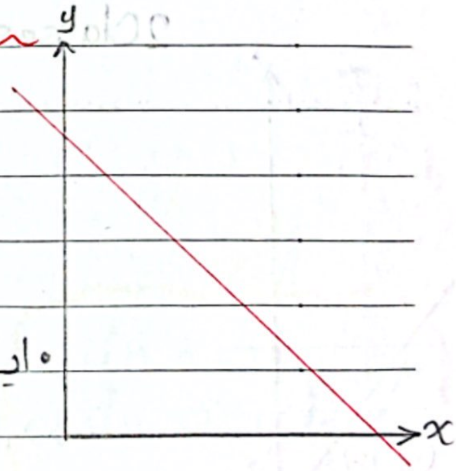
$\theta_1 = 1 - \frac{0.1}{4} [21.98] = 0.45$ #

Decision boundry:



Decision boundry , الخط الفاصل بين ال classes
بيطلع من ال model الى حللناه في الحاجات السابقة

*Decision boundry of logistic Regression:



$\cdot h = \dfrac{1}{1+e^{-z}} \Longrightarrow z = \ln \dfrac{h}{1-h}$

$\cdot z = \theta_0 + \theta_1 x \longleftarrow$ Decision boundry الأهم لل

ايه الفرق بينها وبين ال Threshold ؟

Assume Threshold $= 0.5$

$z = \ln \dfrac{0.5}{1-0.5} = \ln \dfrac{0.5}{0.5} = \ln 1 = 0$ # at threshold ①

افتراض ال كمثال : $z = \theta_0 + \theta_1 x \Longrightarrow z = 1 + 2x$ ②

من ①، ② $\quad 1 + 2x = 0$

هو دة ال decision boundry

الموضوع: ______ التاريخ: / /

Ex: Assume the model is trained by logistic Regression

$z = 1 + 2x_1 + 3x_2$ and threshod $= 0.5$

find the decision boundry

Solution

$$z = \ln \frac{h}{1-h}$$

$$z\Big|_{threshold} = \ln \frac{0.5}{1-0.5} = \ln 1 = 0$$

$$z\Big|_{threshold} = 0$$

$$z = 1 + 2x_1 + 3x_2$$

$1 + 2x_1 + 3x_2 = 0 \leftarrow$ Decision boundry

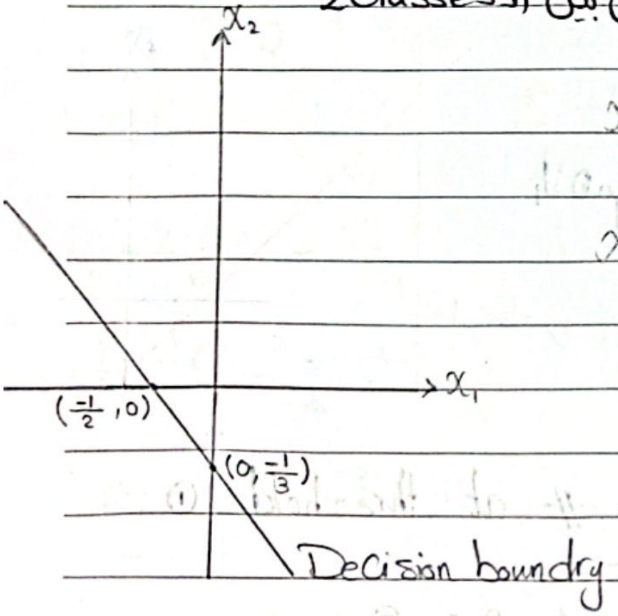
هي دي المعادلة الي هتفصل بين ال 2Classes

$x_1 = 0 \rightarrow 1 + 3x_2 = 0$

$x_1 = 0, x_2 = \frac{-1}{3} \rightarrow (0, \frac{-1}{3})$

$x_2 = 0 \rightarrow 1 + 2x_1 = 0$

$x_2 = 0, x_1 = \frac{-1}{2} \rightarrow (\frac{-1}{2}, 0)$



التاريخ: / /   الموضوع:

Ex: Assume the model is trained by logistic Regression

$\theta_0 = 1$, $\theta_1 = -4$, $\theta_2 = 3$ and threshold = 0.7

find the decision boundry

Solution

$$z = \ln \frac{h}{1-h}$$

$$z\Big|_{\text{threshold}} = \ln \frac{0.7}{1-0.7} = 0.8$$

$$z\Big|_{\text{threshold}} = 0.8 \quad , \quad z = 1 - 4x_1 + 3x_2$$

$$1 - 4x_1 + 3x_2 = 0.8 \rightarrow \text{Decision boundry}$$

Proof:

$$h = \frac{1}{1+e^{-z}}$$

$$(1+e^{-z})h = 1$$

$$1+e^{-z} = \frac{1}{h}$$

$$e^{-z} = \frac{1}{h} - 1 = \frac{1}{h} - \frac{h}{h}$$

$$e^{-z} = \frac{1-h}{h}$$ Take ln to both sides

$$-z \cancel{\ln e} = \ln \frac{1-h}{h}$$

$$z = -\ln \frac{1-h}{h} \rightsquigarrow z = \ln \frac{h}{1-h}$$

$\ln e = 1$

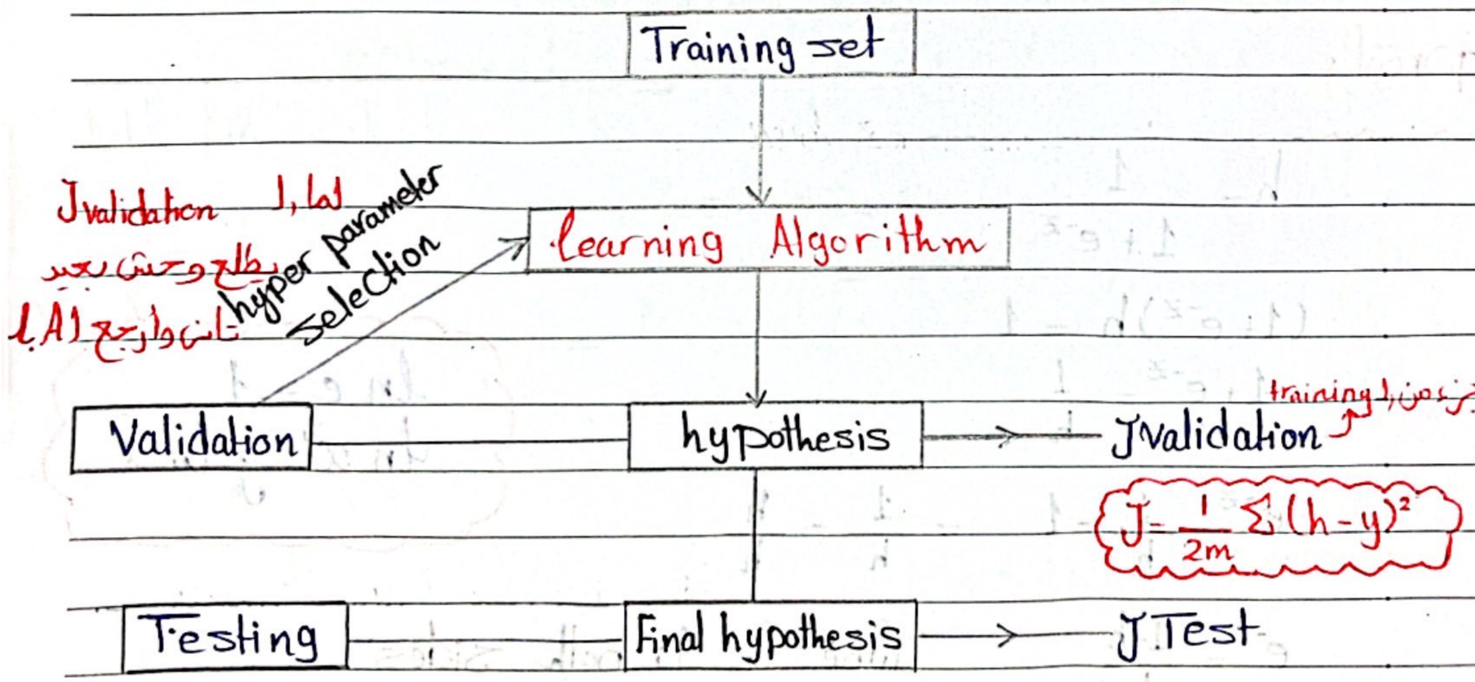
$\ln x^y = y \ln x$

التاريخ: / /   الموضوع:

* Model Evaluation:

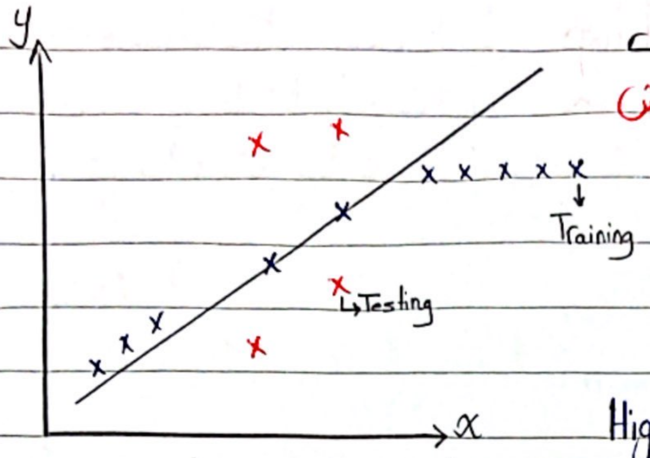
Dataset ⇝ 3 parts
- → Training set (70 %)
- → Validation set (15 %)
- → Testing set (15 %)

ex → $m = 1000$
- → Training → 700
- → Validation → 150
- → Testing → 150



$$\frac{1}{2m} \sum (h-y)^2 = \begin{cases} J_{Training} & ① \\ J_{validation} & ② \\ J_{Testing} & ③ \end{cases}$$

عندنا 3 J هنا ؟

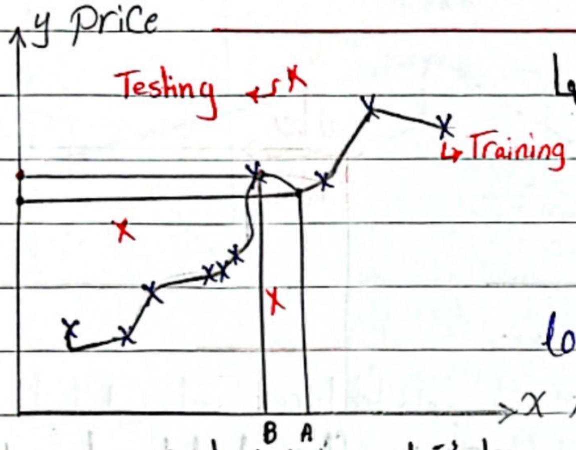


لو أنا صممت اشتغل linear Regression وأجيب خط مستقيم هيطلع حلو ولا وحش؟ وحش

$\underbrace{J_{Training}\uparrow \; , \; J_{Testing}\uparrow}_{\text{underfitting}}$

High Training Error↑ ↔ High Testing Error↑

أحلها ازاي؟ اقعد زود parameter لحد ما تجيب شكل الـ Curve وتظبط



عمل تعرجات وعدى بالـ Training data كلها فأصبح

$\underbrace{J_{Training} = 0 \; , \; J_{Testing}\uparrow}_{\text{overfitting}}$

low Training Error=0 ↔ High Testing Error↑

هنلاحظ كمان حاجات مش منطقية في الرسمة ← لو لاحظت A الـ Area↑ من B لكنه سعرها أقل ودة طبعا مش صح

Exam question: How to Know that your model has the overfitting issue?

Ans: 1- if the Training Error is low and the Testing Error is high

الـ underfitting حلها سهل هنزود الـ parameters لحد ما الدنيا تظبط
لكن الـ overfitting أصعب وحوارها كبير

Model Evaluation.

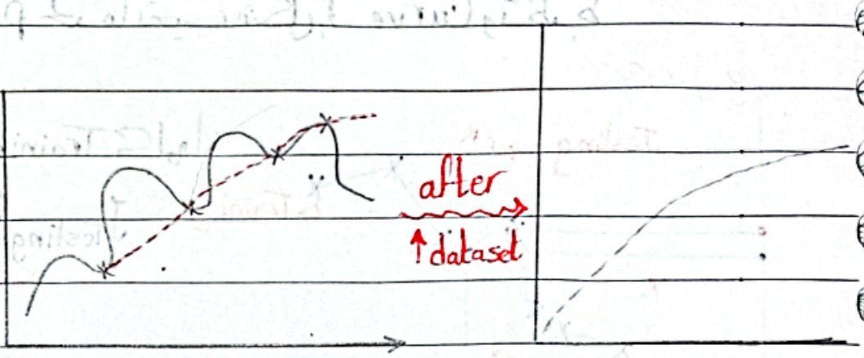
Q) How to over Come over fitting: حسب المشكلة هننقي الحل

الحلول دي بالترتيب ابدأ بالأول وشوف
لو اتحلت خلاص لو متحلتش روح اللي بعدها

1) Increase dataset
2) feature selection
3) Regularization
4) Cross Validation

[1] Increase dataset:

هنا كان في فراغات بين النقط
لما زودنا الـ dataset الفراغات
اتملت فالـ Curve بقى أحسن من التعرجات
دي.



[2] feature selection:

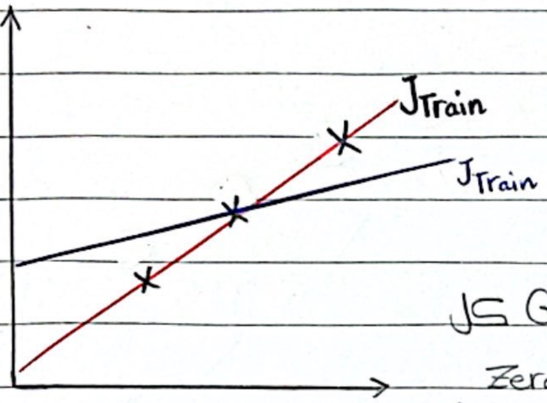
معناها إن باخد features وأسيب features
قليلة ← insufficient dataset ; features كتير
هبدأ أشوف ايه الحاجات المهمة وأخدها وأجيب
لما ماتحسن يعني اقعد أدرس كل x وشوف
مين مهم (مساحة الشقة، عدد الأوضة ...) ✓
لكن مثلاً عدد الشبابيك مهم؟ لأ فادي x skip

x_1	x_2	x_3	...	x_{100}	y

التاريخ: / /　　　　　　　　　　　　　　　　　　　　　　　الموضوع:

3 Regularization:

مشكلة الـ overfitting كانت ايه؟ $J_{Train}=0$, J_{Test} ↑↑
مش الـ gredient diecent كل شوية بيقعد يقلل الـ J؟ يعني شغال صح بس دة أثر علينا
لإنه خلى الـ J_{Train} قليلة جدًا لكن الـ J_{Test} عالية وهنا بيجي دور الـ Regularization
← هيوسطن الدنيا شوية ← هيعلى J_{Train} ↑، ويقلل J_{Test} ↓



الـ overfitting بيمثله الخط الأحمر ليه؟ عشانه
الـ Data على الخط تبيكال لإنه زي ما قولنا
J_{Train} قليل جدًا فهو راح مغير ورسم الخط
الأزرق فإيه اللي حصل؟ زود الـ J_{Train}
طب ازاي ومعادلة الـ Gradient descent كل
هدفها أصلًا تخلي الـ Error أقل حاجة ← Zero
طب ايه الحل؟ هنغير في المعادلة دي شوية بقا

$$J=\frac{1}{2m}\sum(h-y)^2 + \boxed{}$$

regularization Parameter

لـه بيخلي الخط الأزرق أحسن لإنه هيخلي J_{Train} لها قيمة
والـ J_{Test} بتقل وهنكمله المرة الجاية ونحسبه.

الموضوع: L(6) التاريخ: / /

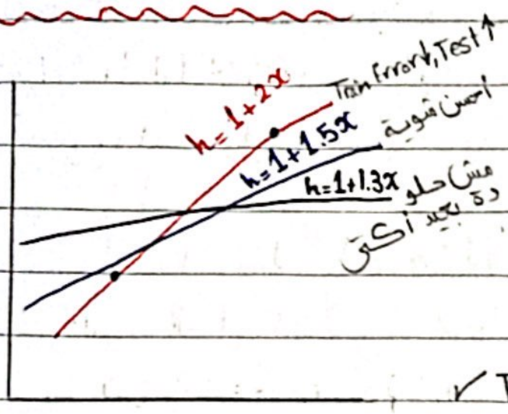
Overfittings

J_{Train} قليلة جداً , J_{Test} عالية

1- inCrease dataset
2- feature selection
3- Regularization
4- Cross validation

• Regularization:

ال Gradint descent بيسعى إنه يقلل J_{Train} أصلاً
ودة حلو فعلا بس ايه الوحش فيه؟ لو اديته data
غريبة هتظهر المشكلة هنا ← بايظ model
هنا احنا مش عايزين الخط يـ fit ال data 100%
← هنزود ال Train error شوية ونقلل Test error
* الى بالأحمر كويس جداً بالنسبة للـ Gradient descent
* الى بالأزرق هو الصح ليه؟ عشان قلل ال Testing Error ✓
هنغير في المعادلة بقا



Ex:

x	y	h
1	3	3
2	5	5

$h = 1+2x$

↳ $J = \frac{1}{2m}\sum (h-y)^2 = Zero$

كدة ال Gradient descent شغال صح وتمام بس
زى ما قولنا أى data غريبة هيطلع Error كبير
هنغير شوية في المعادلة بقا
ليه θ_1 بس؟ $h = \theta_0 + \theta_1 x$
احنا عايزين نعمل ايه؟ نميل الخط
فالمفروض أتحكم في مين؟ θ_1
لإنه θ_0 هي جزء مقطوع من
محور الصادات مش أكتر XX

$$\boxed{J = \frac{1}{2m}\sum (h-y)^2 + \underbrace{\frac{1}{2m}\lambda\theta_1^2}_{\text{Regularization parameter}}}$$

الموضوع: من الرسمة الي فاتت التاريخ: / /

at: $h = 1+2x$

x	y	h	$(h-y)^2$
1	3	3	0
2	5	5	0

$\boxed{0}$

$J = 0 + \frac{1}{4}(4)$

$J = 1$ # الخط الازرق في الرسمة

كويس للـ Gradient descent لكن فعليا وحش للـ Test

at: $h = 1 + 1.5x$

x	y	h	$(h-y)^2$
1	3	2.5	0.25
2	5	4	1

$\boxed{1.25}$

$J = \frac{1}{4}(1.25) + \frac{1}{4}(1.5)^2$

$J = 0.875$ الخط الأحمر

دة أحسنهم فعلا من حيث الـ Error

at: $h = 1 + 1.3x$

x	y	h	$(h-y)^2$
1	3	2.3	0.49
2	5	3.6	1.96

$\boxed{2.45}$

$J = \frac{1}{4}(2.5) + \frac{1}{4}(1.3)^2$

$J = 1.0351$ الخط الأسود

دة أسوء واحد لأنه زاد حتى عن الـ Error بتاع الـ Gradient descent

$$J = \frac{1}{2m}\sum(h-y)^2 + \frac{1}{2m}\lambda\theta_1^2$$

$\theta_0 = ?$

$\theta_1 = ?$

$$\theta_{0new} = \theta_{0old} - \frac{\alpha}{m}\sum(h-y)$$ هي هي كدة كدة متأثرتش

$$\theta_{new} = \theta_{old} - \alpha\frac{\partial J}{\partial \theta}$$

جابه بالتفاضل بالنسبة لـ θ_1

$$\theta_{1new} = \theta_{1old} - \alpha\left[\frac{2}{2m}\sum(h-y)x + \frac{2\lambda}{2m}\theta_1\right]$$

$$\theta_{1new} = \theta_1 - \frac{\alpha}{m}\sum(h-y)x - \frac{\alpha\lambda}{m}\theta_1$$

$$\theta_{1new} = \theta_1\left[1 - \frac{\alpha\lambda}{m}\right] - \frac{\alpha}{m}\sum(h-y)x$$

بيخلينا ناخد جزء من θ_1

لأنه هيكون كدة مضروب في θ_1

Regularization

الموضوع: ________ التاريخ: / /

If we have 2 features → x_1, x_2

$$J = \frac{1}{2m} \sum (h-y)^2 + \frac{\lambda}{2m} [\theta_1^2 + \theta_2^2]$$

* Overfitting in Classification:

- Training accuracy
- Testing accuracy

* logistic Regression:

كنا شغالين بالـ J وفي الكلاسيفكيشن بنشتغل بالـ accuracy

x	y	h
	1	1 ✓
	0	0 ✓
	0	1 x
	0	0 ✓
	1	0 x

h → after threshold

$$accuracy = \frac{3}{5} * 100$$

3 → الحاجات الصح (هي هي الـ h = y)

5 → عددهم

• لو أنا شغال Regression قولنا ازاي أعرف overfitting؟
← $J_{Training}$ قليلة جداً ، $J_{Testing}$ عالية ← في حالة بنتعامل مع J

• طب بالـ accuracy أعرف ازاي؟ ← Overfitting
← Training accuracy ↑ ← عالية يعني كويسة
← Testing accuracy ↓ ← قليلة ← لما دخلت data غريبة

underfitting: ← Training accuracy ↓ , Testing accuracy ↓ الاتنين قليلين

التاريخ: / /　　　　　　　　　　الموضوع:

Regularization in Case of logistic Regressions:

معادلة ال J هتبقى زي ما هي لأن بحسب Cost

$$J = \frac{\sum Cost}{m} + \underbrace{\frac{\lambda}{2m}}_{\text{Regularization parameter}} \theta_1^2$$

$$\theta_0 = \theta_0 - \frac{\alpha}{m} \sum (h - y)$$

$$\theta_1 = \theta_1 \left[1 - \frac{\alpha \lambda}{m}\right] - \frac{\alpha}{m} \sum (h - y) x$$

وخلاص كده كده نفس المعادلات الاختلاف الوحيد هيكون حسبة ال h بس لأن حسبتها بتختلف من ال linear لل logistic

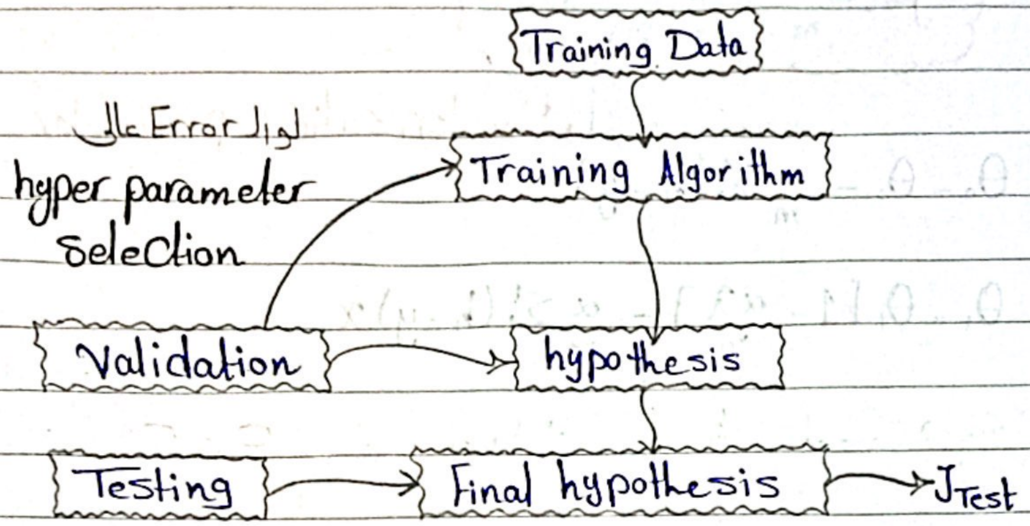
$$h = \frac{1}{1 + e^{-z}} \leftarrow \quad \hookrightarrow h = \theta_0 + \theta_1 x$$

الموضوع: ______ التاريخ: / /

[4] Cross Validation:

اخدنا المرة اللي فاتت الـ evalution model

وقسمنا الـ data ← Training , Validation , Testing



هنا في مشكلة لو الـ data قليلة ← الـ Validation عمال يظبطه وبتاع بعدين جينا الـ Test لقيناه باظ هنا هنروح لـ

→ Cross Validation

• Cross Validation K fold:

↳ parameter

معظم الـ projects بتحدد الـ K ← $5 \leq K \leq 10$

Ex: $m = 1000$ (dataset rows), $K = 5$ → هرسم 5 عينات

$\frac{1000}{5} = 200$ ← هيتقسم 5 أجزاء ← كل جزء من الـ 5

كل جزء فيه 200 ←

Test	Train	Train	Train	Train
200	200	200	200	200
Train	Test	Train	Train	Train
200	200	200	200	200
Train	Train	Test	Train	Train
200	200	200	200	200
Train	Train	Train	Test	Train
200	200	200	200	200
Train	Train	Train	Train	Test
200	200	200	200	200

هقلب الـ Test على الـ data يعني أول مرة أول 200 row دول الـ Test والـ 800 Train ، تاني مرة تاني 200 row هم الـ Test وأول 200 والباقي ← Train وهكذا بالتالي أنت خليته يتدرب على كله ودي حاجة حلوة

الموضوع: التاريخ: / /

من الشكل اللي فات كل مستطيل هيطلع JTest
← كدة عندي كام JTest ؟ 5
كام accuracy ؟ 5

← هعمل الـ report بمين ؟ Average → JTest
accuracy

* الخطوات اللي عملتها دي ضمن الـ model evaluation بعد ما ظبطت الـ model
خلاص هتاخد الـ data كلها على بعضها تدربها على الـ learning Algorithm
لإنه كانت المشكلة إن الـ model كل شوية يعمل Validation يحاول يظبطه وفي
الآخر الـ Error كبير فاحنا هنظبطه في الطريقة اللي فاتت دي وبعدين أدخل
الـ data كلها في الآخر على الـ learning Algorithm كإنها Training data

Evalution Metric: metric مصطلح بيحدد حاجة حلوة ولا وحشة ⟵

I Regression:

$$MSE = J = \frac{1}{2m} \sum (h-y)^2$$

$$RMSE = \sqrt{J} = \sqrt{\frac{1}{2m} \sum (h-y)^2}$$ Root Mean Square Error

ex: Data training ⟶ $h = 1000 + 1000x$

Test data

x	y	h ($1000 + 1000x$)	$h-y$	$(h-y)^2$
2	2300	3000	700	490000
3	3400	4000	600	360000
4	5200	5000	-200	40000
5	5600	6000	400	160000
			Σ	1050000

$$MSE = \frac{1}{2m} \sum (h-y)^2 = \frac{1}{8} * 1050000 = 131250$$

$$RMSE = \sqrt{MSE} = \sqrt{131250} = 362.28$$

2 Classification: ⇝ accuracy, recall, precision, F1-score

Confusion matrix: binary Classification → 2 classes ($\underbrace{\text{yes} \quad \text{No}}$)

$$\text{Accuracy} = \frac{TP + TN}{TP+TN+FP+FN} * 100$$

$$\text{recall} = \frac{TP}{TP+FN} * 100$$

$$\text{precision} = \frac{TP}{TP+FP} * 100$$

$$\text{F1-score} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

		Actual: Yes	Actual: No
predicted	Yes	TP	FP
	No	FN	TN

positive ← P ← yes لو predicted ال

Negative ← N ← No لو predicted ال

بيتكونوا من خانتين

T / F　　P / N

↳ predicted

True ← yes مع yes لو

False ← no مع yes لو

True ← no مع no لو

• **Example:**

		Actual: Spam	Actual: Not spam
predicted	Spam	120 TP	20 FP
	Not-spam	10 FN	50 TN

• $\text{accuracy} = \frac{120+50}{120+50+10+20} * 100$

↳ 85 %

• $\text{recall} = \frac{120}{120+10} * 100 = 92.3\ \%$

• $\text{precision} = \frac{120}{120+20} * 100 = 85.7\ \%$

• $\text{F1-score} = \frac{2 * 85.7 * 92.3}{85.7 + 92.3} = 88.9\ \%$

ال recall مهتمية بال False Negative
يعني لو الدراسة بتاعتي معتمدة على ← بستخدم recall
الناس الي قولت إنهم Negative وإجابتي غلط
يعني لو بدرس مرض المهم بالنسبة لي recall
ليه؟ لإن كارثة لو أنا قولت معندهوش المرض وهو عنده فمركز أعلى recall ← FN↓ تقل

$$\text{recall} = \frac{TP}{TP + FN}$$

ال precision مهتمية بال
أنا بنك وجه حد قال عايزك تعلى model
يقولي الشخص ده هيقدر يسدد القرض ولا لا قبل ما أديه
فالموديل بيقول positive يعني هيدفع بس False يعني إجابتي غلط
التركيز على إني أعلى ال precision ← FP↓ تقل

$$\text{precision} = \frac{TP}{TP + FP}$$

مهتم بالاتنين FP & FN

accuracy ← balanced Data

F1-Score ← imbalanced Data

يعني dataset فيها
→ 8 yes, 2 no
كده غير متوازنة

K-nearest neighbour (KNN)

الجار الأقرب ← K ← hyper parameter

↳ Classification
↳ Regression

$(x_1, y_1) \longrightarrow (x_2, y_2)$

$$D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

• ده مش Algorithm بيعمل training

• Classification :

① Find distance
② Find rank ⇝ ascending order
③ output according to K (majority)

• Regression :

① Find distance
② Find rank
③ output according to K (average)

Classification Example:

$\sqrt{(167-170)^2+(51-57)^2} \leftarrow$

Distance	Rank	x_1	x_2	Class
6.7	5	167	51	UW
13	8	182	62	N
13.4	9	176	69	N
7.6	6	173	64	N
8.2	7	172	65	N
4.1	4	174	56	UW
1.4	1	169	58	N
3	3	173	57	N
2	2	170	55	N

UW: under-weight
N: Normal
find the output when:
$x_1 = 170$
$x_2 = 57$
K = 3 ← هاخد أقرب ③

Steps:

① Distance
② Rank → رتب تصاعدي حسب الأقرب
③ K = 3 → 3 neighbour خد أقرب
↳ 1.4, 2, 3 ⟹ Normal
↳ output اعتمادًا على الـ neighbours

if K = 5 → $\underbrace{1,2,3}_{\text{Normal}}$, $\underbrace{4,5}_{\text{UW}}$

الـ output هيتحدد على أساس الأغلبية
↳ predicted output : Normal

Regression Examples:

x_1	x_2	y	Distance	Rank
7	7	6	$\sqrt{(7-2)^2+(7-1)^2} = 7.81$	4
7	4	3	$\sqrt{(7-2)^2+(4-1)^2} = 5.83$	3
3	4	2	$\sqrt{(3-2)^2+(4-1)^2} = 3.16$	2
1	3	1	$\sqrt{(1-2)^2+(3-1)^2} = 2.23$	1

find the output when:

$x_1 = 2$

$x_2 = 1$

$K = 3$

Solution:

predicted output $= \frac{3+2+1}{3} = 2$

↳ according to K

↳ average → y

↳ Note ← ال regression بيعتمد على ال output

ما ازاي اختار K؟

1. $K = \sqrt{N}$
   ↳ Num of training example

2. Cross validation

To get K with best performance

(accuracy ↑, error ↓)

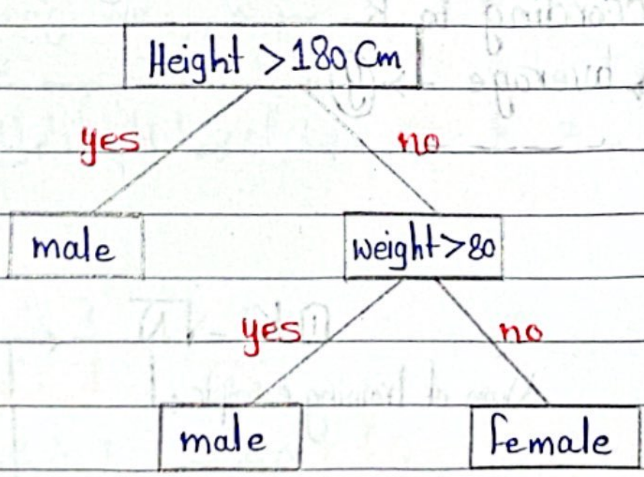
Decision Tree:

- predictive model
- used for:
  - ↳ Regression (CART Algorithm)
  - ↳ Classification (ID3 Algorithm)

Ex:

يتحول ال dataset إلى Decision Tree

weight	height	class output
		Male Female



Height = 190	Height = 175	Height = 150
weight = 85	weight = 90	weight = 60
↝ Male	↝ Male	↝ Female

* Feature selection:

المثال اللي فات عندي weight و height بس في الـTree بدأنا بـ Height طب ليه؟

XX random — X مش عشوائيـة

Measured by:
- ↳ Information Gain ✓
- ↳ Gini index

· Entropy: impurity في شوائب — عكسها X pure

$0 \leq E \leq 1$

height	weight	o/p

قسم على أساس weight

Height	weight	Class
		female
		female
		female
		female

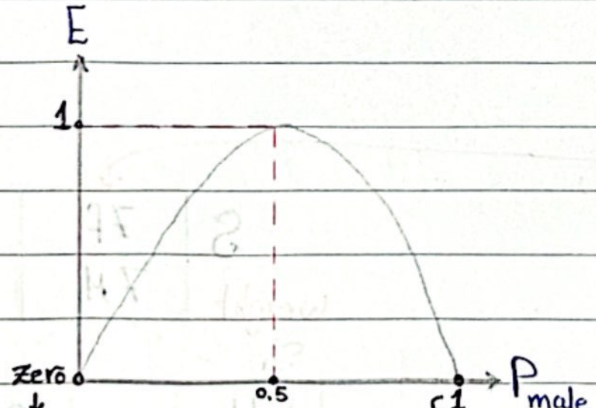
$E=0$

كويسة و E=0 يعني خالية من الشوائب لإنها طلعتلي Class واحد بس female

قسم على أساس height

Height	weight	Class
		female
		male
		male
		female

$E=1$



$P_{male}=0$, $P_{female}=1$

$P_{male}=0.5$, $P_{female}=0.5$

$P_{male}=1$, $P_{female}=0$

أسوء حاجة لإنه مش عارف هو male ولا female

مفيش حاجة هتطلع E=0 بس عايز أقل Entropy فبحسبه للـfeature

$$\boxed{E = -\sum P \log_2 P}$$

P: probability

height	weight	Class
		Female
		male
		female
		female

$\rightarrow p(\text{male}) = \frac{1}{4}$, $p(\text{female}) = \frac{3}{4}$

$$\underset{\text{dataset}}{E(S)} = -\left(\frac{1}{4}\log_2\frac{1}{4} + \frac{3}{4}\log_2\frac{3}{4}\right)$$

Information Gain (IG):

measures reduction in Entropy ($E\downarrow$)

IG $\uparrow$ → best

height	weight	output

S: 7 Female, 7 Male ← الداتا اللي ال output فيها

هنقسمها على أساس height

- S_1: 4M, 4F
- S_2: 3M, 3F

S: 7F, 7M

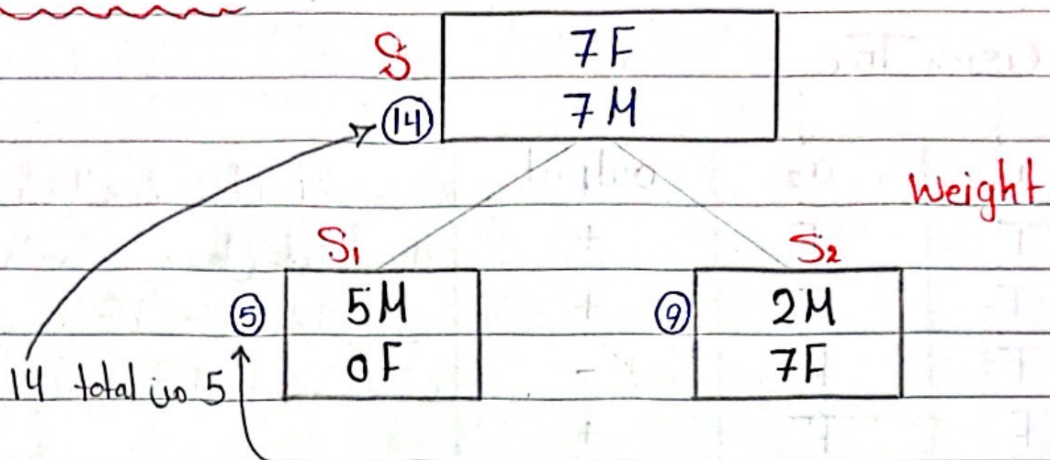
weight

- S_1: 5M, 0F
- S_2: 2M, 7F

- No a best split في ال set يعني مفيش
- IG $\downarrow$ ← الناتج class واحد بس
  يعني في شوائب $E\uparrow$

- Entropy reduced
- IG $\uparrow$

IG Calculation هحسبه للتقسيمة الأحسن ع



$$IG(S, weight) = E(S) - \frac{5}{14}E(S_1) - \frac{9}{14}E(S_2)$$

$$E(S) = -\left[\frac{7}{14}\log_2\frac{7}{14} + \frac{7}{14}\log_2\frac{7}{14}\right] = 1$$

$$E(S_1) = -\left[\frac{5}{5}\log_2\frac{5}{5} + zero\right] = 0$$

$$E(S_2) = -\left[\frac{2}{9}\log_2\frac{2}{9} + \frac{7}{9}\log_2\frac{7}{9}\right] = 0.764$$

$$IG = 1 - \frac{5}{14}(0) - \frac{9}{14}(0.764) = 0.508$$

الموضوع: ______ التاريخ: / /

Example:

Draw decision Tree

instance	a_1	a_2	output
1	T	T	+
2	T	T	+
3	T	f	−
4	f	f	+
5	f	T	−
6	f	T	−

هحل المسألة على 2 parts
أول واحد إني هشتغل على أساس a_1
وأحسب IG بعدين على أساس
a_2 وأحسب IG وأشوف أنهى
أحسن

Solution

- a_1 → T, F
- a_2 → T, F
- output → +, −

a_1 هقسم على أساسها

الداتا ست الأصلية

S (6): 3 +, 3 −

T بتوع a_1
هشوف T في a_1 معاها كام +, −

F في a_1 معاها كام +, − ؟

S_1 (3): 2 +, 1 −

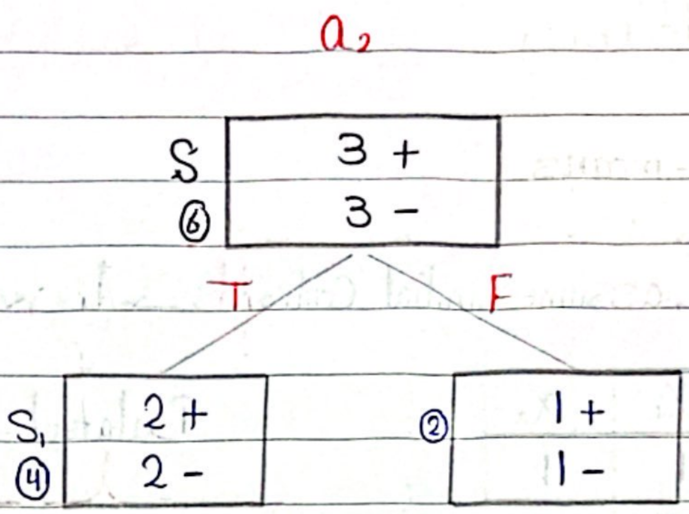
S_2 (3): 1 +, 2 −

$$E(S) = -\left[\frac{3}{6}\log_2\frac{3}{6} + \frac{3}{6}\log_2\frac{3}{6}\right] = 1$$

$$E(S_1) = -\left[\frac{2}{3}\log_2\frac{2}{3} + \frac{1}{3}\log_2\frac{1}{3}\right] = 0.9183$$

$$E(S_2) = -\left[\frac{1}{3}\log_2\frac{1}{3} + \frac{2}{3}\log_2\frac{2}{3}\right] = 0.9183$$

$$IG(S, a_1) = 1 - \frac{3}{6} * 0.9183 - \frac{3}{6} * 0.9183 = 0.0817$$



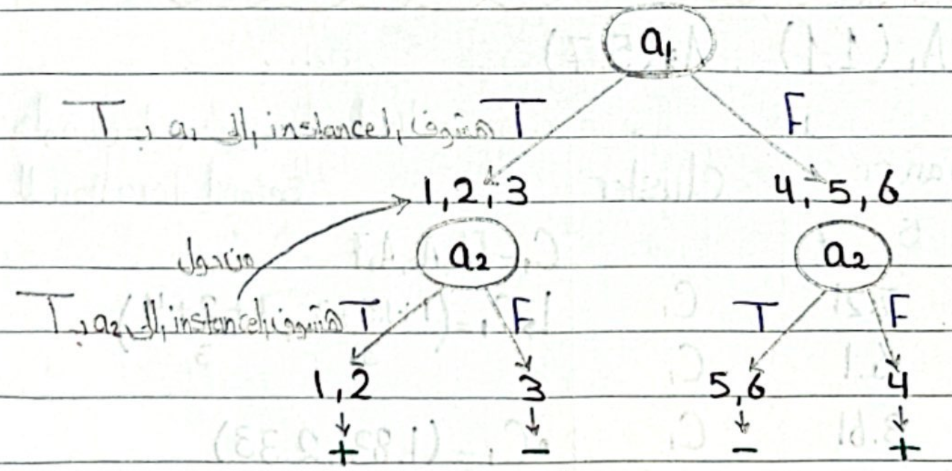
$E(S) = 1$

$E(S_1) = -\left[\frac{2}{4}\log_2\frac{2}{4} + \frac{2}{4}\log_2\frac{2}{4}\right] = 1$

$E(S_2) = -\left[\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\log_2\frac{1}{2}\right] = 1$

- $IG(S, a_2) = 1 - \frac{4}{6} * 1 - \frac{2}{6} * 1 = 0$

Decision Tree: $IG(S, a_1) > IG(S, a_2) \rightarrow a_1$ the best كبداية



Ex: Draw Decision Tree:

عندي كل feature 3 S يعني هحل 3 تقسيمات $\{a_1, a_2, a_3\}$ كل واحدة ليها IG خاص

instance	a_1	a_2	a_3	Classification output
1	True	Hot	High	No
2	True	Hot	High	No
3	false	Hot	High	Yes
4	false	Cool	Normal	Yes
5	false	Cool	Normal	Yes
6	true	Cool	High	No
7	true	Hot	High	No
8	true	Hot	Normal	Yes
9	false	Cool	Normal	Yes
10	false	Cool	High	Yes

Solution

$a_1 \{\text{true}, \text{false}\}$, $a_2 \{\text{Hot}, \text{Cool}\}$, $a_3 \{\text{High}, \text{Normal}\}$, o/p $\{\text{yes}, \text{no}\}$

a_1

S: 6 yes, 4 no

S_1 (True): 1 yes, 4 no

S_2 (False): 5 yes, 0 no

$$E(S) = -[\frac{6}{10}\log_2\frac{6}{10} + \frac{4}{10}\log_2\frac{4}{10}] = 0.971$$

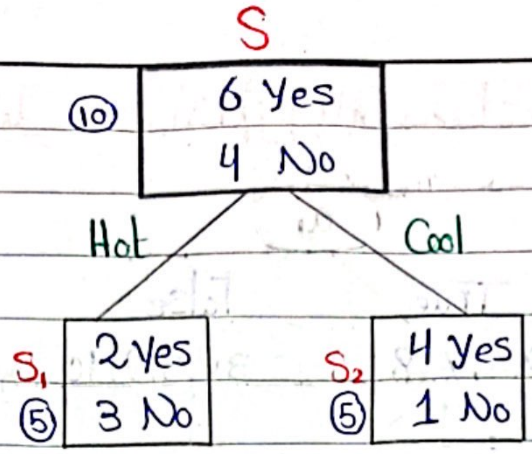
$$E(S_1) = -[\frac{1}{5}\log_2\frac{1}{5} + \frac{4}{5}\log_2\frac{4}{5}] = 0.7219$$

$$E(S_2) = -[\frac{5}{5}\log_2\frac{5}{5} + 0] = 0$$

$$IG(S, a_1) = 0.971 - \frac{5}{10} * 0.7219 - \frac{5}{10} * 0 = 0.61$$

التاريخ: / / الموضوع:

a_2



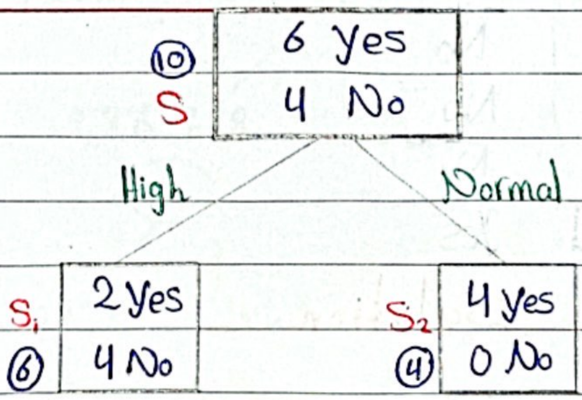
$E(S) = 0.971$ حسبناها فى a_1 خلاص

$$E(S_1) = -\left(\frac{2}{5}\log_2\frac{2}{5} + \frac{3}{5}\log_2\frac{3}{5}\right) = 0.971$$

$$E(S_2) = -\left(\frac{4}{5}\log_2\frac{4}{5} + \frac{1}{5}\log_2\frac{1}{5}\right) = 0.7219$$

$$IG(S,a_2) = 0.971 - \frac{5}{10} * 0.971 - \frac{5}{10} * 0.7219 = 0.1246$$

a_3



$E(S) = 0.971$

$$E(S_1) = -\left(\frac{2}{6}\log_2\frac{2}{6} + \frac{4}{6}\log_2\frac{4}{6}\right) = 0.913$$

$$E(S_2) = -\left(\frac{4}{4}\log_2\frac{4}{4} + 0\right) = 0$$

$$IG(S,a_3) = 0.971 - \frac{6}{10} * 0.913 - \frac{4}{10} * 0 = 0.42$$

الموضوع:

التاريخ: / /

$IG(S,a_1)$ هو الأعلى يعني أقل Entropy ← The best هبدأ ب a_1

a_1

True: 1,2,6,7,8 هنا متنوع الـ output بين Y/N
هعمل ايه بقا؟

False: 3,4,5,9,10 → output واحد بس → Yes

الـ rows بتاعة 1,2,6,7,8 هاخدها
⊆ الـ features subset ← a_2, a_3
طب ليه ماخدتش a_1؟ لأن خلاص بنيت بيها
كالأفضل فلو أخدتها معاهم هتدخل loop

Subsets:

instance	a_2	a_3	O/P
1	Hot	High	No
2	Hot	High	No
6	Cool	High	No
7	Hot	High	No
8	Hot	Normal	Yes

هعتبرها S ← dataset
وهحسب لـ a_2, a_3 ← IG وأشوف
أنهي أحسن أكمل به بعد a_1

Solution

a_2

S (5): 1 yes, 4 No

Hot → S_1 (4): 1 yes, 3 No

Cool → S_2 (1): 0 yes, 1 No

$E(S) = 0.7219$ ← بالنسبة للـ dataset الجديدة هنا

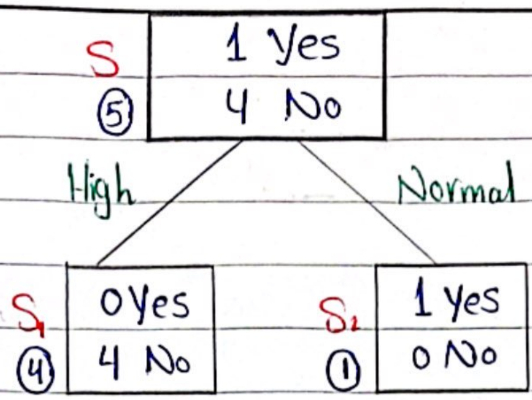
$E(S_1) = 0.8113$

$E(S_2) = 0$

$IG(S, a_2) = 0.07286$

التاريخ: / /   الموضوع: ________

a_3



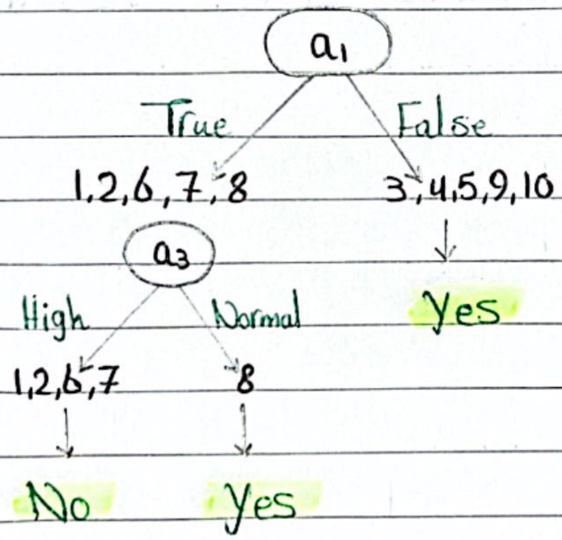
$E(S) = 0.7219$

$E(S_1) = 0$

$E(S_2) = 0$

$IG(S, a_3) = 0.7219$

✓ $IG(S,a_3)$ هو الأعلى ← يعني أقل Entropy ← The best هكمل بـ a_3 بعد a_1



* كمعلومة لو مثلا بعد ما قسمت بناء على a_3 وطلع برده الoutput فيه احتمالين وقسمت على a_2 طلع برده output Y/N مفيش حاجة واضحة هتعمل ايه؟ هتاخد الoutput بناء على الـ majority ← الأغلبية يعني

unsupervised learning
↳ Clustering
↳ K-means

Ex: $K=2$, assume initial Centroids → A_1, A_4

instance	x_1	x_2
A_1	1	1
A_2	1.5	2
A_3	3	4
A_4	5	7
A_5	3.5	5
A_6	4.5	5
A_7	3.5	4.5

Dataset

Cluster 1 (1,1) | Cluster 2 (5,7)

كمثال (3,8) عايز نشوف تكون مع أنهي.

→ Euclidean Distance:

$$D = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

بيستخدموا عشان أجيب ال distance بين نقطتين

Solution

First iteration: $A_1(1,1)$, $A_4(5,7)$

Dataset			Distance		Cluster
			(1, 1)	(5, 7)	
A_1	1	1	0	7.21	C_1
A_2	1.5	2	1.12	6.1	C_1
A_3	3	4	3.61	3.61	C_1
A_4	5	7	7.21	0	C_2
A_5	3.5	5	4.72	2.5	C_2
A_6	4.5	5	5.31	2.06	C_2
A_7	3.5	4.5	4.3	2.92	C_2

داوقتي عايز أجيب ال new Centroids لل second iteration

$C_1 = \{A_1, A_2, A_3\}$

$\hookrightarrow C_1 = \left(\frac{1+1.5+3}{3}, \frac{1+2+4}{3}\right)$

• $C_1 = (1.83, 2.33)$

$C_2 = \{A_4, A_5, A_6, A_7\}$

$\hookrightarrow C_2 = \left(\frac{5+3.5+4.5+3.5}{4}, \frac{7+5+5+4.5}{4}\right)$

• $C_2 = (4.12, 5.38)$

↳ Second iteration: $C_1(1.83, 2.33)$, $C_2(4.12, 5.38)$

الموضوع: ________ التاريخ: / /

الunsupervised يعني أنا معايا data من غير label بس عايزة أقسمها

لـ groups فبستخدم ← K-Means

عدد الgroups

هو مدينى dataset كل instance بيكون فى شكل زوج مرتب

$A_1\ (x_{A_1}, y_{A_1})$
$A_2\ (x_{A_2}, y_{A_2})$
⋮

هو بيحددلك هيدأ بمين كـ Clusters من الdataset دى

فهو فى المثال دة حددلى ← $C_1: A_1(1,1)$, $C_2: A_4(5,7)$

فهنعمل ايه بقا؟ جدول بكل instance عادى زى الـ dataset

هعمل Column لـ A_1 و Column لـ A_4 وأجيب المسافة بين

كل instance مع كل Centroid من الى متحددين

$$D = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

هجيب الـ distance بين A_1 وكل الinstance $(A_1 : A_7)$

هجيب الـ distance بين A_4 وكل الinstance $(A_1 : A_7)$

بعد ما جبت الـ distance هشوف بقا المسافة بين A_1 و A_2 أصغر؟

ولا بين A_4 و A_2؟ ← بين A_1 و $A_2 = 1.12$ ← A_2 هتكون تبع C_1

وهكذا هقسم النقط بناء على الـ distance لـ C_1, C_2

بعد الـ 1^{st} iteration طلع ← $C_1 = \{A_1, A_2, A_3\}$, $C_2 = \{A_4, A_5, A_6, A_7\}$

عايزة أجيب new centroids

الـ x بتاعتهم الـ y بتاعتهم

$C_1 = \left(\frac{1+1.5+3}{3_{\text{عددهم}}}, \frac{1+2+4}{3_{\text{عددهم}}}\right)$

x y

$, C_2 = \left(\frac{5+3.5+4.5+3.5}{4}, \frac{7+5+5+4.5}{4}\right)$

الموضوع: L (10) التاريخ: / /

*Support Vector Machines (SVM)

- Classification
- Regression.

Notes:

$$\begin{bmatrix}2\\3\end{bmatrix} \underset{\text{dot product}}{\cdot} \begin{bmatrix}5\\4\end{bmatrix} = \begin{bmatrix}10\\12\end{bmatrix}$$

ضرب كل رقم في المناظر له .

$$\begin{bmatrix}2\\5\end{bmatrix} + \begin{bmatrix}5\\4\end{bmatrix} = \begin{bmatrix}7\\9\end{bmatrix}$$

- $2x_1 + 3x_2 + 5 = 0$ Write in vector form

$$\begin{bmatrix}2\\3\end{bmatrix} \cdot \begin{bmatrix}x_1\\x_2\end{bmatrix} + 5 = 0$$

- $w_1 x_1 + w_2 x_2 + b = 0$

$$\begin{bmatrix}w_1\\w_2\end{bmatrix} \cdot \begin{bmatrix}x_1\\x_2\end{bmatrix} + b = 0$$

Dot product

$$\underbrace{\vec{w}}_{\text{W vector}} \cdot \underbrace{\vec{x}}_{x \text{ vector}} + b = 0$$

الموضوع: ____ التاريخ: / /

Case Study:

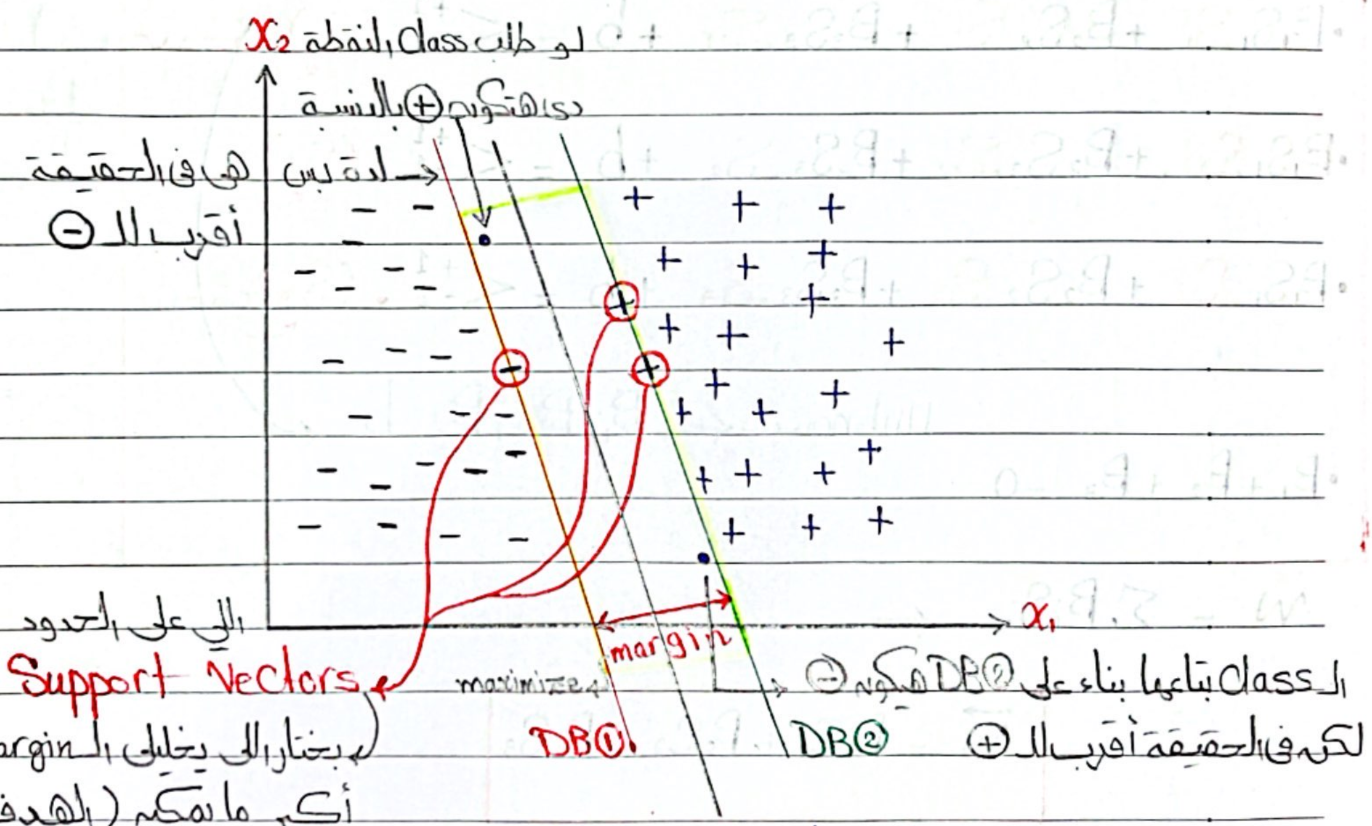
Binary Classification $y \in \{+1, -1\}$

↳ 2 Features

x_1	x_2	y
		+1
		-1
		⋮
		⋮

فايدة الـ SVM إنه بيطلع أحسن Decision boundry أفصل بيها بين الـ 2 Classes الي عندي

-1 ⟷ +1



Best Decision Boundry

(Hyper plane)

$$\vec{w} \cdot \vec{x} + b = 0$$

?↲ ?↲

- Given 3 support vectors — حسب عدد الـ Vectors هكون معادلات
  - S_1
  - S_2
  - S_3

→ Find the hyper plane equation

Solution

$$\underbrace{\vec{w}}_{?} \cdot \vec{x} + \underbrace{b}_{?} = 0$$

المعادلات الى هشتغل بيها

- $\beta_1 S_1 \cdot S_1 + \beta_2 S_2 \cdot S_1 + \beta_3 S_3 \cdot S_1 + b = \left\{ \begin{matrix} +1 \\ -1 \end{matrix} \right.$ حدد حسب (S_1)
- $\beta_1 S_1 \cdot S_2 + \beta_2 S_2 \cdot S_2 + \beta_3 S_3 \cdot S_2 + b = \left\{ \begin{matrix} +1 \\ -1 \end{matrix} \right.$ حدد حسب (S_2)
- $\beta_1 S_1 \cdot S_3 + \beta_2 S_2 \cdot S_3 + \beta_3 S_3 \cdot S_3 + b = \left\{ \begin{matrix} +1 \\ -1 \end{matrix} \right.$ حدد حسب (S_3)

Unknowns ← $\beta_1, \beta_2, \beta_3, b$

- $\beta_1 + \beta_2 + \beta_3 = 0$

$W = \sum \beta_i S_i$ ⇐

$$\boxed{\vec{W} = \beta_1 S_1 + \beta_2 S_2 + \beta_3 S_3}$$

∴ hyper plane equation :=

$$\vec{W} \cdot \vec{x} + b = 0$$

Example: Assume: $S_1 = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$, $S_2 = \begin{bmatrix} 2 \\ -1 \end{bmatrix}$ $\Longrightarrow$ +ve Class

$S_3 = \begin{bmatrix} 4 \\ 0 \end{bmatrix}$ $\Longrightarrow$ -ve Class

Find hyperplane equation.

Solution

$\beta_1 S_1 S_1 + \beta_2 S_2 S_1 + \beta_3 S_3 S_1 + b = 1$

$\beta_1 S_1 S_2 + \beta_2 S_2 S_2 + \beta_3 S_3 S_2 + b = 1$

$\beta_1 S_1 S_3 + \beta_2 S_2 S_3 + \beta_3 S_3 S_3 + b = -1$

$\beta_1 + \beta_2 + \beta_3 = 0 \Longrightarrow$ هنجيب معادلة متغير بدلالة الباقي

عوض في ال 3 معادلات بقا بعد كده

$S_1 . S_1 = \begin{bmatrix} 4 \\ 1 \end{bmatrix} \xrightarrow{\text{اجمع}} 4 + 1 = 5$

$S_1 . S_2 = \begin{bmatrix} 4 \\ -1 \end{bmatrix} \xrightarrow{\text{اجمع}} 4 - 1 = 3$

$S_1 . S_3 = 8 + 0 = 8$

$S_2 . S_2 = 4 + 1 = 5$

$S_2 . S_3 = 8 + 0 = 8$

$S_3 . S_3 = 16 + 0 = 16$

$5\beta_1 + 3\beta_2 + 8\beta_3 + b = 1$

$3\beta_1 + 5\beta_2 + 8\beta_3 + b = 1$

$8\beta_1 + 8\beta_2 + 16\beta_3 + b = -1$

$\beta_1 = -\beta_2 - \beta_3 \longrightarrow$ من رابع معادلة فوق، هناخدها ونعوض في ال 3 معادلات فوقه

$5(-\beta_2 - \beta_3) + 3\beta_2 + 8\beta_3 + b = 1 \Longrightarrow -2\beta_2 + 3\beta_3 + b = 1$

$3(-\beta_2 - \beta_3) + 5\beta_2 + 8\beta_3 + b = 1 \Longrightarrow 2\beta_2 + 5\beta_3 + b = 1$

$8(-\beta_2 - \beta_3) + 8\beta_2 + 16\beta_3 + b = -1 \Longrightarrow 0\beta_2 + 8\beta_3 + b = -1$

$\beta_3 = -0.5 \ / \ \beta_2 = 0.25 \ / \ \beta_1 = -0.25 + 0.5 = 0.25$

$b = 3$

$(0.25 * 2) + (0.25 * 2) - \frac{1}{2} * 4$

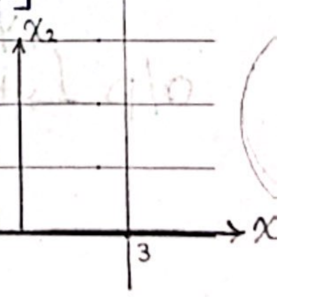
$* \ \vec{W} = \beta_1 S_1 + \beta_2 S_2 + \beta_3 S_3 = 0.25 \begin{bmatrix} 2 \\ 1 \end{bmatrix} + 0.25 \begin{bmatrix} 2 \\ -1 \end{bmatrix} - 0.5 \begin{bmatrix} 4 \\ 0 \end{bmatrix} = \begin{bmatrix} -1 \\ 0 \end{bmatrix}$

hyperplane equation $\vec{W} . \vec{x} + b = 0$

$\begin{bmatrix} -1 \\ 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + 3 = 0 \longrightarrow -x_1 + 3 = 0$

$x_1 - 3 = 0$

$\boxed{x_1 = 3}$



Naive Bayes: هو Algorithm مبني على Conditional probability

Day	outlook	Temprelure	Humidity	Wind	play Tennis
D_1	Sunny	Hot	High	Weak	No
D_2	Sunny	Hot	High	Strong	No
D_3	OverCast	Hot	High	Weak	Yes
D_4	Rain	Mild	High	Weak	Yes
D_5	Rain	Cool	Normal	Weak	Yes
D_6	Rain	Cool	Normal	Strong	No
D_7	OverCast	Cool	Normal	Strong	Yes
D_8	Sunny	Mild	High	Weak	No
D_9	Sunny	Cool	Normal	Weak	Yes
D_{10}	Rain	Mild	Normal	Weak	Yes
D_{11}	Sunny	Mild	Normal	Strong	Yes
D_{12}	OverCast	Mild	High	Strong	Yes
D_{13}	OverCast	Hot	Normal	Weak	Yes
D_{14}	Rain	Mild	High	Strong	No

Test : New instance

outlook = Sunny
Temperature = Cool
Humidity = High
wind = Strong

Solution

o/p → Yes (9) → $p(Yes) = \frac{9}{14}$

o/p → No (5) → $p(No) = \frac{5}{14}$

الموضوع: ______ التاريخ: / /

Outlook	Yes	No
Sunny	$\frac{2}{9}$	$\frac{3}{5}$
overCast	$\frac{4}{9}$	$\frac{0}{5}$
rain	$\frac{3}{9}$	$\frac{2}{5}$

Temprature	Yes	No
Hot	$\frac{2}{9}$	$\frac{2}{5}$
Mild	$\frac{4}{9}$	$\frac{2}{5}$
Cool	$\frac{3}{9}$	$\frac{1}{5}$

Humidity	Yes	No
High	$\frac{3}{9}$	$\frac{4}{5}$
Normal	$\frac{6}{9}$	$\frac{1}{5}$

Wind	Yes	No
Weak	$\frac{6}{9}$	$\frac{2}{5}$
Strong	$\frac{3}{9}$	$\frac{3}{5}$

New instance : outlook = Sunny
Temprature = Cool
Humidity = High
wind = strong

yes(total) yes(Sun) yes(Cool) yes(High) yes(strong)

$$P(Yes|instance) = \frac{9}{14} * \frac{2}{9} * \frac{3}{9} * \frac{3}{9} * \frac{3}{9} = 0.0053$$

$$P(No|instance) = \frac{5}{14} * \frac{3}{5} * \frac{1}{5} * \frac{4}{5} * \frac{3}{5} = 0.0206$$

$$P(No|instance) > P(Yes|instance)$$

Then Classification output : No.