

الأسبوع السادس

Advice for Applying Machine Learning

- المحاضرة ديه من اهم المحاضرات اللي في الكورس ، لأنها مش بتعرفك طريقة جديدة ،لكن بنفهمك ازاي تتعامل مع المشاكل اللي هتقابلك ، وازاي تستخدم كل اللي اتعلمنته قبل كدة
 - الأهم من تعلم العديد من افكار و خوارزميات الـ ML هو فن تطبيقها ووضعها في اطارها المناسب
 - قد تجد اكثراً من تكينك لحل نفس المشكلة ، وتكون بشكل صحيح ، لكن يفرق كتير في الوقت المستخدم لكل منهم ، فيجب عليك اختيار التكنيك المناسب في المكان المناسب
 - فهن Shawf مشاكل ، و هنحاول حلها بطرق معينة ، ومش هتحلل ، فهنحاول شوفلها طريق تاني و هكذا
-

- فمثلاً لو عندي توقع اسعار بيوت بناء على معلوماتها ، فممكن اعمل linear regression بالمعادلة ديه

$$J(\theta) = \frac{1}{2m} \left[\sum_{i=1}^m (h_\theta(x^{(i)}) - y^{(i)})^2 + \lambda \sum_{j=1}^m \theta_j^2 \right]$$

- طيب عملنا الريجريشن بالـ **training data**، ولما جربنا بالـ **test data** لقينا ان الاسعار الناتجة بعيدة عن الاسعار الحقيقية ، نعمل ايه ؟
 - أحد الحلول انك تزود العينة شوية ، وتجيب صفوف اكتر : بس المشكلة ان جمع المزيد من الداتا حاجة متعبة جداً ، زائد انها متضمنش الحل ، لأن لو كان اصلاً التكنيك المستخدم غير مناسب للعملية ديه فده عبط و تضييع وقت
 - حل تاني انك تزود عدد الـ **features** : اللي هي X لأن عددها القليل ممكن يكون عمل مشكلة underfitting وانت مش واحد بالـ X
 - حل ثالث انك تقلل عدد الـ **features** : لأن وارد يكون اصلاً الجراف عامل overfitting فلازم تخليه يهدى اعضاءه شوية ، وتكون عدد الاكسات اقل
 - ممكن انك تعدل في المعادلة نفسها ، وتخلي فيه اكسات **polynomial** : يعني بدل ما هي اكسات اس واحد ، يبقى فيه اكس 1 تربيع و اكس 1 في اكس 2 و هكذا ، بنفس الداتا ، ممكن يحلها
 - كمان ممكن تغير قيمة لمدا بالزيادة او النقصان : وهي معامل الـ **regulation** يعني قيمة التعريم ، فتقليلها او زيادتها بيخلطي قيم النواتج اكثراً دقة

- لازم سيداتك كخبير في الـ ML

- انك يكون عندك السنس و الحدس , بتاع هل لو عملت كذا كذا (زيادة البيانات مثلا) هيفيد ولا لا , لازم تكون عارفه قبل ما تعمله , لأن عمله اصلا بيكلف وقت و فلوس , فلما انت تخلي الشركة تتكلف فلوس مبالغ في حاجة , ويطلع في الآخر ملهاش لازمة , يبقى انت فاشل
- فيه مبرمجين ممكن تضيع شهور طويلة في احد طرق الحل , وفي الآخر يلاقي ان ملهاش اي لازمة , ويشوف غيرها
- المؤسف ان اغلب المبرمجين بيختارو طرق الحل بشكل عشوائي , يلا نعمل الحل الثاني , لا منفعش يلا نجري الرابع , وطبعا ده كلام فاضي
- عشان كدة هنتعرف علي ما يسمى machine learning diagnostic , يعني فن تشخيص تعلم الآلة , اللي من خلاله اعرف المشكلة فين , وحلها ايه
- وتعريفه هو : اختبار يتم داخل الخوارزميات ، لمعرفة ما الذي يعمل و ما الذي لا يعمل ، وتحديد ما هو الحل الأمثل لتطوير الخوارزم ، ومع انه قد يستغرق وقتا ، إلا انه أقل بكثير من الوقت الضائع في تخمين و تجريب حلول عديدة
-

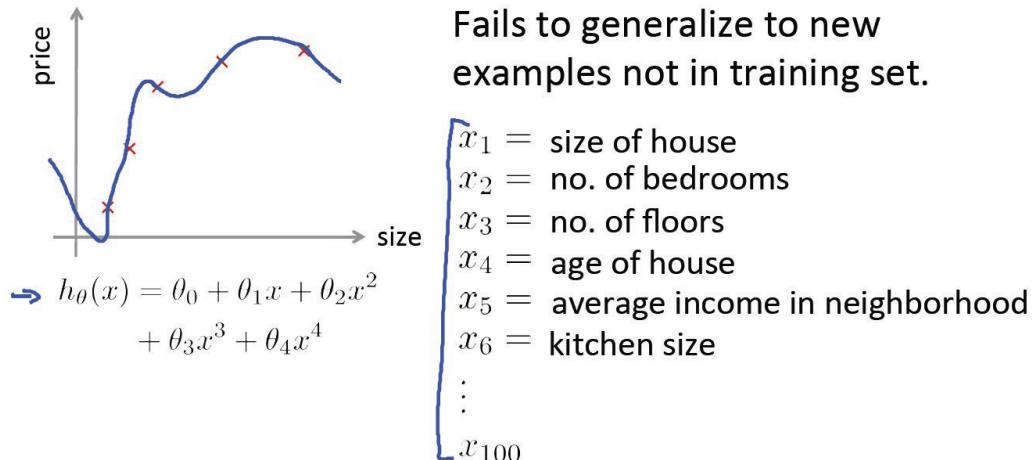
Machine learning diagnostic:

Diagnostic: A test that you can run to gain insight what is/isn't working with a learning algorithm, and gain guidance as to how best to improve its performance.

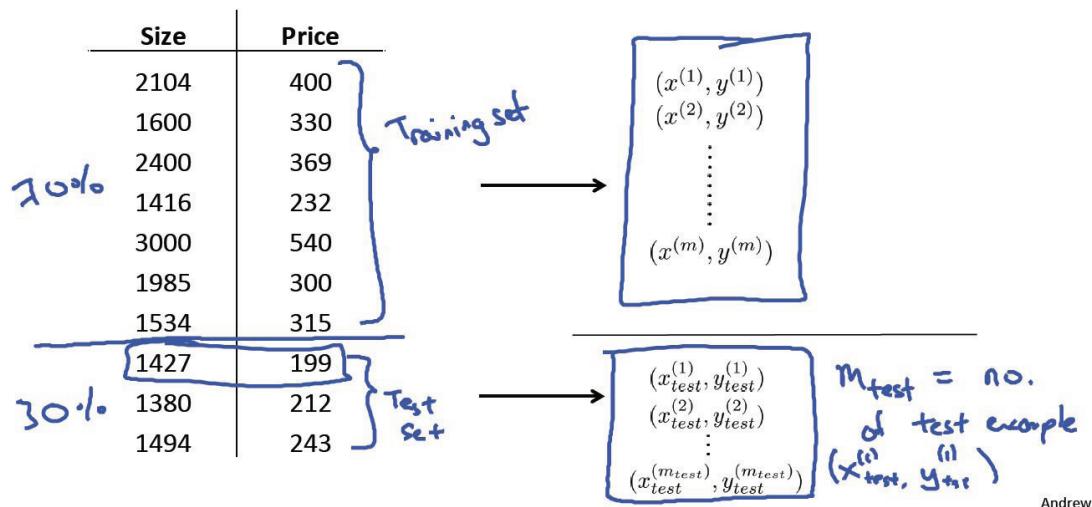
Diagnostics can take time to implement, but doing so can be a very good use of your time.

● طيب نبدأ بما يسمى بـ Evaluating a Hypothesis او تقييم الفرضية

- المقصود بيها ازاي أولاً أعمل تقييم للخوارزم اللي تم عندي عشان اعرف مدي جودته ، وقد ايه هو كويس ولا لا ، وهل فيه OF او UF ولا لا
- فمثلا في المسألة ديه ، عندنا عدد كبير من الـ features ، فده غالباً بيسبب الـ OF والجراف لازق في النقط بشكل مستقر ، فيخلّي نسبة خطأ تقع النقط الجاية أقل كثير



○ اول خطوة تكون بتعيين بيانات التدريب Training Set و بيانات الاختبار Test Set



- وتكون غالبا التوزيع بنسبة 70% للتدريب , و 30% للاختبار , بحيث الـ 70% تخلی الخوارزم يتعلم و يصيغ الـ regression او الـ best fit line
- الاختيار بيكون عشوائي , و خد بالك لو كانت الصفوف مترتبة اصلا بناء على حاجة معينة (من الارخص لlagile مثل) فلازم تعمل shuffle الاول للبيانات قبل ما تقسمهم , والا هتلاقى ان نوعية بيانات التدريب مختلفة عن نوعية بيانات الاختبار
- و خد بالك , ان في حالة الـ OF ان قيمة لبيانات التدريب Training Set بتكون قليلة جدا (لان الجراف بالفعل مجموع كل النقط) بينما قيمة لبيانات الاختبار Test Set (اختبار عملي حقيقي) هتكون كبيرة
- وده بيتم عن طريق اني اجرب قيمة الثيتا اللي طلعت من بيانات التدريب في بيانات الاختبار , وعلى قد ما تكون الـ L كبيرة في بيانات الاختبار على قد ما يكون فيه عك في الشغل

○ فمثلا هنا في معادلة الـ linear regression

$$J_{test}(\Theta) = \frac{1}{2m_{test}} \sum_{i=1}^{m_{test}} (h_{\Theta}(x_{test}^{(i)}) - y_{test}^{(i)})^2$$

او لو هعمل classification بمعادلة الـ logistic regression

$$J_{test}(\theta) = -\frac{1}{m_{test}} \sum_{i=1}^{m_{test}} y_{test}^{(i)} \log h_{\theta}(x_{test}^{(i)}) + (1 - y_{test}^{(i)}) \log (1 - h_{\theta}(x_{test}^{(i)}))$$

هنعمل في كل منهما تعويض بقيمة ثيتا اللي جبناها من بيانات التدريب ، في معادلة باكسات ووايات من بيانات الاختبار ، وعلى اساس قيمة الـ J الخارجة هنعرف شغلنا صح ولا لا

و بالنسبة للـ logistic regression بالتحديد

■ متنساش ان ده classification يعني تقسيم ، يعني يا ده (العميل هيشتري او لا ، الصورة يا تطلع

كلب يا سلعة) ، في الحالة ديه ، الخوارزم بيطلع hx اللي بيكون رقم يتراوح بين الصفر والواحد

ففيه اختبار قد يكون افضل شوية ليه ، اسمه misclassification error 0/1 يعني اختبار التقسيم
الخطي

■ ويقصد بيه اني اشوف كام مرة البرنامج كان كلامه سليم فيها (ان قيمة h كانت اقل من نص ، وكانت الـ y تساوي صفر ، او قيمة h كانت اكبر من نص ، وكانت الـ y تساوي واحد) ، وبرضه اشوف كام مرة البرنامج غلط فيها (ان قيمة h كانت اقل من نص ، وكانت الـ y تساوي واحد ، او قيمة h كانت اكبر من نص ، وكانت الـ y تساوي صفر)
■ وده بيتم بالمعادلة ديه :

$$err(h_{\Theta}(x), y) = \begin{cases} 1 & \text{if } h_{\Theta}(x) \geq 0.5 \text{ and } y = 0 \text{ or } h_{\Theta}(x) < 0.5 \text{ and } y = 1 \\ 0 & \text{otherwise} \end{cases}$$

او ■

$$\text{Test Error} = \frac{1}{m_{test}} \sum_{i=1}^{m_{test}} err(h_{\Theta}(x_{test}^{(i)}), y_{test}^{(i)})$$

■ مع مراعاة ان قيمة err معناها عدد الاخطاء

● طيب عملنا تقييم للخوارزم بتاعنا ، هتتبقي مشكلة اختيار بعض القيم

○ زي قيم الثيتات المبدئية

○ قيمة اللمندا او الفا

○ شكل البولونوميال ، وه يكون بكم اس و بانهي شكل

- و عشان نفهم ده عايزين نفكير حاجة
 - ان وجود قيمة قليلة لـ J من بيانات التدريب مش لازم يكون معناها ان دي حاجة كويستة ، لأن وارد يكون فيه OF فتكون بيانات الـ J لبيانات الاختبار كبيرة و تكون مشكلة
 - طيب لو عندنا بيانات وعايزين نحدد انهي درجة افضل لمعادلة التقسيم هل ه تكون درجة اولي (خطية) ولا تانية (تربيعية) ولا ثالثة (كعوبية) ولا ايه ؟

$$\begin{aligned}
 1. \quad h_{\theta}(x) &= \theta_0 + \theta_1 x \\
 2. \quad h_{\theta}(x) &= \theta_0 + \theta_1 x + \theta_2 x^2 \\
 3. \quad h_{\theta}(x) &= \theta_0 + \theta_1 x + \dots + \theta_3 x^3 \\
 &\vdots \\
 10. \quad h_{\theta}(x) &= \theta_0 + \theta_1 x + \dots + \theta_{10} x^{10}
 \end{aligned}$$

• تحديد درجة المعادلة d يتم بالطريقة ديه :

- نبدا الاول بمعادلة من الدرجة الاولى $d=1$ ونجيب منها قيم ثيتا اللي هنسميها ثيتا $\Theta^{(1)}$ ومنها كمان نجيب قيمة الـ J لبيانات الاختبار (احنا بالفعل استخدمنا بيانات التدريب عشان نجيب ثيتا) واللي هنسميها جي تيست ثيتا $\mathcal{J}_{\text{test}}^{(1)}$
- نعمل نفس الموضوع بالنسبة للمعادلة التربيعية ، ونجيب ثيتا $\Theta^{(2)}$ وجي ثيتا $\mathcal{J}_{\text{test}}^{(2)}$
- نكرر الموضوع لآخر درجة عندنا

المبرهنة $\rightarrow d = \text{degree of freedom}$ ↓
Model selection ↑

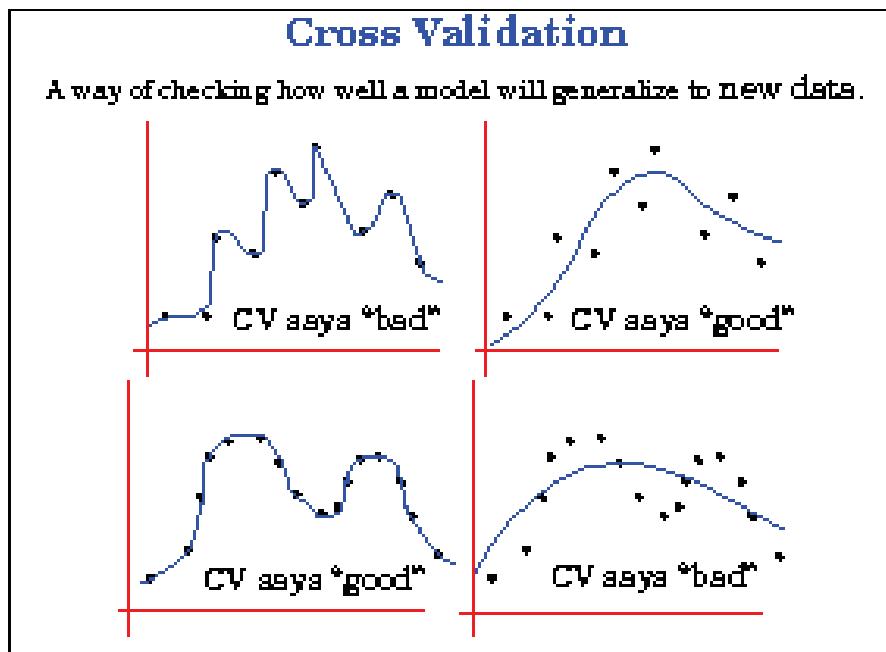
$$\begin{aligned}
 d=1: 1. \quad h_{\theta}(x) &= \theta_0 + \theta_1 x \rightarrow \Theta^{(1)} \rightarrow \mathcal{J}_{\text{test}}^{(1)}(\Theta^{(1)}) \\
 d=2: 2. \quad h_{\theta}(x) &= \theta_0 + \theta_1 x + \theta_2 x^2 \rightarrow \Theta^{(2)} \rightarrow \mathcal{J}_{\text{test}}^{(2)}(\Theta^{(2)}) \\
 d=3: 3. \quad h_{\theta}(x) &= \theta_0 + \theta_1 x + \dots + \theta_3 x^3 \rightarrow \Theta^{(3)} \rightarrow \mathcal{J}_{\text{test}}^{(3)}(\Theta^{(3)}) \\
 &\vdots \\
 d=10: 10. \quad h_{\theta}(x) &= \theta_0 + \theta_1 x + \dots + \theta_{10} x^{10} \rightarrow \Theta^{(10)} \rightarrow \mathcal{J}_{\text{test}}^{(10)}(\Theta^{(10)})
 \end{aligned}$$

Choose $\boxed{\theta_0 + \dots + \theta_5 x^5}$ ↑

- نيجي نقارن بين قيم الجيئات ، ونشوف اقل جيئا فيهم مين ، وفرض كانت جي 5 ، فده معناه ان المعادلة الخامسة هي افضل لهم
- بس فيه مشكلة ، ان لما بيزيدي الاس كتير ، ممكن حتى في بيانات الاختبار تكون النتائج كويستة ، بس مع الواقع تلاقي الارقام غلط ، عشان الـ OF
- عشان نحل المشكلة ديه هنعمل ما يسمى الـ CV

● استخدام بيانات الضبط Cross Validation

- المقصود بيها ان هنقسم البيانات , بدل ما كانت 70% تدريب , و30% اختبار , ه تكون 60% تدريب , 20% ضبط , 20% اختبار
- بيانات الضبط, بتستخدم عشان تساعدنى في ضبط قيم ثيتا , ودرجة المعادلة , عشان اتأكد ان الخوارزم سليم , عشان لما يجي يقيس بيانات الاختبار او اي بيانات جديدة , تبقى كفائته عالية
- وميزته انه بيفضح الـ OF , والـ UF لو شاف اي حاجة زي الصورة ديه



- هنلاقي ان فوق علي الشمال , الـ CV لقي ان فيه OF فلما ظبطننا شوية القيم (سواء بتقليل الدرجة او تغيير اللمنا او الالفا) اتحولت لفوق علي اليمين اللي قال عليها الـ CV كويس
- ولما كان تحت علي اليمين لقي الـ CV انه UF فلما ظبطننا شوية القيم اتحولت تحت علي الشمال اللي قال عليها الـ CV كويس

● فالبيانات تنقسم كدة :

Evaluating your hypothesis

Dataset:

Size	Price	
2104	400	
1600	330	
2400	369	
1416	232	
3000	540	
1985	300	
60%		Training set
1534	315	
1427	199	
20%		Cross validation (cv)
1380	212	
1494	243	
20%		test set

$(x^{(1)}, y^{(1)})$
 $(x^{(2)}, y^{(2)})$
 \vdots
 $(x^{(m)}, y^{(m)})$

$(x_{cv}^{(1)}, y_{cv}^{(1)})$
 $(x_{cv}^{(2)}, y_{cv}^{(2)})$
 \vdots
 $(x_{cv}^{(m_{cv})}, y_{cv}^{(m_{cv})})$

$(x_{test}^{(1)}, y_{test}^{(1)})$
 $(x_{test}^{(2)}, y_{test}^{(2)})$
 \vdots
 $(x_{test}^{(m_{test})}, y_{test}^{(m_{test})})$

$m_{cv} = 10$
 or
 cv
 example
 $(x_{cv}^{(i)}, y_{cv}^{(i)})$

m_{test}

- بيانات التدريب 60% ، واللي هيكون عدد الصفوف M
 - بيانات الضبط 20% ، واللي هيكون عدد الصفوف M_{cv}
 - بيانات الاختبار 20% ، واللي هيكون عدد الصفوف M_{test}
 - ووقتها نجيب التلات جيئات (التدريب ، الضبط ، الاختبار)
- $$J_{train}(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_\theta(x^{(i)}) - y^{(i)})^2$$
- $$J_{cv}(\theta) = \frac{1}{2m_{cv}} \sum_{i=1}^{m_{cv}} (h_\theta(x_{cv}^{(i)}) - y_{cv}^{(i)})^2$$
- $$J_{test}(\theta) = \frac{1}{2m_{test}} \sum_{i=1}^{m_{test}} (h_\theta(x_{test}^{(i)}) - y_{test}^{(i)})^2$$
- ولوقتي نبدا نشوف انهي درجة اعلي بالطريقة ديه :

$$\begin{array}{l}
 \text{1. } h_{\theta}(x) = \theta_0 + \theta_1 x \rightarrow \underset{\text{Min}_{\theta} J(\theta)}{\theta^{(1)}} \rightarrow J_{cv}(\theta^{(1)}) \\
 \text{2. } h_{\theta}(x) = \theta_0 + \theta_1 x + \theta_2 x^2 \rightarrow \theta^{(2)} \rightarrow J_{cv}(\theta^{(2)}) \\
 \text{3. } h_{\theta}(x) = \theta_0 + \theta_1 x + \dots + \theta_3 x^3 \rightarrow \theta^{(3)} \rightarrow J_{cv}(\theta^{(3)}) \\
 \vdots \qquad \vdots \qquad \vdots \qquad \vdots \qquad \vdots \qquad \vdots \\
 \text{10. } h_{\theta}(x) = \theta_0 + \theta_1 x + \dots + \theta_{10} x^{10} \rightarrow \theta^{(10)} \rightarrow J_{cv}(\theta^{(10)})
 \end{array}$$

$\Delta = 4$

Pick $\theta_0 + \theta_1 x_1 + \dots + \theta_4 x^4$ ←

Estimate generalization error for test set $J_{test}(\theta^{(4)})$ ←

- اني اشوف لكل درجة من درجات المعادلة Δ اجيب قيمة الثيتا اللي بتقلل الـ J بقدر الامكان , وده باستخدام بيانات التدريب
- عشان اقارن بين الجيهات , مش هينفع استخدم نفس الجيهات بتاعت بيانات التدريب , ولا هينفع استخدام نفس بيانات التدريب اللي جبت بيها الجيهات, هجيب ساعتها جيهات باستخدام بيانات الضبط و طبعاً قيم الثيتا اللي جبتها من بيانات التدريب , واشوف انهي اقل ل فيهم
- لما الاقي ان افضل J هي مثلاً الدرجة الرابعة , اعمل check تاني ببيانات الاختبار , وهلاقيها فعلاً سليمة

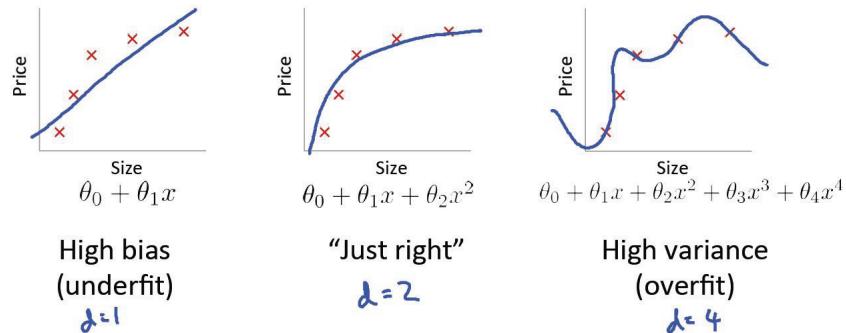
● يعني من غير بيانات الضبط باعمل التالي :

- اشوف قيم ثيتات من كل درجة (اللي بتجيب اقل قيمة للـ J)
 - اطبق الثيتات ديه في بيانات الاختبار عشان اشوف انهي اقل ل
 - وده مشكلته ان مع بيانات حقيقية بلاقي مشاكل
 - لكن مع بيانات الضبط
 - اشوف قيم ثيتات من كل درجة (اللي بتجيب اقل قيمة للـ J)
 - اطبق الثيتات ديه في بيانات الضبط عشان اشوف انهي اقل ل
 - اشف قيمة الـ J مع بيانات الاختبار عشان اتأكد انها قليلة
-

● التحيز و التنويع

- طيب لو تم اختيار الدرجة المناسبة لكن تظل الحسابات فيها مشاكل , فممكن نشك في موضوع الانحياز Bias او Variance التنويع

- التحيز الزائد بيعمل UF لأن الخوارزم بيتجاهل كتير من البيانات ، او بيعمل لها اهتمام قليل شوية ، و ده يؤدي لـ UF لأن الخوارزم مش هيكون يمثل اغلب البيانات ، وده غالباً بييجي من الدرجات الاقل للمعادلة (المعادلة الخطية مثلاً) ، او صغر حجم الثيتات
- التنوع الزائد معناه حساسية عالية لاي تغير في البيانات ، فالخوارزم يحال يجمع كل البيانات مع بعض في جراف واحد ، فيعمل OF ، وده غالباً بييجي من الدرجات العالية للمعادلة ، او زيادة قيمة الثيتا



- و عشان نفهم الكلام ده عملياً ، تعالى نقارن بين قيمتين

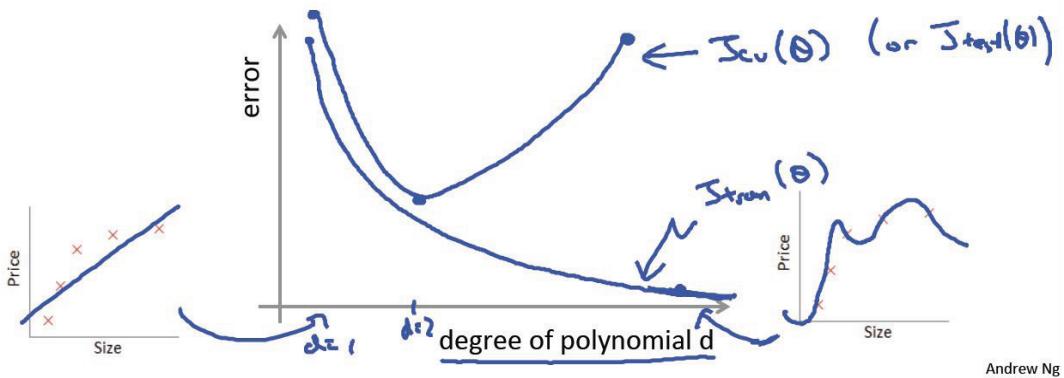
$$J_{train}(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_\theta(x^{(i)}) - y^{(i)})^2$$

قيمة J الخاصة ببيانات التدريب

$$J_{cv}(\theta) = \frac{1}{2m_{cv}} \sum_{i=1}^{m_{cv}} (h_\theta(x_{cv}^{(i)}) - y_{cv}^{(i)})^2$$

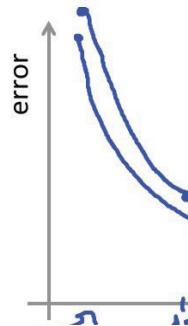
قيمة J الخاصة ببيانات الضبط

- هنا لاحظ في الرسم ده حاجة مهمة :

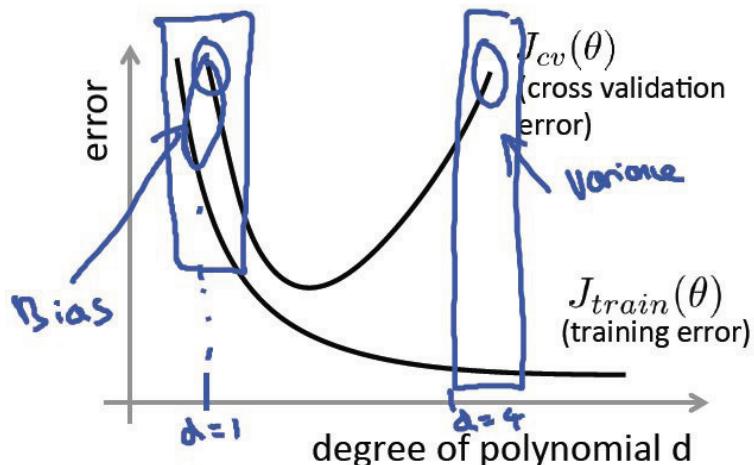


- ان اولاً محور اكس هو عبارة عن درجة المعادلة المستخدمة ، بينما محور واي قيمة الخطأ لـ UF
- ان كل ما يزيد قيمة الدرجة كل ما كان الرسم رايج لـ OF وكل ما قلت كل ما كان رايج لـ UF

- ان دايما قيمة لبيانات التدريب بتقل كل ما تزيد درجة المعادلة ، وده منطقى ، لأن اساسا معاملات الاكسات في المعادلة (الثيتات) بتتطب على اساس بيانات التدريب ، فالخوارزم هيلم كل النقط في جراف معين و ممكن قيمة لـ θ توصل لصفر
- ان قيمة لبيانات الضبط في تكون قليلة في حالتين ، درجة قليلة للمعادلة عشان الـ UF و درجة كبيرة برضه عشان الـ OF ، وبالتالي فيه قيمة في النص اللي تعتبر القيمة الـ $Optimum$ ليها ، واللي المفروض نستخدمها لاحظ ان حتى في الجزء الاول من الرسم قيمة لـ الضبط اعلى من التدريب ، وده منطقى عشان صعب جدا تلاقي ان كفاءة البيانات في الضبط اكبر من التدريب اللي معنولة عليها اساسا

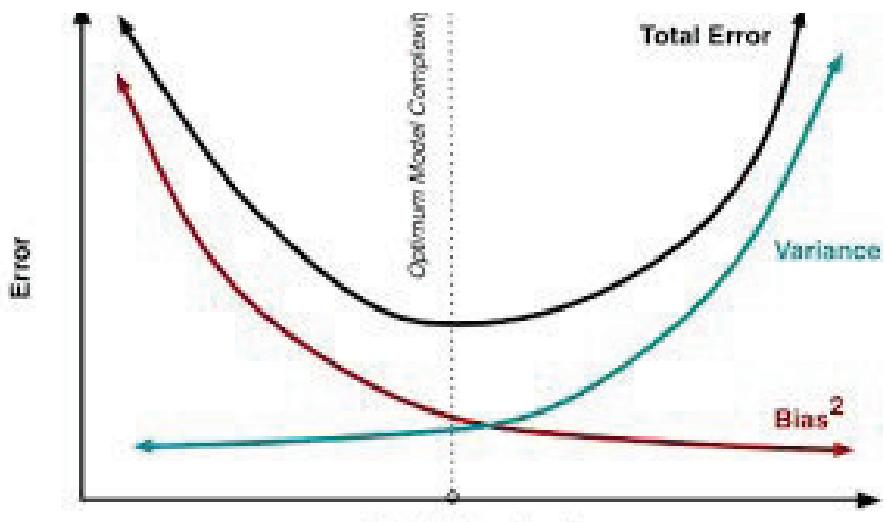


- و لما يكون عندي مشكلة في البيانات ، وانا مش قادر احدد هي UF ولا OF ، ساعتها اشوف الرسم الجاي

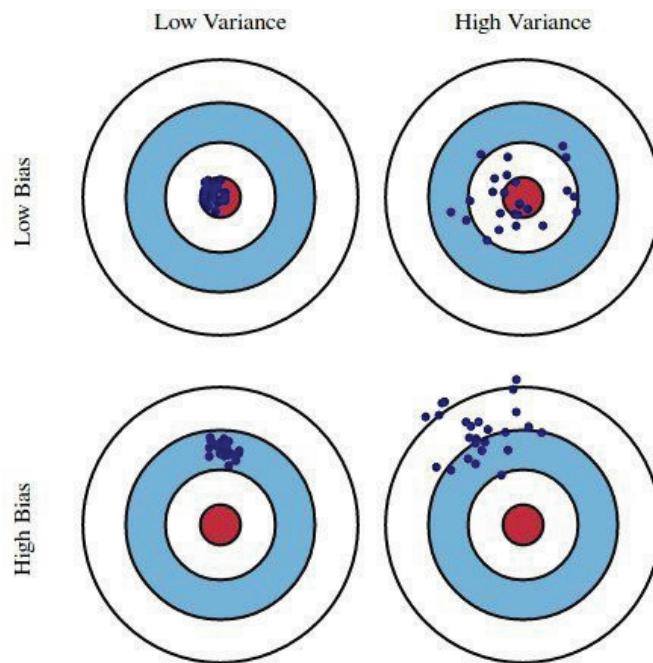


- لو كان كلام من لـ الضبط والتدريب كبيرة ببقي اعرف ان درجة المعادلة قليلة ، وده معناه انه UF
- لو كان لـ الضبط كبيرة ، ولـ التدريب قليلة ، ببقي انا في درجة عالية ، والـ OF ه سبب المشكلة

- و علاقه التنوع Variance بالانحياز Unbias علاقه عكسيه تماما ، في البداية الدرجات القليلة تنوع قليل (ارقام غير متماشية مع المتغيرات) ، وانحياز كبير عن الحقيقة ، بينما في الدرجات الكبيرة ، تلاقي تنوع كبير (الخوارزم بيروح و بيجي مع كل نقطة) ، لكن انحيازه قليل لانه جايب الارقام بالضبط



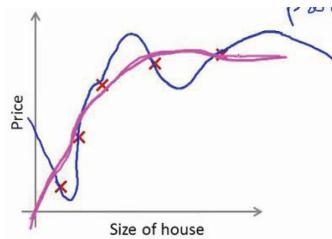
• و هنا هنلاقي مثال عملی لفهم العلاقة بينهم :



- فوق علي الشمال انحراف قليل , فاغلب النقط في قلب الهدف , تتوها قليل , فمش رايحة بعيد عن التمرکز , وديه افضل حالة
 - فوق علي اليمين , الانحراف قليل , فاغلب النقط بتدور حولين الهدف , لكن لزيادة التنوع فهي بتروح و تجي تحت علي الشمال , انحراف كبير , فهي بعيدة عن الهدف , لكن مش متتوعة , فكلها متمركزة في نقطة بعيدة شوية
 - تحت علي اليمين الانحراف كبير , فهي بعيدة , ولان تتوها كبير فهي متوزعة
-

• تأثير الـ **Regularization** في الانحراف و التنوع

- الاول نفترض ايه هو الـ λ , عن Regularization , هو التعديل اللي يحصل للبيانات عشان يجنبها مصير الـ OF , عن طريق اضافة قيمة لمدافي الاخر , عشان يتحوّل الخط الازرق للخط الاحمر هنا



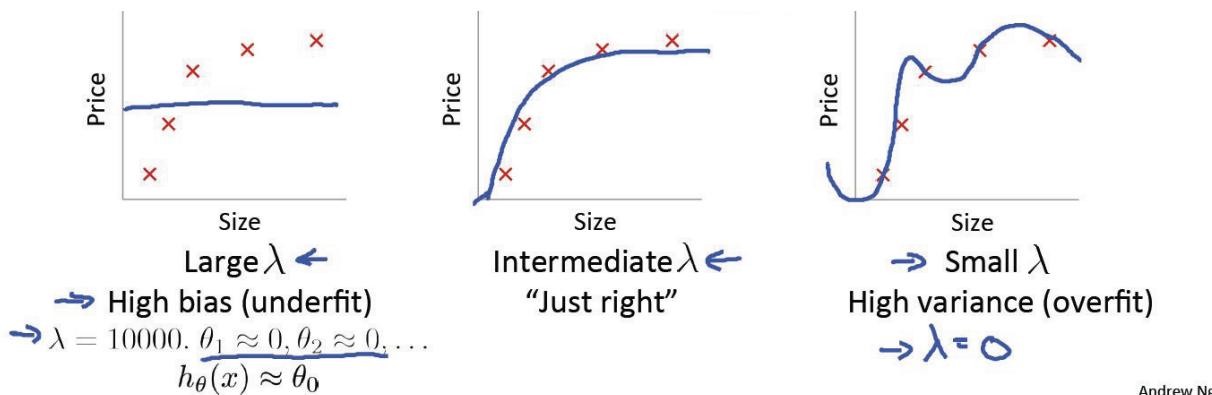
- لو كانت الـ λ تساوي ده

$$h_{\theta}(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$$

- ساعتها بتكون المعادلة بالشكل ده

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

- و عشان ندرس تأثيرها علي كل من الانحراف و التوع , عايزين نعرف هي اللمنا بتعمل ايه اساسا
- زي ما شاييفين في المعادلة اللي فوق , اللمنا بتضرب في مجموع الثيتات سكوير (باستثناء ثيتا صفر), والخوارزم بيحاول يقل قيمة J للاكبر قدر



- فلو كانت اللمنا كبيرة جدا , الخوارزم هيضرط يقل جدا قيمة كل الثيتات عدا ثيتا صفر اللي مش مضروبة فيها , وده هيخلி قيمة كل الثيتات تقريبا بصفر , الا ثيتا صفر اللي ه تكون بقيمة عاديّة , فيكون Hx يساوي تقريبا ثيتا صفر , فع تكون معادلة مش خطية , لكن ه تكون خط افقي مستقيم يساوي ثيتا 0 , زي الشكل الشمال , وه تكون UF
- ولو اللمنا قلت جدا , فده معاناها ان الـ Regularization مش موجود تقريبا , وبالتالي هنفضل الخوارزم OF زي ما في الشكل اليمين
- اما لو اللمنا متوسطة , فمش هتنقل الثيتات كتير , لكن هنضبطها بحيث يتلاشي الـ OF من غير ما ندخل في الـ UF

• طيب نحدد قيمة لمدا المناسبة ازاي ؟

- الاول نبدا نحدد قيم لمدا اللي هنجربها , ويفضل نبدا برقم صغير زي 0.01 ونسميه لمدا 1, بعدها نضاعفه يكون ويكون لمدا 2, ونضاعفه تاني وهكذا , لغاية لما نوصل لرقم كبير زي 10 اللي ه تكون لمدا 12

1. Try $\lambda = 0 \leftarrow$
2. Try $\lambda = 0.01$
3. Try $\lambda = 0.02$
4. Try $\lambda = 0.04$
5. Try $\lambda = 0.08$
- ⋮
12. Try $\lambda = 10$

- بعدها نجريها واحدة واحدة في مع بيانات التدريب , عشان نجيب قيمة ثيتا اللي هتعمل اقل قيمة لـ J , في المعادلة الخاصة بالرجيولاشن اللي هي

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

- كدة هيكون عندنا ثيتا 1 اللي جت لما عملنا لمدا 1 , وثيتا 2 اللي جت لما عملنا لمدا 2 , وهكذا
- مش هبص علي اقل قيمة لـ J دلوقتي , لأن زي ما شفنا انها بتعمل مشاكل , همسك كل ثيتا من غير ما اخد المدا الخاصة بيها , واحطها في نفس المعادلة اللي فوق مع بيانات الـ CV مش التدريب , بس المعادلة المرادي هتون من غير قيمة للمدا , ونشوف اقل J فيهن ه تكون كام

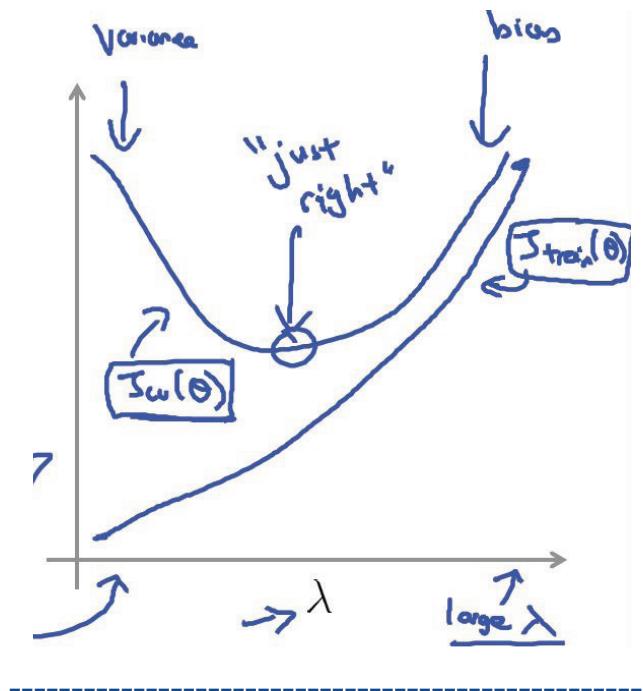
$$J(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_\theta(x^{(i)}) - y^{(i)})^2 .$$

- الـ J الاقل تدل علي افضل ثيتا , وبالتالي , المدا المرتبطة بيها ديه هي الافضل في الاستخدام
- وعشان اتأكد , هجرب نفس الثيتا في بيانات الاختبار , بنفس المعادلة اللي فوق اللي من غير لمدا
- يعني لما هاجي اطبق بشكل حقيقي مع بيانات العمل , هاستخدم المدا ديه اللي جابت افضل ثيتا , اللي جابت اقل J , لكن احنا هنا مؤقتا بنوقف تطبيق المدا عشان اتأكد هل الثيتا اللي خرجت هي بالفعل المثالية اللي مش هتعمل OF ولا UF

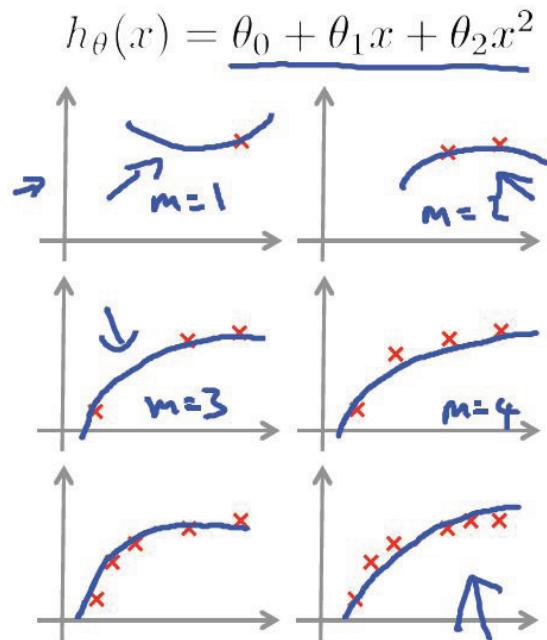
• طب ايه علاقة المدا بالتنوع والانحراف

- هنلاقي ان كل ما نقل المدا, معناه تلاشي قيمة الريجيوليزشن , وبالتالي المعادلة هتفصل في قيمة الـ OF زي ما هي , صحيح قيمة الـ J للتدريب قليلة لكن ه تكون كبيرة في J الضبط زي ما شيافين علي الشمال
- الـ OF معناها ان التنوع كبير , والانحراف قليل

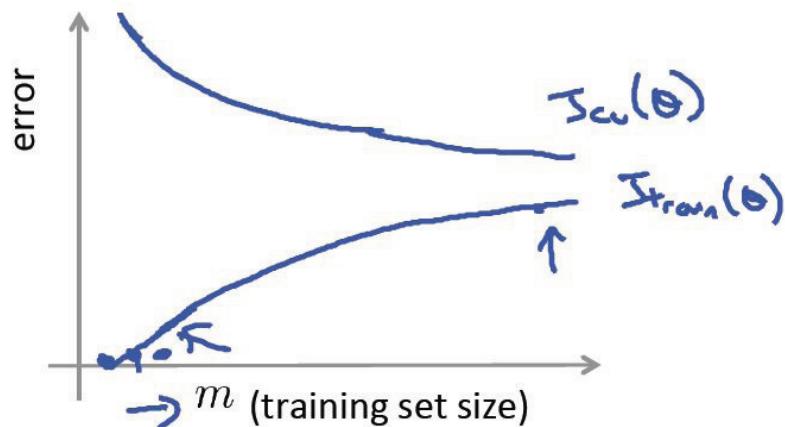
- لكن لو زادت قيمة اللمنا ، فهتقل قيمة الثيتات ، يعني الـ L هتحول لـ UF تدريجيا ، وده بيزيود قيمة كل من L التدريب و الضبط ، وده معناه انحراف كبير و تنوع قليل
- فالحل المثالي اني اخذ قيمة L اللي في النص ، اللي بتقلل قيمة L الضبط لاقل مقدار
- طبعا عيني لازم تكون علي L الضبط مش التدريب ، لأن الضبط هي اللي معناها بيانات العميل الجديدة



- طيب ماذا عن تأثير عدد العينة m على كفاءة الخوارزم
- لو شفنا مثلا علاقة عدد عناصر بيانات التدريب m بنسبة الاخطاء اللي بتيجي من الدالة L
- هنلاقي ان لو المعادلة تربيعية ، وعدد العناصر 1 بس ، هتكون زي الشكل فوق على الشمال
- لو عدد العناصر اتنين ه تكون فوق على اليمين
- كل ما تزيد العناصر ، كل ما تختلف قيمة ثيتات ، بس عشان تعمل لها best fit و بالتالي قيمة الاخطاء هتزيد غصب عنها



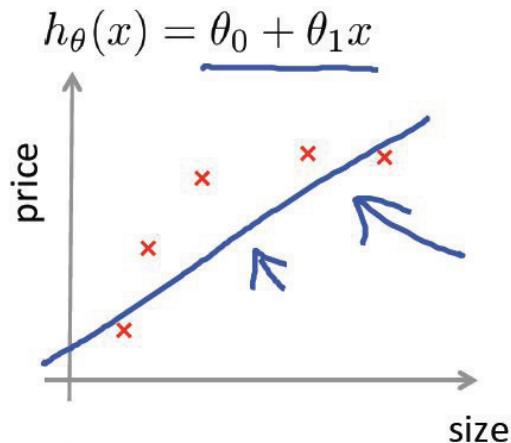
- فنلاحظ في الرسم ان كل ما تزيد قيمة m كل ما تزيد نسبة الاخطاء J في بيانات التدريب



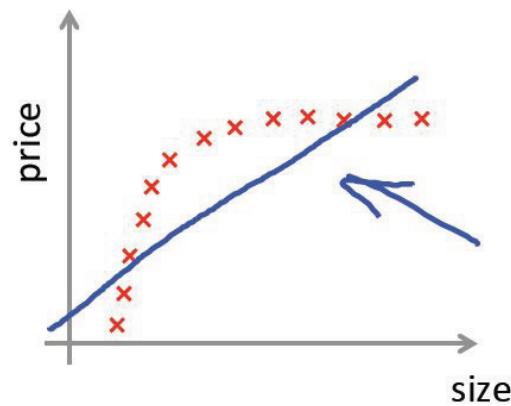
- لكن من البديهي ان بيانات الضبط CV وضعها مختلف ، لأن مع العدد القليل جدا من بيانات التدريب ، صحيح ان L التدريب قليلة (لان الجراف لازق في النقط) لكن فعليا L الضبط هتبقي كبيرة جدا ، ليه ؟ لأن اساسا الخوارزم المعمول بـ m قليل ، مستحيل يقدر يتوقع اي قيم مشافهاش ، وبالتالي قيمة L الضبط ه تكون ضخمة جدا
- كل ما تزيد قيمة m لبيانات التدريب ، كل ما تقل قيمة L لبيانات الضبط لأن الخوارزم بقى اقدر انه يتوقع بيانات مشافهاش

• ماذا عن الانحراف الكبير :

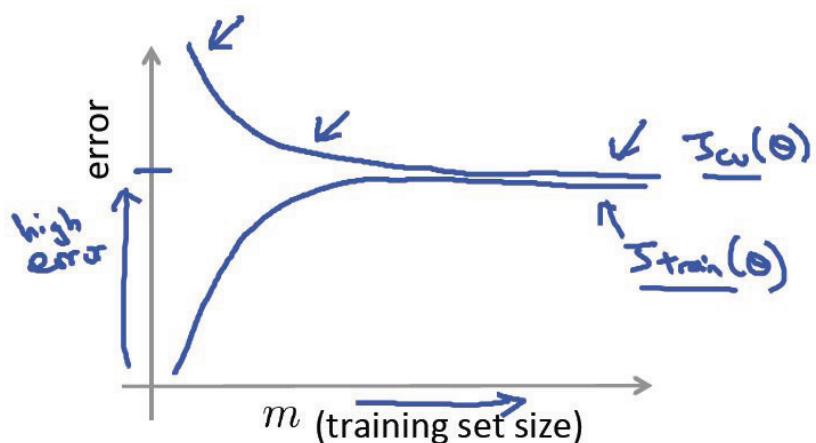
- في حالة كان الانحراف bias اصلا قليل وان المعادلة من الدرجة الأولى ، يعني خطية والمعادلة UF
- ساعتها لو كان العدد m قليل فهيتعمل best fit line بينهم بالشكل ده



- ولو نفس الدالة الخطية ، و عدد البيانات m زاد فبرضه هيتعمل معادلة خطية ، ووقتها هتكون نسبة الاخطاء اعلى



- فمعني كدة ، لما كان عدد بيانات التدريب m قليلة جدا ، كنا ممكن نعمل خط بينهم ، فقيمة J التدريب قليلة ، و القيمة ديه بتزيد تدريجيا كل ما ازود عدد عناصر التدريب ، لغاية لما توصل لرقم شبه ثابت
- بینما مع العدد القليل للـ m قيمة J الضبط كبيرة جدا ، وبتقل شوية بشوية لغاية لما تقرب من قيمة J التدريب

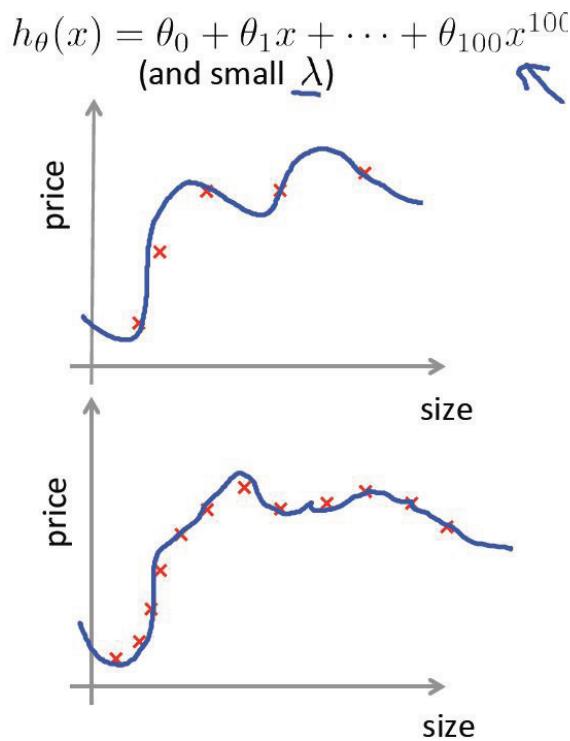


- وده يعرفنا حاجة باللغة الأهمية ، إن في حالات معينة ، زيادة عدد العناصر m مش يصلح الخوارزم ، ولا هيفقل قيمة J لكن تضيع وقت و فلوس على الفاضي

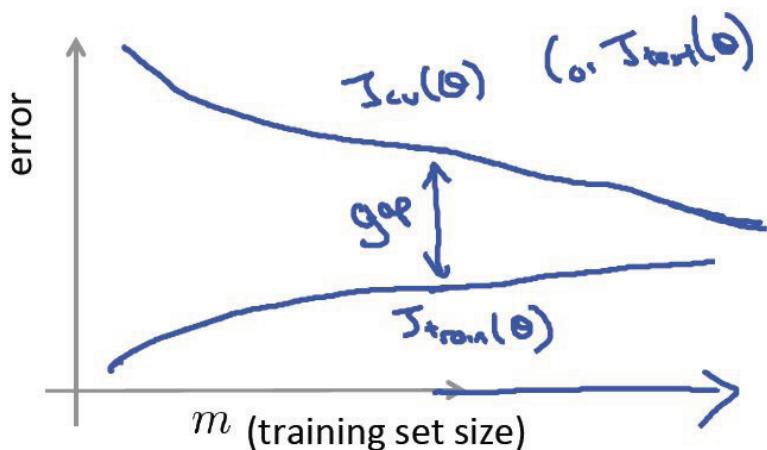
- فلازم يكون عندي السنس ده ، اني احدد امتى محتاج اجيب بيانات تانية ، و امتى محتاج اعدل في الخوارزم

• **ماذا عن التنوع الكبير :**

- لو كان قيمة التنوع كبيرة ، فمعنى كدة ان المعادلة ه تكون باسس كتير ، و بالتالي ه تكون OF



- فهناقي ان سواء عدد العناصر كتير او قليل ، فالخوارزم هيعرف يعملهم تتبع بالكامل ، بس اكيد ان مع العدد الاقل الاخطاء اقل ، وكل ما يزيد العدد تزيد الاخطاء ، بس برضه هنفضل قليلة



- وبالتالي هنلاقي ان جراف ل تدريب قليلة و بتزيد زيادة بسيطة ، بس ل الضبط اصلا كبيرة جدا (عشان الـ OF) و تدريجيا مع زيادة عناصر التدريب الخطأ بيقل تدريجيا ، بس بيظل كبير عشان مشكلة الـ OF

- و خد بالك ان زيادة البيانات هنا حاجة مفيدة , لأنها بشكل ما بتقلل قيمة L الضبط
-

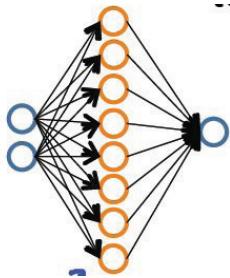
- طيب بعد ما اتعلمنا عدد من التكتيكات لعلاج المشاكل , عايزين نعرف ايه بيستخدم لايه :

- زيادة العينة شوية ونجيب صفوف اكتر :
 - يستخدم بس في حالة الـ OF و ان عندي تنوع كبير , لكن لو التنوع قليل او الانحراف كبير , وعندي UF فزيادة العينة يبقى تضيع وقت
 - تقليل عدد الـ $features$:
 - برضه في نفس حالة الـ OF
 - زيادة عدد الـ $features$:
 - غالبا بتحل مشكلة الـ UF و الانحراف الكبير , لأن زيادة عدد الـ $features$ بيخلطي الخوارزم عنده القدرة انه يتبع النقط بشكل افضل
 - زيادة عدد و اس الاقسات $polynomial$:
 - وده برضه هيحل مشكلة الـ UF و الانحراف الكبير , لأن كل ما يزيد اس المعادلة , كل ما يكون ليها القدرة انها تتمايل اكتر و تتجنب الـ UF
 - زيادة قيمة لمدا :
 - تحل مشكلة الـ OF
 - تقليل قيمة لمدا :
 - تحل مشكلة الـ UF
-

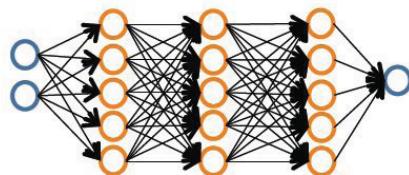
- تصميم الـ NN



- الـ NN الصغيرة , بعدد وحدات أقل و طبقات معدودة
- تكون اسهل في الاستخدام
- اسرع في التعامل
- ارخص
- غالبا بتعمل UF
- غير مناسبة للاعداد الضخمة



- الـ NN الكبيرة , بعده وحدات اكير
- تكون اصعب في الاستخدام
- ابطئ في التعامل
- أغلي
- غالبا بتعمل OF
- مناسبة للاعداد الضخمة
- غالبا باستخدم لما عشان اعجل الـ OF



- طيب علي اي اساس اختيار عدد الطبقات
 - يفضل ساعتها لما تقسم البيانات تكون عامل حساب بيانات الضبط , و تجرب مرة بطبقة واحدة , تقيس قيمة JCV
 - ومرة طبقتين و مرة 3 طبقات و تزود , لغاية لما تشوف ايهم افضل JCV
-

• تصميم نظام الـ ML

- من المهم تعلم كيفية تصميم منظومة ML بشكل سليم وتحديد اولويات الخطوات
- هنعمل نموذج , وهو عمل فلتر للايميلات السبام

From: cheapsales@buystufffromme.com
 To: ang@cs.stanford.edu
 Subject: Buy now!

Deal of the week! Buy now!
 Rolex w4tchs - \$100
Med1cine (any kind) - \$50
 Also low cost M0rgages
 available.

From: Alfred Ng
 To: ang@cs.stanford.edu
 Subject: Christmas dates?

Hey Andrew,
 Was talking to Mom about plans
 for Xmas. When do you get off
 work. Meet Dec 22?
 Alf

- من الواضح ان النموذج اليمين ايميل حقيقي , بينما اليسر هو سبام ,

- اللي هنعمله هو اننا هنختار الـ **features** اللي هي الاكسات ، وهي عبارة عن كلمات تكرر كتير في الايميلات السبام

Building a spam classifier

Supervised learning. $x = \text{features of email}$. $y = \text{spam (1) or not spam (0)}$.

Features x : Choose 100 words indicative of spam/not spam.

deal, buy, discount, address, now, ...

- فهنختار كلمات زي (deal, discount, buy) و نتعامل علي اساس انها بتجي مع السبام كتير ، بينما كلمات زي (اسم الشخص) و نقول انها ممكن تكون ايميل حقيقي (مع ان الواقع غير كدة)

- هنقول ان مصفوفة الاكسات هتكون كدة

$$x = \begin{bmatrix} 0 & \text{address} \\ 1 & \text{buy} \\ 1 & \text{deal} \\ 0 & \text{discount} \\ \vdots & \vdots \\ 1 & \text{now} \end{bmatrix} \quad x \in \mathbb{R}^{100}$$

$x_i = \begin{cases} 1 & \text{if word } j \text{ appears} \\ 0 & \text{otherwise.} \end{cases}$
 From: cheapsales@buystufffromme.com
 To: ang@cs.stanford.edu
 Subject: Buy now!
Deal of the week! Buy now!

- بحيث ان فيكتور اكس ، يكون فيه اصفار ووحيد ، هل الكلمة دي موجودة ولا لا

- لاحظ ان عدد الكلمات الحقيقي اللي بيتعمل للخوارزم بيزيد عن 50 الف كلمة ، ومش احنا اللي بنختارهم ، لكن الخوارزم نفسه بيشوف انهي كلمات اتكررت اكتر
- كمان اللي بيعutto سبام ، بيتعمدو يغيرو شوية حروف ، يعني تبقي كدة : d1scount , m0re , w@tch عشان يفأتو من الفلتر ، فلازم الفلتر ياخد باله من ده كمان
- عشان كدة تكتنيكات عمل الفلتر بتكون كذا حاجة زي :

- استخدام تكتنيك : طبق العسل اللي هو ايميل خدعة معمول من مبرمجي الـ ML عشان يخلوه يقع في ايد اللي بيعutto سبام (زي مثلاً يسجلوه في المنتديات و كدة) عشان يجي عليه اكتر قدر من الايميلات السبام
- محاولة تحليل مصدر الايميل و مبعوت من اي دومين عشان احياناً بيفيد
- محاولة التوصل للكلمات القريبة من تلك التي تستخدم بكثرة فيها لفلترة الايميلات بيها ، زي discounts
- جاية من dealer ، كلمة discount جاية من dealer
- محاولة ايجاد الكلمات القريبة في الـ spelling المستخدمة لخداع الفلتر ، d1scount

• الأخطاء الناشئة :

- ومع جميع تلك الفلترات ، وارد يكون فيه مشاكل في الفلترة ، عشان كدة فيه مجموعة من النصائح :
 - اعمل خوارزم بسيط ، مش لازم يكون معقد
 - اعمل اختبار للفلتر بتاعك علي بيانات الضبط ، عشان تشووف مدي كفائتها

- استخدم learning curves عشان تعرف هل محتاج بيانات اكتر ولا لا (ايميلات اكتر) ■
 - مهم اني ابص بعيني علي نواتج الفلترة ، عشان اشوف مدى صحتها ، وashوف الفلترة الغلط (الايميل اللي انقال عليه سبام وهو لا ، او العكس) عشان اشوف هل فيه باترن معين الفلتر اتلغبط فيه او لا ، او هل فيه كلمات في سبام حقيقي مخدش باله منها ■
 - طيب عملنا الفلتر ، وجربناه علي 500 ايميل ، ولقينا ان فيه اخطاء في تقييم 100 ايميل مثلا ، نعمل ايه ؟
 - لازم نتصص على الـ 100 ايميل بنفسنا و نشوف فيهم حاجتين مهمتين :
 - اولا نوعيتم ايه ، لو لقينا مثلا ان الـ 100 دول متوزعين كالتالي :
 - ادوية : 12 ايميل
 - ردود وهمية : 4
 - سرقة كلمة سر : 53
 - اخر : 31
 - معني كدة ان الفلتر فيه عجز كبير في تقييم نوعية سرقة كلمة السر واني محتاج اقوىه
 - الحاجة الثانية اني اشوف ايه من الـ features اقدر ازوده عشان يحسن الاداء ، هل مثلا لو عملت موضوع الـ spelling او ركزت علي وجود صور معينة و هكذا
 - و النصيحة انك ممكن تعمل فلتر سريع من غير ما تضيع وقت في برمجته ، لكن تخلي الوقت افضل يضيع في انك تراجع الفلتر عشان تشوف ايه الصح و الغلط
 - ممكن استخدام نوعية برامج اسمها Stemming Software والتي بتعمل مقارنة بحروف الكلمات عشان الخوارزم يقدر يحدد ايه الكلمات اللي شبه بعض و ممكن يحطها في البلاك ليست زي :

Discount , discount , discounted , discounting
 - بس خد بالك البرنامج ده زي ما ليه ميزة فهو ليه عيوب ، فممكن مثلا كلمة universe ، university هما كلمتين مشابهتين في الحروف ، لكن مختلفتين في المعنى ، فممكن البرنامج يجيب كلمتين علي اساس شبه بعض وهما ملهموش علاقة ببعض
 - احيانا بيكون من المناسب تجرب استخدام Stemming Software ، واجرب من غيره ، واقارن بين قيمة JCV لده و ده ، وashوف انهي افضل
 - كمان احيانا باخلي البرنامج يدقق علي الكابيتال و السمول (كلمة Man تفرق عن man) ، او يتعامل علي انها نفسها هي هي ، برضه قيمة JCV هي اللي هتحدد اعمل كدة ولا لا
-

• القيم المنحرفة Skewed Classes

- يقصد بها أن فيه بعض الحالات حساب الدقة لا يكون معبرا عن كفاءة الخوارزم
- يعني مثلا ، لو كان فيه 1000 مريض انا شاكك اصابتهم بالسرطان ، و فعليا كان 5 منهم هما المصابين و 995 اصحاء

- فلو ان الخوارزم اظهر لي ان 100% من المرضى اصحاء , فكدة يكون الخوارزم جاب 995 نتيجة صح (الاصحاء) , و 5 نتائج غلط (كل المرضى) , وده معناه كفاءة 99.5%
- ومع الرقم العالى للكفاءة accuracy الا ان الخوارزم ده فاشل تماما في عدم ايجاد اي مريض من المرضى , وده معناه ان رقم الكفاءة accuracy احيانا بيكون مضلل misleading
- فيه برضه مصيبة تانية ممكن تحصل , اني الخوارزم يقول ان كل الـ 1000 عندهم سرطان , وكمدورة ضمنا ان الخمسة اللي عندهم سرطان اتكلهم فعلاً عندهم سرطان , بس المصيبة ان فيه 995 واحد جاتتهم بيانات غلط في خوارزميات تانية , ممكن نسبة خطأ أكبر من كدة , بس مش ه تكون مضللة زي ديه , فالنوع ده اسمه القيم المنحرفة Skewed Classes
- طيب عايزين نختار رقم تاني يعرفنا الصح فين , هيكون قيمة الضبط precision , والاستدعاء recall

• جدول الصحة :

- عايزين او لا نتعرف على جدول البيانات الصحيحة والخاطئة اللي بيكون كالتالي :

		Actual class	
		1	0
Predicted class	1	True Positive	False Positive
	0	False Negative	True Negative

- لو عايزين نقارن بين القيم المتوقعة (0 , 1) والقيم الحقيقية (0 , 1) , هنلاقي التالي
 - لو القيمة المتوقعة 1 , وهي اصلاً 1 , تبقى True Positive
 - لو القيمة المتوقعة 0 , وهي فعلاً 0 , تبقى True Negative
 - لكن لو القيمة المتوقعة 1 , وهي اصلاً صفر , يبقى False Positive
 - كما لو القيمة المتوقعة 0 , وهي اصلاً 1 , يبقى False Negative
 - وكأن لو صحت التوقعات يبقى True , لو اخطأ التوقع يبقى False

- فلو ان فيه الف مشتبه فيه بالسرطان فعلياً فيه 5 مرضى , والخوارزم طلع ان فيه 4 مرضى , اتنين منهم مرضى فعلاً و اتنين مش مرضى
 - يبقى الـ TP يساوي 2
 - الـ TN يساوي 993

- اـ FP يساوي 2 ■
- اـ FN يساوي 3 ■

• قيمة الضبط Precision

- قيمة الضبط تساوي : المتوقع الايجابي الصحيح على مجموع كل المتوقع الايجابي

$$\frac{TP}{TP + FP}$$

- يعني في المثال اللي فات هيكون 2 علي 4 يعني 50% , وده رقم معيب جدا
- وده معناه في المثال ده: يا تري كام واحد من اللي قلنا عندهم سرطان , فعلا عنده سرطان
- و معناه في المنطق: نسبة قد ايه من اللي الخوارزم قال انه صح . طبع فعلا صح
- لاحظ ان القيمة ده لا تساوي قيمة FN على $FN+TN$

• قيمة الاستدعاء Recall

- قيمة الاستدعاء تساوي : المتوقع الايجابي الصحيح على مجموع كل الارقام اللي هي اصلا ايجابية سواء اتعرفت او لا

$$\frac{TP}{TP + FN}$$

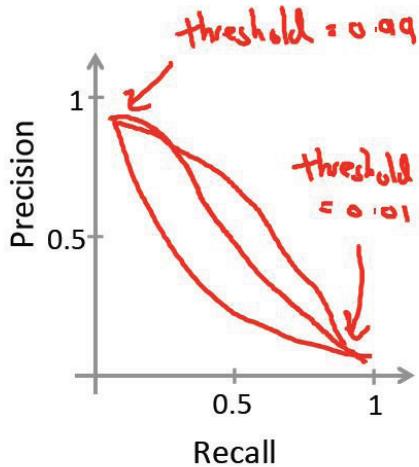
- يعني في المثال اللي فات هيكون 2 علي 5 يعني 40% , وده رقم زفت
 - وده معناه في المثال ده: يا تري كام واحد من اللي عندهم سرطان فعلا الخوارزم شافه و عرفه
 - و معناه في المنطق : يا تري الخوارزم اكتشف نسبة قد ايه من اللي فيه الحاجة المطلوبة
-

• تطبيق قيم الضبط/الاستدعاء في الـ ML

- عشان نفهم علاقة الضبط و الاستدعاء , في تعليم الالة , تعالى نفكـر اـ logistic regression
- لو هعمل خوارزم عشان اتوقع مريض بالسرطان , وهقول ان لو قيمة اـ $y = 0$ هو غير مريض , ولو بـ 1 يبقى مريض
- ولو طبع hx اكتر من نص هيبقـي تبع 1 , يعني مريض , واقل من نص يبقى غير مريض

• الحالـة الأولى :

- بفرض ان لسبب معين , مهم جدا اني مقولش لاي حد صحيح انه مريض , ومفيش مشكلة لو فيه مريض مقولتش انه مش مريض , يعني بالعربي مش هقول لحد انه مريض الا لو كنت متأكد جدا
 - ساعتها هرفع قيمة α threshold اللي هي العتبة المحددة , بدل ما تكون نص هتكون 0.7 , وده هي عمل حاجة مهمة
 - ان بعض اللي كنت شاكك انهم مرضي (0.6 مثلا) مش هقولهم انهم مرضي , وده هي زود قيمة α 0 و يقل قيمة 1 شوية , صحيح انه هينفذ مجموعة من الاصحاء من اني اخضمهم , لكن هيعرض حياة بعض المرضى الحقيقيين للخطر اني مش هقولهم
 - دلوقتي لأن قيمة الضبط معناها (نسبة قد ايه من اللي الخوارزم قال انه صحيحة . طبعاً فعلاً صحيحة) فكدة قيمة الضبط هتزيد شوية لاني هقلل عدد الاصحاء اللي هقولهم انتو مريض
 - لكن لأن قيمة الاستدعاء معناها (يا ترى الخوارزم اكتشف نسبة قد ايه من اللي فيه الحاجة المطلوبة) , فقيمة الاستدعاء هتقى شوية , لأن ممكن يكون فيه مريض وانا مش هقولهم
 - وطبعاً لو خليت العتبة 0.8 يبقى قيمة الضبط هتزيد اكتر و الاستدعاء هتقى اكتر
 - فزيادة قيمة العتبة بيعلي الضبط و يقلل الاستدعاء
- الحالة الثانية:(العكس)
- بفرض ان لسبب تاني , مهم جدا اني مفوتتش اي حد مريض , يعني اي حد شاكك فيه لازم اقوله انه مريض , ومفيش مشكلة لو فيه صحيح قلتله انت مريض مش ازمة كبيرة , يعني بالعربي اي حد هشك فيه هقوله انت مريض من سبيل الاحتياط
 - ساعتها هقلل قيمة α threshold اللي هي العتبة المحددة , بدل ما تكون نص ه تكون 0.3 , وده هي عمل حاجة مهمة
 - ان بعض اللي كنت شاكك انهم اصحاء (0.4 مثلا) هقولهم انهم مرضي , وده هي زود قيمة α 1 و يقل قيمة 0 شوية , و ده هينفذ مجموعة من المرضى انهي هعرفهم المرض, لكن هيخص مجموعة اكبر من الاصحاء اللي شاكك في مرضهم و هقولهم و خلاص
 - دلوقتي لأن قيمة الضبط معناها (نسبة قد ايه من اللي الخوارزم قال انه صحيحة . طبعاً فعلاً صحيحة) فكدة قيمة الضبط هتقى شوية لاني هزود عدد الاصحاء اللي هقولهم انتو مريض
 - لكن لأن قيمة الاستدعاء معناها (يا ترى الخوارزم اكتشف نسبة قد ايه من اللي فيه الحاجة المطلوبة) , فقيمة الاستدعاء هتزيد شوية , لأن تقريبا كل المرضى هيعرفوا
 - وطبعاً لو خليت العتبة 0.2 يبقى قيمة الضبط هتقى اكتر و الاستدعاء هيزيد اكتر
 - فتقليل قيمة العتبة بيقلل الضبط و ويزيود الاستدعاء
- وممك نرسم جراف كدة :



- محور اكس الاستدعاء , محور واي هو الضبط
- لما كانت العتبة كبيرة (فوق علي الشمال) كان الضبط كبير و الاستدعاء قليل
- العكس , لما كانت العتبة قليلة (تحت علي اليمين) الضبط قل و الاستدعاء كبر
- ولو هخلي العتبة في النص , كلام من الضبط و الاستدعاء هيفضلو متواسطين
- وبينهم ممكن الجراف يبقى خطى , او كيرفي فوق او تحت حسب اتجاه القيم

• كيفية المقارنة :

- طيب هنا مع وجود قيمتين للضبط و الاستدعاء , هيكون فيه سؤال , اختار علي اساس انهي فيهم , وماذا لو واحدة كبيرة وواحدة صغيرة , اعمل على اساس مين ؟
- يعني مثلا لو عندنا الجدول ده :

	Precision(P)	Recall (R)
Algorithm 1	0.5	0.4
Algorithm 2	0.7	0.1
Algorithm 3	0.02	1.0

- عندنا 3 خوارزم , كل واحد عملني قيمة للضبط و قيمة للاستدعاء , اختار انهي فيهم ؟ ؟
- لو فكرنا اننا نعمل متوسط مثلا بينهم , هتلaci القيم كالتالي :

- خوارزم 1 : 0.45
- خوارزم 2 : 0.40
- خوارزم 3 : 0.51

- فقيمة المتوسط قالت ان افضلهم هو الثالث , مع ان الضبط فيه ضايع , وده يدل ان العتبة قليلة جدا , يعني هو قال ان الجميع عنه سرطان , وطبعا ده غير سليم

■ نفس الموضوع , لو العتبة زادت , هنقول قيمة الاستدعاء و يزيد الضبط , ويطلع متوسط جميل لكن الكفاءة وحشة

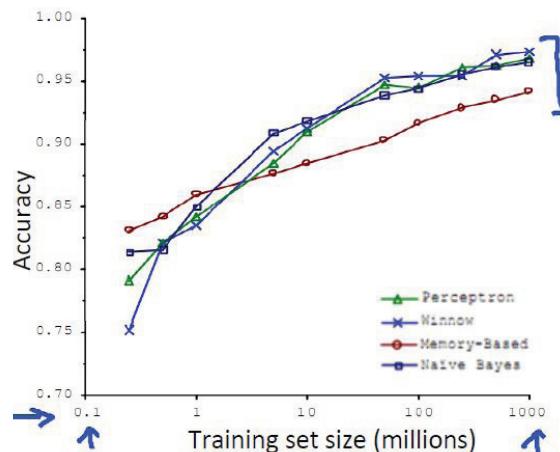
- فالحل استخدام قيمة ما يسمى F1 Score اللي معادلتها كدة :

$$F = \frac{2PR}{P+R}$$

- هنا هناقي ان القيمة النهائية , معتمدة على قيم الضبط و الاستدعاء مع بعض , ولو واحدة فيهم قليلة جدا , الإف هنقول برضه وممكن توصل لصفر
 - و القيمة المثالية , ان الخوارزم يكتشف كل المرضي , ويستبعد كل الأصحاء , و ساعتها كلا من الضبط والاستدعاء هيكونو بـ 1 , وقتها هيكون F بوحدة
 - ولو ايا من الاستدعاء او الضبط بصفر , الـ F هيكون صفر فورا
 - فقيمة الـ F تتراوح بين الصفر والواحد , وهو افضل ممثّل عن كفاءة الخوارزم
-

● إعداد البيانات :

- علمنا من قبل ان الكبيات الكبيرة من البيانات لا تقيد دائما , لكن في بعض الاحيان , تكون الكميات الكبيرة من البيانات مهمة للغاية
- من سنوات عديدة , قام العالمان banko & brill باختبار اعداد من البيانات على اكتر من خوارزم , ووجدو ان كفاءة جميع الخوارزميات تتزايد مع كثرة البيانات بالشكل ده :



- فكان من الواضح ان مع الاعداد الكبيرة للبيانات , الكفاءة بتزداد ليهم كلهم , ده اللي خلامهم يقولو : مش هيفرق مين هيستخدم خوارزم ايه , المهم مين اللي معاه بيانات اكتر ■

**“It’s not who has the best algorithm that wins.
It’s who has the most data.”**

- وهنا ييجي سؤال مهم ، من شوية قلنا ان فيه حالات زيادة البيانات مش هتفيد في حاجة ، وهنا بيقولو ان زيادة البيانات حاجة مهمة ، نصدق مين فيهم :
- الحقيقة ان فيه حالتين مهمتين لازم ندرسهم ونفهم الفرق بينهم :

○ الحالة الأولى

- لو المشكلة عندي اني هملأ فراغ في كلمات مثل زي :

In the breakfast I ate ----- eggs

- وكانت الاختيارات اني هحط واحدة من الكلمات ديه :

Too - two - to

- فهنا لو ان الخوارزم شاف كمية كبيرة من البيانات زي ديه ، هيقدر يستنتج اي مسالة زي كدة

○ الحالة الثاني

- لو المشكلة عندي اني عايز اجيب سعر بيت عن طريق مساحته بس ، من غير اي معلومات تانية ، هنا عدد الـ features قليل ، ولو جبت 500 الف بيانات زي كدة ، الخوارزم مش هيعرف يجيب البيانات بدقة ابدا

- يعني هنا زيادة البيانات مش هيفيد حاجة الا ضياع الوقت فيها

- طب نفرق ازاي بينهم ، ونعرف امتى زيادة البيانات مفيدة ولا لا :
- في الحالة الأول لو عرضناها لبشرى متخصص (مدرس انجليزي) و قلنا له يا ترى هحط كلمة ايه ، هيعرف يجاوبها فورا ، وهذا زيادة البيانات هتفيد
- في الحالة الثانية لو عرضناها لبشرى متخصص (سمسار) و قلنا له سعر البيت الـ 200 متر ، مش هيعرف يجاوب طبعا لانه يحتاج يعرف بيانات تاني وهذا زيادة البيانات مش هتفيد

● و نختم حاجة مهمة :

- البيانات الكبيرة ، بتكون مفيدة في حالة عندي features كثيرة ، لأن
- البيانات الكثيرة بتقلل التنوع Variance
- والـ features الكثيرة بتقلل الانحراف bias
- والحاله المثاليه ، ان يكون كلام من التنوع والانحراف قليلين مع بعض ، فاتجنب الـ OF & UF مع بعض
- بينما لو واحدة كثيرة وواحدة قليلة هقع في مشكلة OF او UF