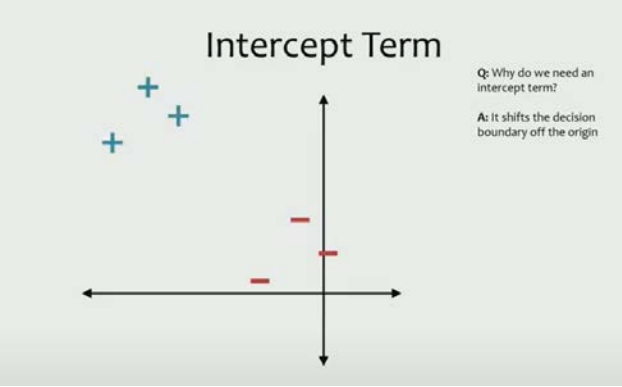
بسم الله الرحمن الرحيم

أهلا بيكم يا شباب .. انهارده هنفكر في ال linear regression .. وقبل ما نوصل هناك .. هنلف كدا لفه سريعه علي ال perceptrons و هنكتشف هو ليه ال perceptron بيشتغل يعني .. ليه بنتكلم علي ال linear regression ... اول واحد هو اننا نتكلم علي regression algorithm و الفرق ما بين ال regression and classification ان الأوتبوت هيبقا real number بدل ماكان integer .. بالاضافه للسب الاولاني .. انو بيبقي مثال علي اننا نبدأ نفكر في ال learning problems بطريقه مختلفه و في framework مختلفه .. بحيث هنا انك بتعمل recasting our learning problem كإنها optimization problem .. الفكره ديه اللي هي "Casting learning as an optimization problem is going to carry through a bunch of different methods and models we will think about"

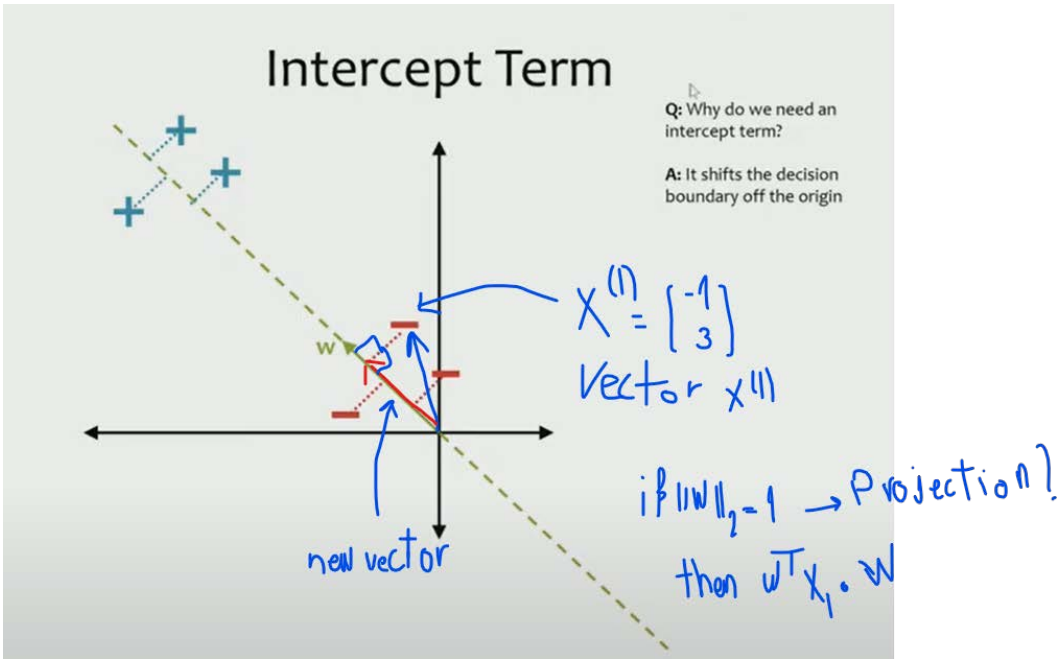
أسيمنت 3 نزل يا جميل .. يلا أبدأ فيه .. هو كلو نظري مفهوش برمجه

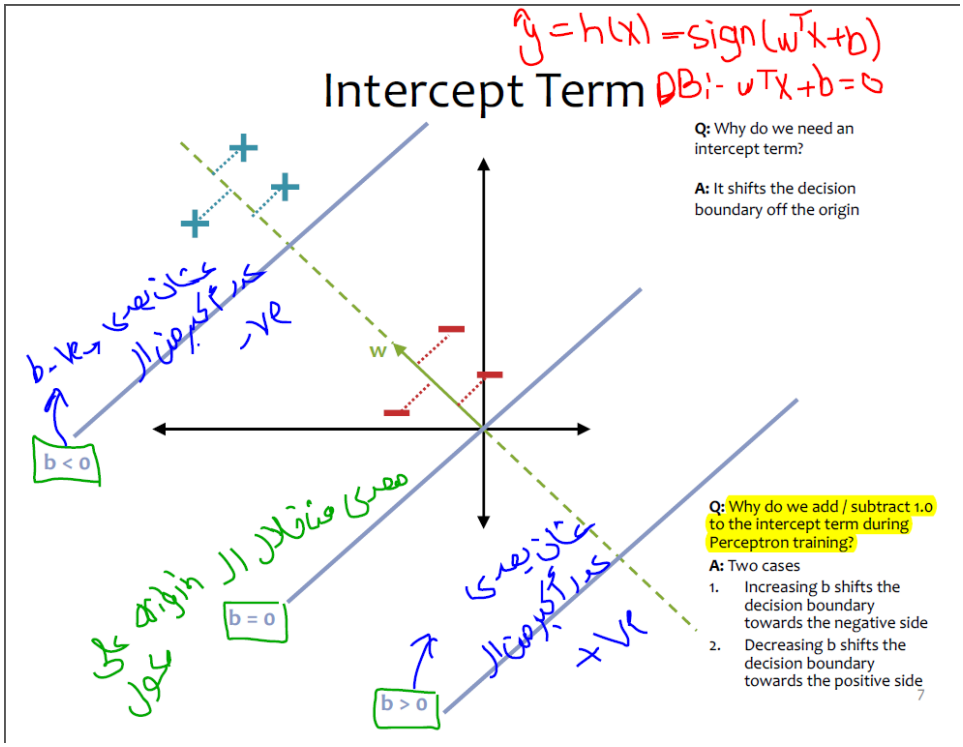
يلا نبدأ

احنا هنبدأ من اننا نفكر ازاي ال intercept term بيلعب دور في ال perceptron algorithm .. اول سؤال .. انت ليه محتاج intercept term اصلا ... عشان هو بي shift ال decision boundary بعيد عن ال origin .. لو انت مكنش عندك intercept term ..في الحاله ديه كل ال decision boundaries هتعدي بال origin ... تعال نفكّر في داتا سيت ..



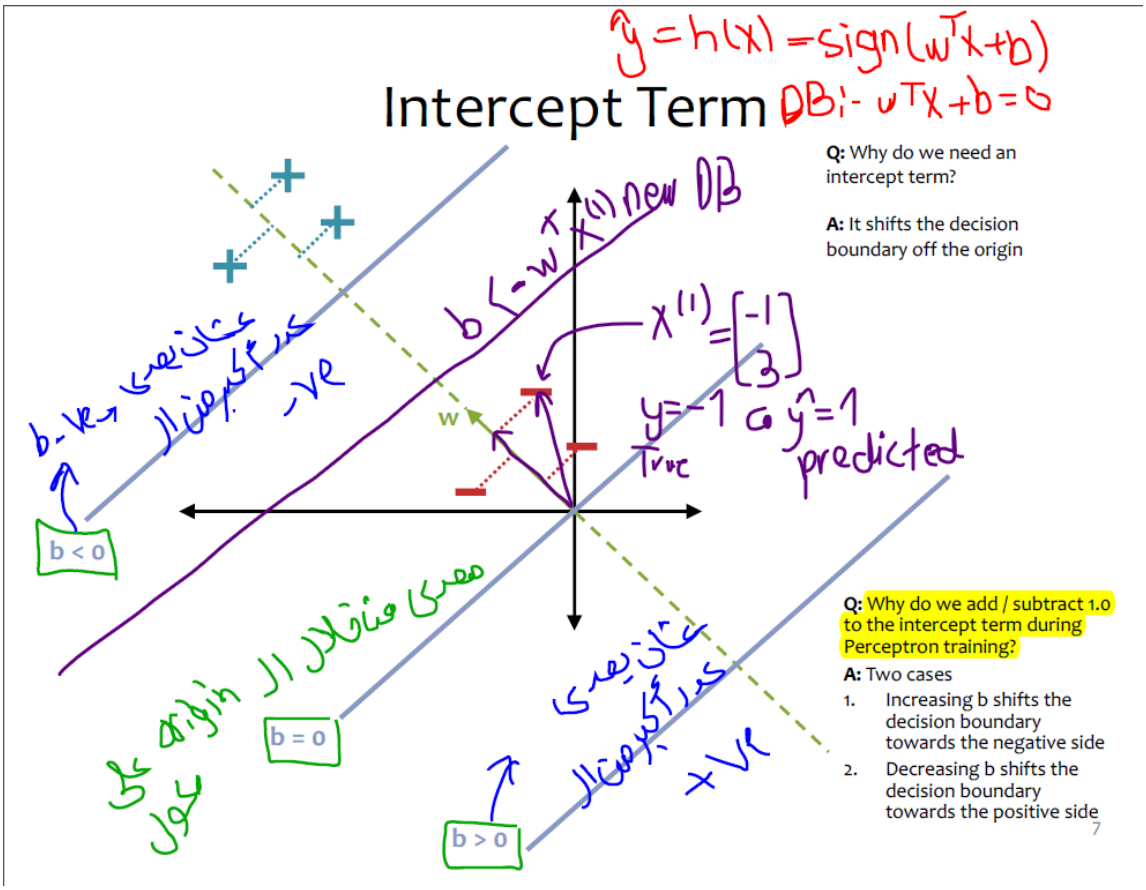
3 + و 3 - .. تخيل ان عندك فيكتور اسمو W .. اختارناه ... تخيل انك ترسم خط في نفس الدايركشن بتاع ال W vector رايح للمالانهايه .. أول ما ترسم الخط ده .. عاوزين نفكر هيبقا الشكل عامل ازاي بعد ما نعمل projection لكل النقط اللي عندنا علي الخط الأخضر في الصوره ده ...





الدكتور سأل سؤال ... دلوقت ال b بتساوي صفر .. وجالك إكس 1 .. و ال wTx1 طلعت بوزيتيف .. فدلوقت هي بقت misclassified .. ايه هي القيمه الجديده اللي هتخلي ال wT.X1 بقت negative لما اجيلها تاني اللفه الجايه في ال iterations بتاعت التريننج ..؟؟

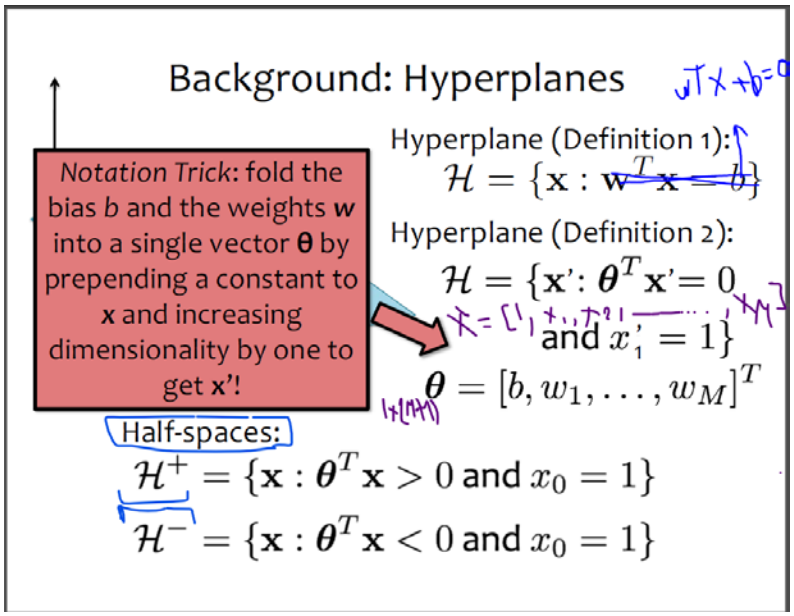
فدلوقت انت بتقول ان wTX1+b بتطلع بوزيتيف .. يبقا انا محتاج اعدل علي ال bias انو يبقا نيجاتيف عشان اقدر احتوي النقطه الجديده اللي جايالي .. ففي الحاله ديه انت عندك معادلة الباوندري كوندشن هي اصلا wTx+b = 0 .. فبالتالي انت لوقلت ان b = -wT.X1 .. في الحاله ديه الخط الجديد هيعدي علي طول بالنقطه اللي اسمها X1 ... انما عشان يقدر يحتويها ... فهيبقا اقل من ال –wT.X1 ..



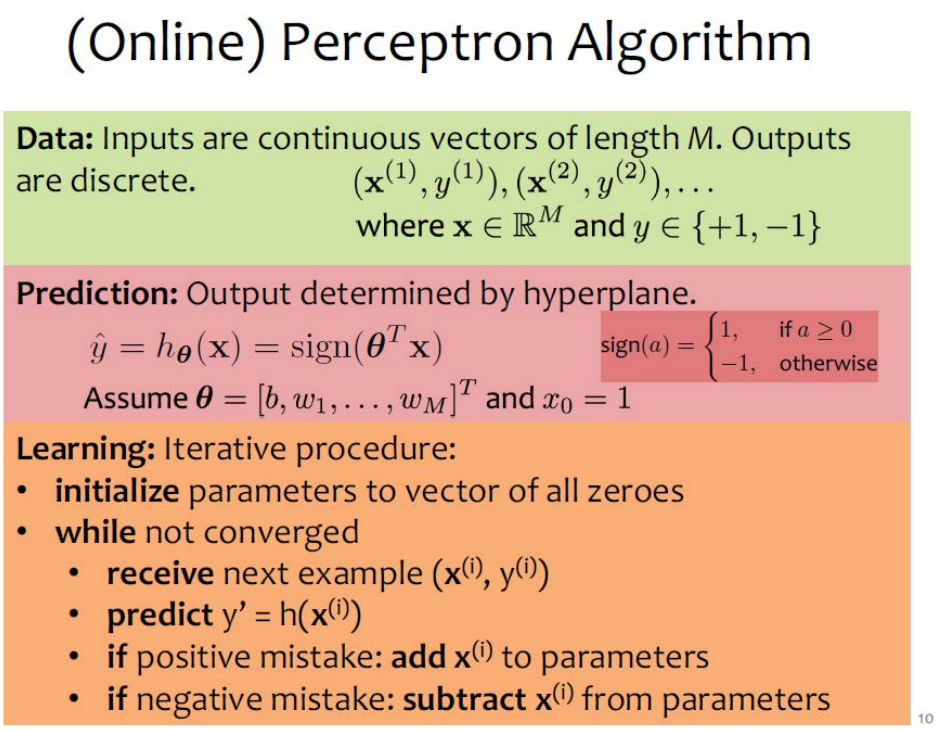
فلما هتبقا قيمة ال bias سالب أكتر .. بيقا هتقدر تحتوي الرقم السالب اللي عندك اللي انت عملتلو misclassification .. طيب اللي نقدر نسألو هنا .. هو ليه ال perceptron مش بيشتغل علي طول .. كل ما يجيلك نقطه و تغلط فيها تقوم حاطط ال b = +- wT.x .. الرد ان اننا مش عاوزين نبعد اوي كدا .. احنا محتاجين نروح بس 1 في الناحيه اللي احنا محتاجينها .. فيا تجمع 1 او تطرح 1

طيب ال inductive bias أول حاجه ان ال decision boundary لازم تبقي linear .. تاني حاجه ان ال most recent mistakes هي ال most important .... فبالتالي تخيل عندك a perceptrion و عندك 100 مثال .. وبعدين يقوم عامل كلو صح .. بس هوبا دابل كيك .. تجيلو نقطه يعملها غلط .. .. ايه اللي هيحصل .. يعني احنا علي مدار 1000 مثال .. مغيرناش البارمترز بتاعتنا .. هنيجي علي البطيخه ديه ونقوم مغيرنها .. هو فعلاً ده اللي هيحصل .. واحتمال كمان لما يغير تلاقي ال 1000 مثال الي فرحان بيهم دول .. كلهم بقو في الناحية الغلط ... فاحنا محتاجين ن prioritize ال most recent mistakes يعني ..

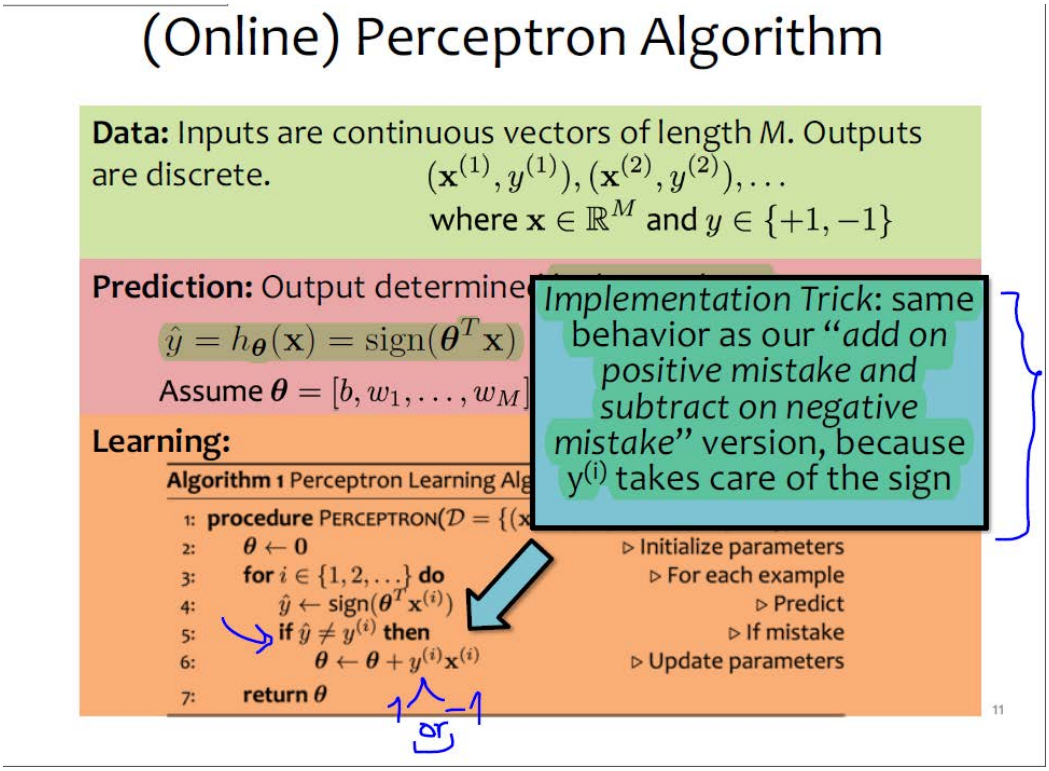
الدكتور عاوز يستعرض notation trick



دلوقت نقدر ن redefine our perceptron algorithm باستخدام التعريف اللي هو ..

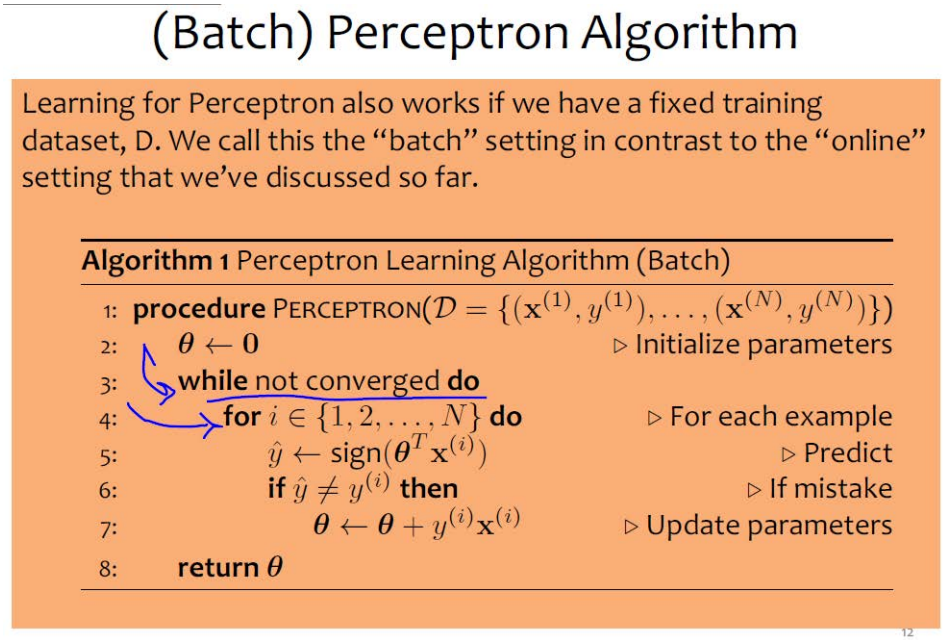


احنا عندنا شوية إكسات .. وعندنا شوية وايات ... الواي ديسكريت يا واد هه .. واحنا هنا هنفترض ان الoutput متحدد بهايبر بليين اللي هو sign(Theta transpose * x) .. ال learning هنا بقا اول حاجه حط البارمترز في الاول بصفر .. وبعدين ابدأ لف .. وانت بتلف بتبص علي البريدكشن بتاعك .. لو كان بوزيتيف . اجمع ال Xi مع الثيتا .. لو نيجاتيف اطرحه من الثيتا .. لإن دلوقت لما هتيجي تجمع الإكسات .. انت حرفيا بتجمع 1 علي ال b .. لإن أول entry من الإكس اللي عندك هو 1 .. ولما هتيجي تطرح فانت فعلياً هتطرح 1 من ال b ..



والصوره اللي فاتت ديه انضف طريقه ت implement بيها االل algorithm ده ..

دلوقت هنخش في ال batch perceptron algorithm .. وهنسيب ورا ضهرنا ال online learning setting بتاع ال perceptron .. وهنرجع لل setting بتاع لما كان عندنا fixed training set D .. ففي ال training time. ... ابعت كل الداتا سيت بتاعتك للي هي بتتكون من N training example .. نفس الألجورذم المستخدم هو هو بس انت هتفضل تعيد وتزيد تعيد وتزيد لحد ما ت converge ..



بس الدكتور هنا قال .. لازم تكون سألت نفسك سؤالين .. يعني ايه الألجورذم ي converge .. و هل الالجورذم ده هي converge اصلا .. بس قبل ما نجاوب هنعوز نعرف شوية حاجات و بعدين نرجع تاني ..

الدكتور راح اتكلم سريعاً علي ان ال Perceptron algorithm ممكن تعمله derivation عن طريق طريقتين ... اول واحده انك ت extend the online perceptron algorithm to the batch setting .. تاني طريقه انك ت apply SGD .. عشان تقلل ال hinge loss علي ال linear separator

في extensions of perceptron ..

Extensions of Perceptron

- **Voted Perceptron**
  - generalizes better than (standard) perceptron
  - memory intensive (keeps around every weight vector seen during training, so each one can vote)
- **Averaged Perceptron**
  - empirically similar performance to voted perceptron
  - can be implemented in a memory efficient way (running averages are efficient)
- **Kernel Perceptron**
  - Choose a kernel K(x', x)
  - Apply the **kernel trick** to Perceptron
  - Resulting algorithm is **still very simple**

linear & nonlinear DB

- **Structured Perceptron**
  - Basic idea can also be applied when **y** ranges over an exponentially large set
  - Mistake bound **does not** depend on the size of that set

14

Perceptron Exercises

Question:

*The parameter vector **w** learned by the Perceptron algorithm can be **written as a linear combination** of the feature vectors $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(N)}$.*

$w = \alpha_1 x^1 + \alpha_2 x^2 + \cdots + \alpha_N x^{(N)}$

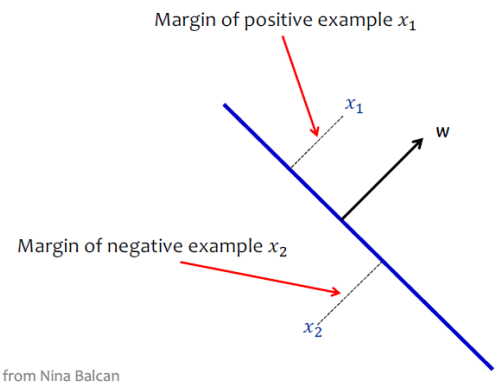
A. *True, if you replace "linear" with "polynomial" above*
B. *True, for all datasets*
C. *False, for all datasets*
D. *True, but only for certain datasets*
E. *False, but only for certain datasets*

15

دلوقت عاوزين نشوف ازاي الحاجات ديه بتشتغل .. فهنحتاج نحط شوية تعريفات كدا ... أول واحد هو ال geometric margin .. و هنا انت هتعرف 3 مارجنز .. أول واحد بيقول .. "The margin of an example with respect to a linear separator W is the distance from X to the plain" ... بمعني ان ال plane هو ال decision boundary. فلو عندك قيمه لل W فهي ليها ديسيجن باوندري .. المارجن هي المسافه من X1 لل plane .. والمسافه ديه بتعتمد علي ال W ..

Geometric Margin

Definition: The margin of example x w.r.t. a linear sep. w is the distance from x to the plane $w \cdot x = 0$ (or the negative if on wrong side)



Slide from Nina Balcan

تاني تعريف هو إن ال margin gamma w لشوية set of examples هنسميهم S .. with respect to a linear separator W .. هي ل smallest margin over all of the points .. فكّر فيها كالآتي .. انت عندك W اللي بتديك الديسيجن باوندري .. دلوقت تخيل انك بترسم 2 parallel lines علي الديسيجن باوندري بتاعك .. لحد ما يخبط في نقطه من النقط اللي عندك .. وأقف هنا واكتب المسافه اللي انت خدتها .. وهي ديه المارجن بتاعتك ..

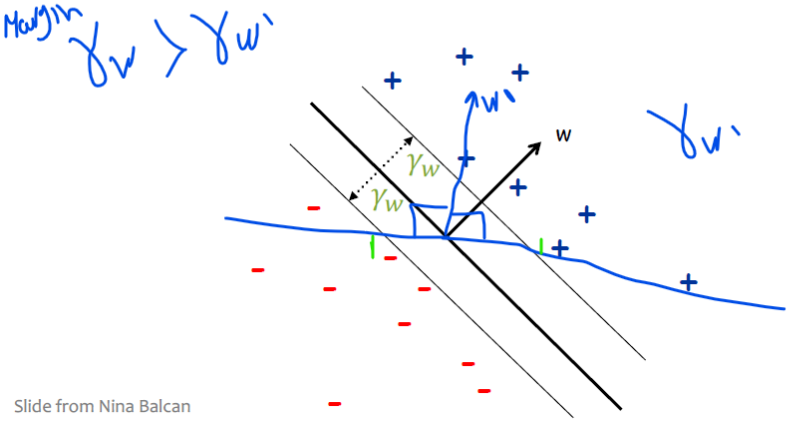
تالت تعريف .. المارجن جاما ل set of examples بالنسبه لل examples دول . هي الماكسمم Gamma W علي كل ال possible linear separators

تخيل عندك 2 linear separator ..

Geometric Margin

Definition: The margin of example x w.r.t. a linear sep. w is the distance from x to the plane $w \cdot x = 0$ (or the negative if on wrong side)

Definition: The margin γ_w of a set of examples S wrt a linear separator w is the smallest margin over points $x \in S$.



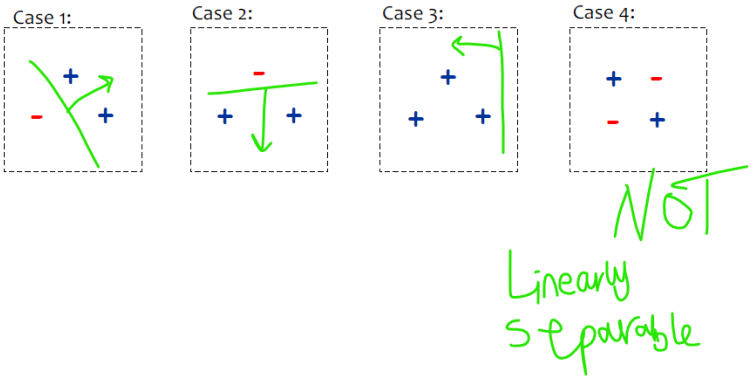
Slide from Nina Balcan

أكبر جاب ما بين ال + و ال - .. خد نص المسافه بتاعت المساحه الفاضيه ديه .. و هي ديه المارجن ..

هنحط تعريف كمان: في البايناري كلاسيفيكيشن بروبلم .. احنا بنقول a set of examples ... بيبقوا linearly separable لو في decision boundary يقدر يفصل النقط اللي عندنا ... مثلاً::

Linear Separability

Def: For a **binary classification** problem, a set of examples S is **linearly separable** if there exists a linear decision boundary that can separate the points



20

طيب تعال نتكلم علي ال perceptron mistake bound .. ده بيقول ان لو الداتا ليها a guarantee .. وكل النقط واقعه جوا كوره margin gamma نص قطرها R ... والكوره متسنتره عند ال origin .. في الحاله ديه ال perceptron هيعمل اقل من أو بيساوي R/gamma^2 غلطات .. ده بيقول ايه الكلام ده .. بيقول ان في خاصيه مهمه للألجورذم ده ... فأخيراً احنا هنا هنقدر نعرف ال converge هنا .. هنقول ان ال patch perceptron خلاص converged algorithm لما يبطل يعمل غلطات علي التريننج داتا .. perfectly classify the training data ... طب ايه اللي بيضمنهولك ال perceptron mistake bound .. بيضمنلك ان ل linearly separable dataset .. في الآخر ال perceptron algorithm هيبطل يعمل غلطات .. ففي الحاله ديه مش هي update the parameters again .. فهنا هنقول اننا converged خلاص ..

Analysis: Perceptron

Perceptron Mistake Bound

Guarantee: If data has margin γ and all points inside a ball of radius R, then Perceptron makes $\leq (R/\gamma)^2$ mistakes.

(Normalized margin: multiplying all points by 100, or dividing all points by 100, doesn't change the number of mistakes; algo is invariant to scaling.)

Def: We say that the (batch) perceptron algorithm has **converged** if it stops making mistakes on the training data (perfectly classifies the training data).

Main Takeaway: For **linearly separable** data, if the perceptron algorithm cycles repeatedly through the data, it will **converge** in a finite # of steps.

Slide adapted from Nina Balcan

22

تعال نكتب الكلام ده كرياضه ...

Analysis: Perceptron

Perceptron Mistake Bound

Theorem 0.1 (Block (1962), Novikoff (1962)).

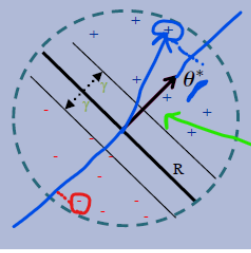
Given dataset: $\mathcal{D} = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{N}$.

Suppose:

1. Finite size inputs: $||x^{(i)}|| \leq R$
2. Linearly separable data: $\exists \boldsymbol{\theta}^*$ s.t. $||\boldsymbol{\theta}^*||_2 = 1$ and $y^{(i)}(\boldsymbol{\theta}^* \cdot \mathbf{x}^{(i)}) \geq \gamma, \forall i$

Then: The number of mistakes made by the Perceptron algorithm on this dataset is

$$k \leq (R/\gamma)^2$$



gure from Nina Balcan

23

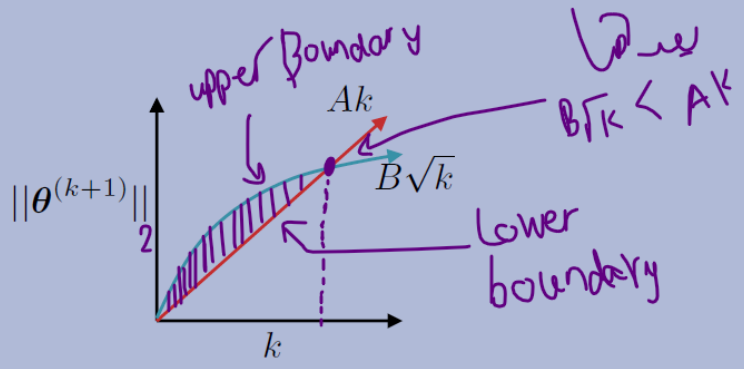
تعال نبص علي الاثبات .. بتاع ان ال perceptron mistake bound ... ال theta(k+1) ... البارمترز بعد ال kth mistake ... احنا هنجيب upper bound and lower bound .. لل L2 norm of the parameters after the kth mistake .. ال upper bound هيبقا قيمة B مضروب في ال sqrt k .. ايه بقا هم الكيرفين اللي عندك ..

Analysis: Perceptron

Proof of Perceptron Mistake Bound:

We will show that there exist constants A and B s.t.

$$Ak \leq ||\boldsymbol{\theta}^{(k+1)}|| \leq B\sqrt{k}$$



25

A هي السلوب بتاع الخط .. منعرفش ايه هي ال B .. بس هو هيبقا كيرف وخلاص يعني طالع اسرع من ال A .. بس بسبب ال sqrt هيبطأ و ي cross .. ال statement true بس في قيمه عند ال k هيحصل الكروس .. عندها ال statement isn't true .. فبالتالي مينفعش يكون في قيمه لل k تعدي الكروس بتاع الكيرفين .. فأكبر رقم لل k .. هو هيبقا الرقم اللي بيتقاطعو عندو ..

يلا علي الريجريشن بسرعه عشان زهقت .. الهدف واضح وبسيط .. عندك شوية داتا .. و الإكس هو فيكتور و الواي هو سكيلر .. اتعلم فانكشن اللي هي كيرف .. اللي هتبقا احسن حاجه ... خد بالك ان الواي هنا هي scalar .. مش integer .. قبل كدا الواي كانت في الكلاسيفيكيشن ديسكريت ... الواي دلوقت هي رقم حقيقي ... هنا ال h(x) مش هتبقا كلاسيفير عادي .. ده هيديك كيرف .. امثله كتيره الحقيقه .. انا نمت في آخر المحاضره .. فابقا اسمعها تاني . هه