

لو فاكرين المره اللي فانت اتكلما علي ال Gaussian mixture .. بس في limitation .. اننا بنفترض Gaussian distribution .. فاحنا انهارد هنتشيل اللمتيشين ده و هنروح لل Ilogistic regression .. ده generalization لل Gaussian .. وفي شركات كتيره بتستخدمو .. احنا هنتشوف استخدام بسيط لحاجه زي كذا قدام .. ده برضو مهم لما هنجي نتكلم علي ال NNs ..

المره اللي فانت اتكلما علي ال Mixture of Gaussians .. في الكونتكت بتاع الكلاسيكيشن .. الفكره ان عندنا ديفرنت كلاسيك .. ولكل كلاس عندنا داتا متسنتره في مكان واحد .. بس نتيجة ان في نويز في القرايات بتاعت السنسورز في نويز في الداتا فبالتالي الداتا بقت متوزعه علي منطقه و احنا هنا فارضين ان النويز ليها Gaussian distribution .. الدكتور رسم ديه:



ايه اللي هيجصل لما الداتا ميكونش ليها كلاسترز ليها الشكل بتاع الجواسين ده .. الداتا ممكن يكون ليها أي شكل .. هنعمل ايه في الحاله ديه ... ايه هو الدستريبيوشن اللي نقدر ناخده في الاعتبار غير جواسين .. في فاميلي أوف ديستريبيوشنز اسمها Exponential family .. في مطعم الكورسات هتلاقي فيها الدستريبيوشن ده .. وديه فيها جواسين و إكسبوننشال و برنولي و كاتيجوريكال .. و حاجات كتير .. العيله ديه بتقول ان ال  $p(x|\text{parameter } \theta)$  .. واسمها إكسبوننشال عشان ال pdf ليها إكسبوننشال .. الإكسبوننت نفسو بيبقا  $T(x) * \theta$  و اطرح منو ترم  $A(\theta)$  و اجمع عليهم ترم  $B(x)$  ...

## Exponential Family

- More generally, when  $\Pr(x|c_k)$  are members of the exponential family (e.g., Gaussian, exponential, Bernoulli, categorical, Poisson, Beta, Dirichlet, Gamma, etc.)

$$\Pr(x|\theta_k) = \exp(\theta_k^T T(x) - A(\theta_k) + B(x))$$

where  $\theta_k$ : parameters of class  $k$

$T(x), A(\theta_k), B(x)$ : arbitrary fns of the inputs and params

- the posterior is a sigmoid logistic linear function in  $x$

$$\Pr(c_k|x) = \sigma(w^T x + w_0)$$

دلوقت اللي بيعمل الفرق ما بين جواسين و برنولي و جاما .. هو ال  $T(x), A(\theta), B(x)$  .. هم دول ... كل الدستريبيوشنز في العيله ديه بيتكتبو بنفس الفورميلا وبيبقا الاختلاف في ال 3 فانكشنز اللي فوق .. وعشان كلهم ليهم نفس الفورم ... ال posterior برضو بتبقا sigmoid .. وده الجمال كلو .. فمثلا في Mixture of Gaussian .. بنبدأ بال prior و ده كان عباره عن categorical distribution .. و بعدين عندك likelihood وده class conditional distribution of Gaussian .. وبعدين ال posterior عشان تعمل prediction عن الكلاس اللي النقطه بتتنتمي اليها .. هيبقا برضو sigmoid logistic function ... أي distribution ليها نفس الفورم ديه .. علي طول ال Posterior هتبقا a sigmoid logistic function بتبقا linear في قيمة الإكس ..

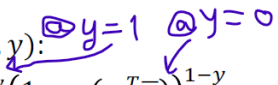
بناءً على كذا هن formulate حاجه اسمها probabilistic discriminative models .. وهنا الفكره هي ان بما إنك عارف ان ال posterior distribution هي بتبقا على هيئة logistic function فبدل ما تعمل assumption ان الداتا بتتعرض لنويز و التويز ديه بيبقا ليها specific distribution وبعدين تحاول ت estimate البارامترز بتاعت الديستريبيوشن ده و لما تيجي تحسب ال posterior تروح على طول لل logistic function وديه بتعتمد بشكل اساسي على ال W and W0 .. فالسؤال هنا هيبقا هل هنقدر ن estimate W and W0 على طول .. بدل ما كنا في ال Mixture of Gaussian نعمل فينتنج لل Mixture of gaussians عن طريق اننا ن learn the mean and covariance matrix of each Gaussian وبعدين من هناك تحسب ال posterior اللي بتخليك تقدر تقول ال W and W0 بيقوا ايه in terms of the mean and the covariance matrix ..... بس ممكن احنا منحتاجش كل الخطوات ديه اصلاً .. ونقدر نروح على طول اننا ن estimate W and W0 عشان احنا عارفين كذا ان ال posterior هيبقا ليها الفورم بتاعت السجمويد .. فده فعلاً اللي هنعملو .. هنا هنعمل كذا عن طريق اننا نعامل ال  $p(C|X)$  انها هي ال likelihood بتاعت ال classes وبعدين ن optimize البارامترز عشان ن maximize الكلاس ده . بشكل عام احنا عارفين ان في البايرني كلاسيفيكيشن هتبقا logistic sigmoid لو عندك أكثر من كلاس هتبقا softmax .. وده هو ال posterior

## Probabilistic Discriminative Models

- Instead of learning  $\Pr(c_k)$  and  $\Pr(x|c_k)$  by maximum likelihood and finding  $\Pr(c_k|x)$  by Bayesian inference, why not learn  $\Pr(c_k|x)$  directly by maximum likelihood?
- We know the general form of  $\Pr(c_k|x)$ :
  - **Logistic sigmoid** (binary classification)
  - **Softmax** (general classification)

طيب فيالتالي ده هودينا للتكنيك اللي اسمو لوجستيك ريجريشن .. عشان لكل الديستريبيوشنز اللي موجوده عندنا ليها لوجستيك سجمويد فاحنا بشكل أساسي هنحاول نعمل فينتنج او اننا نلاقي ايه هي احسن logistic function اللي بت capture the posterior للداتا اللي عندنا .. فلما هنموديل اليوس تريور .. ال Modeling هنا بيبقا  $x$  and  $y$  in terms of .. عشان الهدف من ال posterior انو يحسب ال probability that a point belongs to a specific class  $y$  .. لو الكلاس ده كان يا صفر يا واحد ... فاحنا نقدر نستخدم expression زي الي ف السلايدز عشان نتكلم على ال posterior ... و ال expression ده لل posterior بتاع أي كلاس ..

## Logistic Regression

- Consider a single data point  $(x, y)$ : 

$$w^* = \operatorname{argmax}_w \sigma(w^T \bar{x})^y (1 - \sigma(w^T \bar{x}))^{1-y}$$
- Similarly, for an entire dataset  $(X, y)$ :
 
$$w^* = \operatorname{argmax}_w \prod_n \sigma(w^T \bar{x}_n)^{y_n} (1 - \sigma(w^T \bar{x}_n))^{1-y_n}$$

Objective: negative log likelihood (minimization)

$$L(w) = -\sum_n y_n \ln \sigma(w^T \bar{x}_n) + (1 - y_n) \ln(1 - \sigma(w^T \bar{x}_n))$$

Tip:  $\frac{\partial \sigma(a)}{\partial a} = \sigma(a)(1 - \sigma(a))$

ودلوقت عاوزين نجيب أحسن  $W$  تقدر ت maximize ال posterior .. عندك شوية داتا فيها الإنبوت إكس و الأوتبوت واي .. وانت عاوز تلاقي ال  $W$  اللي تخليك متأكد إن لمعظم الداتا بوينتر ان ال probability of the correct class  $y$  هيبقا اعلي حاجه عندك .. ال  $W$  هتلاقي بتوديك ناحية expression و ده بيكون جواه البارمترز بتاعت ال underlying distribution زي جاوسين مثلا او جاما او دستريبيوشن من ضمن الإكسبوننشال فاملي .. بس دلوقت انت مش مهتم ايه هو الدستريبيوشن اللي موجوده فيه الداتا .. احنا بس مجرد هنعمل فنتنتج  $W$  بشكل مباشر بناءً علي إن في underlying distribution from the exponential family .. فهنا احنا عندنا ال expression لداتا بوينت واحد .. " في السلايد يعني " .. المعادله اللي بعدها هو انك بقيت طالع علي داتا سبت .. الفرق ما بين الاتنين انك هيبقا عندك حاصل ضرب لل posteriors لكل الداتا بوينتس دول .. احنا بس هنعاول ن optimize ال posterior عشان الاقي احسن  $W$  .. المشكله ان النوع ده من ال optimization مش بيتم بشكل مباشر عشان انت عندك عملية ضرب .. فهنستخدم نفس الفكرة اللي علي طول بنعملها هو اننا ناخد ال log بتاع الأوبجكتيف ده مش بيبغير مكان ال maximum بس بيحول ال product summation وبعدين ه apply اللوج للترمات اللي عندي و ده هيدين الإكسبرشن الأخير في السلايد .. هنعاول بس الماكس للمنييم عشان بنفضل اننا نشغل علي minimization problem .. كذا احنا هنمشي علي نفس الخطوات بتاعت الموديلز قبل كذا .. نحسب المشتقه ونساويها بالصفر .. لما بتيجي تحسب المشتقه لازم تاخد بالك ان المشتقه بتاعت السجمويد نفسها سهله .. إكسبرشن بسيط يعني .. السجمويد في واحد ماينص السجمويد .. استخدمها يعني في حساباتك .. دلوقت الهدف بتاعك انك تحل ال optimization problem عشان تلاقي ال  $W$  اللي بتقال ال objective function بتاعتك ... خلينا بس نوضح حاجه الأول قبل ما نكمل .. التكنيك ده اسمو logistic regression بس فعلاً اللي انت بتحاول تعملو انك تعمل classification .. اللي هو لما جينا نتكلم في أول الكورس قلنا ان منين ما بيكون عندنا كلاسيكز بتقا categorical في الحاله ديه بتبقا classification انما لما بتحاول ت predict حاجه بتبقا numerical هي ديه الرجريشن .. هنا ال logistic regression هو تكنيك applied to a classification problem .. فده بيبقا ليه سبب ورا الكلام ده .. انك لما بتفكر في اللي انت بتعملو فعلاً هو نوع من انواع الرجريشن عشان في الآخر انت بتحاول انك ت estimate the posterior اللي هي  $p(C|X)$  وديه نفسها اصلاً رقم ما بين الصفر والواحد .. فده عشان كذا ده بيبقا awkward شويه انو رجريشن متطبق علي كلاسيفيكيشن .. بس الحقيقه انك حولت الكلاسيفيكيشن للريجريشن عن طريق انك بتقول انا هحاول predict البروبابلي لكل كلاس و البروبابلي مجرد رقم فبالتالي كذا بقت رجريشن ..

## Maximum likelihood

- Convex loss: set derivative to 0

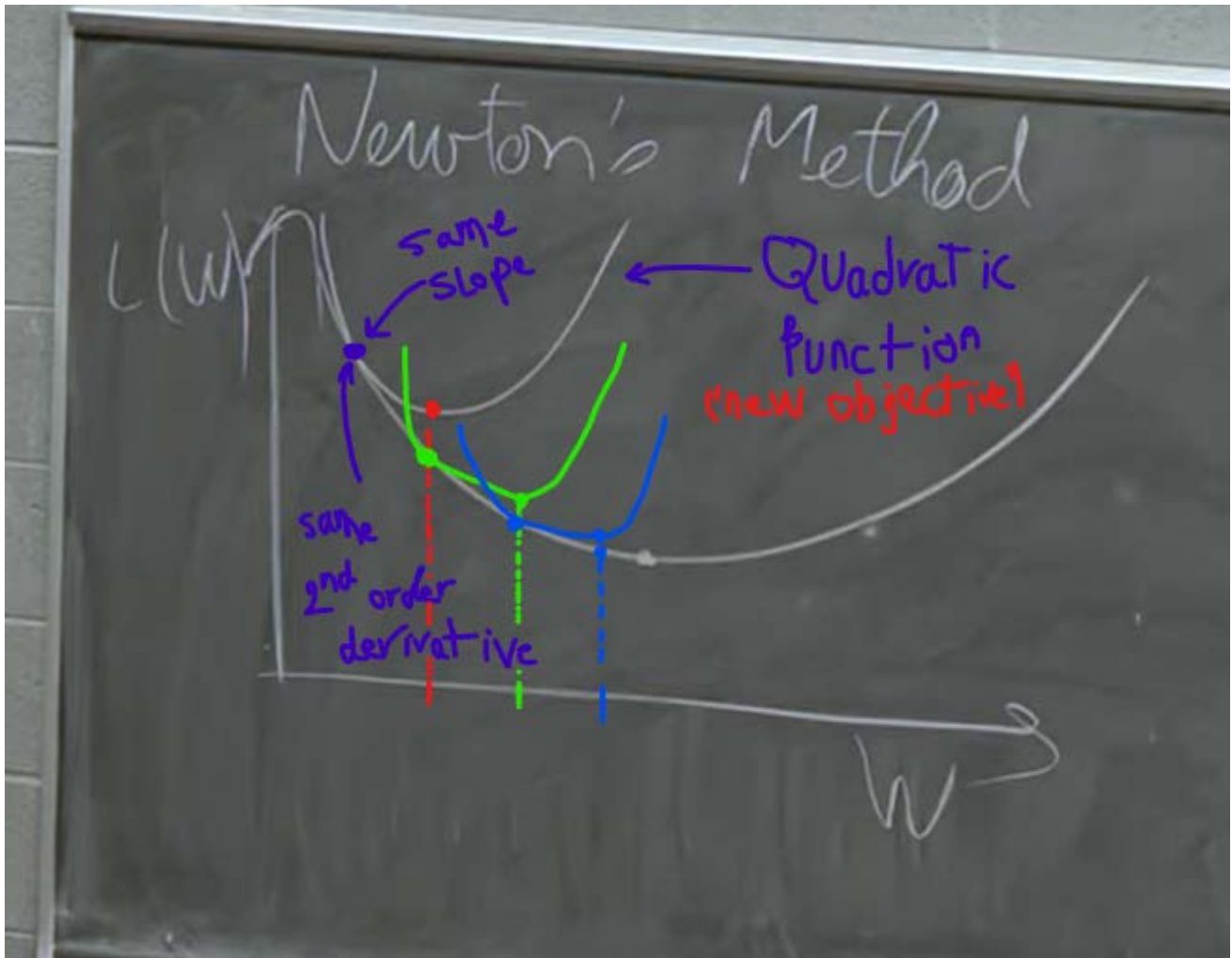
$$0 = \frac{\partial L}{\partial w} = - \sum_n y_n \frac{\cancel{\sigma(w^T \bar{x}_n)} (1 - \sigma(w^T \bar{x}_n)) \bar{x}_n}{\cancel{\sigma(w^T \bar{x}_n)}} - \sum_n (1 - y_n) \frac{(1 - \cancel{\sigma(w^T \bar{x}_n)}) \sigma(w^T \bar{x}_n) (-\bar{x}_n)}{1 - \cancel{\sigma(w^T \bar{x}_n)}}$$

$$\Rightarrow 0 = - \sum_n y_n \bar{x}_n - \sum_n y_n \cancel{\sigma(w^T \bar{x}_n)} \bar{x}_n + \sum_n \sigma(w^T \bar{x}_n) \bar{x}_n + \sum_n y_n \cancel{\sigma(w^T \bar{x}_n)} \bar{x}_n$$

$$\Rightarrow 0 = \sum_n [\sigma(w^T \bar{x}_n) - y_n] \bar{x}_n$$

- Sigmoid prevents us from isolating  $w$ , so we use an iterative method instead

كدا اشتقيت و ساويت بالصفر .. شلنا شوية ترمات طيرناهم .. ووصلت لآخر اكسبرشن عندي ... اللي انا عاوز اعملو من البدايه اني اجيب  $W$  .. إني اعزل ال  $W$  من المعادله .. عشان اعرف اجيلو closed form expression .. بس هنا صعب تخزج ال  $W$  برا ال summation و برا ال sigmoid ... هتبقا صعبه .. السجمويد هي اللي عملا لك أزمه يعني هنا .. تعال نرسم كيرف يوضحلنا ال objective عبارته عن ايه وبعدها هنروح ل iterative method نعرف نتعامل بيها مع المعادله ديه .. اللي اترسم هو convex function .. لو جيت تستخدم ال Gradient descent .. الدكتور بيقول ان الطريقه ديه مش بتبقا إيفشنت و بيبقا في مشكله في حجم ال step .. تعال نشوف طريقه ثانيه .. اللي هي Newton's method .. منين ما يبقا عندك كونيبيكس أوبجكتيف .. تقدر تستخدمها .. وبتبقا اسرع من ال GD و عدد خطوات اقل .. و انت مش محتاج فيها step length .. بتعمل iterative reweighted least square .. الدكتور هيرسم حاجه ع الصوره .. هنا بدل ما بنعمل fitting the line .. فهنا اللي ممكن اعملو انك تبدأ ت fit a quadratic function .. فهي ممكن فعلاً ت approximate الكيرف بشكل احسن ..



الدكتور يقول هنا ليه منفكرش في ال quadratic function كإنها هي ال new objective وتعال ن minimize الفانكشن ديه .. فانت هتجيب ال bottom point وخذها هي النقطة الجديد .. هتديك W .. وبعدين هتجيبك نقطه علي الكيرف الأصلي ارسم فانكشن تانيه من عند النقطة ديه .. و خذ المنيم بتاعها .. وبعدين هيجيبك W جديد .. و تفضل تلف وتلف وتلف .. انت هنا معندكش ستيتب لينث .. انت منين ما يبقا عندك quadratic function انا حل optimal for the minimum in closed form .. وهنا انك تحل ل what is the minimum for the quadratic function هو هو اللي كنا بنعملو في ال linear regression لما كنا بن minimize squared loss .. ده كان quadratic objective و هو كان convex ف كان هو هو المشكله ديه .. فاللي احنا بنقولو ... عند كل خطوه احنا بن approximate a function with a quadratic ... حولها ال linear regression problem عشان تلاقي ال minimum بتاع الكوادراتيك كيرف ده و يدبك estimate وبعدين ت refit the function و تفضل تتحرك .. دلوقت لما بن update W عن طريق اننا نطرح ال H-1 مضروب في ال gradient of the loss function .. اللي انت بتعملو فعلياً هو انك بتخذ خطوه علي الكيرف انك تحل quadratic approximation and finding the minimum



دلوقت هنشوف ازاى نحسب ال hessian .. هي عبارة عن second order derivatives .. بس ايه اسهل طريقه نحسب بيها ال hessian في الكونتكست بتاع ال Logistic regression .. هو مجرد  $\bar{X} * R * \bar{X}^T$  ... الإكس بار .. هو الداتا سيت .. أول روو هو روو أوف 1 .. فكل كولوم هو داتا بوينت .. فمنيين ما تحط ال bar notation .. انت بتحط الواحد علي الداتا بوينتس يعني .. وال R هي diagonal matrix أول entry فيها هو ال  $\sigma(w^T \bar{x})$  و هكذا ...

## Hessian

$$\begin{aligned}
 H &= \nabla(\nabla L(w)) \\
 &= \sum_{n=1}^N \sigma(w^T \bar{x}_n)(1 - \sigma(w^T \bar{x}_n)) \bar{x}_n \bar{x}_n^T \\
 &= \bar{X} R \bar{X}^T
 \end{aligned}$$

Dataset ↗

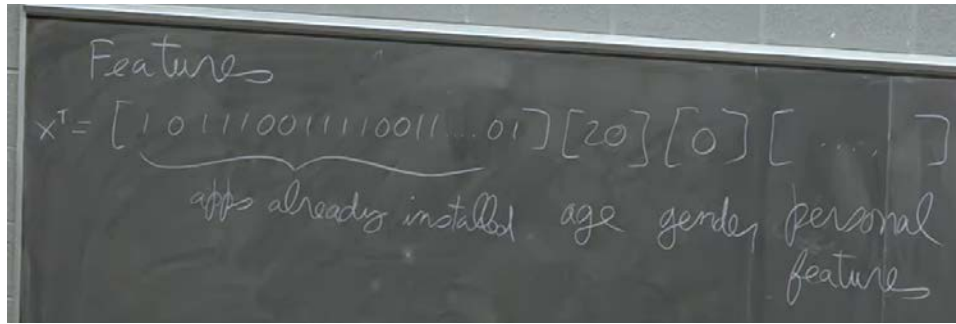
where  $\underline{R} = \begin{bmatrix} \sigma_1(1 - \sigma_1) & & \\ & \ddots & \\ & & \sigma_N(1 - \sigma_N) \end{bmatrix}$

and  $\sigma_1 = \sigma(w^T \bar{x}_1), \quad \sigma_N = \sigma(w^T \bar{x}_N)$

## Case study

- Applications: recommender systems, ad placement
- Used by all major companies
- Advantages: logistic regression is **simple, flexible and efficient**

لو احنا عملنا ريكومنديشن لأبب عشان نحملو مثلاً .... خد في الاعتبار features كتيره عنك .. وبيعلمو التوقعات بناءً علي كذا ... تعال نشوف مثال عندنا binary features طويل عريض كذا .. مالوش آخر .. عبارة عن واحد و اصفار .. كل entry في الفيكتر ده بيعبر عن apps already installed ... ديه اللي انت الريدي حملتها ... لما بتعمل recommendation بتبقا مهتم بالحاجات ديه ... 1 بيقا حملت وسطبت .. صفر بيقا محملتش .. الفيكتر ده هيبقا طوله مليون ع الاقل .. لإن عدد الأبلكيشنز كتير جداً .. هيبقا كمان عندك فيتشر اللي age و فيتشر gender .. وشوية personal features كذا .. " عندكو حمام مشوي .. اغرف يا راجل اغرف" ...



أول ما يبقا معنا الفيتشرز بالطريقه ديه .. احنا هنعوز نعمل كلاسيفيكيشن .. احنا خلاص عملنا التريننج يعني بتاع اللوجستيك ريجريشن موديل .. موبايلك اهو هيعمل ريكومنديشن .. احنا عاوزين نعرف لما نرشدك الأبلكيشن ده اليوزر هيعملو فعلا ولا لا ... لو عندك 2 كلاسيز .. يا واحد يا صفر .. فيتبص علي الكونديشن بتاع السجمويد لو كان اكبر من أو بيساوي ال 0.5 .. في الحاله ديه هتبقا 1 غير كدا هتبقا 0 ... وده شفافه في mixture of Gaussians ... linear separator هناك .. وده بيكافئ انك تقول ان ال  $w^T x$  أكبر من أو بيساوي الصفر ولا لا ... لو بصلتها من علي الإكس أكسيز يعني بدل الواي أكسيز ..

## App Recommendation

- Flexibility: millions of features (binary & numerical)

– Examples:

- Efficiency: classification by dot products

Multiple classes:

$$c^* = \operatorname{argmax}_k \frac{\exp(w_k^T \bar{x})}{\sum_{k'} \exp(w_{k'}^T \bar{x})}$$

$$= \operatorname{argmax}_k w_k^T \bar{x}$$

Two classes:

$$c^* = \begin{cases} 1 & \sigma(w^T \bar{x}) \geq 0.5 \\ 0 & \text{otherwise.} \end{cases}$$

$$c^* = \begin{cases} 1 & w^T \bar{x} \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

– Sparsity:

– Parallelization:

لما بيكون عندك multiclass classification problem .. في الحاله ديه الحسابات هتستخدم فيها softmax ... الفرق ان بيبقا عندك فيكتور  $w$  لكل كلاس  $K$  .. هيبقا عندك ال  $\operatorname{argmax} w^T x$  .... احنا هنا بنستخدم one vector  $w$  for each class  $K$  .. في 2 كلاسيز .. بنستخدم a single  $w$  vector for all .. طيب ده الموضوع بسيط .. بس تقدر كمان تخلي الموضوع ده efficient عن طريق انك تبص علي حاجتين ال sparsity and parallelization ..

اول حاجه ال sparsity .. منين ما بتحسب الدوت برودكت انت بس هدفك انك تضرب ال non-zero entries و فكك من ال zero entries .. لو جينا نبص علي الفيتشرز اللي الدكتور كان حاططها ع الصبوره .. اللي هي في الصوره فوق ... معظم الناس عندها 20 أبلكيشن بس مثلاً يعني .. فبالتالي الفيتشر فيكتور اللي كان حجمو في رينج المليون أبلكيشن .. فده معناه ان الفيكتور ده هيبقا sparse عشان معظم ال entries اللي فيه هتبقا باصفار .. انت كمان تقدر ت parallelize ال computation .. فتقدر تستخدم GPU عشان تعمل elementwise products .. فده مش محتاج ريسورسز كتيره ..

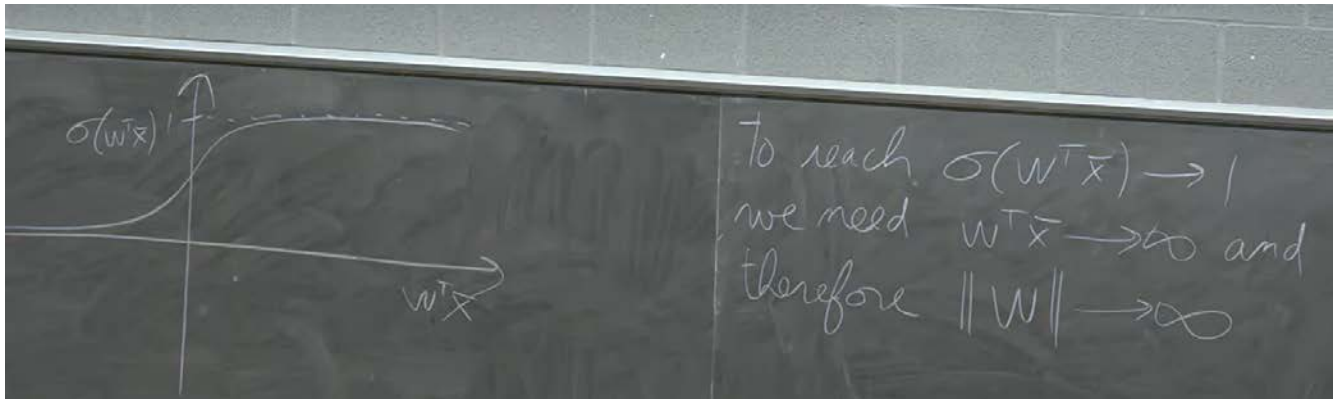
تعال نشوف المشاكل ... اللوجستيك ريجريشن .. بيشغل كويس بس ممكن ي overfit بسهولة .. جزء من المشكله عشان اللوجستيك ريجريشن الأوبتيمائيزيشن فيه هو كونفكيس .. فسهل انك تلاقي ال global optimum .. فبالتالي الجلوبال أوبتيمم بيقدر ي fit the data بشكل جامد جداً .. فهو عرضه لل overfitting .. لما بيبقا عندنا overfitting مع اللوجستيك ريجريشن اللي بيحصل ان عند نقطه ما .. عندنا كلاس هو الكلاس الصح . ولما هعوز أ fit الداتا للكلاس ده بشكل جامد جداً .. فده معناه انو هيحاول يخلي ال probability of the class close to 1 .. ولما بتعمل كدا .. الويتس نفسها هتبقا كبيره جداً .. ممكن توصل لإنفنتي وبالتالي ال hessian is singular .. وبالتالي مش هتعرف تاخذلها الإنفرس وبالتالي مش هتعرف تطبق .. newton's



تعال نرسم صوره عشان نفهم اللي بيحصل ده ازاي ... الدكتور رسم السجمويد .. و الإكس أكسيس هو ال  $wTx$  ... انت عندك شوية داتا وانت عارف ايه هو ال correct class فلو انت بتأوفر فينتنج .. انت هتحاول تخلي ال  $p(\text{class})$  اقرب حاجه للواحد .. فلو احنا بنحاول نخلي البروبابلتي اقرب للواحد .. فهتلاقى انك رحت لل 1 asymptotically .. فبالتالي هتلاقى ان عندك  $wTx$  كبيره جداً ..

عشان توصل لل 1  $\rightarrow \text{sigmoid}(wTx)$

انت محتاج  $wTx$  يوصل لإنفتني فبالتالي ال  $\|w\|$  هيوصل هو كمان لإنفتني



فعشان توصل للاحتمالية الواحد .. الإكس بت correspond to our features .. ال  $W$  هي اللي هتكتب فعلاً يعني ... طب بالنسبه لل hessian .. هي ال  $X \bar * R * X \bar.T$  .. والإكس بار هنا هي ال  $\text{sigmoid} * (1 - \text{sigmoid})$  علي طول ال diagonal ... بما إن السجمويد هتوصل للواحد ... هتلاقى الداياجونال رايح ناحية الصفر بسبب ال  $1 - \text{sigmoid}$  .. فهيتبند ناحية الصفر فبالتالي ال hessian هي tend to zero .. وديه هتبقا مشكله في ال hessian .. لو ده حصل اعرف ان بيحصل أوفر فينتنج .. ايه اللي نقدر نعملو لو حصل أوفر فينتنج في الكونتكتست ده .. الحل كالعاده هو ال regularization ... حط penalty علي ال weights الكبيره .. فالأوبجكتيف الجديد هو هو اللي فات بس حط عليه الترم الجديد ده .. ال hessian هتبقا هي هي بس ضيف عليها  $\lambda$  .. ده الفرق اللي هيساعد حوار الأوفر فينتنج .. هيساعد ال singularity ..

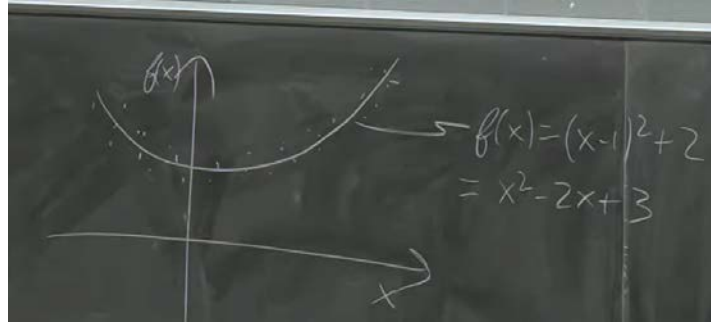
اللي عملناه لحد دلوقت اننا عملنا generalization ل mixtures of gaussians ل logistic regression .. اللي هو احنا خلاص مبقناش نفترض انو جاوسين ديستريبوشن .. بس بقا في ليميتيشن ان بقا عندنا كلاسيفاير الباوندري بتاعتو لينير .. لو 2 كلاسيز .. الباوندري هتبقا  $wTx$  .. حتي لو عندنا multi-class .. بس في معظم المشاكل بنحتاج non-linear boundary .. هل نقدر ن generalize الكلام ده ل non-linear separators ؟ ....

## Generalized Linear Models

- How can we do non-linear regression and classification while using the same machinery?
- Idea: map inputs to a different space and do linear regression/classification in that space

لو هتعوز حاجه non-linear هتحتاج انك تعقد الدنيا شوية في الماث .. وده هيوطينا ناحية ال generalized linear models .. فانت هت map الإنبوتس ل new space .. ومن هنا هتعمل linear regression and classification in that space ... الإنبوتس اللي هي الإكسات يعني .. ده هيطلعلك حاجه non-linear و المابنج ده ببينا non-linear mapping .. في ال new space بعمل linear regression .. فلو طلعلعنا خط في الدومين الجديد .. ده هيمثل ال non-linear curve في ال non-linear domain ..

تعال نرسم صوره .. تعال نبص علي quadratic function ... عندنا شوية داتا .. مينفعش احط لاین عشان هي شبه كيرف .. فالترو أندرلاينج فانكشن هو هيبقا مثلاً  $f(x) = (x+1)^2 + 2$  ... quadratic .. مثلاً يعني .. ازاي أقدر أ approximate الفانكشن ديه باستخدام ال non linear regression ..



فهي ديه الفكره mapping ...

## Basis functions

- Use non-linear basis functions:
  - Let  $\phi_i$  denote a basis function
 
$$\begin{aligned}\phi_0(x) &= 1 \\ \phi_1(x) &= x \\ \phi_2(x) &= x^2\end{aligned}$$
  - Let the hypothesis space  $H$  be
 
$$H = \{x \rightarrow w_0\phi_0(x) + w_1\phi_1(x) + w_2\phi_2(x) | w_i \in \mathbb{R}\}$$
- If the basis functions are non-linear in  $x$ , then a non-linear hypothesis can still be found by linear regression

شوية الفايات هي الإنيوتس دلوقت .. وبعدين نلاقي أحسن فانكشن لل hypothesis space ...

## Common basis functions

- Polynomial:  $\phi_j(x) = x^j$
- Gaussian:  $\phi_j(x) = e^{-\frac{(x-\mu_j)^2}{2s^2}}$
- Sigmoid:  $\phi_j(x) = \sigma\left(\frac{x-\mu_j}{s}\right)$   
 where  $\sigma(a) = \frac{1}{1+e^{-a}}$
- Also Fourier basis functions, wavelets, etc.