

التعلم العميق Deep Learning

○ الدرس الأول :	التعلم العميق و الشبكات العصبية
• الأسبوع الأول :	مقدمة للتعلم العميق
• الأسبوع الثاني :	أساسيات الشبكات العصبية
• الأسبوع الثالث :	الشبكات العصبية المجوفة
• الأسبوع الرابع :	الشبكات العصبية العميقة
○ الدرس الثاني :	تطوير الشبكات العميقة : المعاملات العليا
• الأسبوع الأول :	السمات العملية للتعلم العميق
• الأسبوع الثاني :	الحصول علي القيم المثالية
• الأسبوع الثالث :	ضبط قيم الشبكات العميقة
○ الدرس الثالث :	هيكلية مشاريع الـ ML
• الأسبوع الأول :	استراتيجيات الـ ML - 1
• الأسبوع الثاني :	استراتيجيات الـ ML - 2
○ الدرس الرابع :	الشبكات العصبية الملتفة CNN
• الأسبوع الأول :	أساسيات الشبكات العصبية الملتفة
• الأسبوع الثاني :	حالات عملية من الشبكات العصبية الملتفة
• الأسبوع الثالث :	التعرف علي الأشياء
• الأسبوع الرابع :	التعرف علي الوجه
○ الدرس الخامس :	الشبكات العصبية المتكررة RNN
• الأسبوع الأول :	مفهوم الشبكات العصبية المتكررة
• الأسبوع الثاني :	المعالجة اللغوية الطبيعية NLP
• الأسبوع الثالث :	نماذج التتابع

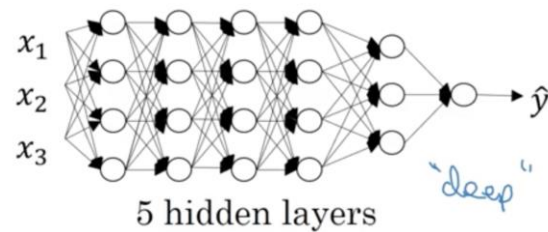
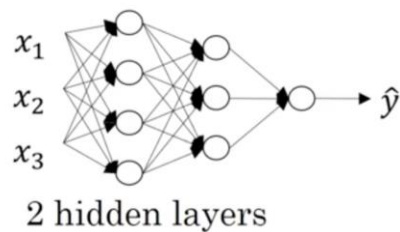
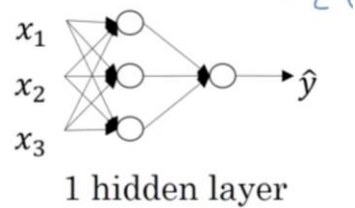
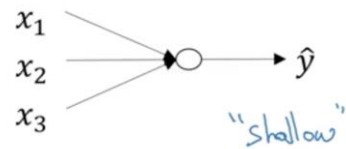
الأسبوع الرابع : الشبكات العصبية العميقة

- رؤية الشبكات العميقة علي انها متسلسلة من الشبكات العصبية مرتبطة ببعض
- بناء و تدريب شبكة عميقة من عدة طبقات
- تحليل مصفوفة الشبكات العميقة للتأكد من عملها كما ينبغي
- فهم كيفية استخدام الكاش لتمرير بيانات من المسار الامامي و الخلفي
- فهم دور الهابير برايمتر في التعلم العميق

* * * * *

ما هي الشبكات المجوفة و العميقة ؟

الشبكات المجوفة هي التي لا يكون بها اي طبقات خفية , او طبقة خفية واحدة (لا تنس اننا لا نحسب طبقة المدخلات) مثل هذه



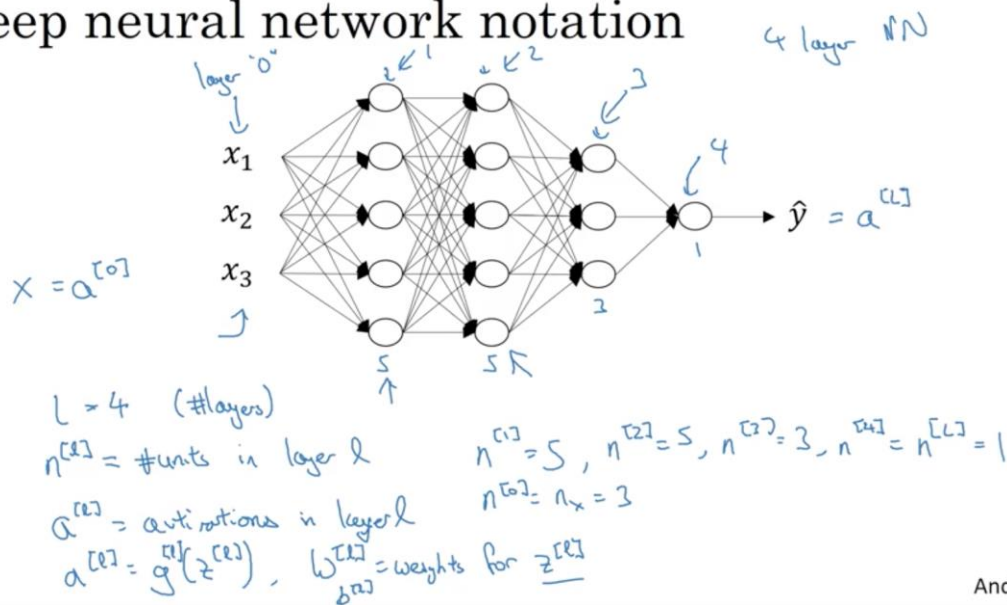
Andrew Ng

اما العميقة هي التي تحتوي علي عدد من الطبقات الخفية

وخلينا نتعرف علي عدد من الرموز الخاصة بيها :

لو الشبكة العميقة عندنا كدة :

Deep neural network notation



فأولا حرف L يشير لعدد الطبقات

حرف n فوقه رقم , يشير لعدد الخلايا في كل طبقة بالرقم اللي فوقه

يعني n1 يساوي 5 (اول طبقة هي اول طبقة خفية لان المدخلات مش طبقة) , n4 يساوي 1 (المخرج) , والمدخلات ممكن نقول عليها n0

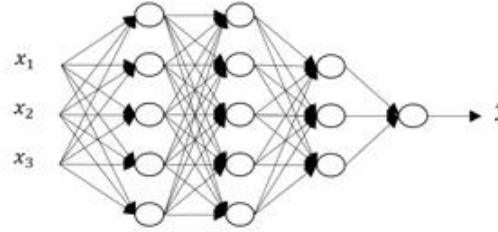
قيمة الاكتيفاشن a كمان بيكون ليها رقم , حسب الطبقة , و تعتبر هي دالة g داخل فيها z بنفس الطبقة

كمان بيكون فيه مصفوفة b , بنفس رقم الطبقة

اخيرا الاكسات ممكن نقول عليها a_0

*_**

طيب ماذا عن معادلة المسار الامامي في الشبكات العميقة :
لو نظرنا للشبكة العميقة ديه :



هناك ان قيمة z تساوي حاصل ضرب w_1 في x زائد b

$$x : z^{[1]} = \underline{w}^{[1]} x + \underline{b}^{[1]}$$

بينما الاكتيفاشن a تساوي

$$\underline{a}^{[1]} = \underline{g}^{[1]}(z^{[1]})$$

ونفس الفكرة لـ z لباقي الطبقات , وهتكون الصيغة العامة

$$Z^{[l]} = W^{[l]} a^{[l-1]} + b^{[l]}$$

$$a^{[l]} = g^{[l]}(Z^{[l]})$$

مع تذكر ان واي النهائية (واي هات) هتساوي $g(z)$ زي النهائية

ولو هنتعامل مع كل عناصر العينة , فهنخلي كلا من Z و A حروف كايبتال عشان تشمل كل z و a لكل عنصر من العناصر , فالمعادلة هتكون

$$X = A^{[0]}$$

$$Z^{[1]} = W^{[1]}X + b^{[l]}$$

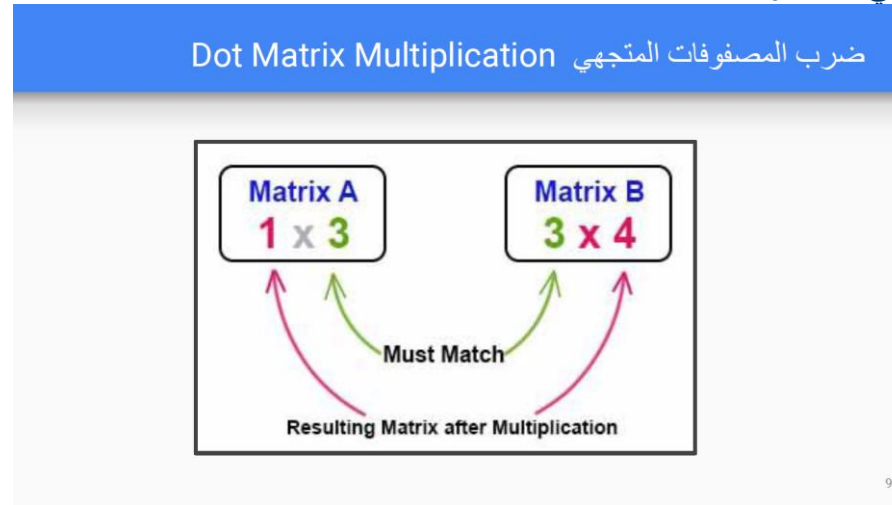
$$A^{[1]} = g^{[1]}(Z^{[1]})$$

* * * * *

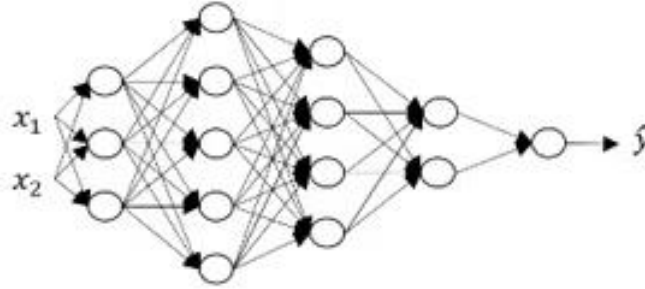
أحد الأدوات اللي بتأكدلنا ان الشغل سليم , اننا نتأكد من ابعاد المصفوفات المستخدمة في الجمع و الضرب , التأكد من ضبط الابعاد , بيأكدلنا ان العناصر مترتبة بشكل سليم

مننشاش ان جمع او طرح المصفوفات لازم تكون الابعاد متطابقة , في ضرب المصفوفات لازم عمود المصفوفة الاولى يتساوي مع صف الثانية

و ان وقتها بيكون الناتج عدد صفوف الاولى في عدد عواميد الثانية



لو هنطبق ده علي الشبكة العميقة عندنا هنا



هنمسك اول طبقة , وهتكون معادلة قيمة الـ z_1 كدة :

$$z^{[1]} = \boxed{w^{[1]} \cdot x} + \boxed{b^{[1]}}$$

مصفوفة الـ z_1 (وهي نواتج الطبقة الاولى) هتكون 3 في 1 , بينما مصفوفة الـ x 2 في 1 , يعني كدة :

$$3 \times 1 = W * 2 \times 1$$

وده لازم يخلي مصفوفة الـ w (اللي احنا منعرفهاش و عايزين نعرف ابعادها من الطريقة هنا) لازم هتكون 3×2 , والـ b لازم تكون بنفس ابعاد z

$$z^{[l]} = \underbrace{w^{[l]} \cdot x}_{(3,1) \leftarrow (3,2) \quad (2,1)} + \underbrace{b^{[l]}}_{(3,1)} \\ \underbrace{(n^{[l]}, 1)}_{(n^{[l]}, n^{[l-1]})} \quad \underbrace{(n^{[l]}, 1)}_{(n^{[l]}, 1)} \\ \begin{bmatrix} \vdots \\ \cdot \end{bmatrix} = \begin{bmatrix} \vdots \\ \vdots \end{bmatrix} \begin{bmatrix} \cdot \end{bmatrix}$$

و بالرموز هتكون :

$$n1 \times 1 = n1 \times n0 * n0 \times 1$$

و b هتكون

$$n1 \times 1$$

وبالتالي دائما w و b بيكونو

$$W^{[l]} : (n^{[l]}, n^{[l-1]})$$

$$b^{[l]} : (n^{[l]}, 1)$$

متكلمناش عن a ..

متنساش ان a هي سيجمويده او تانش للـ z وبالتالي هتكون بنفس ابعاد الـ z فورا

وفي حالة انك بتعمل مسار خلفي BP هيكون كلا من dw و db مساوي للاصل , يعني كمان dw هيساوي نفس ابعاد w و db زي b

طيب ماذا عن الحروف الكابيتال , وهي المخصصة لمصفوفات العينة بالكامل ؟

اولا الـ Z هي تجميعة زدات صغيرة بالشكل ده :

$$[z_{11} z_{12} \dots z_{1n} \\ z_{21} z_{22} \dots z_{2n} \\ \vdots \vdots \vdots \vdots \vdots \\ z_{n1} z_{n2} \dots z_{nn}]$$

يعني كل Z من كل عنصر من عناصر العينة , بياخد عمود من عواميد Z الكبيرة

وفي حين ان كل Z صغيرة هي عبارة عن مصفوفة عمود واحد وعدد صفوفها هو عدد الوحدات nL

وبالتالي مصفوفة Z الكبيرة هتكون صفوفها هي نفس صفوف Z الصغيرة , بينما عواميدها هتكون عدد عناصر العينة

يعني هتكون

nL x m

ده كان Z , بالنسبة لـ w مفيش حاجة اسمها W كبيرة لانها هي هي

بالنسبة لـ X الكبيرة ؟

هتكون نفس فكرة Z بالضبط , ان X هي عبارة عن عواميد , كل عمود فيها هو x الصغيرة اللي هي اصلا n0x1

وبالتالي X الكبيرة هتكون n0 x m

وبالنسبة لـ b فهيتم تجميعها برضه لكل عينة بحيث تكون n1 x m

وبالتالي الشكل النهائي للكبير سيكون :

$$\rightarrow Z^{[1]} = W^{[1]} \cdot X + b^{[1]}$$

Diagram illustrating the dimensions of the variables in the equation above:

- $(n^{[1]}, m)$ is the dimension of X .
- $(n^{[1]}, n)$ is the dimension of $W^{[1]}$.
- $(n^{[1]}, m)$ is the dimension of $Z^{[1]}$.
- $(n^{[1]}, 1)$ is the dimension of $b^{[1]}$.
- $(n^{[1]}, m)$ is the dimension of the bias term $b^{[1]}$.

يبجي سؤال , ليه اصلا باعمل شبكات عميقة بطبقات خفية كتير ؟ ايه الميزة فيها عن الشبكات السطحية shallow ؟

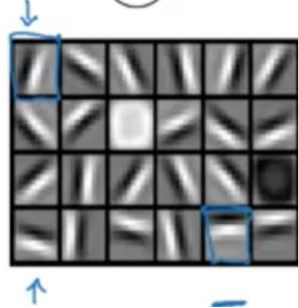
الحقيقة ان الشبكات العميقة بعدد كبير من الطبقات , بتعمل خطوة مهمة , وهي تقسيم المهام شديدة الصعوبة , لمهام بسيطة و سهلة , و تزداد بشكل تراكمي و تدريجي لتصل للمهام الصعبة

يعني مثلا لو عندي صورة عايز اعمل لها detection

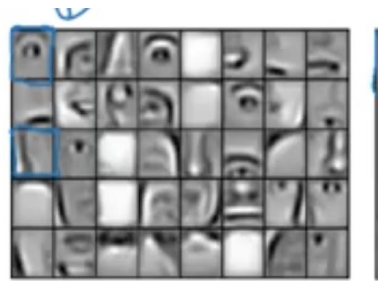


ففحص الصورة كلها عملية شاقة , فهيتم تجزيئها كمهام علي كل طبقة من الطبقات

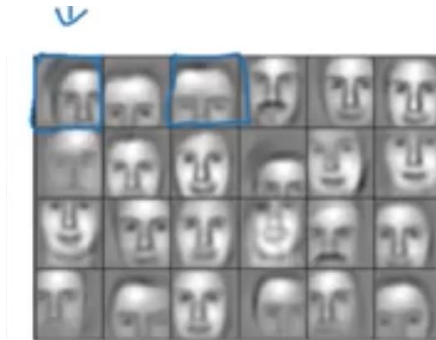
فمثلا الطبقة الاولى هتمسك اجزاء صغيرة من الصورة , وتحللها لمربعات صغيرة زي كدة



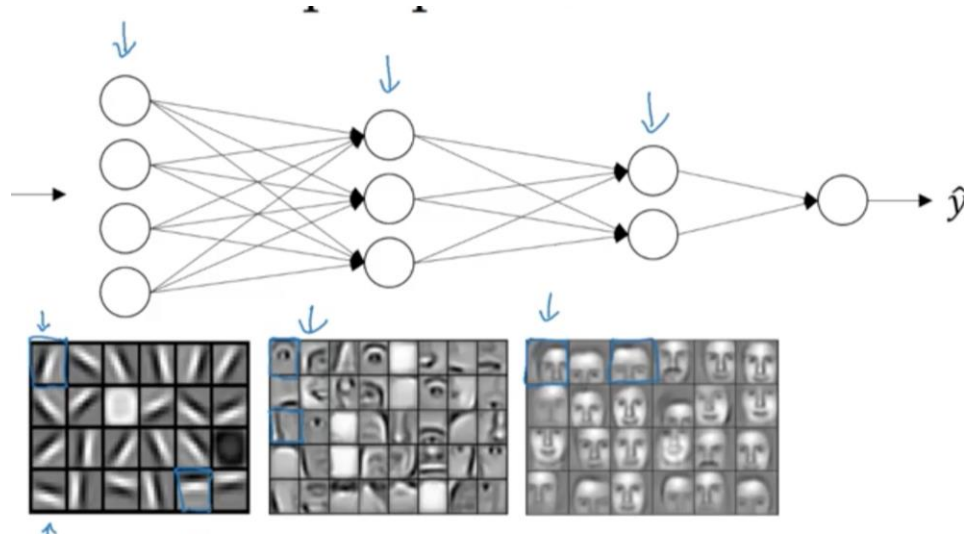
تيجي الطبقة الاولى تسلم نتايجها للطبقة الثانية , اللي تجمعها مع بعض , عشان تشوف هل ينفع تتجمع كجزء معروف , مثلا انف , اذن , فم



وده يتم تسليمه للطبقة الثالثة , اللي ممكن تبدا تقارن بالوجوه نفسها , لما يتك تجميع الاجزاء ديه عشان تعمل وجه كامل



وديه الصورة الكاملة :



فعليا الموضوع مش بيتم بالسلسلة ديه , يعني ممكن تجميع اجزاء الوجه نفسه يتم علي 5 مراحل مثلا , وبالتالي ممكن عدد الطبقات يزيد عن 20 طبقة , وكل ما تزيد الطبقات , كل ما تزيد الكفاءة بشكل كبير

الموضوع مش بس علي الصور , فمثلا لو هاعمل audio recognition فلو عندي اربع طبقات مثلا , ممكن الطبقة الاولى تكون بتميز المداخل الاولى للاصوات تسلمها للطبقة الثانية اللي تميز الحروف , تسلمها للطبقة الثالثة اللي هتجمع الحروف و تميز الكلمات , تسلمها للطبقة الرابعة اللي تجمع جمل كاملة

فالفكرة في تقسيم المهام المعقدة لمهام تدريجية ابسط تتم واحدة ورا الثانية

و ده يتماشى بشكل ما مع طبيعة تفكير المخ البشري اساسا , لان تدريجيا العقل بيستوعب اشياء بسيطة , بعدها يقوم بتضخيمها و تعقيدها لفهم ابعاد اعماق في الشئ فيه كمان سبب ثاني وجيه , يجعل الشبكات العميقة اكثر كفاءة

ان بشكل عام , لو تم استبدال شبكة عميقة , بشبكة ثانية سطحية عشان تقوم بنفس المهام , فستتطلب عدد ضخم من الوحدات في الطبقة الوحيدة في الشبكة السطحية , لتقوم بنفس المهام

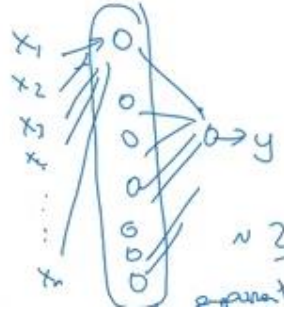
فمثلا لو كنا عايزين نعمل المعادلة ديه بشبكة عصبية

$$y = x_1 \text{ XOR } x_2 \text{ XOR } x_3 \text{ XOR } \dots \text{ XOR } x_n$$

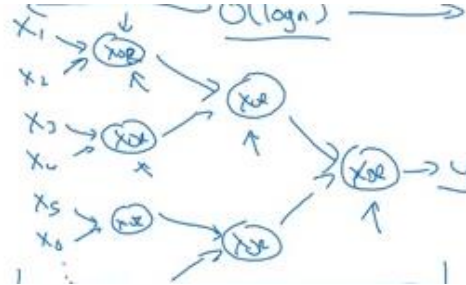
وهي ان الـ y عبارة عن عدد من المدخلات x_1, x_2, \dots , بينهم علاقات XOR مثلا مع بعض

فلو عملنا شبكة سطحية هتكون كدة :

to compare:



هتلاحظ ان عدد الوحدات في الطبقة الخفية الوحيدة كبير جدا , عشان يتناسب مع تجميعية كل x مع باقي الاكسات
بينما ممكن نعملها بسهولة في شبكة عميقة كدة



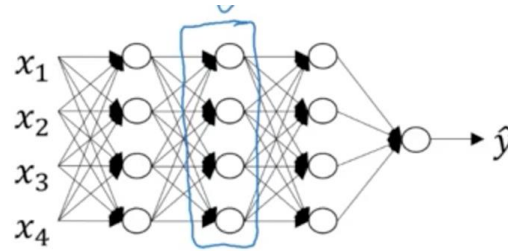
بعدد اقل بكثير من الوحدات , لان اسلوب الشبكة العميقة , جعل من الممكن ربط كل الاكسات بسهولة بعدد وحدات اقل بالشكل الهرمي ده

بعد ما عرفنا اساسيات الشبكات العميقة , تعالي نتعرف علي كيفية بناء الشبكة عبر رص اجزاءها معا

لفهم ده , تعالي نتناول احد الطبقات و نتعرف علي المدخلات و المخرجات

أولا بالنسبة للمسار الأمامي FP

لو مسكنا الطبقة L من الشبكة هنا



فالولا , المعاملات الخاصة بيها هي :

$w(L)$, $b(L)$

$a(L-1)$

بينما المدخل ليها هو

والمخرج هو

$a(L)$

و معادلة Z هي :

$$Z^{[l]} = W^{[l]} a^{[l-1]} + b^{[l]}$$

$$a^{[l]} = g^{[l]}(Z^{[l]})$$

ثانيا بالنسبة للمسار الخلفي BP

نفس الطبقة سيكون المدخل ليها هو :

$da(L-1)$

بينما المخرج

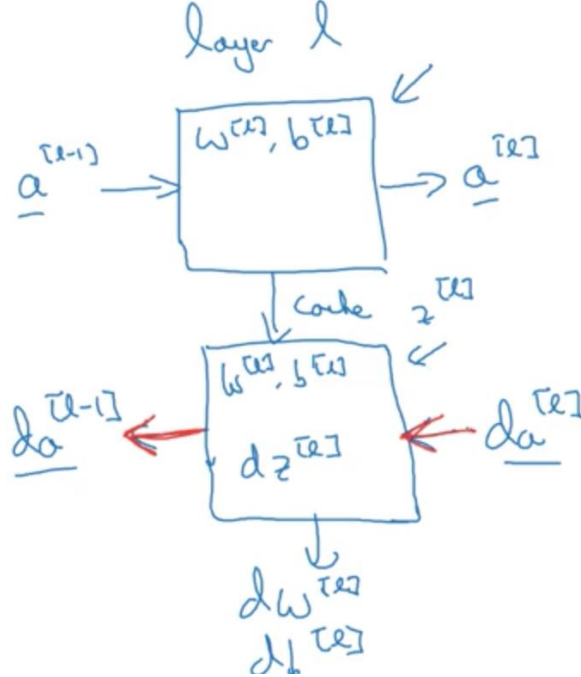
$da(L)$

بينما المعاملات

$dw(L)$

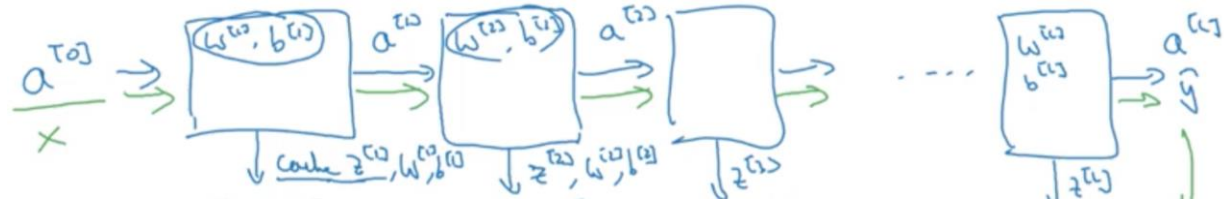
$db(L)$

وبالتالي ممكن اعمل رسم مجمع كدة :



فوق شاييف المدخل و المخرج في المسار الامامي , ويخرج منه كمان z , بينما في المسار الخلفي (الاسهم الحمراء) بالاستعانة بنفس الـ z هستخدم المدخل للحصول علي المخرج و الاشتقاقات

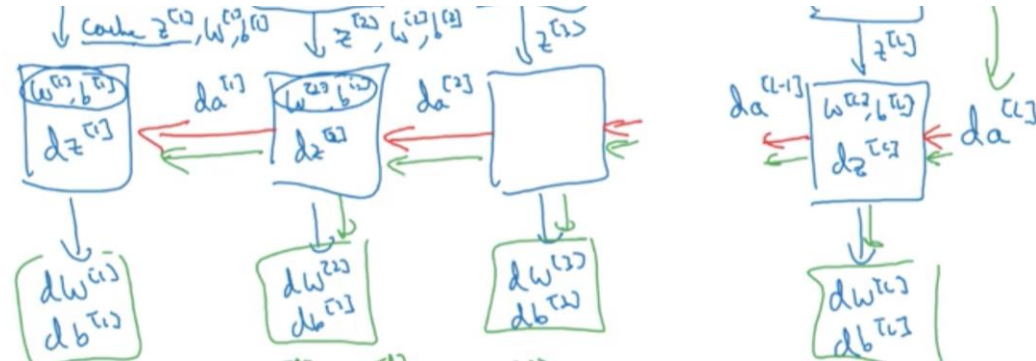
ولما نيجي نربط الطبقات مع بعض . . .



هنلاقي في المسار الامامي , ان بيتم اعطاء a و تخرج a التالية , عبر استخدام w , b , وبيخرج بالتوازي قيمة z (اللي هتستخدم في المسار الخلفي)

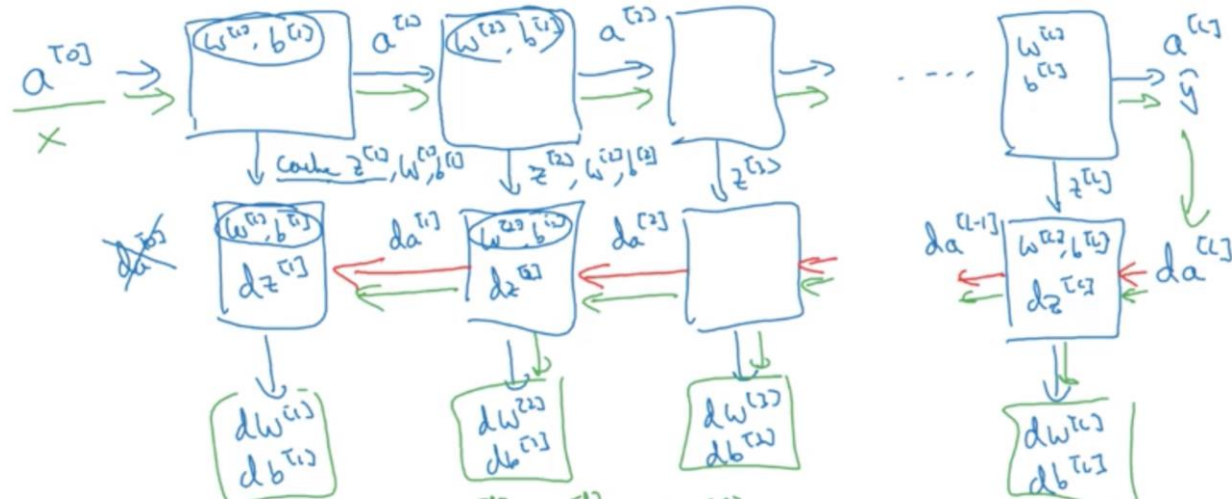
لغاية لما اول لآخر a وهي y هات

وفي المسار الخلفي :



بيتم البدء بـ da النهائية , و احسب منها da السابقة لها , ومنها ارجع للطبقة السابقة , حتي اصل بـ da_1 , لاحظ ان لا يوجد ما يسمى da_0 لان مفيش طبقة قبله وده بيتم باستخدام z من المسار الامامي , والأهم اني باحصل علي dw , db , اللي اقدر اعمل ليهم update لقيم w , b

فيكون الرسم الكامل :



* * * * *

إذن ممكن نلخص كل قوانين المسار الأمامي هنا :

$$z^{[L]} = W^{[L]} \cdot a^{[L-1]} + b^{[L]}$$

$$a^{[L]} = g^{[L]}(z^{[L]})$$

Vertical

$$z^{[L]} = W^{[L]} \cdot A^{[L-1]} + b^{[L]}$$

$$A^{[L]} = g^{[L]}(z^{[L]})$$

والمسار الخلفي هنا :

Input $da^{[L]}$

Output $da^{[L-1]}, dW^{[L]}, db^{[L]}$

$$dz^{[L]} = da^{[L]} * g^{[L]'}(z^{[L]})$$

$$dW^{[L]} = dz^{[L]} \cdot a^{[L-1]}$$

$$db^{[L]} = dz^{[L]}$$

$$da^{[L-1]} = W^{[L]T} \cdot dz^{[L]}$$

$$dz^{[L]} = W^{[L+1]T} \cdot dz^{[L+1]} * g^{[L+1]'}(z^{[L+1]})$$

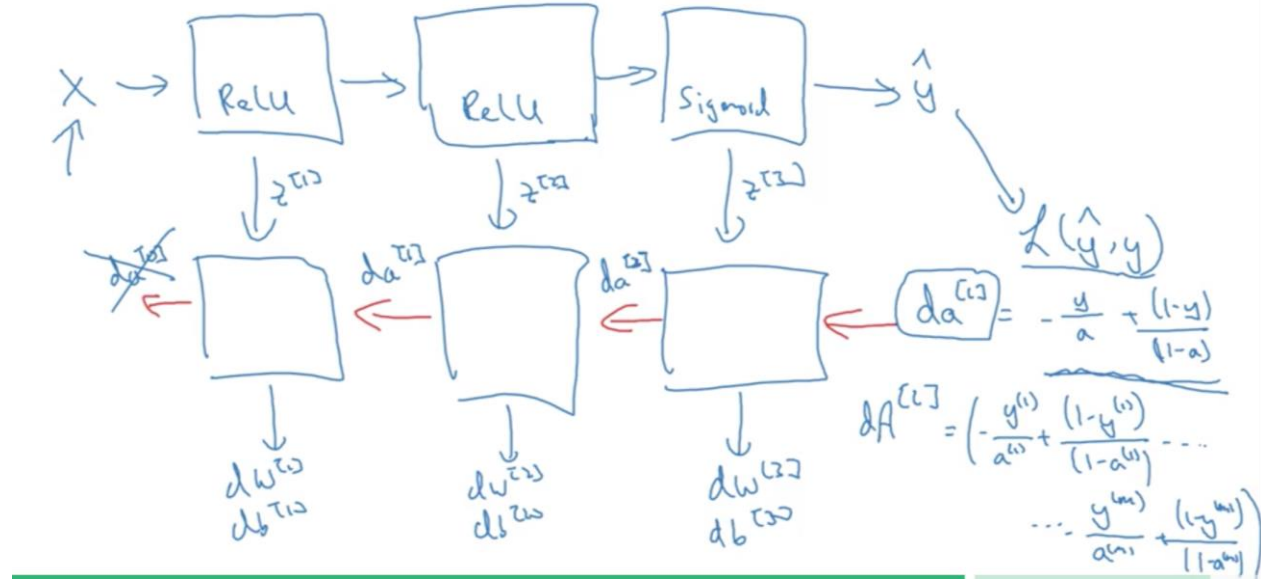
$$dz^{[L]} = dA^{[L]} * g^{[L]'}(z^{[L]})$$

$$dW^{[L]} = \frac{1}{n} dz^{[L]} \cdot A^{[L-1]T}$$

$$db^{[L]} = \frac{1}{n} np.sum(dz^{[L]}, axis=1, keepdims=True)$$

$$dA^{[L-1]} = W^{[L]T} \cdot dz^{[L]}$$

وهتكون الخريطة العامة :



نتلاحظ ان الدالة المستخدمة في اول طبقتين هي ReLU, وفي الاخيرة السيجمويد (اختيار المبرمج لسبب معين)

من الاخيرة هنجيب الواي هات , و منها هنرجع للمسار الخلفي , عشان نجيب الـ da الخاصة بكل خلية , و نتاخذ الـ z من البلوكات اللي فوقها , و نتطلع كمان الـ dw, db

مع ملاحظة ان معادلة الـ da هي :

$$-\frac{y}{a} + \frac{(1-y)}{(1-a)}$$

وبالتالي لكل العينات يعني الـ dA الكابيتال هتكون :

$$dA^{[i]} = \left(-\frac{y^{(1)}}{a^{(1)}} + \frac{(1-y^{(1)})}{(1-a^{(1)})} \dots \dots \dots -\frac{y^{(m)}}{a^{(m)}} + \frac{(1-y^{(m)})}{(1-a^{(m)})} \right)$$

متنشاش ان دايمًا واي علي ايه بيكون سالب , بينما الترم الثاني بيكون موجب , وهيتم جمعهم لكل عنصر من عناصر العينة

المعاملات العليا : Hyper Parameters

المعاملات العادية Parameters هي الـ w, b

لكن المعاملات العليا , وهي كمان شديدة الاهمية في تحديد شكل الـ NN , مثل :

- معدل التعلم α Learning rate
- عدد مرات التكرار No. of iterations
- عدد الطبقات الخفية L
- عدد وحدات او خلايا كل طبقة $n1, n2, n3$
- اختيار دالة الاكتيفاشن : Sigmoid , Tanh , ReLU

فيه معاملات عليا تانية , لكن سندرسها لاحقا

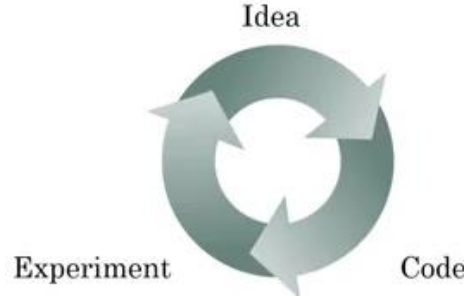
والفارق الاساسي بين المعاملات و المعاملات العليا , ان المعاملات لا يتم تحديدها , لكن بيتم حسابها من الـ NN , وحتى لو حددتها قيم مبدئية عشوائية لـ w, b فهي تتغير و ستصل للقيم المثلي

لكن المعاملات العليا لابد من تحديدها قبل بدء الحساب , وهي لا تتغير اوتوماتيك , ولا يتم عمل update لها , وبالتالي اذا ما تم تحديدها بشكل خاطئ فسيصل الحساب لاماكن بعيدة , دون الوصول لنتائج مناسبة

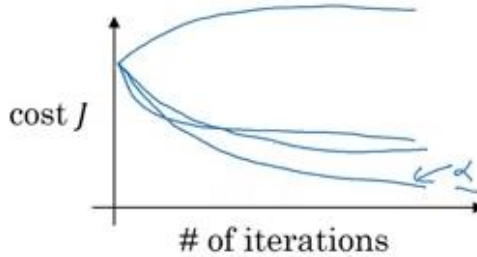
وبالتالي لازم نستخدم ما يسمى الطريقة الامبيريكال في التعامل , يعني الطريقة التطبيقية

يعني ممكن يكون عندي الاول فكرة او اقتراح معين لرقم ما , بعدها ابدأ انفذه عبر كتابة الكود , وبالتجربة اشوف هل مناسب ولا لا

مثلاً نقول اني هختار معادل التدريب الفا هيكون 0.01 , اقوم اعمل تطبيق له , واشوف بالتجربة هل الرقم مناسب ولا لا , حسب الخريطة هنا :



ونقصد بالتجريب هنا ان نرسم علاقة بين الفا و بين معادلة الخطأ J و نختار علي اساسها الرقم المناسب زي هنا :



كنصيحة عامة , في تحديد الارقام الخاصة بالمعاملات العليا هي إنه لا بد من مراعاة الفروق في ارقام المعاملات العليا بين التطبيقات المختلفة للـ DL يعني لو حد شغال في في الـ voice recognition وهو متعود علي ارقام معاملات عليا معينة , فقد تكون غير مناسبة لو استخدم نفس الارقام لو هيشغل علي recommendation system

لان كل تطبيق بيكون ليه احتياجات و عدد طبقات مناسب و معامل تدريب مناسب

بل ان في نفس التطبيق , ممكن تختلف ارقام المعاملات العليا من مسألة لمسئلة

و كذلك مع مرور الوقت و الشهور , ممكن تلاقي ان الخوارزمات اتغيرت

*_**

نهاية الاسبوع الرابع , والكورس الأول