+ **الإجابات الذكية لحل مادة الشبكات العصبية**

Students questions in **Black**. [write in العربیة or english, Don’t use FrancoArab]

- **TA** answers are in **Red**

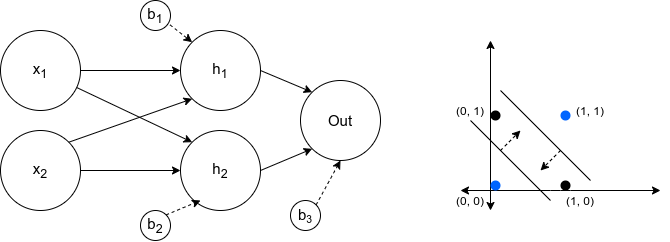
- **Students** answers are in **Blue**

### Q1) مش فاهم أول مسألة في شيت 3 خالص , ال weights جات ازاي وال bias جات ازاي؟

اول حاجة بنعملها فى اسئلة ال design هو اننا نرسم ال points اللى معانا ولحسن الحظ الرسم بسيط وقائم على معرفتنا بال XOR.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

اول حاجة نقدر نستنتجها من النقط ان عندنا 2 inputs وواحد output ولكن زى ما عارفين من اخر سؤال فى شيت 2 فال XOR لا يمكن فصلها بخط واحد (perceptron واحد) فكان الحل الوحيد لمشكلة زى كده هو ان احنا نبنى Multilayer Neural Network



فيها خطين يتمثلوا فى 2 نيورونز (h1, h2) وخط اخير بينتج رقم واحد يتمثل فى out.

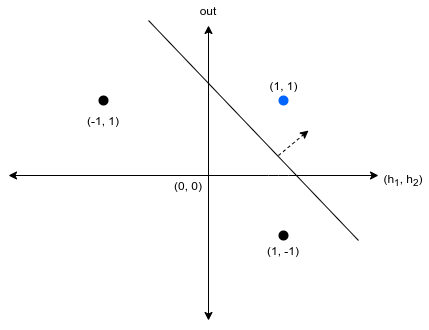
ال h1 وال h2 بنحسبهم عن طريق الرسم اللى على اليمين … فبنشوف النقط اللى يمكن يفصلها كل خط ونحل على اساسهم المتباينات الناتجة من ال Sign … واهم حاجة مننساش ان ال weights ما هى الا المتجه العمودى على كل خط واللى بيعرفنا الاتجاه اللى لو اتحركنا فيه نوصل لقيم موجبة وعكسه إلى قيم سالبة

أول خط h1 وليكون هو الخط القريب من النقطة (1, 1) لو افترضنا المتجه العمودى عليه هيكون [-1, -1] فنقدر نعوض فى النقط اللى احنا عارفين هيفصل بينها صح وهى (1, 1) (1, 0) (0, 1) فى معادلة ال Sign

نقدر نستنتج من المعادلات اللى فوق ال bias او ال ونعمل المثل مع ال h2.

اخيرا مع ال out بنعمل نفس الموضوع ولكن هنحتاج نعيد رسم النقط اللى هنقسم بينها … لو لاحظنا النواتج اللى ممكن تصدر من h1 و h2 هنلاقى انها مش هتخرج من اربع احتمالات:

اول حالة استحالة هتحصل لأنها بتوصف نقطة الخطين هيعتبروها فى الجزء السالب وده غير موجود فى ال 4 نقط اللى معانا من ال XOR … الحالة التانية بتوصف النقطة (0, 0) والحالة التالتة بتوصف النقطة (1, 1) أما الحالة الاخيرة بتوصف النقطتين (1, 0) و(0, 1) كونهم بيكونوا موجبين مع كلا الخطين … بناءا على هذه المعلومات هنرسم الاحتمالات الجديدة ونقسم على أساسها



فينتج عندنا 3 نقط نقدر نقسم ما بينهم بنفس الطريقة اللى فوق ونستنتج ال weights وال bias

### Q2) في مسألة رقم 7 في شيت 2 التلات مسائل متكرر الحل بتاعهم من أول مسألة.. النقطة بقى اللي عايز اسأل عليها..

### dL/db = -ym, dL/dw = -ym\*xm يعني التفاضلين مش واحد وبالتالي المعادلة لازم تبقى معادلتين للابديت مش واحدة بس صح ولا غلط؟

سؤال كويس … على الحقيقة الاجابتين اللى انت كتبتهم نقدر ندمجهم فى حاجة واحدة … لو تتذكر الvector بتاع ال input هتلاقى ان عند الرقم اللى كان بيتضرب فى ال bias كنا بنكتب 1 على امل ان لما ناخد ال dot product الناتج يكون ال bias مجموع لوحده … نفس الفكرة بتطبق هنا وعدم وجود ال xm فى معادلة ال bias مش بسبب انها مش موجودة ولكن لأن قيمتها وقتها بواحد

### Q3) In sheet 3 q1, the question says: “**Tune** **Mathematically**”. Does not this mean to use gradient descent or any other optimization algorithm to solve the problem and not to use the graph to deduce the weights?

It will be stated to use GD if it was required. Here, we tune our assumptions mathematically based on the inequalities we drive from the Sign Function

### Q4) How To estimate ( W , b ) by visualizing the plot in 3D , how to estimate this perpendicular vector on the decision boundary !!? (Q→4 sheet 2 )

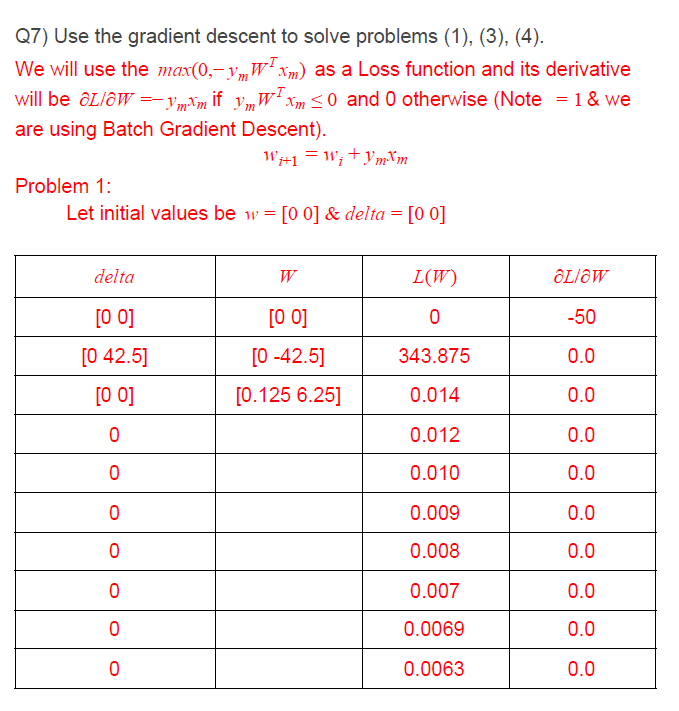
Same as the previous questions, you choose the perpendicular vector on the plane to be the direction (>0) and visually estimate it.

### Q5) Sheet 3 , problem 1 .. how the bias value of h2 will be = -2 while form the inequalities ( 0>W0>-1 ) !? ( note : if i calculated it wrong , will you please write it on a paper and capture it ? ) thanks in advance :)

The bias of h2 is -0.5 not -2

Please note that sign(0)=1 so -2 won’t be the correct Bias

### Q6) can you please explain this again ? I got the idea of the loss function and its derivative but i can’t understand the rest of the problem’s flow ..



This is an old answer (please check the new one), regarding the flow … you will need to revisit the Batch Training Paradigm and remember how the weights are updated

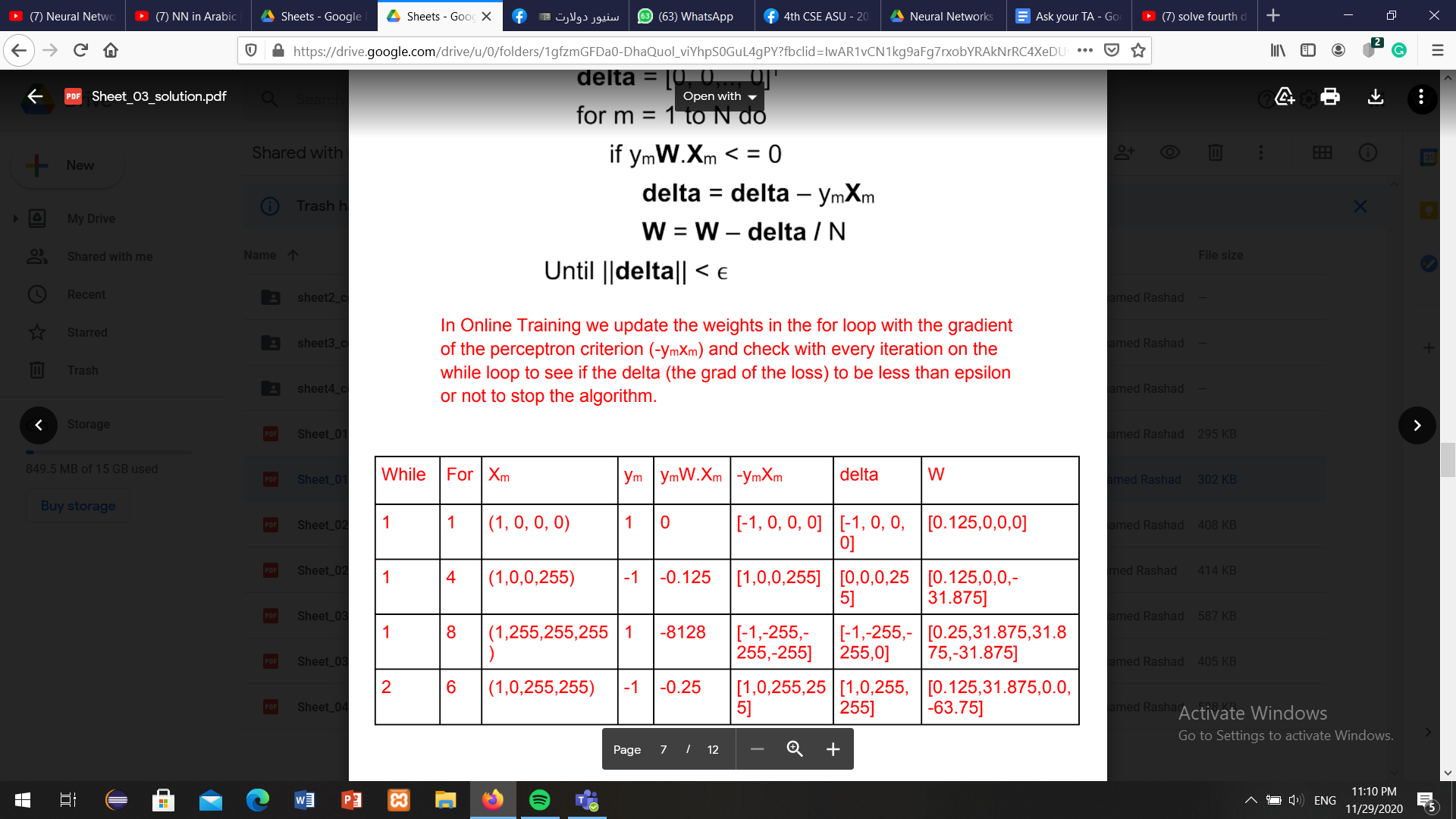
### Q7) In perceptron criteria & hinge loss, max(0, y \* WX), is the Y here the label or the predicted value? Also do they depend on the activation function? Or they always use W\*X which is the value before entering the activation function and producing the final output.

First of all, the loss function lacks a minus .

Regarding the notations, we use y\_hat for the output of the model (or prediction) and y for the label.

Yes, loss functions can change with the change of the activation functions (This should have been covered in Slides04 from page 22 to 29)

In this example we used W\*X because the Sign function is non-differentiable and what we actually care about is its input not the function itself.

Q) 

### Q8) In (Q4) sheet 2, it must be -ve so i think it won't be correct, can i estimate values of weights and bias by trying any values or i should solve it in steps by equations?

### 

In the Second equation we substituted by (255,0,0)>> y =+1

if u want to follow another steps make sure that the classification is correct

Can you tell me what your weights and bias are?

B = 90

W1 = 0.1

W2 = 0.2

W3 = -0.5

I think in number 2 in the answer : sign(1\*255+0\*255+0\*255+100) = -0.087 (y = -ve)

Our equation is X1+X2-2X3+100 so when you substitute with (0,255,255)

0+255-2\*255+100=-155 sign(-155)=-1 correct class

If we used your weights 0.1X1+0.2X2-0.5X3+90 When u substitute with (0,255,255)

0.1\*0+0.2\*255-0.5\*255+90=13.5 sign(13.5)=1 and this point is classified to -1 so this is misclassification

But if W1 = 1 ,W2 = 1 ,W3 = -2 and b = 100, this point (255,0,0) will be misclassified as sign(255\*1+100) = -ve but it must be +ve

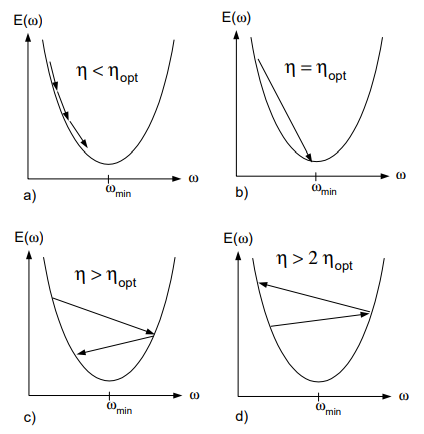
255+100=355 sign(355)=+1

To keep it clear sign(any positive number)=1 sign(0)=1

sign(any negative number)=-1

### Q9) In sheet 2 Q5-> should the learning rate always be between 0 and 1 or is it just for this case and why?

Learning rate must be less than 2\*optimal\_learning\_rate and larger than zero (as if it’s value is negative the GD will go in the direction of the maximum value not min).



### Q10) Why is the weight vector orthogonal to the decision boundary’s hyperplane? And why does it point towards the positive data points?

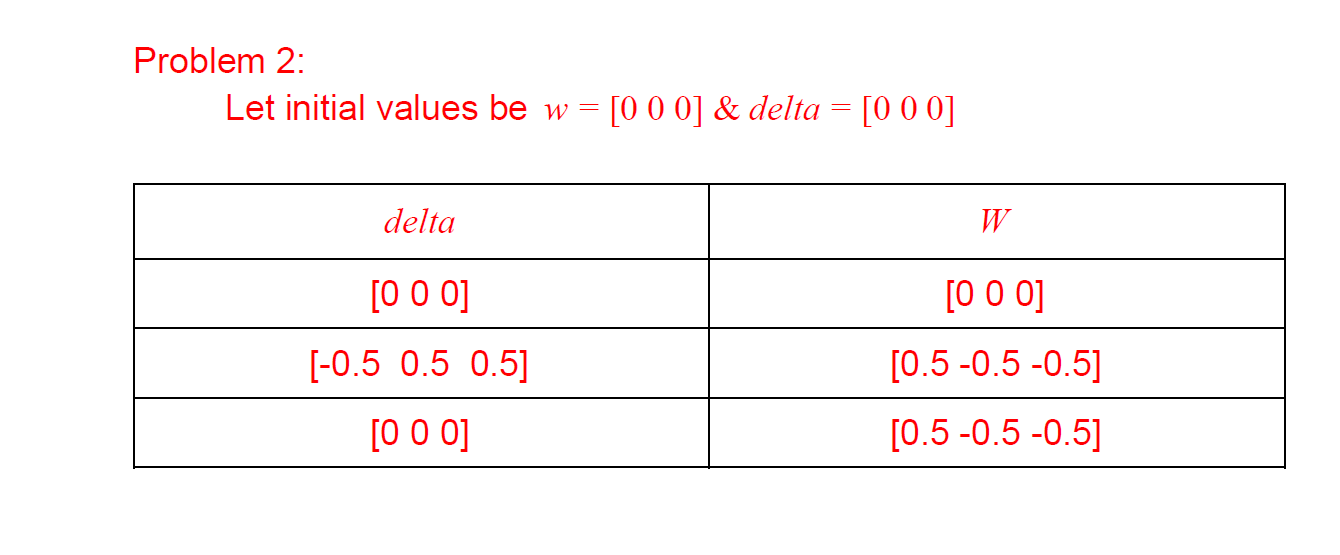
We know that the equation of the hyperplane (decision boundary) is WTX = 0, means that if we substitute with the data points (X vector) that are located at the decision boundary and with the weight vector, the result should be zero

The equation WT. X = 0 is a dot product, which can be written as ||W|| \* ||X|| cos theta = 0,

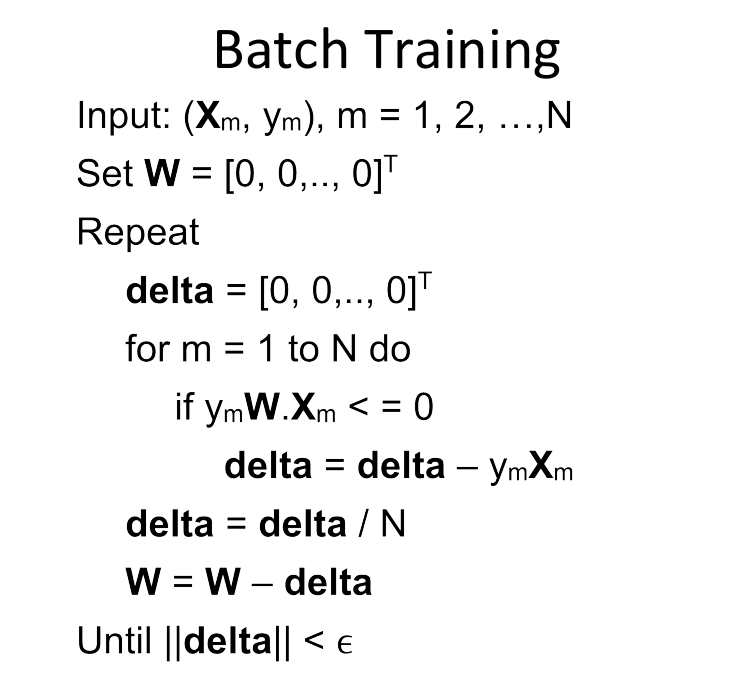
Where theta is the angle between the two vectors and ||W||, ||X|| are the magnitudes.

To satisfy the decision boundary equation, either the magnitude of one of the W, X vectors must be zero, or simply the theta must be equal 90 degrees to have cos theta = 0, then the equation be satisfied.

### Q11 ) could you provide the steps solution for this problem ?



قبل اى شئ لازم نفتكر ان فى حالة السؤال مقالش طريقة ال GD فأحنا بنفترض اننا شغالين ال GD العادى وان احنا شغالين Batch Training … فأول حاجة هنحتاج نعملها نكتب قدامنا ال process



ونفتكر ان ال if condition موجودة هنا عشان ال grad بتاع ال loss كان مقسوم لجزئين جزء منه بصفر فى حالة ال ywx كانت أكبر من الصفر وكان بقيمة yx- لما تبقى أصغر من الصفر.

النواتج فى المسألة قامت على اننا بدأنا بالتعويض بال weights بصفر ونتج عن كده ال pred كان دائما بصفر فكنا دائما بندخل فى ال if … الاربع النقط اللى كانوا معانا فى المسألة كانوا كالأتى

1. نقطة (-1 ,-1) وال label بتاعها 1
2. نقطة (1 ,-1) وال label بتاعها 1
3. نقطة (-1 ,1) وال label بتاعها 1
4. نقطة (1 ,1) وال label بتاعها -1

احنا هنفرض ان ال bias هو اول قيمة وان ال inputs اول رقم فيها واحد عشان يناظر ال bias.

مع اول نقطة هنلاقى الدلتا بقيت [-1 1 1] وبما ان ال weights مش هيحصلها update الا بعد ما ال for loop تخلص فال points هيحصلها accumulation بقيمة ال -yx فهنلاقى الدلتا هتوصل ل [-2 2 2] بعدها هنقسم على 4 (عدد ال samples اللى عندنا) فالناتج هيكون

Delta = [-0.5 0.5 0.5]

وبالتبعية ال weights هتكون سالب قيمة الدلتا مجموعة على الاصفار فهينتجلنا نفس ال vector بس بالسالب …

W = [0.5 -0.5 -0.5]

كده نكون خلصنا اول iteration فى ال while loop … هنعيد العملية تانى لأن قيمة ال delta كا vector اكبر من ال epsilon.

فى ال iteration التانية هنلاحظ ان كل النقط لما بنعوض بيهم مع ال weights الجديدة بيطلع حاصل ضرب ال ywx بموجب دائما فمش بندخل فى ال if condition نهائى والدلتا بتفضل بصفر وبالتبعية مش هيحصل update فى ال weights وال while loop هتنتهى لأن قيمة ال delta بصفر.

### Q12) Can we use steepest gradient descent to find the local minima and solve the problem manually without a computer program? If so, how do we compute the learning rate?

فى حل بالألة عن طريق التعويض زى اللى موجودة فى الفيديو هنا

<https://www.youtube.com/watch?v=U7AdSSnLXkA>

وفى حلول عن طريق الرسم زى ما موجود فى ال slides

### Q13)اعرف منين اتجاه الperpendicular vector للboundary بيشاور فوق ولا تحت؟

الاتجاه على حسب ال parameters بتاعت ال boundary … اللى احنا بنسميها فى السكاش ال weights.

فى الغالب احنا اللى بنحددها فى مسائل ال design ولكن لو كانت موجودة فاحنا هنحول المعادلة لضرب vectors عن طريق ال dot product … فهيكون ال parameters vector هو المتجه العمودى على ال boundary … التعويض برضو هيأكد اكتر على النتائج.

يعنى هتبقى فى اتجاه الclass اللى فيه points اكتر مثلا؟

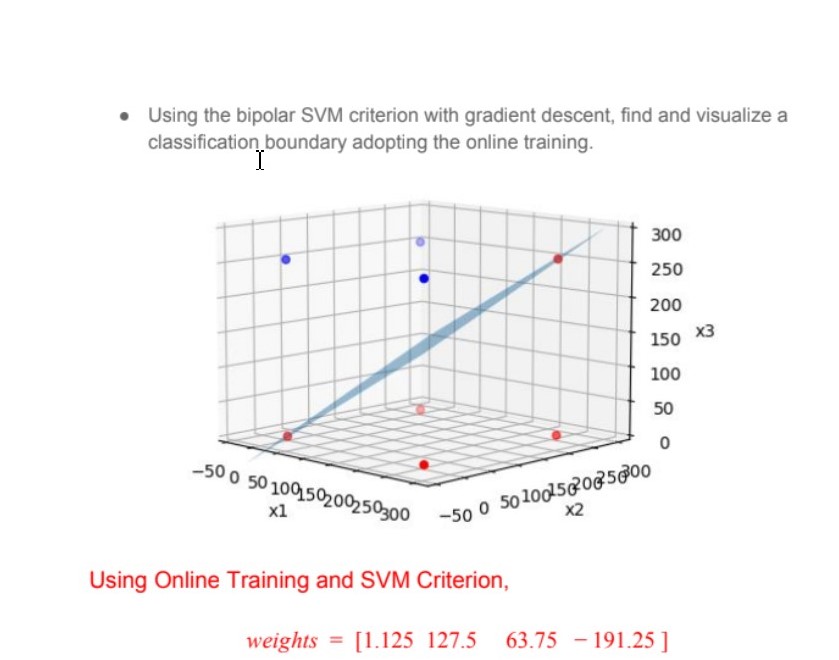
لا فى اتجاه ال class اللى انت حددت انه 1

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### Q14)how do we conclude the bias, w1 and w2 from the graph? I have seen the answers of the prev questions but still not clear

لو جزء التخيل فيه مشكلة كبيرة ممكن تستخدم المتباينات اللى بنستنتجها من الpoints الموجودة فى المسألة … هتضمن لك ان الاجابة تكون صح.

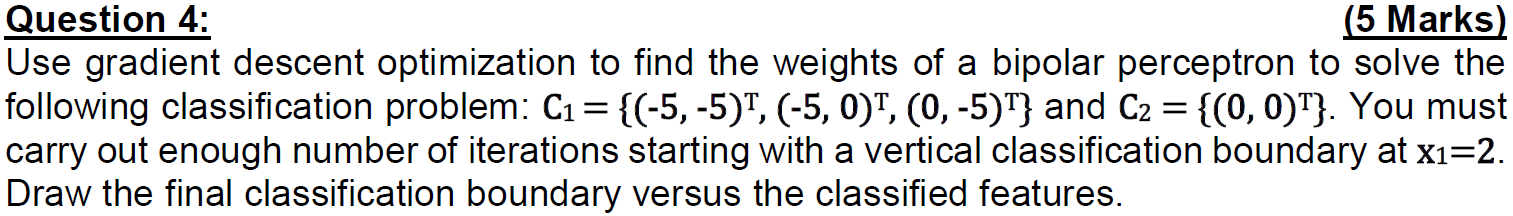
### Q15) how did we get the weights in problem 5 sheet 3 ?



We updated the solution of this problem in the sheet model answer, try reviewing it as it’s more detailed and quite comprehensive.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### Q16) In the following question, how to get the initial weights from the statement “vertical classification boundary at x1= 2”?

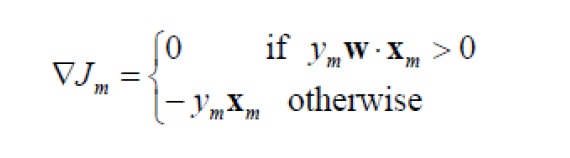


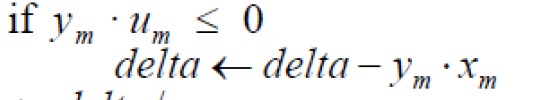
Vertical classification boundary means that the equation of classification boundary is (x1= constant). In the problem, the classification line passes through x1= 2. Then, the equation becomes: x1-2= 0. By analogy to the general equation of the classification boundary: w1x1+w2x2+b= 0, you can easily get: w1= 1, w2= 0, b= -2.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### Q17) for the gradient descent , the new weight equals the old weight **-** Alpha\*gradient

### For the bipolar perception criteria , the gradient it self contains negative sign



So should the figure below be delta + yx , or is it correct like that?  


No it’s correct, because we subtract the delta when we update the weights so the signs match up.

### Q18) how to find the mag of delta to check if it is less than epsilon?

We calculate its Norm:



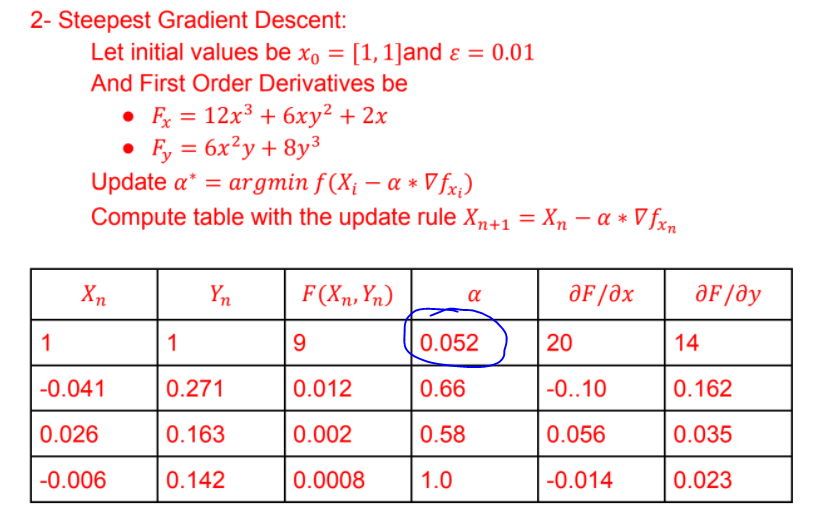
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### Q19) What is the general approach to find constraints on learning rate for convergence ? In sheet 2 pb 5 it was possible to take Xo common factor.. which is not possible if f(x)=X^4.. what defines convergence ?

The main constraints is shown in the second figure of the solution, the learning rate must be between 0 and 2\*optimal\_learning\_rate

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### Q20) How did we obtain the first optimal alpha (0.052)?

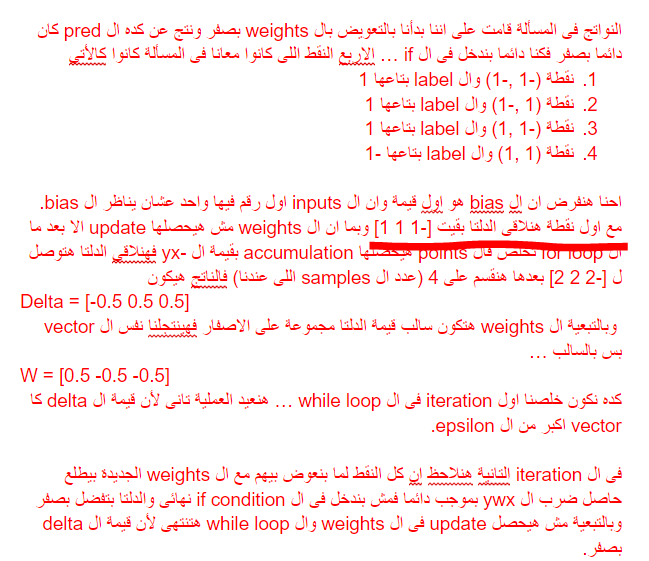


We computed the min value of

Differentiate the previous function and find the roots that makes it equal to zero.

### Q21) When we get y\_hat of the first example with the initial weights (0, 0, 0): y\_hat = 1 (same as the label y) so the gradient of the loss will be 0 because y.W.X > 0… so shouldn’t delta remain (0, 0, 0) and only be updated at the last datapoint where the prediction is 1 but the label is -1?

### Then by the end of the first iteration, we would have delta = ¼(1, 1, 1)?



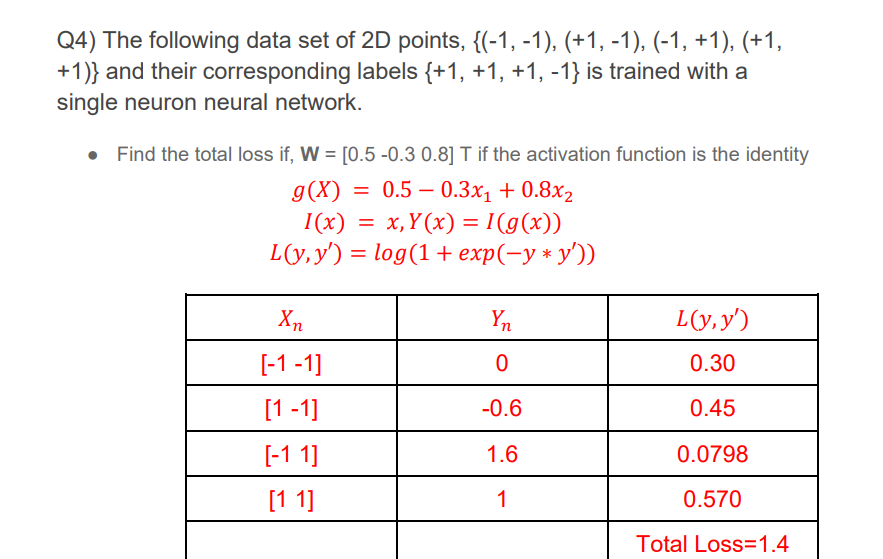
With yWX we will get an output of zero with the initial values, but if we remembered the grad of the loss function we find that when yWX = 0 that grad will be with a value not a zero, the value will be -yx and that’s why we accumulated in the first step.

**Nice thing to know**: Initialization with zero is not always a good idea because of something like this, Sign(0) is either undefined or equal to zero (based on the definition you will use).

Q21) why did we use here the logi

egression loss fn not the bipolar perceptron loss function, although bostic rth can have identity activation function,

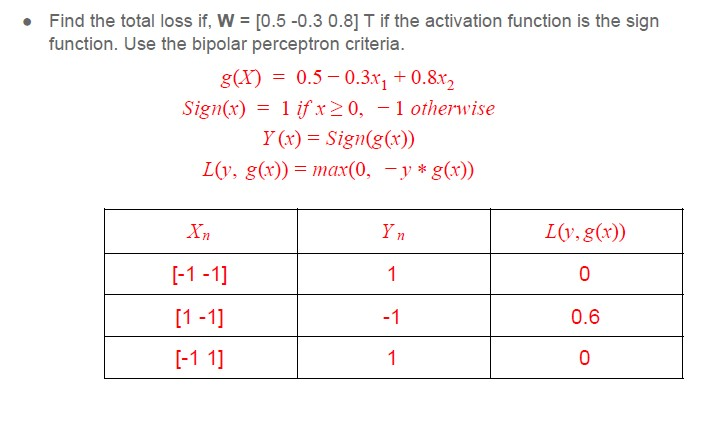
In other words why didn’t we use the max(0, -ymWxm)



Because in the same question point 3, we are asked to use the perceptron criteria … so we choose to introduce you to a new loss function and avoid redundancy (They both work).

### Q22) why in Loss Function we used Loss=max(0, − y \* g(x)) ?

I think it should be --> Loss = max(0, − y \* y^)) and it’s given to use SIGN as activation Function ,so y^=SIGN(g(x)) then   
Loss = max(0, − y \* SIGN(g(x))) is that right ?



One of the main features with loss functions is that they have to be differentiable, Sign is not differentiable so we had to take it out.

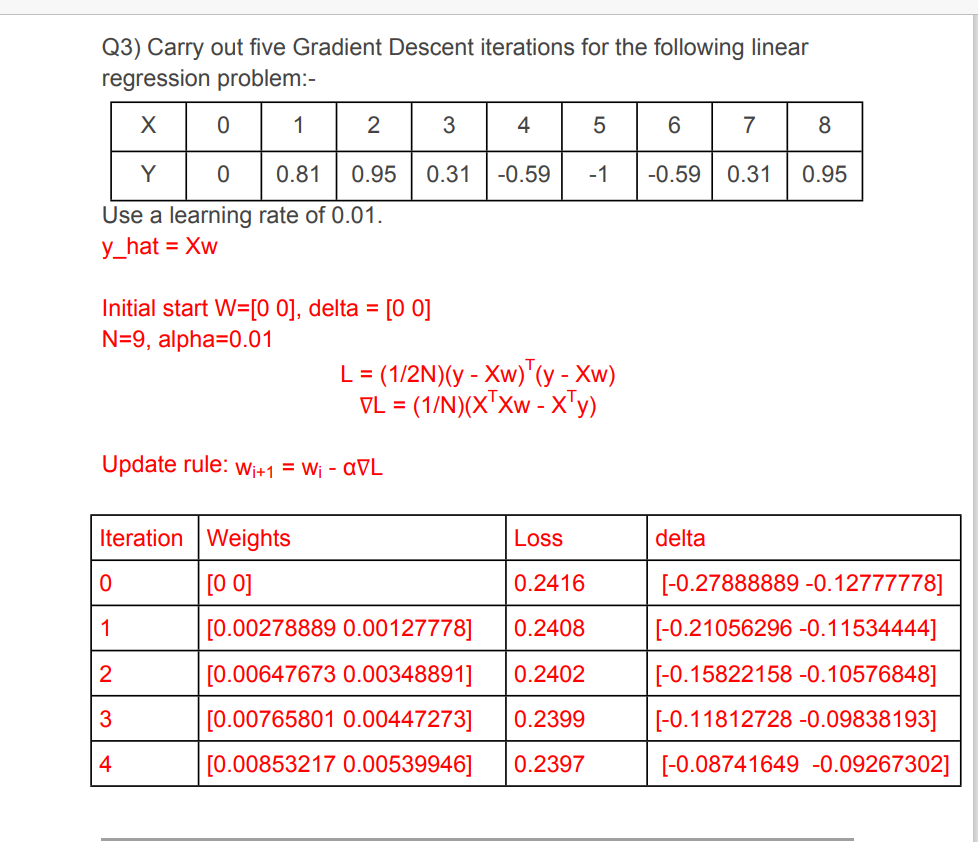
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Q23) how is the loss calculated in the first step although the weight = zero if we substituted in the loss function

Also I want to know how to substitute, is it correct?

Iteration 0: L= 1/18 [ [0]- [1;0;1;2;3;4;5;6;7;8] [0] ] T

[[0]- [1;0;1;2;3;4;5;6;7;8] [0]



U are substituting the weights instead of y

y=[0, 0.81, 0.95, 0.31, -0.59,-1,-0.59, 0.31, 0.95]

X=[[0, 1], [1, 1], [2, 1], [3, 1], [4, 1], [5, 1], [6, 1], [7, 1], [8, 1] ]

The X is a 2 dimension input first is data and the second is the bais>>

For example X[0]=[0,1]>> 0 is the data given in the Question

>>1 is the bais

L= 1/18 [ y- X [0 0] ] T[y- X [0 0]]

We don’t solve here by batch training?

Yes we use batch Training

I still can’t get how in the first loss calculation there’s a value : L=1/18 [ 0 - [0 1] [0 0] ] T [ 0 - [0 1] [0 0]] should give 0 ?

No it’s an example to just illustrate what is the 1 but u got to substitute with the whole array

Every time I will substitute with the whole array of X to get the loss?

Yes

### Q24) Just making sure I understand Steepest GD correctly:

### Obtain f(xi - 𝛼∇fxi) which will be a function of alpha.

### Differentiate this function with respect to alpha and get f(1).

### Differentiate f(1)(𝛼) to get f(2)(𝛼).

### Equate f(1)(𝛼) = 0 and solve it using the Newton-Raphson method (which the Dr. mentioned in the last lecture):

### 𝛼n+1= 𝛼n- (f(1)(𝛼n) / f(2)(𝛼n))

### However, how do I know that the alpha obtained minimizes f(1)(𝛼) and not maximizes it?

Here, you are using Newton Rahpson to optimize for the alpha … it’s quite overwhelming i should say but it can work.

We know that we are moving to min not max by analyzing the function and seeing if it’s convex or not … if it’s convex then we only have a min that we are moving toward.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### Q26)If he said find the eigenvalues, should I always get the hessian?

In the sheet problem, Eigenvalues of the hessian matrix were required to check the convexity of the learning process.

So it depends on the question and its requirements.

But for the convexity of the learning process, we examine the eigenvalues of the Hessian Matrix

### Q27) In newton raphson method if we have F as the cost function Should I use Xn+1 = Xn - F/F` to minimize the cost function directly Or Xn+1 = Xn - F`/F`` to minimize the gradient of the cost function

Good Question, Actually it depends on what you want to do … In GD we want to optimize F and to do so we need the input that makes it’s **grad** equal to zero (because then we are on a min value).

The process of Newton Raphson kicks from there as its main goal in life to get the input where the value of function equals zero, that’s why we used it with the gradient and that’s why it’s possible to be used in the steepest GD algorithm to get the best alpha.

### Q28) How did we conclude the value of the bias and the weights directly from the visualization without solving inequalities like in the lec slides ?

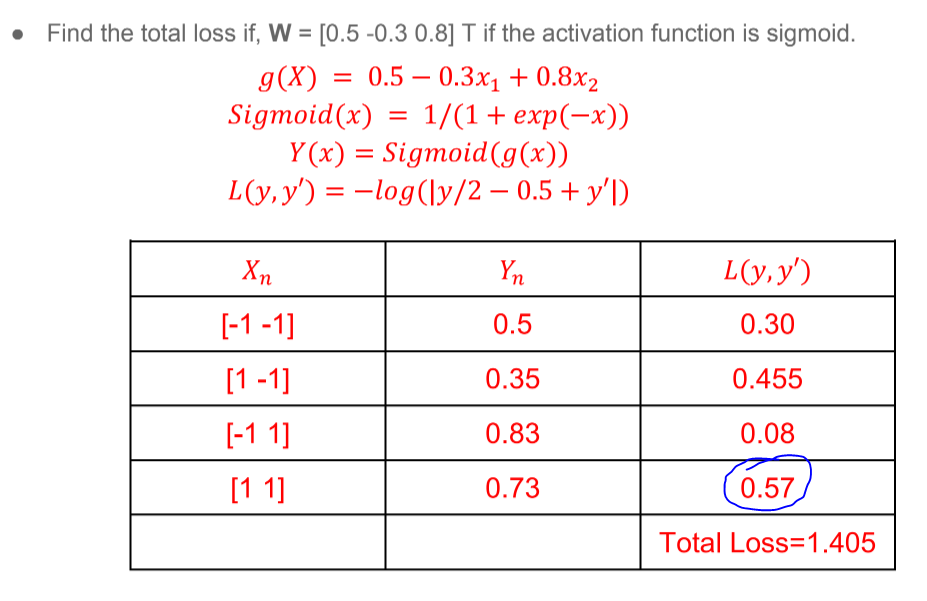
Depends on how the problem and the visualization looks like, in the example you attached, we concluded that the weight vector = [-1 -1] as we assumed it’s a vector going downwards and with angle 45 with the x-axis as it’s perpendicular to the decision line that we draw, that was assumed to be with angle 135

It's just a process of assumption , the important thing is to verify your assumption with the points and to get the classification right.

If you didn’t get lucky with the assumption or you didn’t see a proper one, you should solve the inequalities analytically.



Q29) Shouldn’t this value be 0.136 instead? Since -log(0.73) = 0.136?



Loss = -log(|y/2-0.5+yn|)

y=-1 from the question

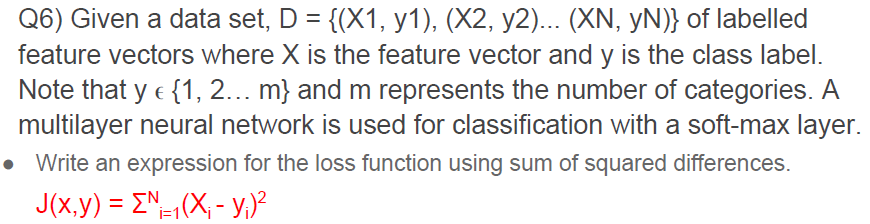
yn=0.73 >>sub>>-½ -0.5+0.73=-0.27 >> don’t forget the abs

log(0.27) =-0.57

L=-log(0.27)=0.57

**How is yn = 0.73? Isn’t yn = sigmoid(g(x)) = sigmoid(0.5-0.3+0.8) = sigmoid(1) = 0.27?**

**Sigmoid (g(x))=1/1+exp-g(x)**

Q30)  
  
مش المفروض تكون sigma(1-Yr)^2 ?  
لاننا نحسب لاكبر احتمال فقط

Okay, you are confusing two losses

It’s either sum of squared differences, and it's in the answer,

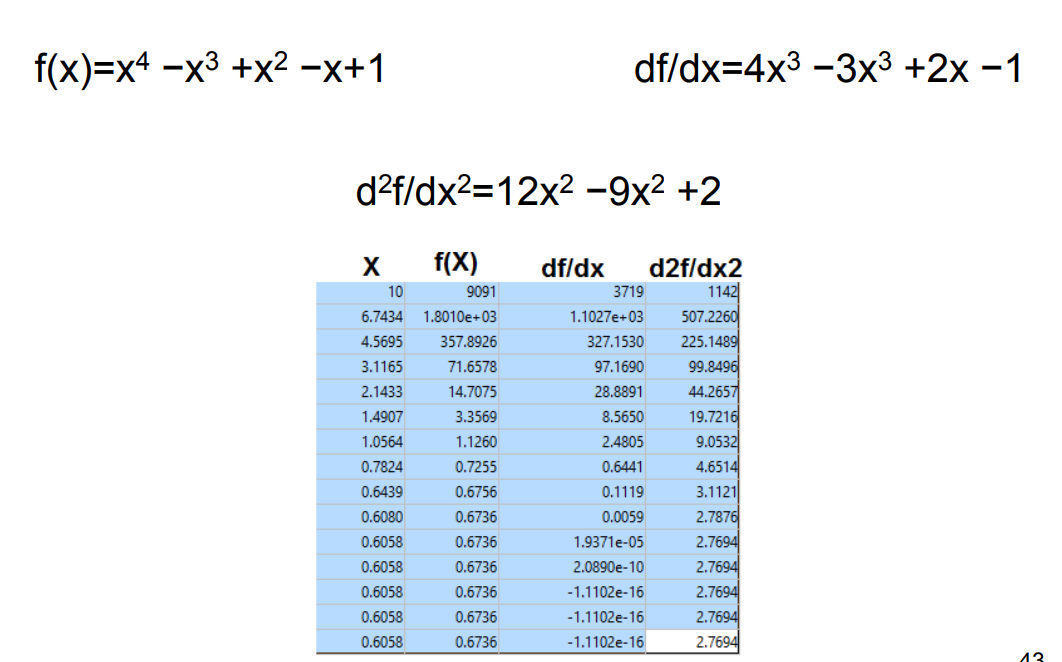
Or log likelihood, and as we are in multi classification we use the multi-class log likelihood: L = - log(y’r)

And in this particular question it was asked for both of them, so you answer as it’s required.

But if you are asked for the loss for a multi classification problem with softmax without stating whether its log likelihood or sum of squared differences. You should use the suitable one for the probabilistic distributions resulting from the softmax layer, which is L = - log(y’r)

And in either ways, the answer **sigma(1-Yr)^2** you are proposing is not correct

Q 32)



What is the form here because i use all possible rules no one is right and the div df / dx is wrong

What form

Xi+1 = x - f(x)/f `(x)

Or xi+1 = x - alpha \* f `(x)

They are both right but you are using them wrong, the first one will be f’(x)/f’’(x) as the function we want it to be zero is f’(x) not f(x).

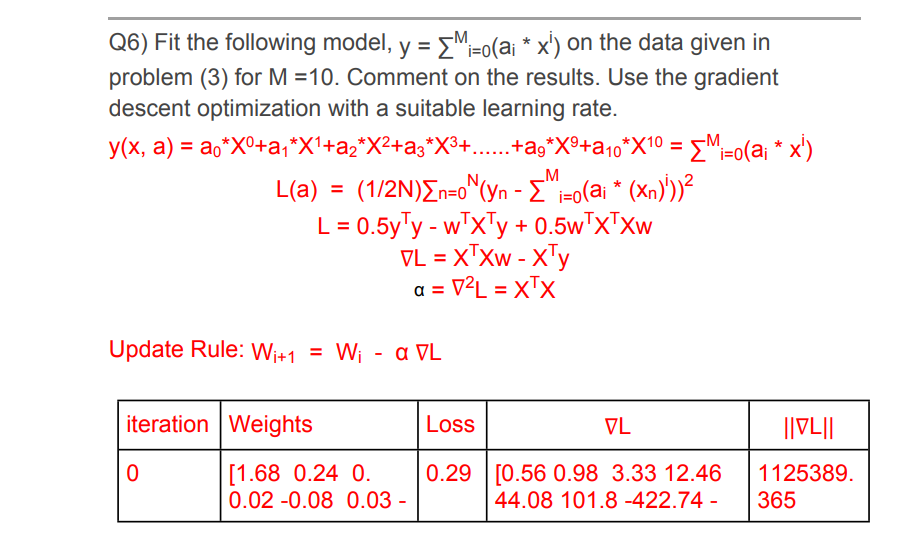
The second rule is also correct as you will substitute alpha for 1/f’’(x) and this will return you to the first formula as they are both the same thing actually.

Thanks alot

Q31) what does the given function refer to? Y = sum(ai\*xi)

Isn't the loss function supposed to be the difference between the true value and the output of my model? So why did we take the difference with this function?

Also how is the weight calculated in the first iteration?



This function is the model output, it’s polynomial curve fitting (non linear output)

**please check page 11 at slide 5 (regression)**

Isn't the loss function supposed to be the difference between the true value and the output of my model? So why did we take the difference with this function?

we took the difference between the labels and this function because, of course you know now :), this function is the model output

W is computed from the learning rule after the first iteration

The first weights are initialized with zeros

Check the problem code for more info:

https://drive.google.com/file/d/1uq8Kfksa6E5spYtTRsKyoatEJ2dnLnok/view?usp=sharingWhy did we calculate (1,255,255,255) first before (1,0,255,255)? And why do we keep repeating values of Xm that we already computed?

The table is only updated when the weights are updated … as this is online training we update the weights in the for loop but we do this when yWx<=0 … the iterations we didn’t include had no weights or delta update as the yWx>0.

I understand this, I calculated wrong, but what I’m asking is this: further down the table, we apply Xm=(1,0,0,0) again along with other points, why?

Yes, because we keep iterating over the dataset until we reach the min value (this iteration happens in two levels … one over the while loop that is maintained by delta being bigger than epsilon … and the other over the for loop where we take every sample into consideration.

And when will delta<epsilon? At the end of the table =, delta is still a matrix with big values?

If you made one more step you will find out that delta will be zero and no update will happen to the weights … and that’s why we stopped (we only care about the steps that the weights got updated in)

Then why didn’t we take the weight at delta =0? And which next iteration, when i substitute with the following XM or the next XM that causes yWx<=0 ?

At delta equal zero the weights will be the same … when we run over a full loop over the whole dataset and the delta is zero then we terminate.

Q) in SVM change in bipolar, the condition change will be 1-ymxm<=0, correct?

The condition will be if yWx <= 1 then delta = delta -ymxm otherwise the grad will be zero.

Q) in the above question about online training, when we update delta, we directly sum the previous value of delta with the value in column -YmXm for the current iteration right? If so then the given answers are wrong right, as in the 4th row for example?

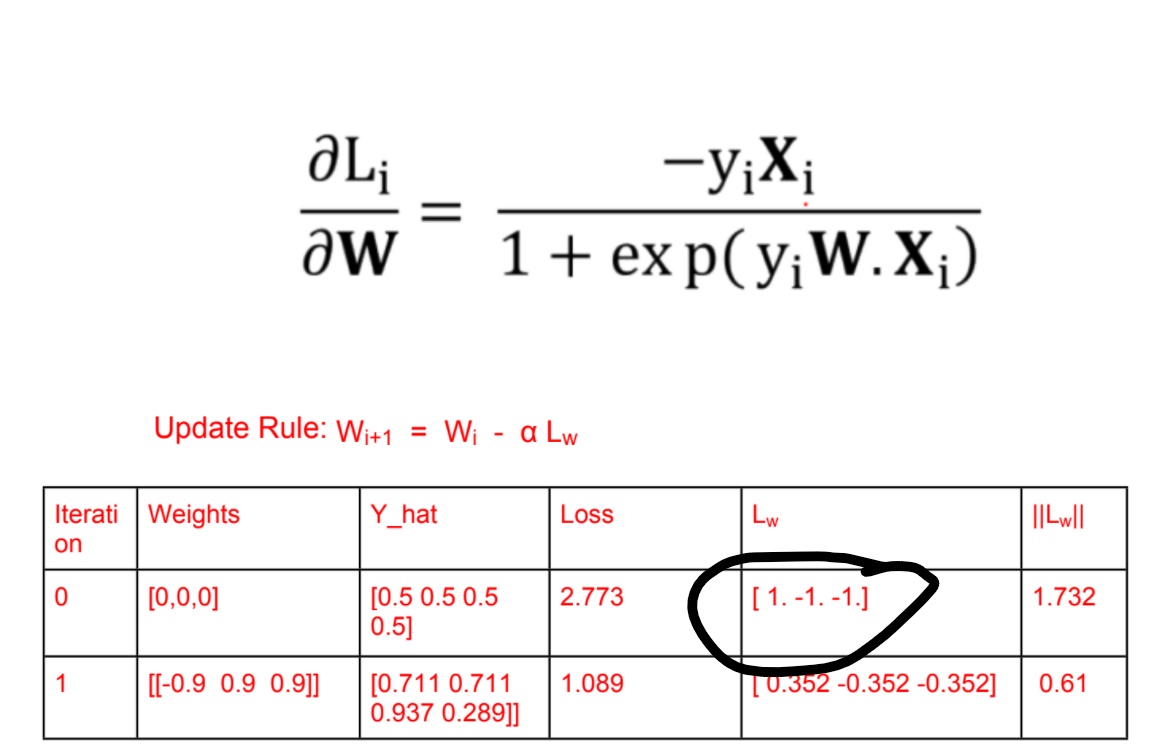
Take into consideration that we zero out the delta with every while loop iteration.

Q) Should I remember all loss functions used with all activation functions or they will be given?

You should remember them. No formula sheet in the midterm exam.

Q) Q8 sheet 4 .. how could i compute the gradient here ? I can’t figure out how I will make a dot product between y and x and they don’t have the same inner dimensions?

The vectors y & x are multiplied element wise multiplication, not dot product.



Q) how did we differentiate here, do we differentiate log or sign?

If we differentiate the log then there should be the term y/2 -½ + y\_hat under the fraction

Yes, we differentiate the log, where L = log(y’)

Check slides 5 at page 23

but here the loss = at the picture not log (y hat) ?

Then it’s wrong in the sheet right?

I will check it and get back to you Ok

**Update**: The answer is not wrong, check the slides from page 22

To make things simpler, as we deal with bipolar model where y = +1 or -1

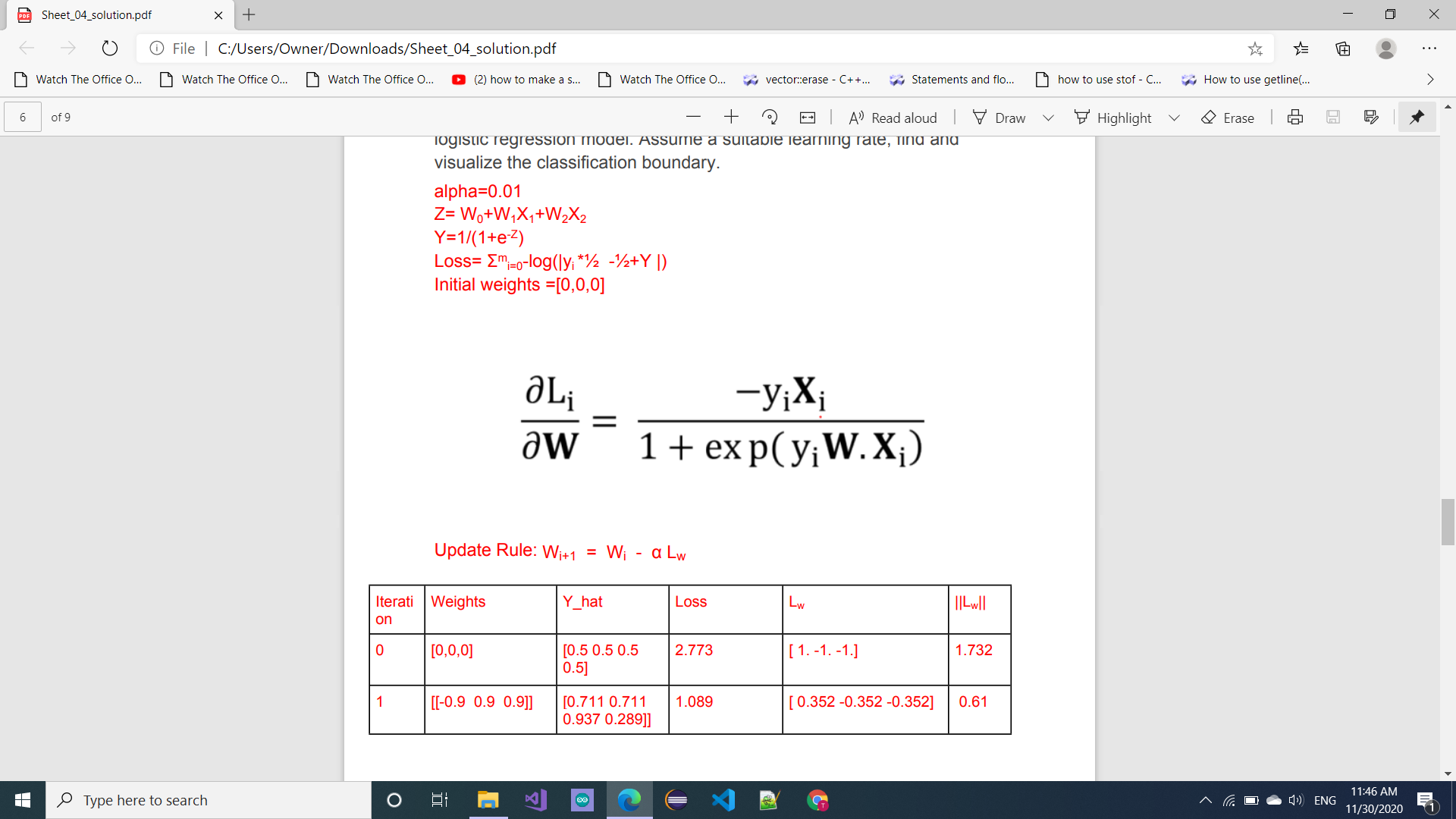
We substitute at the loss

When y = +1 and y = -1

So the loss becomes

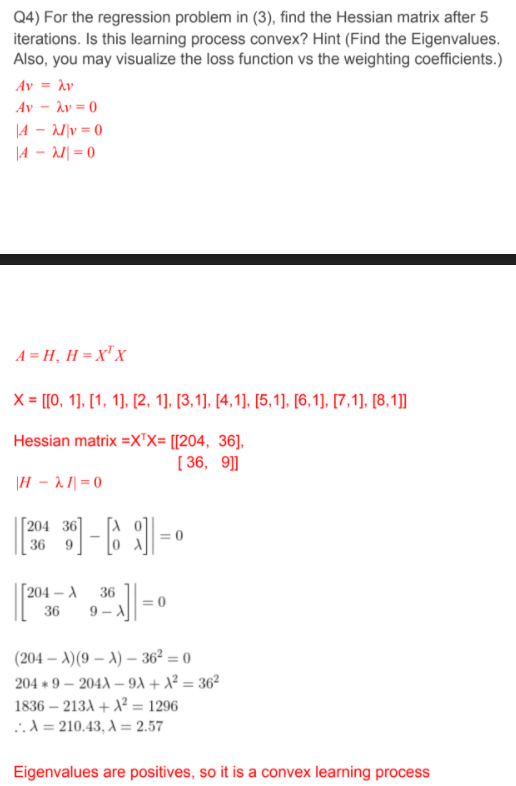
And

So it’s wrong you can use this formula if it’s bipolar perceptron and y = +1 or -1



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

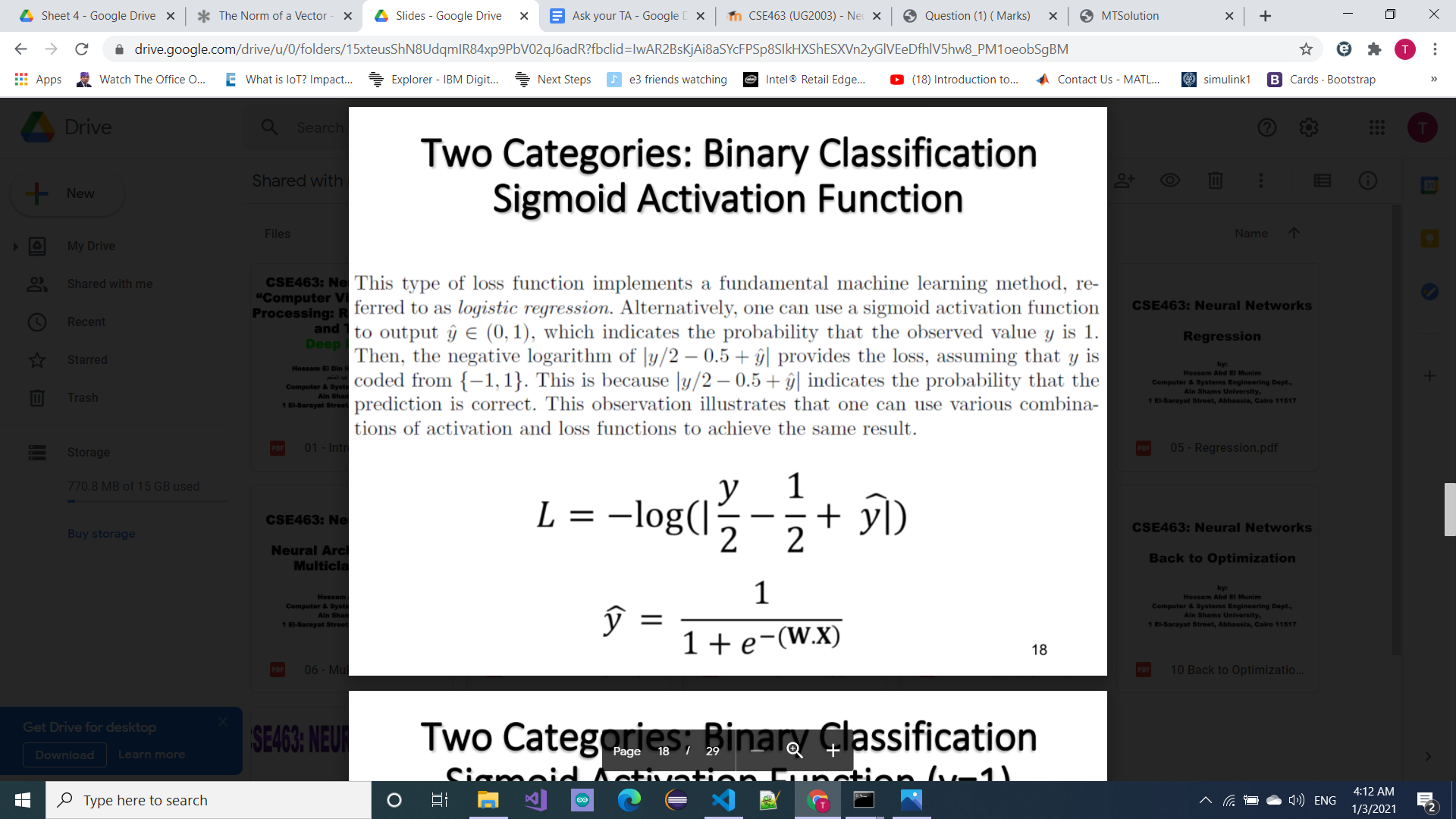
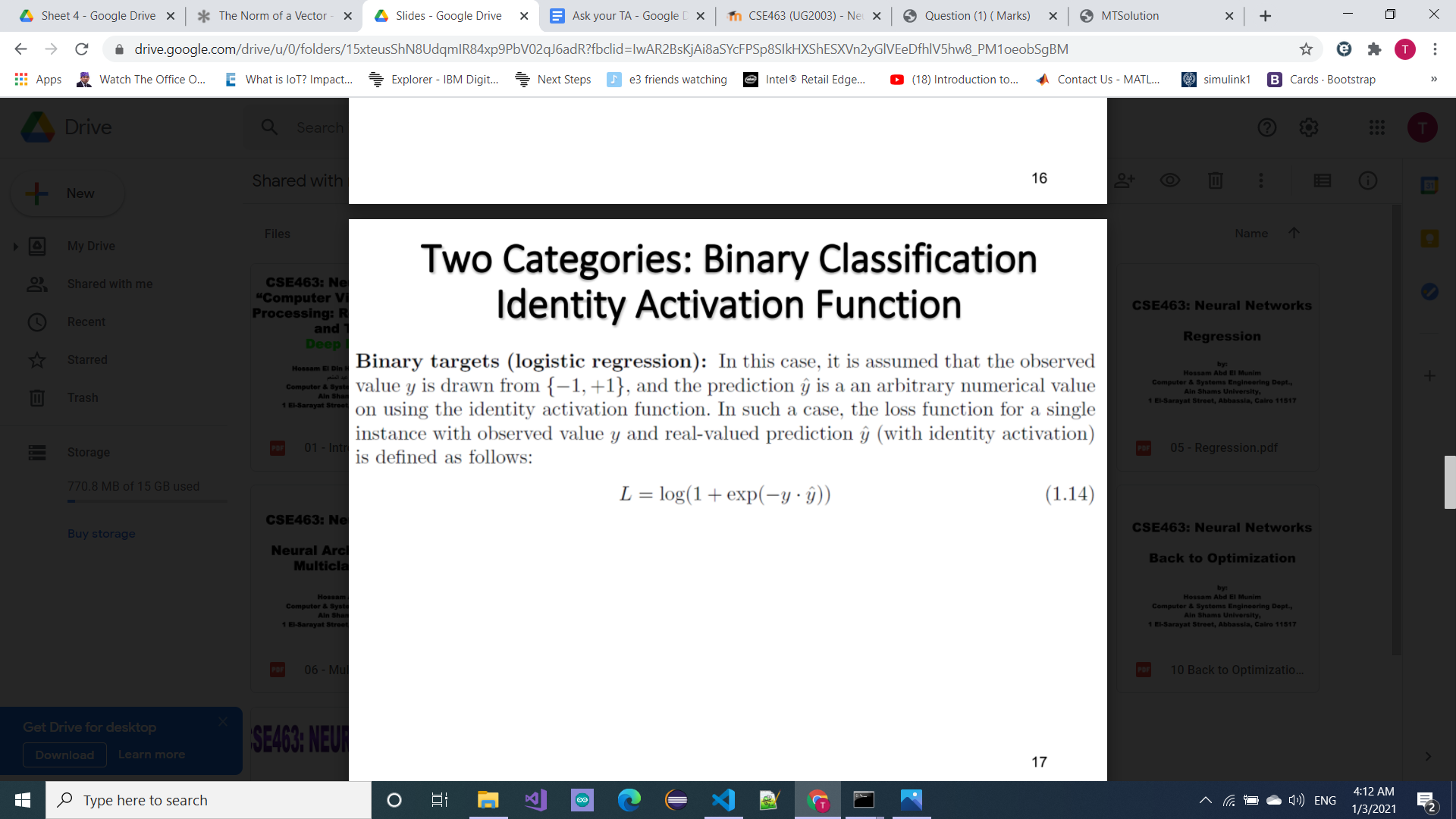
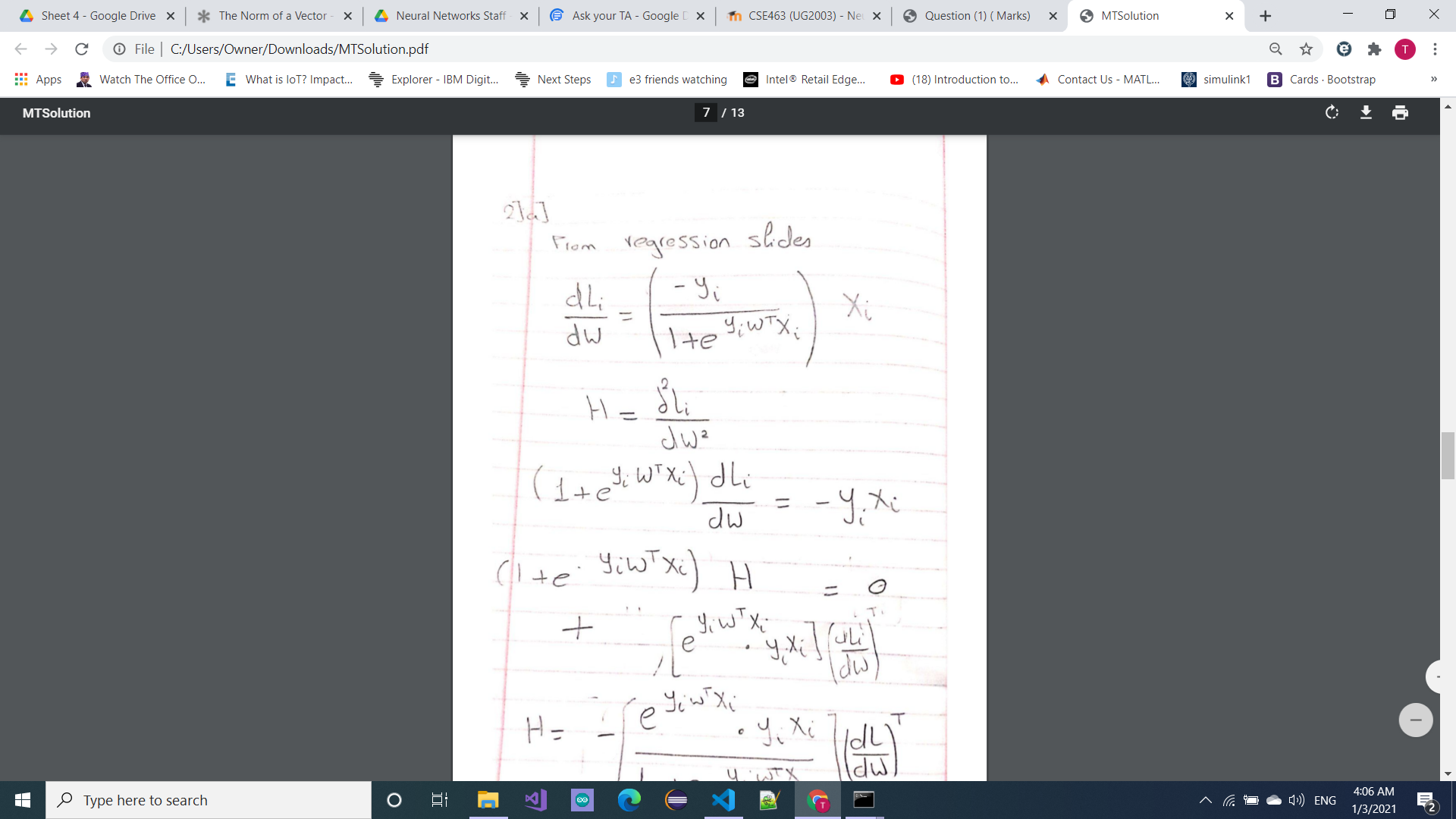
Q) Could you explain what’s happening here? Also we’ve already shown that the loss function used in linear regression is always convex (Sheet 4 Pb. 1) so why do we need to find the Eigenvalues?



Q) in the midterm solution why did we assume that the loss function is L= -log( |0.5\*yi - 0.5+Y|) → for logistic with sigmoid

And not L= log(1+exp(-y\*y\_hat)) -> for logistic with identity

Both functions are mentioned in the slides for logistic regression, and in the exam he didn’t specify the activation function



Both are the same, it’s just that the one with the exponential has the sigmoid substituted inside the first loss function.

Sheet 3 Q4 c and d

مش المفروض ال total loss اقسم في الاخر علي عدد ال training examples؟

هو total اقرب لل sum من average فالقسمة على N مش شايفها افضل اجابة (عموما لو عايز تكتب الاجابتين فى الامتحان مش هيبقى فيه ضرر)

Q) Is there any other loss function to use in back propagation problems except the euclidean distance loss that was used in the sheet solution?

Yes, Question 1 in sheet 7 is a good example for something like this.

(as long as you know the derivative of the loss function you can manufacture any neural network you need) ,So how to know the suitable loss function ( except the case of softmax which is in sheet 7 q 1) ?

Well, it’s an elimination process … in sheet 6 for example we have chosen the square of difference loss function because the outputs were not classes but values … In other circumstances like sheet 7 q1 a loss function like the multiclass log likelihood can be of a great use.

ََQ)

Do I have to memorize the CNN architectures’ parameters in the slides for the final exam?

No, just remember the output shape formula and you will be able to deduce the answers

Q)

In the optimization algorithms (sheet 7 solution), we are updating the parameters after each example

and earlier in online training, we were accumulating the grads of the previous examples in delta before updating instead of updating using the grad of the current example directly

so I am confused when to update using the grad of the current example directly and when to accumulate the grads of the previous examples before updating

Actually starting from the momentum idea in sheet 7 we are accumulating the gradients just like what we saw in the online training.

Q33)

هو ممكن في ال polynomial curve fitting ال input feature vector ميبقاش 1D ؟؟

لو ينفع يبقى اكتر ال y cap هتبقى عاملة ازاي ؟

ال y\_cap هتفضل رقم واحد … لأن ال regression عبارة عن

Y\_cap = W.X = [w1 w2 w3 …] . [x1 x2 x3 …] = w1.x1 + w2.x2 ...

لو اشتغلنا closed form الناتج هيكون vector 1D بينما ال X هتكون matrix وال w هتكون vector 1D

Q) What are the possible loss functions to be used with logistic regression (classification) with tanh activation?

Mean square error is an easy option even if it’s not the best, classification loss functions like the bipolar loss, the hinge loss and the log likelihood loss are also viable solutions but will need some tweaking.

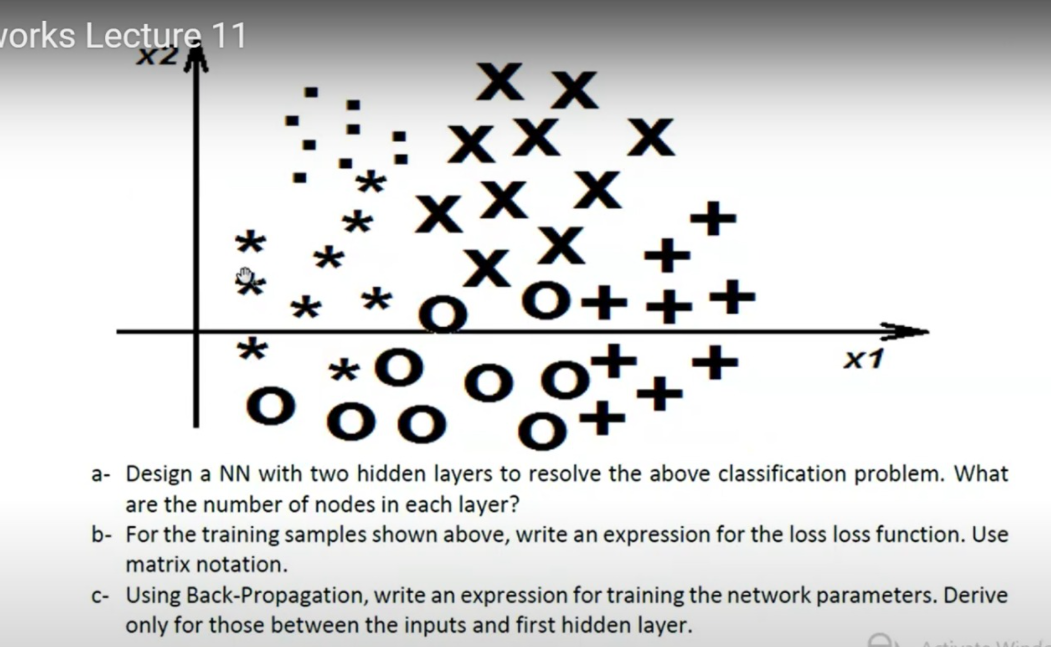
Q) in multiclass bipolar criterion , if two nodes have the same wx, should I update both ? and what if one of them was the correct label , then should I consider no loss ?

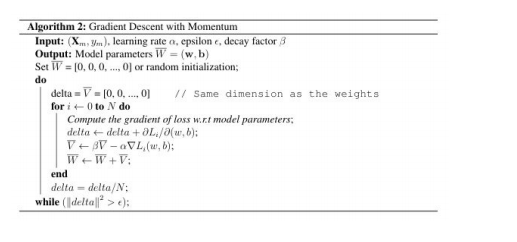
* How to determine number of neurons in hidden layer

There is a huge number of researchers still trying to figure this question out, there is one rule of thumb that says the number of hidden neurons in a hidden layer should be between the number of output neurons and input neurons … it’s not a bad rule of thumb as the number of classes on the output layer should affect the number of lines we draw to classify our data.

Q) In previous exams, When the doctor state to derive an objective function to train the NN “With Tanh activation” What does he mean with that? Also would it be correct to just state that I will be using difference square loss as the objective fn ?

He is just asking to choose a loss function for your problem, i don’t think mse will fail or stop working (it may not be the best solution though)



Q) sheet 7 c.

1. can the momentum algorithm (along with other optimization algorithms) be used with batch/mini-batch training as well?

Yes, you can see the PyTorch docs using momentum in GD optimizers with batch\_size.

2. Why is the V vector initialized on each iteration? shouldn't it be accumulated for the whole training process?

Well, we designed the algorithm like this because we started from the online training paradigm where we update the weights on every sample so the momentum will be the accumulation over the whole dataset.

If we have chosen to start from a batch GD point of view your idea will be valid and the V will accumulate across the whole training process.

Q) I have a problem with how to determine the loss function based on the output, for example if the output is 0->1 ? -1 or +1? -1 -> +1? Different random values as 1,2 3, ?

You can choose a default loss function and work with it like the mse (it won’t be the best answer in some cases but it will work), or you can remember that a log likelihood loss -ln(y/2 - 0.5 + y\_hat) is a good choice for binary classification.

In a 0 to 1 range for the output neuron and -1 to 1 in the labels you can use the log likelihood we just shown before.

With an output range between -1 to 1 you can use the log likelihood with a tweak -ln(y/2 - 0.5 + y\_hat/2 + 0.5)

And when to use soft max?

Multi-Classification

Q) why the performance of the neural network is getting worse with increasing the number of hidden layers. How can we overcome this problem? How can we modify the network?

How can we get better optimization?

There are possibly two reasons for something like this, Vanishing GD and Overfitting … Vanishing GD can be solved with relu or skip connections (didn’t talk about it in the course), Overfitting can be reduced with solutions like dropout, regularization or simply adding more data.

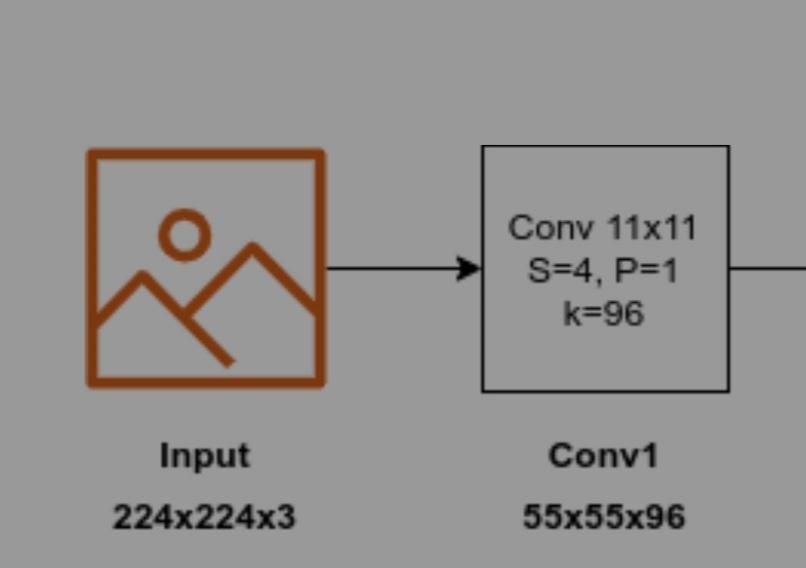
We can get faster training times with normalization

Q) Is this a mistake in the slides?

The output dimensions of the conv layer is

(224-11+2)/4 +1 = 54.75

So the dimension is floor(54.75) = 54 not 55

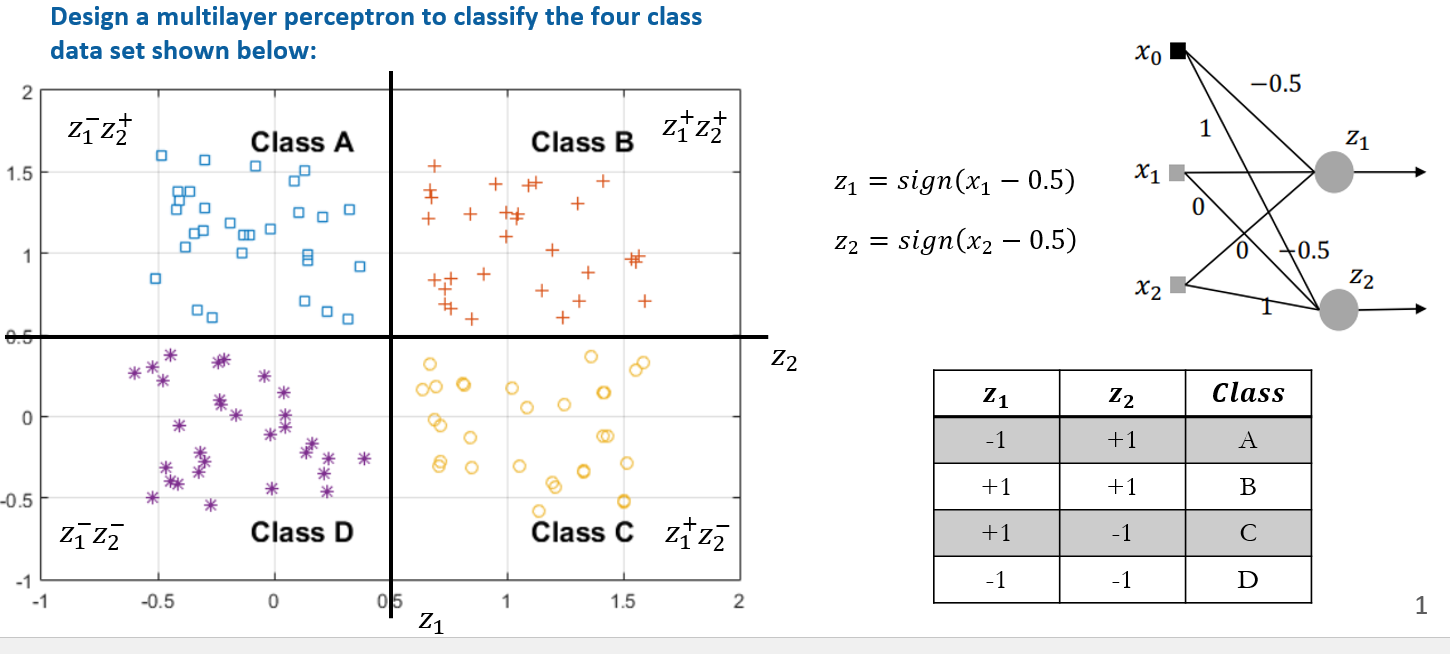


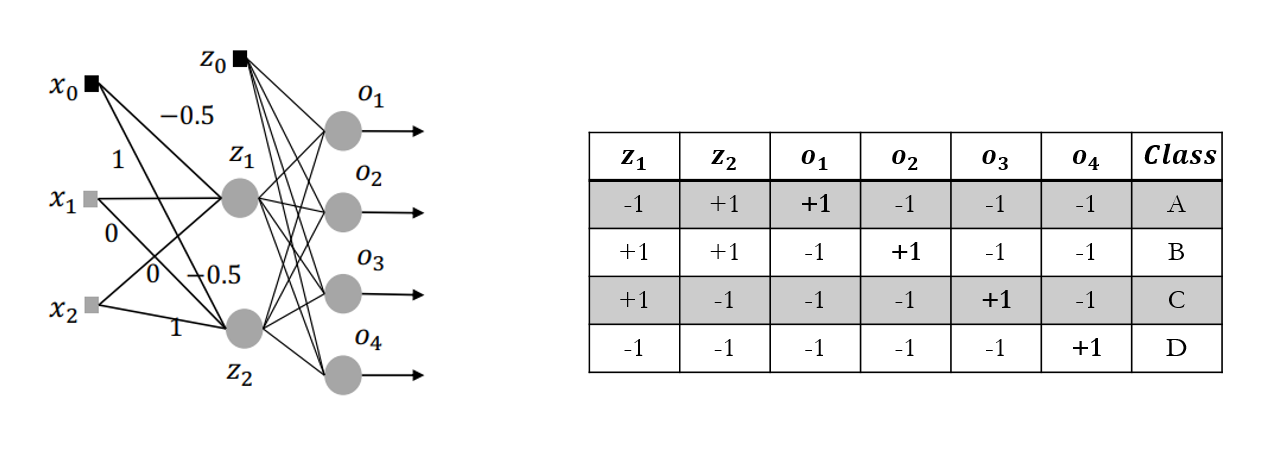
To cover the ¾ part take the ceil of the number so it’s 55

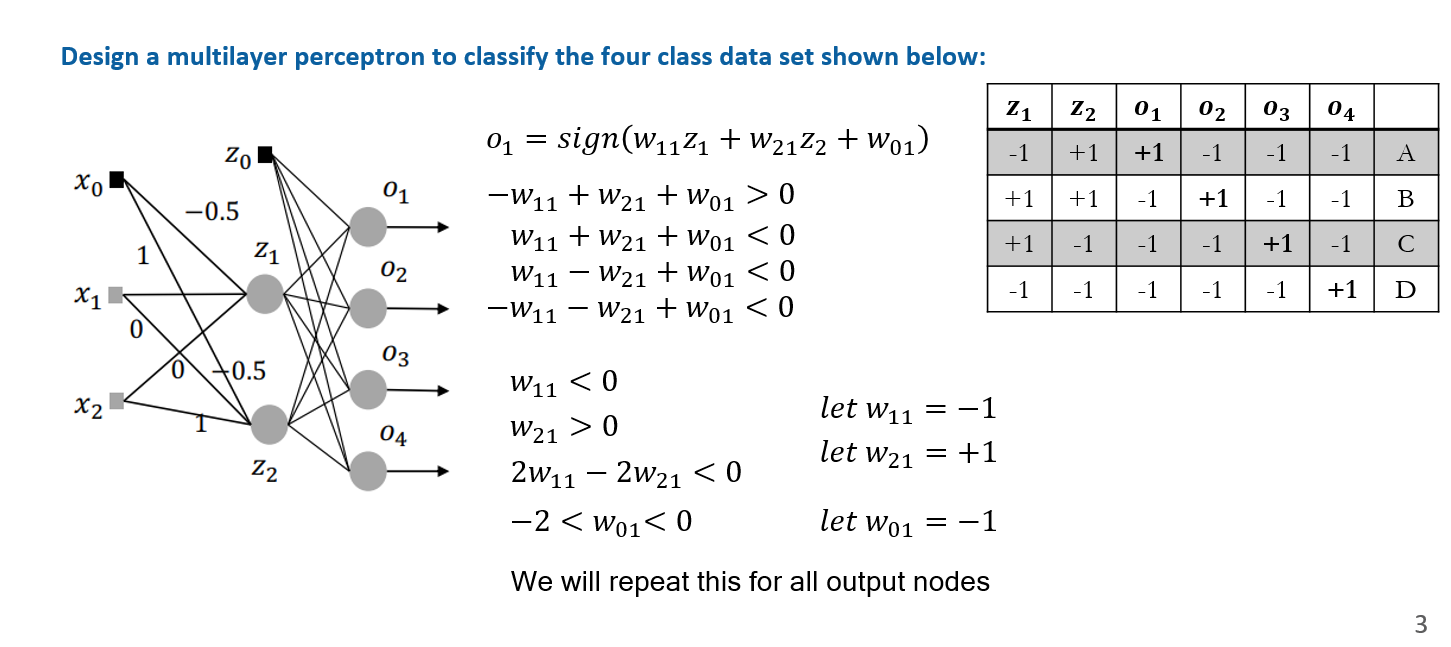
Exams:

a- Design a multilayer perceptron to classify the four class data set shown below. Use a sign activation function and justify your answer.





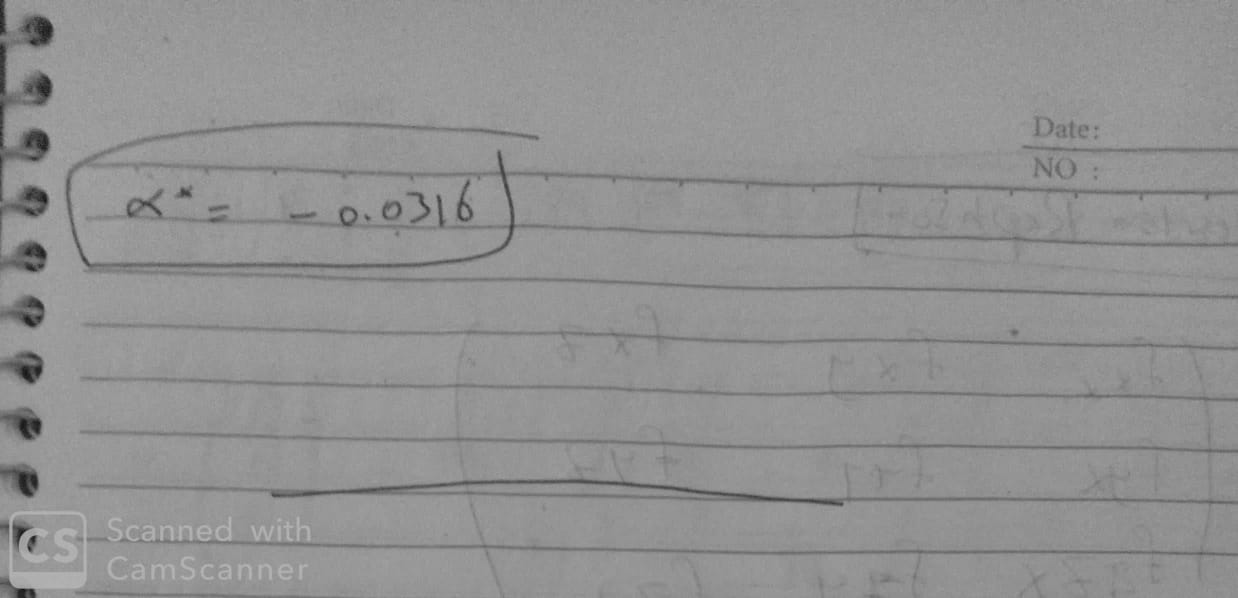
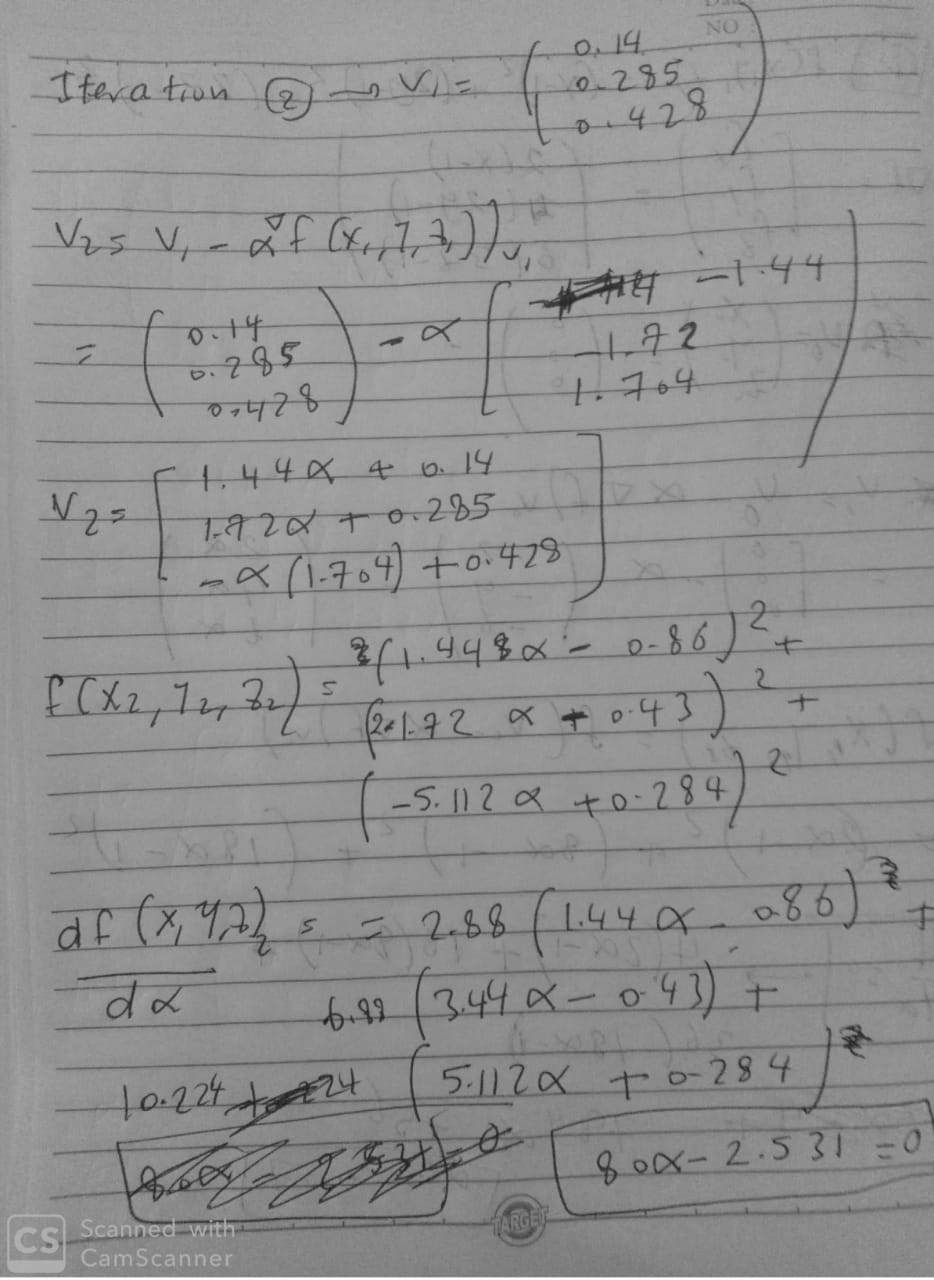
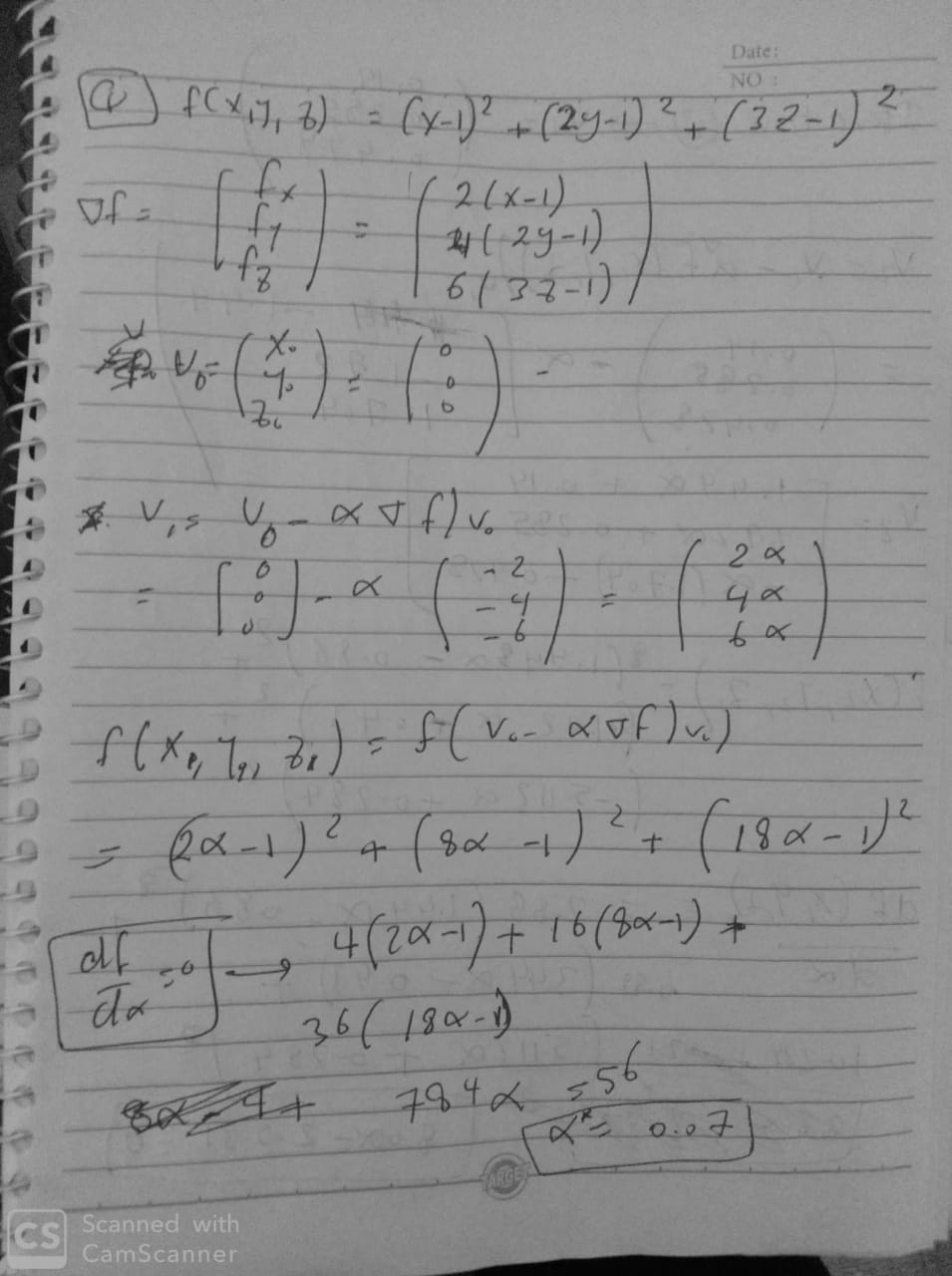




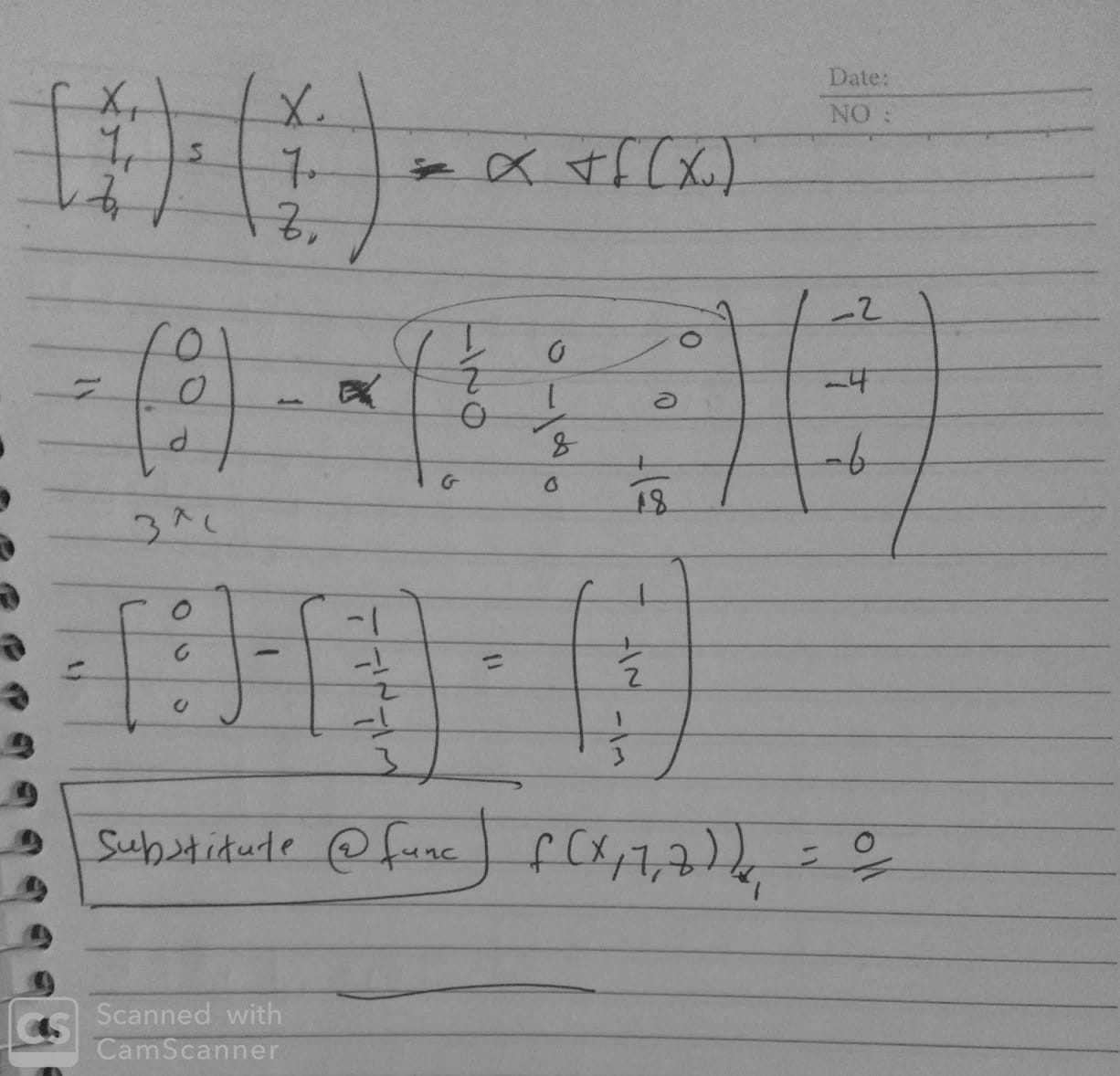
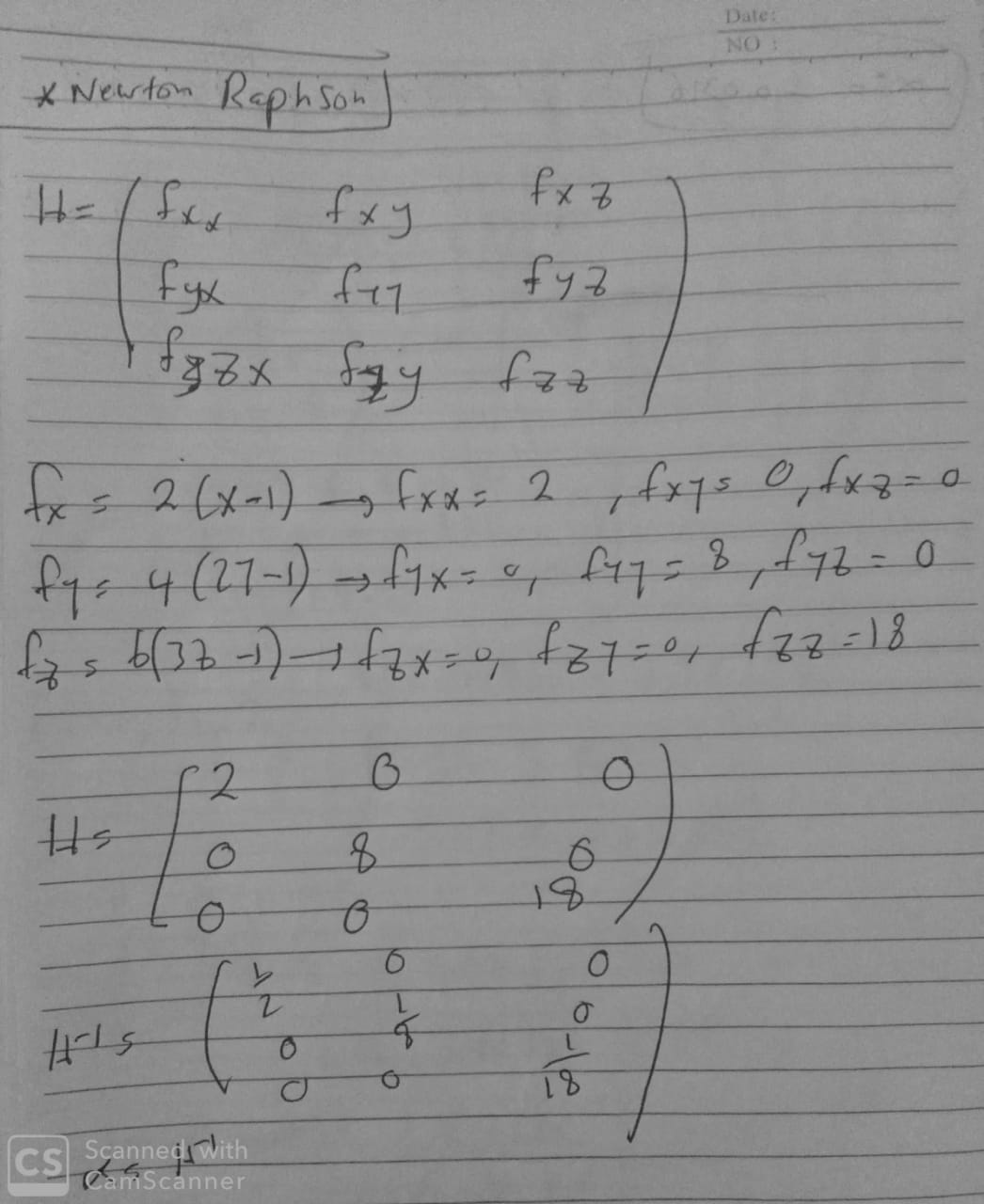
b- A neural network has: two nodes in the input layer, two hidden layers with two nodes each, and one node in the output layer. Derive the learning rule for each network parameter assuming a tanh activation function.

------------------------------------------------------------------------------------------------

Q) b- Use two iterations of the steepest descent algorithm to find the minimum point of the function f(x, y, z) = (x-1)2 + (2y-1)2 + (3z-1)2. Start from the point (0, 0, 0)T.



c- Repeat part (b) using the Newton-Raphson’s algorithm. Use the same initial point and compare the results.



------------------------------------------------------------------------------------------------

Q) A neural network has three input and one output nodes. Two hidden layers are included with 5 nodes each. A tanh activation function is assumed while the network is used as a binary classifier.

a) Draw a schematic diagram for the network specifying all network parameters.

b) Show how to find the network output for arbitrary input in matrix form.

c) For the following training data {(X1,y1), (X2,y2),…, (XN,yN)} find an expression for the training objective function. Note that the given pairs are for inputs and their desired outputs where X ϵ R3 and y ϵ {-1, 1}.

d) Derive the learning rule for each network parameter using the training data given above. You must use matrix notation.

e) Using matrix notation, write a MATLAB code for training this network showing specifically the stopping criterion.

------------------------------------------------------------------------------------------------

a) Explain in brief why the performance of the neural network is getting worse with increasing the number of hidden layers. How can we overcome this problem? How can we modify the network?

Because of the vanishing gradients problem, as the grads go below zero, they can vanish when we backpropagate with the deep neural nets (increasing number of hidden layers) which affects our learning.

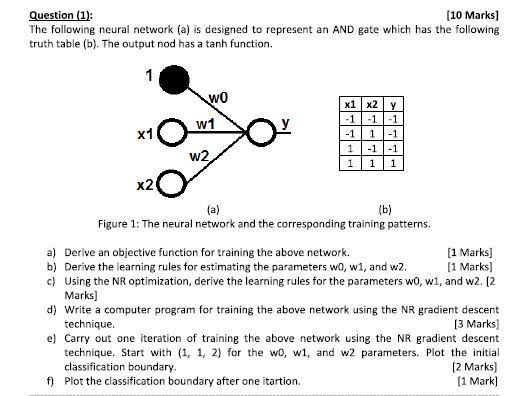
We can overcome vanishing gradient descent by:

* Choosing activation functions with higher gradient value (ReLU).
* Batch normalizations technique.

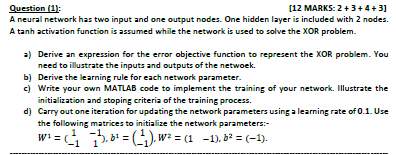
How can we get better optimization?

------------------------------------------------------------------------------------------------

Explain the difference between deterministic and stochastic optimization techniques, give examples for each one.



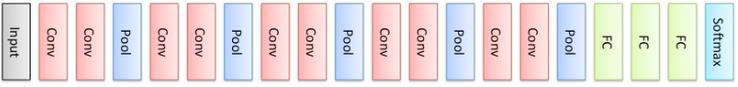
------------------------------------------------------------------------------------------------



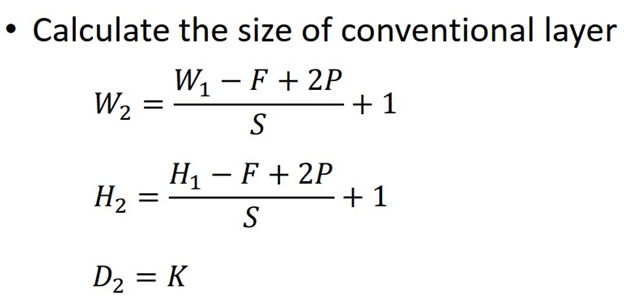
-----------------------------------------------------------------------------------------------

Check sheet 3

For the VGGNET structure given below, assume that the filter size is 3X3 in the convolutional layers. Consider a stride of 1 with zero padding. Discuss and calculate the size of each stage. The input image size will be 256 X 256.



Please Check Sheet 7 Q14



------------------------------------------------------------------------------------------------