Test 01 – Math

Arthur J. Redfern
arthur.redfern@utdallas.edu
Oct 02, 2019

---

0 Instructions

- There are 35 numbered questions with indicated point values that sum to 100
- Write all of your answers clearly on the answer sheet and turn it in
- No reference materials are allowed
- No help from others is allowed
- Correct answers in red

1 Test

Strategy [8 points]

1. [8 points] Circle true or false for each of the following statements

True / False	Many tasks can be framed as a classification or regression problem
True / False	Neural networks are universal function approximators under some mild conditions and can be used to map from data to classes or values
True / False	A function always exists that maps an arbitrary input to arbitrary output
True / False	It's possible to design xNNs to exploit different types of structure in data
True / False	It's not possible to end to end train xNNs that contain max pooling or ReLU layers because these layers are not differentiable
True / False	Software and hardware exists for efficient xNN implementations
True / False	xNNs provide state of the art results for many applications
True / False	A 3 layer xNN is the best choice for most applications

Data [8 points]

2. [4 points] Consider an image classification data set **X** with 2^6 classes and 2^{10} labeled examples per class for a total of 2^{16} labeled examples. What is the total information content of all of the labels?

Information content of all labels = number of labels * information per label
$= 2^{16} \log_2(2^6)$
$= 6 * 2^{16}$ bits

3. [2 points] Consider a dataset of color images where each image **X** is a 3 x 1024 x 2048 tensor composed of elements X(c, h, w) with per channel mean μ_c and variance σ_c^2. How would you transform the data set **X** to a per channel 0 mean unit variance data set $\mathbf{X}_{norm}$? $\mathbf{X}_{norm}$ =

$\mathbf{X}_{norm}(c, :, :) = (\mathbf{X}(c, :, :) - \mu_c)/\sigma_c$, c = 0, 1 and 2

4. [2 points] List the 2 image data augmentation strategies used during training data pre processing in the example code for training CNNs.

Random crop and left right flip

Weight initialization [2 points]

5. [2 points] Let's say I know the mean μ and variance σ^2 of an individual weight in a network, but I know nothing else about it. What is the entropy maximizing distribution to sample from to initialize this weight?

The entropy maximizing distribution is Gaussian with mean μ and variance σ^2

Feature extraction – CNN layers [11 points]

Consider a CNN style 2D convolution layer $\mathbf{Y}^{3D} = \mathbf{f}(\mathbf{H}^{4D} \otimes \mathbf{X}_{padded}^{3D} + \mathbf{V}^{3D})$ where $\otimes$ is used to denote CNN style 2D convolution and

Input:	$\mathbf{X}^{3D}$ with dimensions N_i x L_r x L_c
Pad:	P_r (= **sum** of top + bottom pad), P_c (= **sum** of left + right pad)
Padded input:	$\mathbf{X}_{padded}^{3D}$ with dimensions N_i x $(L_r + P_r)$ x $(L_c + P_c)$
Filter:	$\mathbf{H}^{4D}$ with dimensions N_o x N_i x F_r x F_c (no striding)
Bias:	$\mathbf{V}^{3D}$ with dimensions N_o x M_r x M_c and constant per n_o
Nonlinearity:	**f** of type ReLU
Output:	$\mathbf{Y}^{3D}$ with dimensions N_o x M_r x M_c

6. [2 points] What are P_r and P_c such that $M_r = L_r$ and $M_c = L_c$?

$P_r = F_r - 1$
$P_c = F_c - 1$

7. [3 points] When padding is chosen such that $M_r = L_r$ and $M_c = L_c$, what are the dimensions (rows x columns) of each of the matrices that result from the above CNN style 2D convolution

operation ⊗ lowered to matrix matrix multiplication $\mathbf{Y}^{2D} = \mathbf{H}^{2D}\,\mathbf{X}_{filter}{}^{2D}$ when there are N_o rows in $\mathbf{Y}^{2D}$?

$\mathbf{Y}^{2D}$ dimensions are N_o x (L_r*L_c)
$\mathbf{H}^{2D}$ dimensions are N_o x $(N_i*F_r*F_c)$
$\mathbf{X}_{filter}{}^{2D}$ dimensions are $(N_i*F_r*F_c)$ x (L_r*L_c)

8. [2 points] How many MACs are required in the standard matrix multiplication based implementation of CNN style 2D convolution with the pad chosen as above (not including the bias and nonlinearity)?

Number of MACs = $L_r*L_c*N_o*N_i*F_r*F_c$

9. [2 points] Assume that the layer is part of a network and trained for a 3 x 32 x 64 input **X**. Is the convolution operation mathematically compatible with a 3 x 96 x 96 input **X**? Circle yes or no.

Yes / No

10. [2 points] Consider the L_r x L_c output feature map at channel n_o. Are the same filter coefficients used for mapping inputs to outputs for all L_r*L_c output pixels in feature map n_o? Circle yes or no.

Yes / No

Feature extraction – RNN layers [7 points]

Consider a standard RNN layer $\mathbf{y}_t^T = \mathbf{f}(\mathbf{x}_t^T\,\mathbf{H} + \mathbf{y}_{t-1}^T\,\mathbf{G} + \mathbf{v}^T)$ with

Output at time t:	$\mathbf{y}_t$ with dimensions 1 x N_o
Nonlinearity:	**f** of type ReLU
Input at time t:	$\mathbf{x}_t^T$ with dimensions 1 x N_i
Input weight matrix:	**H** with dimensions N_i x N_o
Output at time t–1:	$\mathbf{y}_{t-1}$ with dimensions 1 x N_o
State weight matrix:	**G** with dimensions N_o x N_o
Bias:	**v** with dimensions 1 x N_o

and the sequential set of inputs $\{\mathbf{x}_0^T, \mathbf{x}_1^T, \mathbf{x}_2^T, \mathbf{x}_3^T, \mathbf{x}_4^T\}$ and outputs $\{\mathbf{y}_0^T, \mathbf{y}_1^T, \mathbf{y}_2^T, \mathbf{y}_3^T, \mathbf{y}_4^T\}$ with $\mathbf{y}_{-1}^T = \mathbf{0}^T$.

11. [2 points] Can all of the input terms $\{\mathbf{x}_t^T\,\mathbf{H}\}$ for t = 0, …, 4 be computed parallel (at the same time)? Circle yes or no.

Yes / No

12. [2 points] Can all of the state terms $\{\mathbf{y}_{t-1}^T\,\mathbf{G}\}$ for t = 0, …, 4 be computed parallel (at the same time)? Circle yes or no.

Yes / No

13. [3 points] Assume that there's an error in output $\mathbf{y}_1$. What other output(s) will potentially be in error because of this?

Output(s): $\mathbf{y}_2$, $\mathbf{y}_3$, $\mathbf{y}_4$

Feature extraction – self attention layers [14 points]

Consider a single headed self attention layer $\mathbf{Y}^T = \mathbf{A}^T \mathbf{X}^T \mathbf{H}$ with

Output matrix:	$\mathbf{Y}^T$ with dims M x N composed of M output vectors, N features each
Attention matrix	$\mathbf{A}^T$ with dims M x M where each row is a valid pmf
Input matrix:	$\mathbf{X}^T$ with dims M x K composed of M input vectors, K features each
Weight matrix:	$\mathbf{H}$ with dims K x N

14. [4 points] Circle true or false for each of the following statements

True / False The attention matrix $\mathbf{A}^T$ is input data independent.

True / False The attention matrix $\mathbf{A}^T$ mixes $\mathbf{X}^T$ across vectors.

True / False The weight matrix $\mathbf{H}$ mixes $\mathbf{X}^T$ across features.

True / False If the attention matrix $\mathbf{A}^T$ is an identity matrix then multiple input vectors contribute to each output vector.

15. [4 points] Consider the term $\mathbf{X}^T \mathbf{W}_q \mathbf{W}_k^T \mathbf{X}$ where $\mathbf{X}^T$ is defined as above, $\mathbf{W}_q$ is K x P and $\mathbf{W}_k^T$ is P x K. What is the constraint on P such that the number of MACs required to compute $\mathbf{X}^T \mathbf{W}_q \mathbf{W}_k^T \mathbf{X}$ is less than the number of MACs required to compute $\mathbf{X}^T \mathbf{W}_{qk} \mathbf{X}$ where $\mathbf{W}_{qk}$ is K x K?

MKP + PKM + MPM < MKK + MKM

2KP + PM < KK + KM

P < (KK + KM) / (2K + M)

Side note: if K >> M then ~ P < K/2 (so this answer is also ok) and if K = M then P < 2K/3

Consider a hybrid self attention – dense layer $\mathbf{Y}^T = \mathbf{f}(\mathbf{A}^T \mathbf{X}^T \mathbf{H} + \mathbf{1}\,\mathbf{v}^T)$ where $\mathbf{f}()$ is a ReLU function, $\mathbf{1}$ is a M x 1 vector of 1s, $\mathbf{v}^T$ is a 1 x N vector of bias values and other terms are defined as above.

16. [2 points] Circle true or false for each of the following statements

True / False This generalizes self attention to an affine transformation.

True / False This has the ability to 0 out negatively aligned features within vectors.

17. [4 points] Taking a similar approach, write down an equation for a hybrid self attention – RNN layer that enables mixing across vectors, across features and across time. Assume input $\mathbf{X}_t^T$ at time t.

$\mathbf{Y}_t^T = \mathbf{f}(\mathbf{A}_t^T \mathbf{X}_t^T \mathbf{H} + \mathbf{Y}_{t-1}^T \mathbf{G} + \mathbf{1}\,\mathbf{v}^T)$ where $\mathbf{G}$ is N x N

Feature extraction – pooling layers [2 points]

Consider an input feature map **X** of dimension N_i x L_r x L_c where N_i, L_r and L_c are all divisible by 4.

18. [2 points] For a 4x4/4 average pooling layer, what are the dimensions of the output **Y**?
    The dimensions of **Y** are N_i x (L_r/4) x (L_c/4)

Feature extraction – nonlinearity choices [8 points]

19. [1 points] Circle true or false for each of the following statements
    True / False   It's possible to have a deep neural network without nonlinearities

20. [4 points] Assume that inputs to ReLU are independent random variables with uniform pmfs that can be represented by a 9 bit signed integer in {-256, …, 255}. If the output of ReLU is optimally coded, approximately how many bits (round to the nearest integer) are needed to represent each output?
    257 out of 512 values map to 0
    255 out of 512 values map to {1, …, 255} with the probability of each being 1/512
    $H = -(257/512)\log_2(257/512) - 255\,(1/512)\log_2(1/512) \approx -(1/2)(-1) - (1/2)(-9) = 5$ bits

21. [3 points] What type of common xNN nonlinearity …
    A. Zeros out negative outputs, does not change positive outputs
       ReLU
    B. Constrains the output to (-1, 1)
       Tanh
    C. Constrains the output to (0, 1) and is frequently used as a gate
       Sigmoid

Prediction [8 points]

Consider a network designed for image classification with the following sequential structure

Input $\mathbf{X}_0$ with dimensions N_i x L_r x L_c
Multiple CNN and pooling layers
Feature map $\mathbf{X}_d$ with dimensions N_d x (L_r/D) x (L_c/D)
Global average pooling layer with N_d x 1 output $\mathbf{x}_p$
Dense layer $\mathbf{x}_p^T \mathbf{H} + \mathbf{v}^T$ with no nonlinearity
Output $\mathbf{y}^T$ with dimensions 1 x C where C = the number of classes

22. [2 points] Circle true or false for each of the following statements
    True / False   The global average pooling layer allows the dense layer to be mathematically compatible with feature map $\mathbf{X}_d$ given input $\mathbf{X}_0$ with ~ arbitrary rows and cols

23. [2 points] What are the dimensions of the dense layer weight matrix **H**?

    N_d x C

24. [2 points] What is the relationship between N_d and C for top performing image classification networks?

    $N_d > C$

25. [2 points] What is the arithmetic intensity of the dense layer (ignore the bias) in terms of MACs / data movement?

    $(N_d\ C) / (C + N_d + N_d\ C) \approx 1$ for large N_d and C

Error computation [8 points]

26. [2 points] Write the equation for softmax for transforming 1 x N input $\mathbf{x}^T$ to 1 x N output $\mathbf{p}^T$ using n to index elements within the input and output vectors. p(n) = <ans>, n = 0, …, N – 1

    $p(n) = (1/\Sigma_n e^{x(n)})\ e^{x(n)}$, n = 0, …, N – 1

27. [2 points] Circle true or false for each of the following statements

    True / False KL divergence is a method for mapping 2 pmfs to a divergence

    True / False KL divergence reduces to cross entropy when the true pmf is 1 hot

28. [2 points] What does it do (in a few words) to the output of softmax if the input is scaled by a constant > 1?

    It makes the output a peakier pmf

29. [2 points] Write the equation for MSE for mapping an 1 x N network output $\mathbf{y}^T$ and a 1 x N true output $\mathbf{y}^{T*}$ to an error e. e =

    $e = (1/N)\ (\mathbf{y}^T - \mathbf{y}^{T*})\ (\mathbf{y} - \mathbf{y}^*)$

Back propagation [8 points]

30. [2 points] Circle true or false for each of the following statements

    True / False A graph for back propagation can automatically be constructed from the graph for forward propagation

    True / False For end to end training with back propagation, it's ok if a few layers are not differentiable or sub differentiable.

31. [6 points] Consider a scalar residual building block with input x, output $y = x + f_2(f_1(f_0(x)))$ and assume that de/dy, the sensitivity of the error e with respect to the output y, is given. Further, define the following terms:

$x_0 = x$
$x_1 = f_0(x_0)$, df_0/dx_0 is known
$x_2 = f_1(x_1)$, df_1/dx_1 is known
$x_3 = f_2(x_2)$, df_2/dx_2 is known
$y = x + x_3$

Write de/dx, the sensitivity of the error with respect to the input, in terms of de/dy and the above known terms.

$de/dx_0 = (dx_1/dx_0)\ (dx_2/dx_1)\ (dx_3/dx_2)\ (de/dx_3) = (df_0/dx_0)\ (df_1/dx_1)\ (df_2/dx_2)\ (de/dy)$
$de/dx = de/dy + de/dx_0 = de/dy + (df_0/dx_0)\ (df_1/dx_1)\ (df_2/dx_2)\ (de/dy)$

Weight update [16 points]

Given:

A is symmetric positive definite
α is a scalar
Operator $\partial/\partial(\mathbf{h} - \mathbf{h}_0)$ applied to $e(\mathbf{h}_0) = \mathbf{0}$
Operator $\partial/\partial(\mathbf{h} - \mathbf{h}_0)$ applied to $(\mathbf{h} - \mathbf{h}_0)^T \mathbf{g} = \mathbf{g}$
Operator $\partial/\partial(\mathbf{h} - \mathbf{h}_0)$ applied to $0.5\ (\mathbf{h} - \mathbf{h}_0)^T \mathbf{A}\ (\mathbf{h} - \mathbf{h}_0) = \mathbf{A}\ (\mathbf{h} - \mathbf{h}_0)$

32. [4 points] Let error $e(\mathbf{h}) = e(\mathbf{h}_0) + (\mathbf{h} - \mathbf{h}_0)^T \mathbf{g} + 0.5\ (\mathbf{h} - \mathbf{h}_0)^T \mathbf{A}\ (\mathbf{h} - \mathbf{h}_0)$. What is the optimal choice of $\mathbf{h} - \mathbf{h}_0$ to minimize the error?

$\partial e/\partial(\mathbf{h} - \mathbf{h}_0) = \mathbf{0} + \mathbf{g} + \mathbf{A}\ (\mathbf{h} - \mathbf{h}_0)$
$= \mathbf{0}$
$\mathbf{h} - \mathbf{h}_0 = -\ \mathbf{A}^{-1}\ \mathbf{g}$

33. [4 points] Now force $\mathbf{h} - \mathbf{h}_0 = -\ \alpha\ \mathbf{g}$ such that error $e(\mathbf{h}) = e(\mathbf{h}_0) - \alpha\ \mathbf{g}^T \mathbf{g} + 0.5\ \alpha^2\ \mathbf{g}^T \mathbf{A}\ \mathbf{g}$. What is the optimal choice of α to minimize the error?

$\partial e/\partial\alpha = -\ \mathbf{g}^T \mathbf{g} + \alpha\ \mathbf{g}^T \mathbf{A}\ \mathbf{g}$
$= \mathbf{0}$
$\alpha = (\mathbf{g}^T \mathbf{g})\ /\ (\mathbf{g}^T \mathbf{A}\ \mathbf{g})$

34. [4 points] Under what conditions is the gradient descent update (the update in problem 33) equivalent to the Newton's method update (the update in problem 32)?

$A^{-1} = \text{diag}(\alpha, \ldots, \alpha)$

35. [4 points] Assume there's a critical point at $\mathbf{x}_c = [x_c(0), x_c(1), \ldots, x_c(1023)]$ and for each element $x_c(n)$ and small positive perturbation Δ

- It's equally likely that the function at $x_c(n) + \Delta$ is greater than or less than the function at $x_c(n)$

- It's equally likely that the function at $x_c(n) - \Delta$ is greater than or less than the function at $x_c(n)$
- These properties hold independently for each vector element $x_c(n)$

What is the probability that $\mathbf{x}_c$ is a local minima? Saddle point? Local maxima?

P($\mathbf{x}_c$ is a local minima) = $(1/4)^{1024}$
P($\mathbf{x}_c$ is a saddle point) = $1 - 2(1/4)^{1024}$
P($\mathbf{x}_c$ is a local maxima) = $(1/4)^{1024}$