Test 01 – Math

Arthur J. Redfern
arthur.redfern@utdallas.edu
Feb 17, 2020

0 Instructions

- There are 26 numbered questions with indicated point values that sum to 100
- Write all of your answers clearly on this test and turn it in
- No reference materials are allowed
- No help from others is allowed
- Correct answers in red

1 Test

Strategy [5 points]

1. [5 points] Circle true or false for each of the following statements

[✓ True] / False   Many tasks can be cast as a mapping from data to classes or values
[✓ True] / False   Under some mild conditions, neural networks are universal function approximators and can be used to map from data to classes or values
True / [✓ False]   A function always exists that maps arbitrary input to arbitrary output
[✓ True] / False   Inner products are at the heart of the input output mappings for all of the key trainable layers in xNNs
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

Data [10 points]

2. [6 points] Consider an image classification data set **X** with 2^{10} classes and 2^{20} labeled examples per class for a total of 2^{30} labeled examples. What is the total information content of all of the labels?

Information content of all labels = number of labels * information per label
= $2^{30} \log_2(2^{10})$ bits

$= 10*2^{30}$ bits
$= 10737418240$ bits

3. [4 points] Consider a dataset of color images where each image $\mathbf{X}$ 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 $\mathbf{X}$ 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

Weight initialization [4 points]

4. [4 points] Let's say I know the final value of a weight in a xNN is an integer in the range [-127, 127] 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 discrete uniform from [-127, 127]; i.e., each of the values in [-127, 127] has probability 1/255

Feature extraction – CNN layers [13 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:	$\mathbf{f}$ of type ReLU
Output:	$\mathbf{Y}^{3D}$ with dimensions N_o x M_r x M_c

5. [4 points] What are P_r and P_c such that $M_r = L_r$ and $M_c = L_c$?

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

6. [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 $\otimes$ lowered to matrix matrix multiplication $\mathbf{Y}^{2D} = \mathbf{H}^{2D}\mathbf{X}_{filter}^{2D}$ when there are N_o rows in $\mathbf{Y}^{2D}$? Use "x" to separate the dimensions in your answer.

$\mathbf{Y}^{2D}$ 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)

7. [4 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 and not playing games to take advantage of multiplying by known 0s)?

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

8. [2 points] Assume that the layer is part of a network and trained for a 3 x 32 x 64 input $\mathbf{X}$. Is the convolution operation mathematically compatible with a 3 x 512 x 1024 input $\mathbf{X}$? 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:	$\mathbf{y}_t^T$ with dimensions 1 x N_o
Nonlinearity:	$\mathbf{f}$ of type ReLU
Input at time t:	$\mathbf{x}_t^T$ with dimensions 1 x N_i
Input weight matrix:	$\mathbf{H}$
Output at time t–1:	$\mathbf{y}_{t-1}^T$
State weight matrix:	$\mathbf{G}$
Bias:	$\mathbf{v}^T$

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 $\mathbf{y}_{-1}^T = \mathbf{0}^T$.

9. [3 points] What are the dimensions of the input weight matrix $\mathbf{H}$, state weight matrix $\mathbf{G}$ and bias $\mathbf{v}^T$? Use "x" to separate the dimensions in your answer.

$\mathbf{H}$ dimensions are N_i x N_o
$\mathbf{G}$ dimensions are N_o x N_o
$\mathbf{v}^T$ dimensions are 1 x N_o

10. [2 points] Which term, $\mathbf{x}_t^T \mathbf{H}$, $\mathbf{y}_{t-1}^T \mathbf{G}$ or $\mathbf{v}^T$, forces a sequential dependency in the computation of the RNN layer output given the availability of $\{\mathbf{x}_0^T, \mathbf{x}_1^T, \mathbf{x}_2^T, \mathbf{x}_3^T, \mathbf{x}_4^T\}$?

$\mathbf{y}_{t-1}^T \mathbf{G}$

11. [2 points] Assume that there's an error in output $\mathbf{y}_2$. What other output(s) will potentially be in error because of this?

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

Feature extraction – self attention layers [15 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

12. [3 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.

13. [6 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.

14. [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.

15. [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 [6 points]

16. [4 points] Consider an input feature map $\mathbf{X}$ of dimension N_i x L_r x L_c. If a 3x3/2 max pooling layer is applied to X padded by 2 pixels total in the row and 2 pixels total in the column dimensions, what are the dimensions of the output $\mathbf{Y}$?

The dimensions of $\mathbf{Y}$ are N_i x $(L_r/2)$ x $(L_c/2)$

17. [2 points] What value should be assigned to the padding pixels when using max pooling?

Negative infinity or the most negative value for the data type

Feature extraction – nonlinearity choices [6 points]

18. [6 points] Assume that inputs to ReLU are independent random variables with discrete uniform pmfs that can be represented by an integer in [-63, …, 192]. If the output of ReLU is optimally coded (i.e., an entropy code that can optimally assign fractional numbers of bits to different symbols), on average how many bits are needed to represent each output?

64 out of 256 values map to 0

192 out of 256 values map to {1, …, 192} with the probability of each being 1/256

$H = -(64/256)\log_2(64/256) - 192\,(1/256)\log_2(1/256)$

$= -(1/4)(-2) - (3/4)(-8)$

$= 0.5 + 6$

$= 6.5$ bits

Prediction [5 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

19. [1 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

20. [4 points] What is the arithmetic intensity of the dense layer (ignore the bias and there's no nonlinearity in this case) 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 [9 points]

21. [4 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

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

22. [3 points] Circle true or false for each of the following statements

[✓ True] / False KL divergence is a method for comparing 2 pmfs

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

True / [✓ False] The entropy of a 1 hot pmf is 1

23. [2 points] What does it do (in a few words) to the output of softmax if the input is scaled by a constant c that satisfies $0 < c < 1$?

It makes the output a flatter pmf

Back propagation [8 points]

24. [8 points] Consider a scalar modified residual building block with input x, output $y = x + f_1(f_0(x)) + g_0(x)$ and assume that de/dy, the sensitivity of the error e with respect to the output y, is given. Further, define the following terms:

$u_0 = x$
$u_1 = f_0(u_0)$, df_0/du_0 is known
$u_2 = f_1(u_1)$, df_1/du_1 is known
$v_0 = x$
$v_1 = g_0(v_0)$, dg_0/dv_0 is known
$y = x + u_2 + v_1$

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

$$\begin{aligned} de/dx &= de/dy + de/du_0 + de/dv_0 \\ &= de/dy + (du_1/du_0)\,(du_2/du_1)\,(de/du_2) + (dv_1/dv_0)\,(de/dv_1) \\ &= de/dy + (df_0/du_0)\,(df_1/du_1)\,(de/dy) + (dg_0/dv_0)\,(de/dy) \end{aligned}$$

Weight update [12 points]

Given:

$\mathbf{A}$ 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 $0.5\,(\mathbf{h} - \mathbf{h}_0)^T \mathbf{A}\,(\mathbf{h} - \mathbf{h}_0) = \mathbf{A}\,(\mathbf{h} - \mathbf{h}_0)$

25. [6 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? Show your derivation.

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

26. [6 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? Show your derivation.

$$\begin{aligned} \partial e/\partial \alpha &= -\,\mathbf{g}^T\,\mathbf{g} + \alpha\,\mathbf{g}^T\,A\,\mathbf{g} \\ &= 0 \\ \alpha &= (\mathbf{g}^T\,\mathbf{g}) \,/\, (\mathbf{g}^T\,A\,\mathbf{g}) \end{aligned}$$