CS 4476: Mock Exam 2

Name:

GTID:

This exam contains 13 pages (including this cover page) and 5 (multi-part) questions.

There are 100 total points.

Time Limit: 60 Minutes

Important: Make sure to write your name and GTID on this page **and** your GTID in the box at the top right corner of each following page.

For multiple choice questions fill in the square: □ → ■

Grade Table (for staff use only)

Question	Points	Score
Stochastic Gradient Descent	15	
Neural Nets	25	
CNN Fundamentals and Training	25	
Detection & Segmentation	20	
Transformers & Generation	15	
Total:	100	

1: Stochastic Gradient Descent (15 points)

You would like to train a model to $f(x)$ that takes as input an image x and produces as output a label y. You have access to a set of N training data $\{x_i, y_i\}$, for $i \in [1, N]$. You decide that your model should be of the form of a two layer neural network: $f(x) = W_2(\texttt{ReLU}(W_1 x)))$. The full set of model parameters is denoted, $W = [W_1, W_2]$.

(a) (3 points) Select an optimization objective to help you learn your model.

- ☐ $\max_W \sum_{i=1}^{N} ||f(x_i) - y_i||^2$
- ☐ $\max_W \sum_{i=1}^{N} ||f(x_i) + y_i||^2$
- ☐ $\min_W \sum_{i=1}^{N} ||f(x_i) - y_i||^2$
- ☐ $\min_W \sum_{i=1}^{N} ||f(x_i) + y_i||^2$

(b) (3 points) You decide to fit your model using gradient descent. You begin by computing the gradient over all your data using your current set of parameters, W_1, W_2. Let's denote gradients for the loss function with respect to parameter W_1 as $g_1 = \frac{dL}{dW_1}$ and with respect to W_2 as $g_2 = \frac{dL}{dW_2}$. Which of the following would be the correct update rule for W_2 assuming a step size α.

- ☐ $W_2 \leftarrow W_2 + \alpha g_2$
- ☐ $W_2 \leftarrow W_1 - \alpha g_1$
- ☐ $W_2 \leftarrow W_1 + \alpha g_1$
- ☐ $W_2 \leftarrow W_2 - \alpha g_2$
- ☐ $W_2 \leftarrow W_2 - \alpha g_1$
- ☐ $W_2 \leftarrow W_2 - \alpha g_1 \cdot g_2$

(c) (3 points) What algorithm from this course could you use to compute all necessary gradients, g_1, g_2?

(d) (3 points) As the gradient descent algorithm approaches an optima, it is best practice to do which of the following with the learning rate?

- ☐ Increase it
- ☐ Keep it fixed
- ☐ Decrease it

(e) (3 points) During batch gradient descent, gradients at each step are computed using:

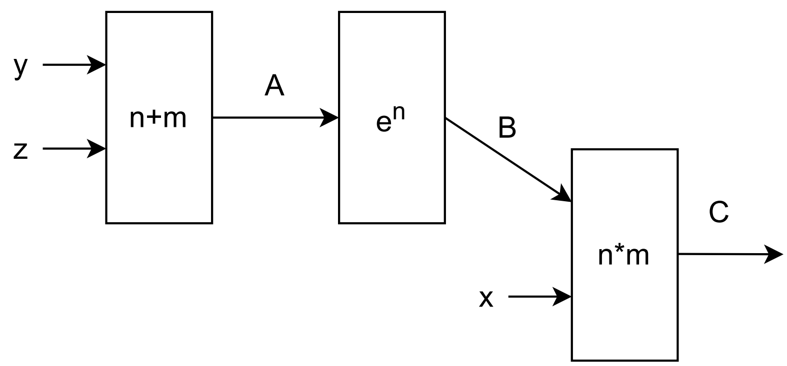
- ☐ Single training example
- ☐ Random subset of the train set
- ☐ Entire training set
- ☐ Fixed subset of the train set

2: **Neural Nets** (25 points)

Below is a computational graph for the function $f(x, y, z) = x * e^{y+z}$. We would like to compute the partial derivative of this function with respect to each input. To get started we will break the function down into three intermediate functions:

$$A = y + z \qquad\qquad B = e^A \qquad\qquad C = x * B$$

From these we can build a feed forward computation graph shown below. For functions with two inputs, n, m, the top input is n and bottom input is m.



Recall the following derivatives, where α is a constant:

$$\frac{\partial}{\partial n} e^n = e^n \qquad\qquad \frac{\partial}{\partial n}(n + \alpha) = 1 \qquad\qquad \frac{\partial}{\partial n}(\alpha * n) = \alpha$$

(a) (2 points) What is the correct expression for $\frac{\partial C}{\partial x}$?

☐ 1 ☐ B ☐ $B * x$ ☐ x

(b) (2 points) What is the correct expression for $\frac{\partial C}{\partial B}$?

☐ 1 ☐ B ☐ C ☐ x

(c) (2 points) What is the correct expression for $\frac{\partial C}{\partial A}$?

☐ $x * e^A$ ☐ A ☐ e^B ☐ e^A

(d) (2 points) What is the correct expression for $\frac{\partial A}{\partial z}$?

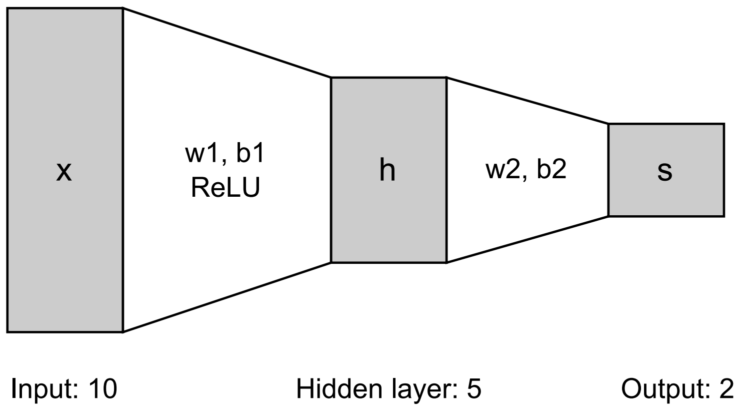
☐ 1 ☐ $y + z$ ☐ z ☐ y

(e) (2 points) Using chain rule, what is the correct expression for $\frac{\partial C}{\partial y}$?

☐ $\frac{\partial C}{\partial x} \cdot \frac{\partial x}{\partial y}$ ☐ $\frac{\partial C}{\partial B} \cdot \frac{\partial B}{\partial A} \cdot \frac{\partial A}{\partial y}$

☐ $\frac{\partial C}{\partial x} \cdot \frac{\partial C}{\partial B} \cdot \frac{\partial B}{\partial A} \cdot \frac{\partial A}{\partial y}$ ☐ $\frac{\partial C}{\partial y} \cdot \frac{\partial B}{\partial y} \cdot \frac{\partial A}{\partial y}$

(f) You are designing a classification network that takes in 10 input features and outputs classification scores for 2 classes. You decide to use 2 fully connected layers. The first layer takes in the 10 input features and has a hidden size of 5 neurons, and has a ReLU activation function. The second layer takes in the 5 hidden neurons and outputs 2 neurons. Both layers have bias.



i. (3 points) How many parameters are there from weights and bias?

☐ 50 ☐ 60 ☐ 67 ☐ 100

ii. (2 points) How many bias parameters are there?

☐ 2 ☐ 7 ☐ 10 ☐ 50

iii. (2 points) What is the role of bias?

iv. (3 points) What would happen if you added another ReLU activation function for the hidden layer?

(g) (2 points) What will the training accuracy and validation accuracy look like when your model is underfitting? What about overfitting?

(h) (3 points) How can overfitting in Neural Networks be prevented or mitigated?

☐ By increasing the number of neurons in each layer-

☐ By reducing the number of training epochs (Early Stopping)

☐ By saving the temporary model with the best training accuracy

☐ By using regularization techniques

☐ By using a larger learning rate

3: CNN Fundamentals and Training (25 points)

(a) (3 points) Which of the following are true when comparing ConvNets and Standard Neural Nets (NN) (i.e. those with only linear layers)? *(Select all that apply.)*

- ☐ ConvNets generally require fewer parameters than an NN.
- ☐ ConvNets are more prone to overfitting than NNs.
- ☐ ConvNets can learn translation invariant features, unlike NNs.

(b) (3 points) Why do we sometimes add padding while applying a convolutional filter? *(Select all that apply.)*

- ☐ To reduce the number of layers in the network.
- ☐ To avoid losing information at edge pixels of the input.
- ☐ To decrease the size of the output features of that layer.
- ☐ To increase the number of model parameters in the layer.
- ☐ To preserve the original size (H,W) of the input after convolution.

(c) Given below is a 4x4 matrix. Answer the following subparts.

M =

3	2	1	1
4	5	0	6
2	1	3	4
3	0	2	0

i. (4 points) The above matrix is passed through a Max-pooling layer with a (2x2) filter, and stride = 2. What is the output?

ii. (2 points) What would the output shape be if the matrix was instead passed through a max-pooling layer with (2x2) filter, and stride = 1?

☐ 2×2 ☐ 3×3 ☐ 4×4

(d) (3 points) Given the following convolution operation described, what is the resulting output shape?

Input: Image with dimensions (100, 100, 3)

Operation: Apply a (5x5x3) filter, where stride = 1, there is no padding, and number of filters = 10.

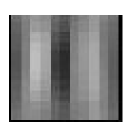
(e) (3 points) How many parameters does this convolution layer have, given the following details?

(Hint: Remember the bias!).

Input: Grayscale image with dimensions (50, 50)

Convolution Layer: Apply a (3x3) filter, where stride = 2, padding = 1, and number of filters = 5.

(f) (2 points) After training your ConvNet you examine the filters learned in the first convolutional layer and realize it looks familiar. Which of the hand-designed filters we have seen before most closely resembles this filter?



☐ Gaussian Filter ☐ Derivative Filter ☐ Constant Filter

(g) (3 points) Suppose you have an input volume of dimension nH x nW x nC. Which of the following statements you agree with? (Assume that "1x1 convolutional layer" below always uses a stride of 1 and no padding.) (Select all that apply)

☐ You can use a 1x1 convolutional layer to reduce nC but not nH, nW.

- ☐ You can use a 1x1 convolutional layer to reduce nH, nW, and nC.
- ☐ You can use a pooling layer to reduce nH, nW, but not nC
- ☐ You can use a pooling layer to reduce nH, nW, and nC.

(h) (2 points) Which of the following do you typically see as you move to deeper layers in a ConvNet? (nH,nW = height, width , nC= number of channels)

- ☐ nH and nW increases, while nC decreases
- ☐ nH and nW decrease, while nC increases
- ☐ nH and nW increases, while nC also increases
- ☐ nH and nW decreases, while nC also decreases

4: Detection & Segmentation (20 points)

(a) (2 points) Which of the following tasks is NOT part of the progression from image classification to object detection?

- ☐ Recognition
- ☐ Localization
- ☐ Segmentation
- ☐ Classification

(b) (2 points) What does the PASCAL criterion for object detection evaluation consider as a true positive?

- ☐ A detected object where the Intersection over Union (IoU) with the ground truth is above 0.5.
- ☐ Any detected object with a confidence score above a threshold.
- ☐ Any detected object that matches the class of the ground truth object.
- ☐ All detected objects, regardless of their overlap with the ground truth.

(c) (2 points) Describe the main advantage of the EdgeBoxes method over Selective Search.

(d) (2 points) What are the four losses used in the Faster R-CNN's single network

---

(e) (2 points) What is the primary advantage of using fully-convolutional networks(FCNs) for semantic segmentation methods over traditional approaches?

- ☐ FCNs provide better interpretability of the segmentation results than traditional methods.
- ☐ FCNs can capture complex spatial relationships and contextual information in the image.
- ☐ FCNs require less memory resources than thresholding-based approaches for real-time segmentation.
- ☐ FCNs require fewer hyperparameters to be tweaked than traditional methods.

(f) (4 points) Below are four True/False question related to image segmentation. Write a brief explanation if the statement is False.

i. The Discriminative Part-based Models (DPM) are based on using a single rigid template for object detection.

ii. The YOLO (You Only Look Once) object detection algorithm divides the image into a grid, and each grid cell predicts bounding boxes and class probabilities.

iii. The IoU-score measures the average performance, while DICE measures the worst-case performance.

iv. Transpose convolution is an efficient way to increase the resolution of feature maps without introducing artifacts.

(g) Answer the following questions assuming the inputs below - a 2D 2x2 image matrix and a single 2x2 convolution filter. We will be performing both a convolution and a **transpose convolution** of the image with the given filter, assuming no padding and a stride of 1.

$$\text{image} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \qquad \text{filter} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

i. (1 point) What is the shape of the output after **convolution** (You do not need to mention the number of channels)?

ii. (1 point) What is the shape of the output after **transpose convolution** (You do not need to mention the number of channels)?

iii. (2 points) What is the value of output[0][0] after **convolution**?

iv. (2 points) What is the value of output[0][0] after **transpose convolution**?

5: Transformers & Generation (15 points)

Transformers

(a) (2 points) Given query (Q), key (K) and value (V) and dimension d, how do we define attention?

☐ $$Attention(Q, K, V) = sigmoid(QK^T)V$$

☐ $$Attention(Q, K, V) = Q \times softmax(\frac{KV^T}{\sqrt{d}})$$

☐ $$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d}})V$$

☐ $$Attention(Q, K, V) = QK^TV$$

(b) (2 points) How does ViT preprocess the images before passing to the transformer layers?

☐ Convert images into small patches, then convert them into single vectors (i.e. tokens) and pass them to transformer layers.

☐ Get image representations in frequency domain and pass them to transformer layers.

☐ Get keypoints in an image and pass the keypoints to transformer layers.

☐ Run a deep CNN over the image and pass the resultant vector to the transformer layers.

(c) (3 points) Which of the following are downsides of vanilla Vision Transformers? (*Select all that apply.*)

☐ They require a lot of data to achieve satisfactory results.

☐ High training time, even for relatively small datasets.

☐ Their individual layers can only see a part of the image.

☐ They cannot be parallelized at all on any type of hardware.

(d) (1 point) Vision Transformers use the same fundamental building blocks as transformers in language (e.g. ChatGPT, GPT-3, etc). True or False?

☐ True

☐ False

Generation

(e) (3 points) Given is the optimization objective of a typical GAN. Select the correct statements from the following. (*Select all that apply.*)

$$\min_G \max_D \left(E_x \sim p_{data}\Big[\log D(x)\Big] + E_{z\sim p(z)}\Big[\log\Big(1 - D\big(G(z)\big)\Big)\Big]\right)$$

- ☐ Discriminator optimizes towards $D(x) = 1$ for fake data.
- ☐ Discriminator optimizes towards $D(x) = 1$ for real data.
- ☐ Generator optimizes towards $D(x) = 1$ for fake data.
- ☐ Generator optimizes towards $D(x) = 0$ for fake data.

(f) (2 points) In the usual unconditional generator-discriminator setup, what is the input to the generator?

- ☐ A random image from training dataset
- ☐ A random class label from the training dataset
- ☐ Gaussian noise
- ☐ None of the above

(g) (2 points) What are the correct statements regarding the Inception Score(IS)?

- ☐ Low IS implies better quality.
- ☐ IS is a combination of two criteria, sharpness and diversity.