EECS 498.007 / 598.005
Deep Learning for Computer Vision
Fall 2019 Sample Midterm Exam

October 21, 2019

Full Name: ____

UM Uniqname (Username): ____________________________

UM-ID (Number): __________________________________

Question		Score
True/False	(8 pts)	
Multiple Choice	(12 pts)	
Short Answer	(12 pts)	
Total	(32 pts)	

- The exam is **1 hour 15 minutes**
- The sample exam should have **5 pages** – make sure you have them all
- The exam is closed-book, closed-internet
- You may use one page (max 8.5" × 11") of handwritten notes
- **This sample exam is significantly shorter than the full exam**

I understand and agree to uphold the Honor Code during this exam.

Signature: __________________________ Date: ________________

Good luck!

1 True / False (8 points)

Fill in the circle next to True or False, or fill in neither. Fill it in completely like this: ●. No explanations are required.

Scoring: Correct answer is worth 2 points. To discourage guessing, incorrect answers are worth -1 points. Leaving a question blank will give 0 points.

1.1 After implementing a neural network, your numeric gradients do not match your analytic gradients. A likely cause of this problem is that you are optimizing the network using Adam instead of SGD.

○ True ○ False

1.2 If the input to a CNN (without batch normalization) is a zero image (all zeros), then the class probabilities will come out uniform.

○ True ○ False

1.3 Turning off L2 weight regularization will likely lead to higher accuracy on the training set.

○ True ○ False

1.4 If a neuron with the ReLU activation function ($y = relu(Wx + b)$) receives input x that is all negative, then the final (not local!) gradient on its weights and biases will be zero (i.e. none of its parameters will update at all).

○ True ○ False

2 Multiple Choice (12 points)

Fill in the circle(s) next to your answer(s). Fill them in completely, like this: ●.
No explanations are required. For each question, mark all answers that apply.

Scoring: Each question is worth 4 points. Each answer within each question is worth one point. Example: If the correct answers are A and B, and you choose A and C, then you receive 2/4 points for the question: one point for correctly choosing A, and one point for correctly not choosing D.

2.1 You start training your Neural Network but the total loss (cross entropy loss + regularization loss) is almost completely flat from the start. What could be the cause?

- ◯ A: The learning rate could be too low
- ◯ B: The regularization strength could be too high
- ◯ C: The class distribution could be very uneven in the dataset
- ◯ D: The weight initialization scale could be incorrectly set

2.2 A VGGNet only uses a sequence of 3x3 CONV with stride 1 pad 1 and 2x2 POOL stride 2 pad 0 layers. It eventually transitions to Fully Connected layers and the classifier. There are 5 POOL layers in total. On ImageNet, the VGGNet takes 224x224 images as input. If we take this VGGNet **trained** on ImageNet, and try to run it at test time with a 32x32 input image (e.g. CIFAR-10 image):

- ◯ A: The code would crash on the very first CONV layer because 3x3 filters with stride 1 pad 1 wouldn't "fit" across 32x32 input
- ◯ B: The amount of memory needed to store the forward activations in the first CONV layer would be reduced by a factor of 7 (since 224/32 = 7)
- ◯ C: The network would run fine until the very first Fully Connected layer, where it would crash
- ◯ D: The network would run forward just fine but its predictions would, of course, be ImageNet class predictions

2.3 A max pooling layer in a ConvNet:

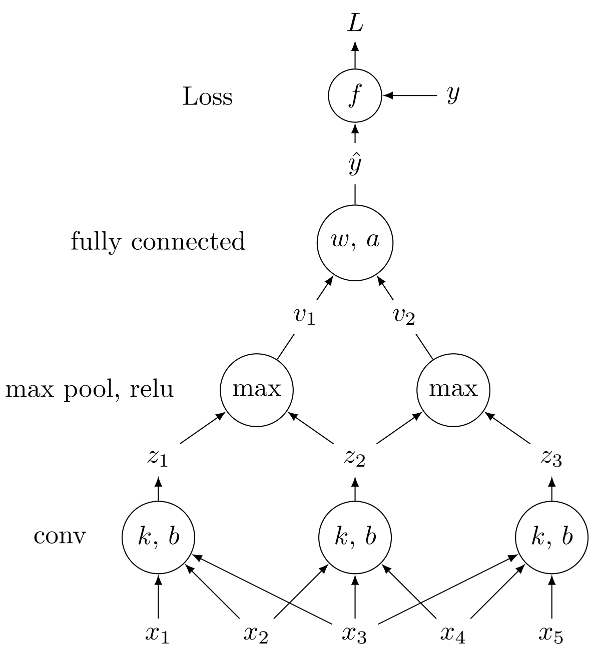
- ◯ A: Is approximately as fast to compute in both forward and backward pass as a CONV layer (with the same filter size and strides).
- ◯ B: Is similar to batch normalization in that it will keep all of your neuron activities in a similar range.
- ◯ C: Could contribute to difficulties when comparing numeric and analytic gradients (higher error than usual, as in the SVM).
- ◯ D: Could contribute to the vanishing gradient problem (recall: this is a problem where by the end of a backward pass the gradients are very small)

3 Short Answer (12 points)

Answer each question in provided space.

3.1 Simple ConvNet (12 points)

Consider the following 1-dimensional ConvNet, where all variables are scalars:



$$L = \frac{1}{2}(y - \hat{y})^2$$

$$\hat{y} = \begin{bmatrix} w_1 & w_2 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} + a$$

$$\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} \max\{z_1, z_2, 0\} \\ \max\{z_2, z_3, 0\} \end{bmatrix}$$

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} k_1 & k_2 & k_3 & 0 & 0 \\ 0 & k_1 & k_2 & k_3 & 0 \\ 0 & 0 & k_1 & k_2 & k_3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} + \begin{bmatrix} b \\ b \\ b \end{bmatrix}$$

(a) **(1 point)** List the parameters in this network.

(b) **(3 points)** Determine the following

$$\frac{\partial L}{\partial w_1} =$$

$$\frac{\partial L}{\partial w_2} =$$

$$\frac{\partial L}{\partial a} =$$

(c) **(3 points)** Given the gradients of the loss L with respect to the second layer activations v, derive the gradient of the loss with respect to the first layer activations z. More precisely, given

$$\frac{\partial L}{\partial v_1} = \delta_1 \qquad \frac{\partial L}{\partial v_2} = \delta_2$$

Determine the following

$$\frac{\partial L}{\partial z_1} =$$

$$\frac{\partial L}{\partial z_2} =$$

$$\frac{\partial L}{\partial z_3} =$$

(d) **(3 points)** Given the gradients of the loss L with respect to the first layer activations z, derive the gradient of the loss with respect to the convolution filter k. More precisely, given

$$\frac{\partial L}{\partial z_1} = \delta_1 \qquad \frac{\partial L}{\partial z_2} = \delta_2 \qquad \frac{\partial L}{\partial z_3} = \delta_3$$

Determine the following

$$\frac{\partial L}{\partial k_1} =$$

$$\frac{\partial L}{\partial k_2} =$$

$$\frac{\partial L}{\partial k_3} =$$

$$\frac{\partial L}{\partial b} =$$

(e) **(2 points)** Suppose we have a general 1D convolution layer

$$\begin{bmatrix} z_1 \\ \vdots \\ z_m \end{bmatrix} = \begin{bmatrix} k_1 & \cdots & k_d & & & \\ & k_1 & \cdots & k_d & & \\ & & & \ddots & & \\ & & & k_1 & \cdots & k_d \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} b \\ \vdots \\ b \end{bmatrix}$$

And we know that

$$\frac{\partial L}{\partial z_i} = \delta_i$$

Determine

$$\frac{\partial L}{\partial k_j} =$$

$$\frac{\partial L}{\partial b} =$$