Introduction to Deep Learning: Midterm Exam
October 24, 2018
Total Points 100

This exam consists of 5 sections, each with multiple questions. You have 75 minutes to complete this exam. You are not allowed to: use a phone, laptop/tablet, calculator or spreadsheet, access the internet, communicate with others, or use other programming capabilities. Disable all networking on your devices ("airplane mode").

Student Name: DAVID CARLSON

Sign below when you are done with the exam.

I adhered to the Duke Community Standard in the completion of this exam. I did not give or receive aid to anyone in the completion of this exam.

Student Signature: _____

Part 1: Short-Answer and Multiple Choice Questions (30 points)

Either provide a short response (i.e. a complete sentence) or circle the correct answers (may be ***one or up to all*** given choices) for each problem. Each question is worth the same number of points, and credit will only be given for completely correct answers.

1. What is necessary for supervised machine learning?
   a. Labeled Training Data
   b. A Model
   c. Learning from Data
   d. Human to teach it

2. What does logistic regression predict (one answer):
   a. Binary (true/false) outcomes
   b. Categorical (multiple categories) outcomes
   c. Real-valued outcomes
   d. Ordinal (1,2,3,4,...) outcomes

3. What does the sigmoid function do?
   a. Converts a real-valued number to a probability of a positive outcome
   b. It's an impenetrable math function that no one understands
   c. Converts a real-valued number to a binary value
   d. Maps a value from negative infinity to infinity

4. When was the multilayer perceptron introduced?
   a. 1940
   b. 1960
   c. 1980
   d. 2000

5. Which one of the following models, when used for image classification, can exceed the performance of humans?
   a. Convolutional neural network
   b. Multilayer perceptron
   c. Logistic regression

6. Which of the following gives the best conceptual meaning of *convolution*?
   a. Shifting a filter to every location in an image
   b. Selecting an atomic element from an image
   c. Stacking a collection of feature maps
   d. Surveying a feature map for a high-level motif

7. What is the "gold standard" validation strategy?
   a. Try on new real-world data
   b. Repeat on training data
   c. Mathematical proof of accuracy
   d. Compare two different models

8. When existing data is used to validate performance, into which of the following groups is data split?
   a. Training set
   b. Validation set
   c. Test set
   d. Optimization set
   e. Learning set
   f. Prediction set

9. In general terms, the basic steps to do learning are:
   i. Obtain a large set of labeled data.
   ii. [What goes here?]
   iii. Determine parameters that minimize the sum over loss.

   Which of the following best describes the missing step 2?
   a. Determine the loss function, which computes loss between true label and model label
   b. Check how well the network predicts the associated labels for new data
   c. Ask an domain expert what they think the parameters should be
   d. Pick the most representative examples of the labeled data, given the context of the data

10. What does transfer learning mean in the context of medical imaging?
   a. Weights of convolutional layers learned from ImageNet transfer to medical images, so we only need learn new parameters at the top of the network.
   b. Just as assigning categories to images in ImageNet required millions of images, so too does analyzing medical images require millions of labeled medical images.
   c. Once the convolutional layers are learned from labeled medical images, the top layers can be inferred from the parameters found with data from ImageNet.
   d. Sufficient labeled radiological images can be used to learn all of the model parameters, so they can be used for ophthalmological or dermatological images.

11. Which of the following are benefits of stochastic gradient descent?
   a. Stochastic gradient descent finds a more exact gradient than gradient descent.
   b. Stochastic gradient descent can update many more times than gradient descent.
   c. Stochastic gradient descent gets near the solution quickly.
   d. With stochastic gradient descent, the update time does not scale with data size.
   e. Stochastic gradient descent finds the solution more accurately.

12. Suppose that a cross-entropy classifier predicts the same probability on each class. If there are 10 class, then what is the loss?
    a. -log(10)
    b. -0.1 log(10)
    c. -log(0.1)
    d. -10 log(0.1)

13. During backpropagation, when a gradient flows through a sigmoid function it cannot change sign.
    a. True
    b. False

14. Which of the following operations are used in pooling? (Choose all that are correct.)
    a. Maximum
    b. Mean
    c. Minimum
    d. Convolution

15. If the training loss is significantly below the validation loss, what issue is happening? (Short Answer)

It is overfitting

16. A collaborator asks you to look at a set of a few hundred scanned images and make predictions on them (i.e. does this patient have lung disease?). Describe an approach you use to effectively apply CNNs to this dataset. (Short answer)

Load pretrained features from Imagenet, then train only the top layers on the new data.

17. Early stopping has often proved a critical technique in deep learning. Briefly describe the motivation for early stopping. (Short answer)

Overfits less and uses less compotational resources.

Problem 2: Convolutional Filtering + Strides/Pools (20 points)

This problem is designed to evaluate whether you understand the basic mechanics of convolutional neural networks.

Consider the convolutional operation below:

0	1	2	3	4	5	6
1	2	3	4	5	6	7
1	2	3	4	5	6	7
0	1	2	3	4	5	6
0	1	2	3	4	5	6
1	2	3	4	5	6	7
1	2	3	4	5	6	7

Image

*

2	-1
-1	2

Filter

=

(a) What is the result of this convolution operation run with the "valid" setting? Make sure you get the resultant size correct. Note that the image matrix is heavily patterned to reduce the amount of math that's needed to be done here. (7 pts)
(b) What is the result if the convolution has stride 2x2? (2 pts)
(c) What is the result if the output is put through a 2x2 max pool (window size 2x2)? (2 pts)
(d) What is the result if the convolution has stride 3x3? (2 pts)
(e) What is the result if the output is done through a 3x3 max pool (window size 3x3)? (2 pts)
(f) Qualitatively, what is the difference between strides and max pools? (i.e. a succinct, written description) (5 pts)

Note that if you mis-interpreted the question you still got credit & benefit of the doubt.

a)

2	4	6	8	10	12
3	5	7	9	11	13
2	4	6	8	10	12
1	3	5	7	9	11
2	4	6	8	10	12
3	5	7	9	11	13

Extra page for problem 2: Convolutional Filtering + Strides/Pools

b)

2	6	10
2	6	10
2	6	10

c)

5	9	13
4	8	12
5	9	13

d)

2	8
1	7

e)

7	13
7	13

(f) Stricles simply skip pixels to downsample, whereas max pooling can capture the max feature in an area. Max pooling can lead to improved spatial invariance. Which approach is better depends on the application.

Problem 3: Stochastic Optimization (20 points)

Write down pseudo-code for:

a) Gradient Descent (4pts)
b) Stochastic Gradient Descent (5pts)
c) Stochastic Gradient Descent with momentum (5pts)

For this problem, assume that our objective is:

$F(\theta) = \frac{1}{N}\sum_{n=1}^{N} f_n(\theta)$, where each n is a separate data point.

The gradient is given as $\nabla F(\theta) = \frac{1}{N}\sum_{n=1}^{N} \nabla f_n(\theta)$.

For each algorithm, define what parameters have to be set, i.e.:

1. Initialize θ_0
2. Choose step-size α
3. For $k = 1, \ldots$

...

With short answers (i.e. 1-2 sentences), please write about:

d) Rationale for using *stochastic* methods rather than full batch methods (2pts)
e) Why and when momentum can be helpful (2pts)
f) What to do when the loss stops improving (2pts)

a) Initialize θ_0
Choose a step-size sequence $\alpha_0, \ldots$
for $k = 0, \ldots$
Calculate $\nabla F(\theta_k)$
Update $\theta_{k+1} = \theta_k - \alpha_k \nabla F(\theta_k)$

b) Initialize θ_0
Choose a step-size sequence $\alpha_0, \ldots$
for $k = 0, \ldots$
Choose random data example i_k
Calculate $\nabla f_{i_k}(\theta_k)$
Update $\theta_{k+1} = \theta_k - \alpha_k \nabla f_{i_k}(\theta_k)$

Extra page for problem 3: Stochastic Optimization

c) Initialize θ_0, and $m_0 = 0$
Choose $\alpha_0, \ldots$
Set momentum param. β
for $k = 0, \ldots$
choose i_k randomly
calculate $\nabla f_{i_k}(\theta_k)$
Update $m_{k+1} = \beta m_k + \nabla f_{i_k}(\theta_k)$
Update $\theta_{k+1} = \theta_k - \alpha_k m_{k+1}$

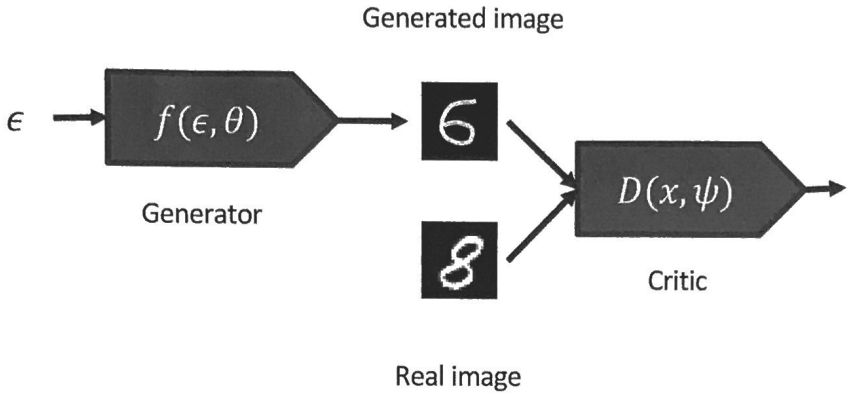
d) Stochastic methods have the same iteration time regardless of the data size, and can get near the solution much faster that traditional methods.

e) Momentum can help in two ways:
i) Dampen highly oscillatory directions
ii) Decrease variance on our gradient estimates which improves convergence.
(i) helps regardless, but (ii) only helps in the stochastic setting.

f) If the loss stops improving, consider two strategies:
(i) Stop. Check the validation loss curve and evaluate if you are finished.
(ii) Reduce step size to get closer to the optimal solution.

Problem 4: Generative Modeling (20 points)

A typical Generative Adversarial Modeling setup is shown below.



In short responses, please describe:

a) What the critic/discriminator is doing qualitatively (2pts)

The critic determines whether an input is a real or generated image.

b) What the generator is doing qualitatively (2pts)

The generator takes a random input vector and uses a deep network to convert it to a synthetic data example.

c) Qualitatively, how does the objective function for GAN work? (2pts)

The critic and generator are playing a game with each other. The generator tries to trick the critic into thinking that it is a real image.

d) What is the objective function for GAN? (2pts)

$$\min_{\theta} \max_{\phi} \; E_{x \sim p_{data}(x)} [\log D_{\phi}(x)] + E_{z \sim p_z(z)} [\log (1 - D_{\phi}(G_{\theta}(z)))]$$

e) When is the generator "optimal?" What is the mathematical definition of the optimal generator? (3pts)

The generator is optimal when the discriminator cannot tell between real and synthetic samples. Mathematically, this implies $G_{\theta}(z) \sim p_{data}$, or that the distributions are identical.

f) Mark below which approach best matches the following statement (1 point each):

(a) GAN (b) VAE (c) Both

i. b involve two networks, one that takes an image x as input, one that produces an image $\hat{x}$ as output

ii. a has its networks (typically) trained in an alternating fashion, as opposed to jointly

iii. b involves a variational approximation

iv. b capable of inference in the original formulation

v. a has famously been used to generate realistic looking images

vi. c can be trained completely unsupervised

g) Why is sampling from a GAN more likely to create images more realistic to the human eye than a VAE? (3 points)

VAEs are designed to minimize mean squared error (at least typically), which does not maintain sharp edges. In contrast, GANs produce sharp edges because blurred edges are a giveaway to the discriminator that its a synthetic image.

Problem 5: Code Interpretation. (10 points)

Suppose we define a network in code:

```
# Model Inputs
x = tf.placeholder(tf.float32, [None, 784])
y_ = tf.placeholder(tf.float32, [None, 10])

# Define the graph
x_image = tf.reshape(x, [-1, 28, 28, 1])

# First convolutional layer.
W_conv1 = tf.Variable(tf.truncated_normal([5, 5, 1, 32], stddev=0.1))
b_conv1 = tf.Variable(tf.zeros([32]))
h_conv1 = tf.nn.relu(tf.nn.conv2d(x_image, W_conv1, strides=[1, 1, 1, 1], padding='SAME') + b_conv1)

# Pooling layer.
h_pool1 = tf.nn.max_pool(h_conv1, ksize=[1, 2, 2, 1], strides=[1, 2, 2, 1], padding='SAME')

# Second convolutional layer.
W_conv2 = tf.Variable(tf.truncated_normal([5, 5, 32, 64], stddev=0.1))
b_conv2 = tf.Variable(tf.zeros([64]))
h_conv2 = tf.nn.relu(tf.nn.conv2d(h_pool1, W_conv2, strides=[1, 1, 1, 1], padding='SAME') + b_conv2)

# Second pooling layer.
h_pool2 = tf.nn.max_pool(h_conv2, ksize=[1, 2, 2, 1], strides=[1, 2, 2, 1], padding='SAME')

# Fully connected layer 1
W_fc1 = tf.Variable(tf.truncated_normal([7 * 7 * 64, 256], stddev=0.1)) # note that 7 * 7 * 64 = 3136
b_fc1 = tf.Variable(tf.zeros([256]))

h_pool2_flat = tf.reshape(h_pool2, [-1, 7*7*64])
h_fc1 = tf.nn.relu(tf.matmul(h_pool2_flat, W_fc1) + b_fc1)

# Map the 256 features to 10 classes, one for each digit
W_fc2 = tf.Variable(tf.truncated_normal([256, 10], stddev=0.1))
b_fc2 = tf.Variable(tf.zeros([10]))

y_conv = tf.matmul(h_fc1, W_fc2) + b_fc2

# Loss
cross_entropy = tf.reduce_mean(tf.nn.softmax_cross_entropy_with_logits_v2(labels=y_, logits=y_conv))
```

[28×28]

For this code, sketch a graph to describe what network is being defined (i.e. a flow chart with a block for each major operation), and give the dimensionality of the image at each layer.

Extra page for problem 5: Code Interpretation.