CS583A: Final Exam (Sample Questions)

Name:

Policy: Books and printed materials are allowed. Do not use electronic divice, including phone, laptop, and tablet.

Hint: (i) $\frac{\partial e^a}{\partial a} = e^a$, (ii) $\frac{\partial \log_e(a)}{\partial a} = \frac{1}{a}$, (iii) $\frac{\partial \frac{1}{a}}{\partial a} = -\frac{1}{a^2}$, and (iv) $\frac{\partial \cos(a)}{\partial a} = -\sin(a)$.

Q1 (3%). Let $\mathbf{A}$ be a real matrix and $\mathbf{b}$ be a real vector. To compute the multiplication $\mathbf{Ab}$ efficiently, which of the following libraries is the the best choice?

A. Level 1 BLAS.

B. Level 2 BLAS.

C. Level 3 BLAS.

D. LAPACK.

Q2 (3%). The input shape is 18×18, the pool size is 3×3, and the pooling has no overlap (equivalently, the stride is 3×3). Then what is the output shape of the pooling?

A. 3×3.

B. 6×6.

C. 15×15.

D. 16×16.

E. 18×18.

Q3 (3%). You want to train a deep convolutional neural network using the CIFAR10 dataset. Which of the following tricks cannot be applied to improve the test error?

A. Using dropout regularization.

B. Using data augmentation.

C. Using ensemble method.

D. Using multi-task learning.

Q4 (3%). We use a deep learning model to predict whether a customer's review is positive or negative. We tokenize the reviews in the word-level and thus need an embedding layer to convert words to vectors. Upon the embedding layer, we build RNN layers and other types of layers. We find serious overfitting because the dataset has merely 10,000 samples. Which of the followings is the best improvement?

A. Pretraining the embedding layer using a big dataset.

B. Using Bi-LSTM to replace SimpleRNN or LSTM.

C. Combing self-attention with RNN.

D. Using stacked-LSTM to increase the model capacity.

Q5 (3%). The following implements a convolutional neural network. The code has a major problem. How can we make correction?

A. Remove the second convolutional layer (Lines 10 to 12), because it is not useful.

B. Put the batch normalization layers (Lines 8 and 11) after the activations (Lines 9 and 12).

C. Put the batch normalization layers (Lines 8 and 11) before the convolutions (Lines 7 and 10).

D. Put the dropout layer (Line 18) before the first dense layer (Line 16).

E. Put the dropout layer (Line 18) before the second dense layer (Line 17).

```
1  from keras import models
2  from keras.layers import Input, Conv2D, MaxPooling2D, Activation
3  from keras.layers import BatchNormalization, Flatten, Dense, Dropout
4
5  input_img = Input((128, 128, 3))
6
7  x = Conv2D(32, (3, 3))(input_img)
8  x = BatchNormalization()(x)
9  x = Activation('relu')(x)
10 x = Conv2D(32, (3, 3))(x)
11 x = BatchNormalization()(x)
12 x = Activation('relu')(x)
13 x = MaxPooling2D((2, 2))(x)
14
15 x = Flatten()(x)
16 x = Dense(128, activation='relu')(x)
17 x = Dense(256, activation='relu')(x)
18 x = Dropout(0.5)(x)
19 x = Dense(10, activation='softmax')(x)
20
21 model = models.Model(input_img, x)
```

Q6 (3%). (Fill the blank.) The training set contains $1,000$ samples. We train a neural network using mini-batch SGD. One epoch amounts to 50 iterations. What is the batch size?
The batch size is _____.

Q7 (3%). (Fill the blank.) What is the output of the following Python program?
Answer: _____

```
1 import numpy
2 a = numpy.random.rand(3, 5) # generate a random matrix
3 b = numpy.random.rand(5, 10) # generate a random matrix
4 c = numpy.dot(a, b)
5 print(c.shape[0])
```

Q8 (12%). (Fill the blanks.) The following code builds a neural network for sentiment analysis.

- Line 10 is an embedding layer. What is the output shape of this layer?
  Answer: _____
- What is (roughly) the number of parameters in the embedding layer?
  Answer: _____
- Line 11 is a SimpleRNN layer. What is the output shape of this layer?
  Answer: _____
- What is (roughly) the number of parameters in the SimpleRNN layer?
  Answer: _____
- Line 12 is a dense layer. What is the output shape of this layer?
  Answer: _____
- What is (roughly) the number of parameters in the dense layer?
  Answer: _____

```
1 from keras.models import Sequential
2 from keras.layers import SimpleRNN, Embedding, Dense
3
4 voc_size = 10000
5 shape_x = 20
6 seq_length = 100
7 shape_h = 50
8
9 model = Sequential()
10 model.add(Embedding(voc_size, shape_x, input_length=seq_length))
11 model.add(SimpleRNN(shape_h, return_sequences=False))
12 model.add(Dense(1, activation='sigmoid'))
```

Q9 (12%). Read a piece of code regarding convolutional neural network and infer the output shapes and number of parameters.

Q10 (10%). A matrix calculus question analogous to those in the quiz.