UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN
Department of Electrical and Computer Engineering

ECE 417 Multimedia Signal Processing
Fall 2018

PRACTICE EXAM 2

Tuesday, December 11, 2018

- This is a PRACTICE exam. In the real exam, you will be permitted to use one sheet (front and back) of handwritten notes.
- In the real exam, no calculators will be permitted. You need not simplify explicit numerical expressions.
- There will be a total of 100 points in the exam. Each problem specifies its point total. Plan your work accordingly.
- You must SHOW YOUR WORK to get full credit.

Problem	Score
1	
2	
3	
4	
5	
6	
Total	

Name: ____

Possibly Useful Formulas

Neural Nets

$$\begin{aligned} a_k &= u_{k0} + \sum_j w_{kj} x_j \\ y_k &= g(a_k) \\ \frac{\partial E}{\partial x_j} &= \sum_k w_{kj} g'(a_k) \frac{\partial E}{\partial y_k} \end{aligned}$$

Logistic Function

$$\sigma(x) = \frac{1}{1+e^{-x}}, \quad \sigma'(x) = \sigma(x)\left(1-\sigma(x)\right)$$

Loss Functions

$$\begin{aligned} E_{MSE} &= \frac{1}{n}\sum_{i=1}^{n} \|\vec{z}_i - \vec{\zeta}_i\|^2 \\ E_{CE} &= -\frac{1}{n}\sum_{i=1}^{n}\sum_{\ell=1}^{r} \zeta_{i\ell} \ln z_{i\ell} \\ E_{CE} &= -\frac{1}{n}\sum_{i=1}^{n} \left(\zeta_i \ln z_i + (1-\zeta_i)\ln(1-z_i)\right) \end{aligned}$$

Affine Transform

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

Barycentric Coordinates

$$\begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix} = \begin{bmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix}$$

LSTM

$$\begin{aligned} &\vec{i}[n] = \text{input gate} = \sigma(B_i \vec{x}[n] + A_i \vec{c}[n-1]) \\ &\vec{o}[n] = \text{output gate} = \sigma(B_o \vec{x}[n] + A_o \vec{c}[n-1]) \\ &\vec{f}[n] = \text{forget gate} = \sigma(B_f \vec{x}[n] + A_f \vec{c}[n-1]) \\ &\vec{c}[n] = \vec{f}[n] \odot c[n-1] + \vec{i}[n] \odot g\left(B_c \vec{x}[n] + A_c \vec{c}[n-1]\right) \\ &\vec{y}[n] = \vec{o}[n] \odot \vec{c}[n] \end{aligned}$$

Topics Covered in This Exam

(a) Neural Nets: logistic, softmax. MSE, Cross-entropy. Back-prop

(b) ConvNets. Back-prop through convolution, max pooling, and ReLU

(c) (Not Covered: SGD, Batch, Mini-Batch, Data Augmentation)

(d) Affine Transforms

(e) (Not Covered: Image interpolation, PWC, PWL, and sinc)

(f) Adversarial examples, adversarial training. (Not covered: autoencoder, VAE, GAN)

(g) Barycentric coordinates

(h) RNN, GRU, LSTM

Problem 1 (16 points)

In class, we have been working with nodes in layers, but a neural net can also be defined as a fully-connected graph, with every node connected to every other node. For example, suppose there is a scalar input x, and

$$\begin{aligned} y_0 &= x \\ a_\ell &= \sum_{k=0}^{\ell-1} w_{\ell k} y_k, \quad 1 \le \ell \le L \\ y_\ell &= \sigma(a_\ell), \quad 1 \le \ell \le L \\ E &= \frac{1}{2}\sum_{\ell=1}^{L}(y_\ell - y_\ell^*)^2 \end{aligned}$$

Define the back-propagation error to be $\delta_\ell = \frac{dE}{da_\ell}$. Find an algorithm that computes δ_ℓ for all $1 \le \ell \le L$.

Problem 2 (17 points)

A convolutional layer leads to convolutional back-propagation. In the neural net literature, however, convolution is sometimes replaced (without comment!) by correlation, resulting in something like the following, where $x[m_1, m_2]$ is the input and $u[m_1, m_2]$ are the network weights:

$$a[n_1, n_2] = \sum_{m_1}\sum_{m_2} u[m_1 - n_1, m_2 - n_2]x[m_1, m_2]$$

Suppose the error, E, is some function whose partial derivatives $\epsilon[n_1, n_2] = \frac{\partial E}{\partial a[n_1,n_2]}$ are known. Define $\delta[m_1, m_2] = \frac{\partial E}{\partial x[m_1,m_2]}$. Find $\delta[m_1, m_2]$ in terms of $\epsilon[n_1, n_2]$.

Problem 3 (16 points)

Consider four points, $\vec{u}_1 = [u_1, v_1, 1]^T$, $\vec{u}_2 = [u_2, v_2, 1]^T$, $\vec{u}_3 = [u_1 + \alpha\cos\theta, v_1 + \alpha\sin\theta, 1]^T$, and $\vec{u}_4 = [u_2 + \beta\cos\theta, v_2 + \beta\sin\theta, 1]^T$. Notice that the slope of the line segment connecting $\vec{u_1}$ to $\vec{u}_3$ is $\frac{a\sin\theta}{a\cos\theta} = \tan\theta$, while the slope of the line segment connecting $\vec{u}_2$ to $\vec{u}_4$ is also $\frac{b\sin\theta}{b\cos\theta} = \tan\theta$. Suppose that there is an affine transform A such that $\vec{x}_1 = A\vec{u}_1$, $\vec{x}_2 = A\vec{u}_2$, $\vec{x}_3 = A\vec{u}_3$, and $\vec{x}_4 = A\vec{u}_4$. Prove that, for any affine transform matrix A, the line segment connecting $\vec{x}_1$ to $\vec{x}_3$ is parallel to (has the same slope as) the line segment that connects $\vec{x}_2$ to $\vec{x}_4$.

Problem 4 (17 points)

Suppose you have a dataset containing audio waveforms, $\vec{x}_i$, each matched with two different one-hot label vectors. The label vector $\vec{y}_i^* = [y_{i1}^*, \ldots, y_{iq}^*]^T$, where $y_{ij}^* \in \{0, 1\}$, is approximated by the network output $\vec{y}_i = [y_{i1}, \ldots, y_{iq}]^T$, where $y_{ij} \in (0, 1)$. The label vector $\vec{z}_i^* = [z_{i1}^*, \ldots, z_{ir}^*]^T$, where $z_{ij}^* \in \{0, 1\}$, is approximated by the network output $\vec{z}_i = [z_{i1}, \ldots, z_{ir}]^T$, where $z_{ij} \in (0,1)$. Both $\vec{y}_i$ and $\vec{z}_i$ are functions of a hidden nodes vector $\vec{h}_i$ as

$$\begin{aligned}
\vec{h}_i &= g\left(W\vec{x}_i\right) \\
\vec{y}_i &= \text{softmax}\left(U\vec{h}_i\right) \\
\vec{z}_i &= \text{softmax}\left(V\vec{h}_i\right)
\end{aligned}$$

where U, V and W are trainable weight matrices, and $g(\cdot)$ is some scalar nonlinearity. Find an error metric E such that, by minimizing E, you can:

- **maximize** the accuracy of $\vec{y}_i$ as an estimate of $\vec{y}_i^*$
- **minimize** the accuracy of $\vec{z}_i$ as an estimate of $\vec{z}_i^*$

Problem 5 (17 points)

The Barycentric coordinates of point $\vec{x}_0 = [x_0, y_0, 1]^T$, as defined by the triangle $\vec{x}_1 = [x_1, y_1, 1]^T$,$\vec{x}_2 = [x_2, y_2, 1]^T$,$\vec{x}_3 = [x_3, y_3, 1]^T$, are the coordinates λ_1, λ_2, λ_3 such that

$$\begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix} = \begin{bmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix}$$

Provide an equation in terms of the six scalars $x_1, x_2, x_3, y_1, y_2, y_3$ specifying the conditions under which the matrix $\begin{bmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ 1 & 1 & 1 \end{bmatrix}$ is singular.

Problem 6 (17 points)

Consider an LSTM defined by

$$\begin{aligned}
\vec{i}[n] &= \text{input gate} = \sigma(B_i\vec{x}[n] + A_i\vec{c}[n-1]) \\
\vec{o}[n] &= \text{output gate} = \sigma(B_o\vec{x}[n] + A_o\vec{c}[n-1]) \\
\vec{f}[n] &= \text{forget gate} = \sigma(B_f\vec{x}[n] + A_f\vec{c}[n-1]) \\
\vec{c}[n] &= \vec{f}[n] \odot c[n-1] + \vec{i}[n] \odot g\left(B_c\vec{x}[n] + A_c\vec{c}[n-1]\right) \\
\vec{y}[n] &= \vec{o}[n] \odot \vec{c}[n]
\end{aligned}$$

where the vector cell is $\vec{c}[n] = [c_1[n], \ldots, c_p[n]]^T$, and where $\odot$ denotes the Hadamard (array) product, e.g., $\vec{o}[n] \odot \vec{c}[n] = [o_1[n]c_1[n], \ldots, o_p[n]c_p[n]]^T$. Find the derivative $\frac{\partial c_j[n]}{\partial c_k[n-1]}$.