College of Staten Island/CUNY

DEPARTMENT OF COMPUTER SCIENCE

CSC 767 Neural Networks and Deep Learning

FINAL EXAM

NOTE: Final Exam is INDIVIDUAL!

Name, Surname: ..... December 14, 2020

Question	Max Points	Your Points
1	40	
2	20	
3	20	
4	20	
Total	100	

Regular Submission on the BB on December 14, 2020, 11:30 pm.

Late Submission on the BB with 20% penalty on December 15, 11:30 pm.

For the first problem you can use a code in Python or solve it by hand. If Python, submit the three files (.pdf, .py and .ipynb) and all outputs. Submit your Problem 1 solution (the manual calculations/code, outputs, analysis etc.). Some more details of hand/Python solution of Problem 1 are specified after the problem description.

Submit the rest of the problems (2, 3 and 4) in .doc or .pdf format.

Do not copy from Internet and paste directly into your exam answers. Do not borrow your colleague responses to the exam paper questions.

1. Execute forward and backward modes of neural network **by hand or by using Python (your choice) for 3 iterations (epochs).** The model is shown in Fig. 1. This network comprises of a hidden layer with 3 ReLU units and a squared-error loss. (Note: Use *tanh* as activation function in the output unit and ReLU for hidden layer)

(40 points)

$$ReLU(x) = \begin{cases} x, & \text{if } x \geq 0 \\ 0, & \text{otherwise} \end{cases}$$

(Hint: Notice that this function is *non-differentiable* around 0. Therefore, when computing your gradients, you will have to compute one gradient if the input is larger than 0, another if it is smaller than 0, and ignore when the input is 0 and thus non-differentiable.)

Step 1: Initialize the weights to these random values shown below

$$\begin{bmatrix} W_{11} & W_{21} & W_{31} \\ W_{12} & W_{22} & W_{32} \end{bmatrix} = \begin{bmatrix} 0.60 & 0.70 & 0.00 \\ 0.01 & 0.43 & 0.88 \end{bmatrix}$$

$$\begin{bmatrix} w_{11} \\ w_{12} \\ w_{13} \end{bmatrix} = \begin{bmatrix} 0.02 \\ 0.03 \\ 0.09 \end{bmatrix}$$

The data is provided in the following matrix where samples are stored in rows.

$$\begin{bmatrix} x_1^T \\ x_2^T \\ x_3^T \\ x_4^T \end{bmatrix} = \begin{bmatrix} 0.75 & 0.80 \\ 0.20 & 0.05 \\ -0.75 & 0.80 \\ 0.20 & -0.05 \end{bmatrix}$$

Their corresponding labels are $y = [1, 1, -1, -1]$.

Step 2: You may use the recursive patterns to update parameters of this network.

HINT:

Pseudo-code for manual calculations:

1. Forward the input and record (i) input to the unit, s_j, and (ii) output of the unit, z_j, for each unit j in all layers.
2. Compute the loss and record it.

   Use $\mathcal{L} = 0.5 \cdot (y_{out} - y)^2$
3. Compute the error signal, $\delta_{out} = \frac{\partial \mathcal{L}}{\partial s_{out}}$ at the output.
4. Propagate δ_{out} backwards to compute $\delta_j = \frac{\partial \mathcal{L}}{\partial s_j}$ at hidden units.

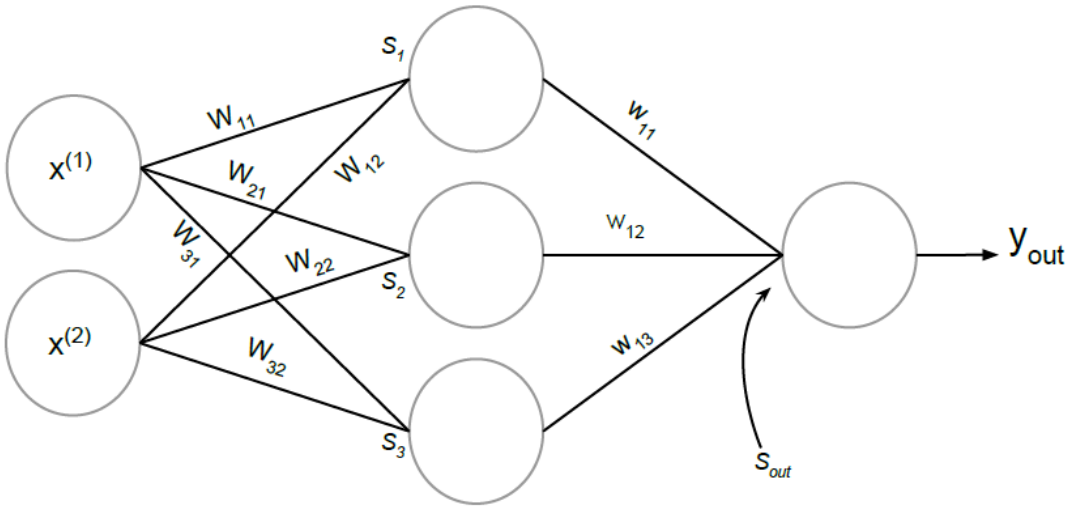

Fig. 1. NN topology

5. Compute gradients (i.e. $\Delta W, \Delta w$) using these and update weights using gradient descent.

   Hint: $W^{(t+1)} = W^{(t)} - \alpha \cdot \Delta W$
6. Repeat 3 times.

Problem 1 Instructions:

All hand solutions MUST contain the details including formulas of pseudocode (Steps 1-6, for 3 iterations) you are using and not only the final values.

All hand solutions (without a code) which contain ONLY final values will be evaluated to ZERO.

All solutions using Python must display the results of all formulated in pseudocode Steps 1-6 for 3 iterations. Submitting only final values will result in ZERO points.

Clearly mark when the problem and the respective steps start. Please, don't be messy.

2. Describe the key advantages of
- Batch normalization
- Self-Normalizing activations and
- Max Norm weight constraints **(required 1 to 2 pages text, excluding graphics, figures, formulas, References, Times New Roman, size 11, single space)**

(max **20 points)**

3. Describe the importance of **Autoencoders** as well as similarities and differences between autoencoders and PCA **(required 1 to 2 pages text, excluding graphics, figures, formulas, References, Times New Roman, size 11, single space)**

(max 20 points)

4. Why regularization is important. What is explicit and implicit regularization (**required 1 to 2 pages text, excluding graphics, figures, formulas, References, Times New Roman, size 11, single space)**

(max 20 points)