Midterm for CSC321, Intro to Neural Networks
Winter 2018, night section
Tuesday, March 6, 6:10-7pm

Name: ______________________________

Student number: ____________________

This is a closed-book test. It is marked out of 15 marks. Please answer ALL of the questions. Here is some advice:

- The questions are NOT arranged in order of difficulty, so you should attempt every question.
- Questions that ask you to “briefly explain” something only require short (1-3 sentence) explanations. Don’t write a full page of text. We’re just looking for the main idea.
- None of the questions require long derivations. If you find yourself plugging through lots of equations, consider giving less detail or moving on to the next question.
- Many questions have more than one right answer.

Final mark: ______ / 15

1. **[2pts]** When tuning hyperparameters, it's important to use a validation set.

   - **[1pt]** Briefly explain what can go wrong if you tune hyperparameters on the training set.

   - **[1pt]** Briefly explain what can go wrong if you tune them on the test set.

2. **[1pt]** Give the definition of a convex function. I.e., let f be a scalar-valued function that takes a vector $\mathbf{x}$ as input. The definition is that f is convex if and only if...

3. **[3pts]** Consider a layer of a multilayer perceptron with logistic activation function σ:

$$z_i = \sum_j w_{ij} x_j + b_i \qquad\qquad h_i = \sigma(z_i)$$

The two layers have dimensions D_1 and D_2, respectively. Let M denote the mini-batch size. Recall the backprop equations for a multilayer perceptron:

$$\overline{z_i} = \sigma'(z_i) \circ \overline{h_i} \qquad\qquad \overline{x_j} = \sum_i \overline{z_i} w_{ij} \qquad\qquad \overline{w_{ij}} = \overline{z_i} x_j$$

Your job is to write NumPy code to implement the backward pass for this layer. Assume the following NumPy arrays have already been computed:

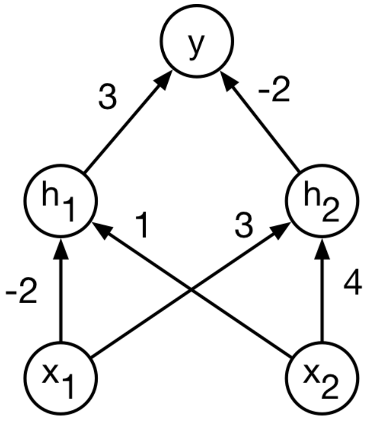
- `X`, an $M \times D_1$ matrix representing the x_j values for a whole mini-batch. (I.e., each row contains the activations for one training example.)
- The matrices `Z` and `H` are defined analogously.
- `W`, a $D_2 \times D_1$ matrix representing the weights. I.e., the (i, j) entry of `W` is w_{ij}.
- `H_bar`, the matrix of error signals $\overline{h_i}$, the same size as `H`.

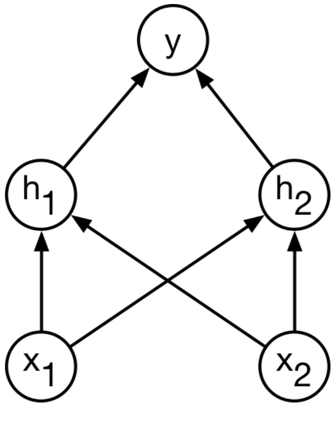
Also, assume you are given a function `sigma_prime` which computes the derivatives of the logistic function, elementwise.

Write NumPy code that computes arrays `Z_bar`, `X_bar`, and `W_bar`, representing error signals for the corresponding variables. `W_bar` should be the average of the derivatives over the mini-batch. Your code should be vectorized, i.e. do not use `for` loops. This can be done in three lines, one for each backprop equation.

4. **[1pt]** TRUE or FALSE: in logistic regression, if every training example is classified correctly, then the cost is zero. Briefly explain your answer.

5. **[1pt]** Consider the following multilayer perceptron shown on the left. All units use the logistic activation function, and there are no biases. On the right-hand diagram, give the weights for an equivalent network (i.e. one which computes the same function). You don't need to explain your answer.





6. **[2pts]** Briefly explain two reasons to use automatic differentiation rather than finite differences to compute the gradient for a neural network during training.

7. **[1pt]** Alice and Barbara are trying to redesign the LeNet conv net architecture to reduce the number of weights. Alice wants to reduce the number of feature maps in the first convolution layer. Barbara wants to reduce the number of hidden units in the last layer before the output. Whose approach is better? Why?

8. **[2pts]** Recall the neural language model architecture from Assignment 1. Suppose there are 100 words in the dictionary, the embedding dimension is 30, the context length is 3 words, and the hidden layer has 60 units. You don't need to explain your answer, but doing so may help you get partial credit.

   (a) **[1pt]** How many parameters are needed for the embedding layer? (You may assume this layer has no biases.)

   (b) **[1pt]** How many weights and biases are needed for the hidden layer?

9. **[2pts]** Recall that a plateau is a flat region in the cost function. Give two examples of plateaux (plual of plateau) that can occur in neural net training, and briefly explain why they are plateaux.