Midterm for CSC321, Intro to Neural Networks
Winter 2018, afternoon section
Tuesday, March 6, 1:10-2pm

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. **[1pt]** We saw that we can perform polynomial regression (i.e. fit the coefficients of a degree-d polynomial of a scalar variable x) using linear regression with a feature map $\boldsymbol{\phi}(x)$. Assume there is no explicit bias parameter, so the model is $y = \mathbf{w}^\top \boldsymbol{\phi}(x)$. What is $\boldsymbol{\phi}(x)$ if we are fitting a degree-4 polynomial? You don't need to explain your answer.

2. **[1pt]** Suppose we have implemented a function that computes the derivative $\mathrm{d}f/\mathrm{d}x$ for a univariate function f. In order to test the correctness of our implementation using finite differences, we compare the function's output to another value. What is that value, and how is it used?

3. **[2pts]** Suppose we have two multilayer perceptrons, A and B. TRUE or FALSE: if A has more units than B, then A must also have more connections than B. Explain why it is true or provide a counterexample.

4. **[3pts]** In this question, we'll design a binary linear classifier to compute the NAND (not-AND) function. This function receives two binary-valued inputs x_1 and x_2, and returns 0 if both inputs are 1, and returns 1 otherwise.

   (a) **[1pt]** Give four constraints on the weights w_1 and w_2 and the bias b, i.e. one constraint for each of the 4 possible input configurations.

   (b) **[2pts]** Consider a slice of weight space corresponding to $b = 1$. Sketch the constraints in weight space corresponding to the two input configurations ($x_1 = 1, x_2 = 0$) and ($x_1 = 1, x_2 = 1$). (Make sure to indicate the half-spaces with arrows.)

5. **[3pts]** Consider a layer of a multilayer perceptron which has ReLU activations:

$$z_i = \sum_j w_{ij} x_j + b_i$$
$$h_i = \mathrm{ReLU}(z_i)$$

(a) **[2pts]** Give the backprop rules for computing the error signals $\overline{z_i}$, $\overline{x_j}$ and $\overline{w_{ij}}$ in terms of the error signals $\overline{h_i}$.

$$\overline{z_i} =$$

$$\overline{x_j} =$$

$$\overline{w_{ij}} =$$

(b) **[1pt]** Consider a pair of units (x_j, h_i). Based on your answer to part (a), for what values of x_j and z_i are we guaranteed that $\overline{w_{ij}} = 0$?

6. **[1pt]** Recall that Autograd includes a module, `autograd.numpy`, which provides similar functionality to `numpy`, except that each of the functions does some additional bookkeeping needed for autodiff. Briefly explain one thing that `autograd.numpy.sum` does which `numpy.sum` does not.

7. **[1pt]** Compute the convolution of the following two arrays:

$$\begin{pmatrix} 4 & 1 & -1 & 3 \end{pmatrix} * \begin{pmatrix} -2 & 1 \end{pmatrix}$$

Your answer should be an array of length 5. You do not need to show your work, but it may help you get partial credit.

8. **[1pt]** Both the neural probabilistic language model and n-gram language models make a Markov assumption. What is this Markov assumption? (One sentence should suffice.)

9. **[2pts]** When we discussed resource constraints for neural nets, we noted that the activations need to be stored in memory at training time, but not at test time. For simplicity, focus on the case of a multilayer perceptron.

   (a) Why do the activations need to be stored at training time (in order to do backprop)?

   (b) Why don't they need to be stored at test time?