Deep Learning CS583 Fall 2020
Final Exam

December 10, 2020

Instructor: Jia Xu

Student name: _____

Student ID: _____

Student email address: _____

- **Read these instructions carefully**
- Fill-in your personal info, as indicated above.
- You have 3 hours.
- There are ten questions. Each question worths the same (1 points).
- Both computer-typed and hand-writing in the very clear form are accepted.
- Submit ONE single pdf containing your solutions to all questions.
- This is an open-book test.
- You should work on the exam only by yourself.
- Submit your PDF/Doc/Pages **by 9:30 PM Dec 10th** on Canvas under Final exam.

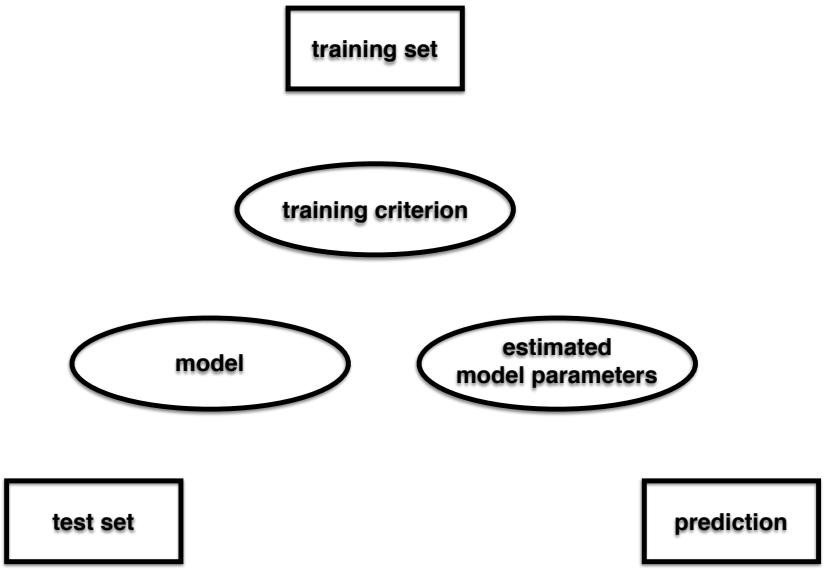
good luck!

1 Question

A classification or a regression system such as those in machine translation, or in other tasks such as question answering, image recognition, speech recognition, follows a workflow. This workflow is composed of several components, as depicted in the graph below.

Your tasks in this question:

1. Connect the boxes with directed edges and indicate the relationship between the components connected with your edges.
2. Write the following terms as instances of their belonging component box, considering no noise in the training set but errors in the prediction output (this can be many-to-many alignment if you think necessary):
   (a) neural networks
   (b) smoothing using linear interpolation
   (c) $P(e) = 0.5$
   (d) maximum likelihood
   (e) mean squared error
   (f) "the cat sat on the matt"
   (g) "matt the on"



2 Question

- Applying back-propagation to train a neural network is guaranteed to find the global optimal.

  A. TRUE. B. FALSE.

- Regardless of the choice of the activation function, it makes the network function as a linear mapping from inputs to outputs to set all weights close to zero in a neural network.

  A. TRUE. B. FALSE.

- A multi-layer feedforward network with linear activation functions is more expressive than a single-layer feedforward network with linear activation functions.

  A. TRUE. B. FALSE.

- Suppose that you are training a neural network for classification, but you notice that the training loss is much lower than the validation (test set) loss. Which of the following can be used to address the issue (select all that apply)?

  A. Use a network with fewer layers

  B. Decrease dropout probability (Dropout turns off neurons with a probability)

  C. Increase the regularization weight

  D. Increase the size of each hidden layer

3 Question

- A training pattern, consisting of an input vector $x = [x1, x2, x3]^T$ and desired outputs $t = [t1, t2, t3]^T$, is presented to the following neural network. What is the usual sequence of events for training the network using the back-propagation algorithm?

  A. (1) calculate $z_k = f(I_k)$, (2) update W_{kj}, (3) calculate $y_j = f(H_j)$, (4) update v_{jv}.

  B. (1) calculate $y_j = f(H_j)$, (2) update v_{ji}, (3) calculate $z_k = f(I_k)$, (4) update w_{kj}.

  C. (1) calculate $y_j = f(H_j)$, (2) calculate $z_k = f(I_k)$, (3) update v_{ji}, (4) update w_{kj}.

  D. (1) calcualte $y_j = f(H_j)$, (2) calculate $z_k = f(I_k)$, (3) update w_{kj}, (4) update v_{ji}.

- The learning rate is a function of the number of training steps t. Why is this function important?

  A. It is not important the actual value of the learning rate will not affect the performance of the system.

  B. The learning rate is decreased by 1% every learning step so that the learning will stabilize and the weights will eventually reach a steady state.

  C. The learning rate depends on the neighborhood of the winning neuron the neighbors of the winning unit are adapted by a smaller amount so that the network learns a topological mapping.

  D. The learning rate is increased by 1% every learning step so that the robot can improve its performance over time.

4 Question

- Is an autoencoder for supervised learning or for unsupervised learning? Explain briefly.

- Is a mixture of Gaussians for supervised learning or for unsupervised learning? Explain briefly.

5 Question

- We have seen that averaging the outputs from multiple models typically gives better results than using just one model. Why is it helpful? Briefly explain.

- Let's say that we're going to average the outputs from 10 models. Of course, we want 10 good models, i.e. models that also perform well individually. What additional property of a collection of 10 models makes that collection a good candidate for output averaging?

6 Question

In Bayesian learning, we consider not just one, but many different weight vectors. Each of those is assigned a probability by which it is weighted in producing the final output.

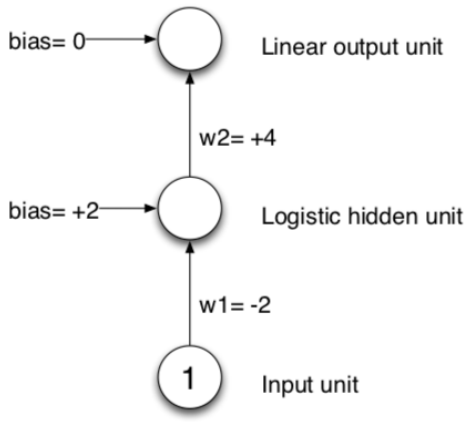
- Write down Bayes' rule as it applies to supervised neural network learning. Clearly define the symbols that you are using.

- Clearly indicate which part of the formula is the "prior distribution", which is the "likelihood term", and which is the "posterior distribution".

- In this context, how is Maximum A Posteriori (MAP) learning different from Maximum Likelihood (ML) learning?

7 Question

Here you see a very small neural network: it has one input unit, one hidden unit (logistic), and one output unit (linear). Let's consider one training case. For that training case, the input value is 1 (as shown in the diagram), and the target output value is 1. We're using the standard squared error loss function: $E = (t - y)^2/2$. The numbers in this question have been constructed in such a way that you don't need a calculator.



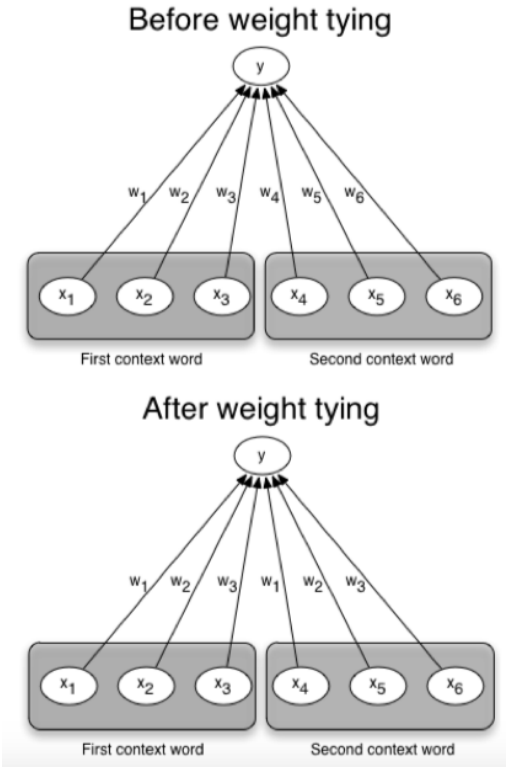
- What is the output of the hidden unit and the output unit, for this training case?

- What is the loss, for this training case?

- What is the derivative of the loss w.r.t. w2, for this training case?

- What is the derivative of the loss w.r.t. w1, for this training case?

8 Question



Suppose that we have a vocabulary of 3 words, "a", "b", and "c", and we want to predict the next word in a sentence given the previous two words. For this network, we don't want to use feature vectors for words: we simply use the local encoding, i.e. a 3-component vector with one entry being 1 and the other two entries being 0.
In the language models that we have seen so far, each of the context words has its own dedicated section of the network, so we would encode this problem with two 3- dimensional inputs. That makes for a total of 6 dimensions. For example, if the two preceding words (the "context" words) are "c" and "b", then the input would be (0, 0, 1, 0, 1, 0). Clearly, the more context words we want to include, the more input units our network must have. More inputs means more parameters, and thus increases the risk of overfitting. Here is a proposal to reduce the number of parameters in the model:

Consider a single neuron that is connected to this input, and call the weights that connect the input to this neuron w1, w2, w3, w4, w5, and w6. w1 connects the neuron to the first input unit, w2 connects it to the second input unit, etc. Notice how for every neuron, we need as many weights as there are input dimensions (6 in our case), which will be the number of words times the length of the context. A way to reduce the number of parameters is to tie certain weights together, so that they share a parameter. One possibility is to tie the weights coming from input units that correspond to the same word but at different context positions. In our example that would mean that w1=w4, w2=w5, and w3=w6 (see the "after" diagram). **Explain the main weakness that that change creates.**

9 Question

- For a fully-connected deep network with one hidden layer, increasing the number of hidden units should have what effect on bias and variance? Explain briefly.

- What is the risk with tuning hyperparameters using a test dataset?

10 Question

Consider the following linear auto-encoder with 1 input and 1 output: $\tilde{x} = w_2 w_1 x$, trained with the squared reconstruction error:

$L(W) = \frac{1}{P} \sum_{i=1}^{P} \frac{1}{2}(x^i - w_2 w_1 x^i)^2$

The scalar training samples have variance 1.

(a) What is the set of solutions (with 0 loss)?

(b) Does the loss have a saddle point? Where?