

Name: _____

USC e-mail: _____@usc.edu

Answer the questions in the spaces provided. **If you run out of space, continue your work on the last two pages, and indicate that your answer is there.** You may use the backs of pages for scratch work only. **Please use pen for ease of grading.** This exam has 7 questions, for a total of 150 points.

Question 1: K-Means and Global Optima (26 points)

Recall the k -means objective function is

$$L(z_{1:n}, \mu_{1:k}) = \sum_{i=1}^n \left\| x^{(i)} - \mu_{z_i} \right\|^2,$$

where $z_{1:n}$ are the cluster assignments, $\mu_{1:k}$ are the corresponding cluster centroids, and $x^{(i)}$ is the i -th input datapoint. Now consider the following dataset consisting of three datapoints in a 2-D plane: $\{(0, 3), (0, 0), (4, 0)\}$.

- (a) (6 points) Suppose you run k -means with $k = 2$ and have the initial cluster centers be $\mu_1 = (4, 0)$ and $\mu_2 = (0, 3)$. What will be the final cluster centers when k -means converges? Show your work and explain how you know that k -means has converged.

- (b) (6 points) Prove that the above initialization converges to the global minimum for $k = 2$. (Hint: Enumerate over all possible cluster assignments and compute the objective for each assignment.)

- (c) (6 points) Suppose we pick two distinct datapoints (out of the three datapoints) uniformly at random as the cluster centers at initialization. Will k -means always converge to the global minimum on this dataset? Justify your answer.

- (d) For this question, you may **not** use k -means as an answer for either part.
- i. (4 points) Name one machine learning method from class (not including k -means) that **is** guaranteed to converge to the global optimum of its objective function. Explain why it is guaranteed.

- ii. (4 points) Name one machine learning method from class (not including k -means) that **is not** guaranteed to converge to the global optimum of its objective function. Explain why it is not guaranteed.

Question 2: Transformers and Multi-Headed Attention (24 points)

Consider a Transformer model with a single 1-headed self-attention layer. As input, we pass in four words, each represented with an embedding of dimension 2. That is, the inputs are vectors $w_1, w_2, w_3, w_4 \in \mathbb{R}^2$.

The single headed self-attention layer is parameterized by matrices $K, Q, V \in \mathbb{R}^{2 \times 2}$. For each w_i , the key, query, and value vectors are defined as $k_i = Kw_i$, $q_i = Qw_i$, and $v_i = Vw_i$, respectively. For simplicity, we provide the following values:

$$\begin{array}{cccc} q_1 = \begin{bmatrix} 3 \\ 7 \end{bmatrix} & q_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} & q_3 = \begin{bmatrix} 0 \\ 2 \end{bmatrix} & q_4 = \begin{bmatrix} 2 \\ 5 \end{bmatrix} \\ k_1 = \begin{bmatrix} 1 \\ 12 \end{bmatrix} & k_2 = \begin{bmatrix} 1 \\ 12 \end{bmatrix} & k_3 = \begin{bmatrix} 1 \\ 12 \end{bmatrix} & k_4 = \begin{bmatrix} 1 \\ 26 \end{bmatrix} \\ v_1 = \begin{bmatrix} 12 \\ 6 \end{bmatrix} & v_2 = \begin{bmatrix} 9 \\ 3 \end{bmatrix} & v_3 = \begin{bmatrix} 3 \\ 6 \end{bmatrix} & v_4 = \begin{bmatrix} 4 \\ 1 \end{bmatrix} \end{array}$$

(Note: The values here are created for ease of calculation. Technically, these values are not possible, since k_1, k_2, k_3 are identical but q_1, q_2, q_3 are not.)

(a) (6 points) Assume this transformer is an encoder. Calculate the output of self-attention for w_2 . Note that your answer should be a vector $\in \mathbb{R}^2$.

(b) (6 points) Now assume this transformer is a decoder. Calculate the output of self-attention for w_3 . Again, note that your answer should be a vector $\in \mathbb{R}^2$.

(c) (4 points) During lecture, we saw that the entire self attention mechanism can be represented in matrix form. Why is it important that we are able to do this?

(d) (4 points) Ryan is training a Transformer language model, but it is taking a long time. He notices that the multi-headed attention layers are slow, so he decides to remove all the multi-headed attention layers from the model. Will the model still perform similarly as before, or will it perform much worse? Explain why.

- (e) (4 points) Transformer models commonly have multiple layers of multi-headed attention, such as having 12 layers with 12 heads each. An alternative would be to have a single layer with 144 heads, thus having the same total number of attention heads. Give one reason why using 12 layers and 12 heads is generally preferable.

(d) (4 points) If Vishesh were to use PCA to address this issue, this would introduce an additional hyperparameter for his classification problem. What is this hyperparameter and how should Vishesh choose a good value for it?

(e) (6 points) An alternative strategy would be to use L_1 regularization instead of PCA. L_1 regularization will cause many of the classifier's weights to be 0, which essentially means those features get ignored. Give one reason for why L_1 regularization might perform better than using PCA.

(Hint: Imagine that Vishesh is performing multiple tasks with the same dataset, e.g., he could either try to predict whether the patient will require surgery in five years, or whether they will have another doctor visit within the next year. Think about what L_1 regularization would do in these two scenarios compared with what PCA would do.)

- (f) (4 points) A local hospital has heard about Vishesh's model and wants to use it to decide which patients should get priority when scheduling doctor's visits. Vishesh is concerned that the model could be unfair to members of different racial groups. Describe one evaluation metric that Vishesh could use to measure the fairness of the model. (You don't need to remember the exact name of the metric. If you do give the name, you should still explain what it does.)

Question 4: Explaining Q-Learning (18 points)

In class, we saw the following update rule for Tabular Q-Learning:

$$\hat{Q}(s, a) \leftarrow (1 - \eta)\hat{Q}(s, a) + \eta(r + \gamma\hat{V}(s')).$$

- (a) (4 points) What is the difference between choosing a large value of η and a small value of η ?

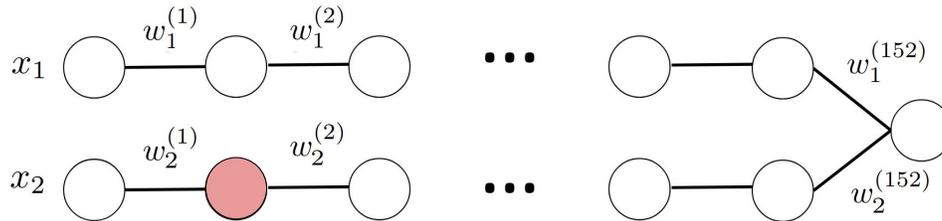
- (b) (6 points) In class, we discussed two quantities, $\hat{V}(s')$ and $V_{\text{OPT}}(s')$. Explain (1) the difference between these two quantities **and** (2) why the Q-learning update rule uses $\hat{V}(s')$ instead of $V_{\text{OPT}}(s')$.

(c) (4 points) In class, we learned about something called ϵ -Greedy. What is the purpose of ϵ -Greedy? What would go wrong if we did not use ϵ -Greedy when running Q-learning?

(d) (4 points) Your friend knows that the choice of ϵ affects the final result of Q-learning, but is confused about how this is possible because ϵ does not show up in the update rule for Tabular Q-learning. Explain which parts of the Q-learning update rule would be directly affected when ϵ is changed, and why.

Question 5: Backpropagation (25 points)

Consider the 152-layer neural network in the figure below. Throughout the question, we assume that the training example is (x, y) where $x = (x_1, x_2) \in \mathbb{R}^2$ and $y \in \mathbb{R}$ (all activations and gradients are with respect to this training example). The network has no bias terms, and it is fully-connected only for the final output node (i.e. the input x_1 is not connected to the red node in the next layer and so on, except for the last output node).



The network has a total of $152 \times 2 = 304$ parameters, namely $w_1^{(1)}, \dots, w_1^{(152)}$ and $w_2^{(1)}, \dots, w_2^{(152)}$. It has two neurons in each layer, except for the final output layer. The network computes a prediction as follows:

- In the first layer, each neuron $o_i^{(1)}$ is computed by first computing $a_i^{(1)} = w_i^{(1)} x_i$, and then applying the activation function h : $o_i^{(1)} = h(a_i^{(1)})$. For example, the red node takes in x_2 , computes $a_2^{(1)} = w_2^{(1)} x_2$, and outputs $o_2^{(1)} = h(a_2^{(1)})$.
- Each subsequent layer before the last layer works the same way, except it takes the previous layer's output as input. So, $o_i^{(j)}$ is computed by first computing $a_i^{(j)} = w_i^{(j)} o_i^{(j-1)}$, then outputting $o_i^{(j)} = h(a_i^{(j)})$.
- The final layer takes in the outputs of the penultimate layer, $o_1^{(151)}$ and $o_2^{(151)}$, and outputs $o^{(152)} = w_1^{(152)} o_1^{(151)} + w_2^{(152)} o_2^{(151)}$.

The activation function $h(a)$ is the same for every neuron.

The network is trained on the standard squared loss, so that $L(w) = \frac{1}{2}(o^{(152)} - y)^2$ for the example (x, y) , where w is the vector of all the weights in the network.

Note: For parts (a), (b), and (c), you can use the notation $h'(a)$ to represent the derivative of the activation function $h(\cdot)$ with respect to some input a in your answer.

- (a) (3 points) Derive the expression for $\frac{\partial L(w)}{\partial o^{(152)}}$.

(b) (4 points) Derive the expression for $\frac{\partial L(w)}{\partial w_1^{(152)}}$.

(c) (10 points) Derive the expression for $\frac{\partial L(w)}{\partial w_1^{(1)}}$.

(Hint: Use the chain rule, for example, $\frac{\partial o_1^{(j)}}{\partial o_1^{(j-1)}} = \frac{\partial o_1^{(j)}}{\partial a_1^{(j)}} \frac{\partial a_1^{(j)}}{\partial o_1^{(j-1)}}$.)

(d) (4 points) Bill decides to use the sigmoid activation function $h(a) = \frac{1}{1+e^{-a}}$ for all nodes in this neural network. He notices that when he is trying to train this model, $\frac{\partial L(w)}{\partial w_1^{(1)}}$ is very close to 0. Using your expression from the previous part, explain why this is the case.

(Hint: The gradient of $h(a)$ is $h'(a) = h(a)(1 - h(a))$. You may use the fact that for any real number a , the quantity $h(a)(1 - h(a))$ is always between 0 and 0.25.)

(e) In class, we learned about a phenomenon where the gradient can often become zero when the computation graph is very long.

i. (2 points) Give the name for this phenomenon.

ii. (2 points) Name one strategy that can be used to combat this phenomenon. (Your answer does not need to apply to the neural network in this question.)

Question 6: Short Response (14 points)

Answer the following questions and **explain your reasoning fully**. You may also draw explanatory diagrams when appropriate.

- (a) (4 points) Wenyang has a CNN model for image classification that uses multiple conv-relu-pool blocks, followed a single feedforward layer. She decides to change all the max pooling layers in her network from 2×2 to 4×4 , while the number of layers, type of layers, kernel sizes, and number of input/output channels will stay the same. The size of the input images also does not change. Will the number of parameters in her model have to increase, decrease, or stay the same?

- (b) (4 points) Suppose we want to run linear regression on the following data points. Is there a unique solution for w that minimizes the linear regression loss function?

x_1	x_2	y
1	2	5
3	6	1
-2	-4	0

(c) (6 points) Lorena is training a supervised learning model and keeps collecting more and more training data. As the amount of training data increases,

- The variance of the model (**gets larger/gets smaller/stays the same**).
- The bias of the model (**gets larger/gets smaller/stays the same**).

Circle one choice for each bullet point and explain your answers.

Question 7: Multiple Choice (20 points)

In the following questions, circle the correct answer(s). There is no need to explain your answer.

- (a) (2 points) **True** or **False**: The sigmoid function $\sigma(z) = \frac{1}{1+e^{-z}}$ from $\mathbb{R} \rightarrow \mathbb{R}$ is a convex function.
- (b) (2 points) **True** or **False**: Decision trees do not suffer from overfitting problems because all x 's assigned to the same leaf node receive the exact same prediction.
- (c) (2 points) **True** or **False**: One difference between k -means and GMM is that k -means uses hard assignments whereas GMM uses soft assignments.
- (d) (2 points) **True** or **False**: The SVM algorithm minimizes L_2 -regularized hinge loss.
- (e) (2 points) **True** or **False**: Adding Dropout is a viable way to regularize a Transformer model.
- (f) (2 points) **True** or **False**: In a Markov Decision Process, if an agent visits the same state twice and takes the same action each time, the probability of transitioning to an end state is guaranteed to be the same for both times.
- (g) (2 points) Which of the following statements is true about the total number of parameters learned by word2vec?
 - A. It is equal to the dimension of one word vector.
 - B. It is proportional to the size of the training dataset.
 - C. It is proportional to the size of the vocabulary.
 - D. word2vec does not learn any parameters.
- (h) (3 points) Which of the following statements about PCA are true? Choose all that apply.
 - A. PCA involves randomly initializing a guess for the best dimension.
 - B. PCA computes a non-linear transformation of the original dataset.
 - C. PCA requires computing eigenvalues and eigenvectors of the covariance matrix.
 - D. PCA can be used to visualize data.
- (i) (3 points) Which of the following are best practices for responsible machine learning? Choose all that apply.
 - A. It is important to identify the right metrics to quantify the performance of the ML model (such as accuracy, precision, recall, performance on sub-groups, etc.)
 - B. The model may latch onto spurious correlations to make their predictions, so should be tested extensively in use cases that directly impact many people.
 - C. The model should be tested on diverse data that reflects the use cases in deployment.
 - D. The model should not be monitored or updated after deployment.

[This page provides extra space for answers]

[This page provides extra space for answers]