Applied Machine Learning Final Exam (CIS 419/519)

Instructions

Write your answers on paper with a pen. **Write your name, and your section number (CIS 419 or CIS 519)** prominently on the first page of your answers at the top left.

No devices except for a calculator. You may use one handwritten sheet of paper as a cheatsheet, with both sides.'"

The exam contains 3 sections, roughly arranged by question type:

- Section A has 1 or 2-point questions that are each intended to take 1-2 mins to complete (total: 18 pts, 11 questions, ~20 minutes).
- Section B contains 3,4-point questions that are each intended to take 2-4 mins to complete. (total: 29 pts, 9 questions, ~35 minutes)
- Section C contains 5, 6, and 7-point questions that are intended to take between 4 and 8 mins to complete. As a thumb rule, an x-pt question should take not much longer than x mins. (total: 33 pts, 6 questions, ~45 minutes)

The exam is out of 80 points, and you have 120 minutes. Manage your time well! These expected time suggestions are intended to help: if you find yourself sinking a lot more time into a problem, it may be a better strategy to stop and return later to take a fresh look.

At the end of 2 hours, you will put down your pens, and proceed to submission.

Submission Process:

Step 1: Scan & Upload Within 15 mins: After the exam, you will have 15 minutes of access to your smartphone for scanning and uploading your solutions to gradescope. Make sure to use a **pdf scanning app** (for example, the Google Drive app has a feature) to scan your answers and upload to Gradescope during these 15 minutes. Make sure your smartphone is charged and ready!

Step 2: Select Pages On Your Own Time: Then, *after the submission window closes,* you will be able to annotate pages for each question on your own time. Please do use a pdf scanner app, and do not upload image files directly: we have noticed that for some reason, Gradescope doesn't allow you to annotate your responses after the submission window if you upload image files. Complete this before 9 p.m. today, May 9.

For in-person exam takers, we will collect your answer sheets as a backup in case scanning went wrong.

All the best!

Section A

1 or 2-point questions, that are each intended to take 1-2 mins to complete (total: 18 pts, 11 questions, ~20 mins)
No need to show your work.

1. [1 pt] Suppose we have a regularized linear regression model: $argmin_w \lambda||Y - Xw||_2^2 + ||w||_1$. What is the effect of increasing λ on bias and variance? Select the best option.
   A. Increases bias, increases variance
   B. Increases bias, decreases variance
   C. Decreases bias, increases variance
   D. Decreases bias, decreases variance

2. [1 pt] The TAs of the course want to develop a neural network which takes as input student information, such as autograder scores for each homework, and frequency of class attendance and predict student performance on this final. Which of the following functions is an appropriate loss function for this problem? (Select all that are appropriate)
   - Binary Cross Entropy
   - Negative Log Likelihood → could also award points for saying this.
   - Mean Squared Error
   - KL divergence.

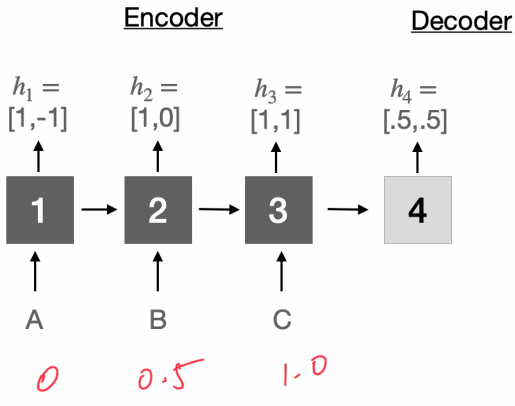
3. [1 pt] For an MDP (S, A, T, γ, R) if we only change the discount factor γ, the optimal policy is guaranteed to remain the same. True/ False?

4. [1 pt] Suppose you are designing a recommendation system for the Penn library book collection. The system will be used to recommend similar books when people checkout book rentals. The library has more than 1 million books, but has only collected <10,000 user reviews + checkout history in total so far. Which one of the following methods would best suit the use case?
   A. Nearest Neighbor Collaborative Filtering
   B. Content-based Recommendation
   C. Matrix Factorization
   D. Popularity-based Recommendation

5. [2 pt] When training RNNs, making sure gradients do not get too small (vanishing gradients) or get too big (exploding gradients) is crucial to successful learning. Select all true statements about controlling gradient magnitudes:

   (1) L2 Regularization helps reduce the vanishing gradient problem.
   (2) LSTM cells can reduce vanishing gradients.
   (3) Adding more hidden layers helps solve the vanishing gradient problem.
   (4) Clipping the gradients helps solve the exploding gradients problem.

6. [2 pts] Consider the following encoder-decoder style model over characters. The goal of the model is read in a list of characters (i.e. A,B,C), and then output the same sequence in exactly that order (i.e. Copy the input sequence as output). In the examples on the right, the input sequence is consumed by an RNN, and the hidden states of the RNN are shown above the states. Assume every decoder hidden state, in light gray, is used to compute dot-product attention with the encoder hidden states, in dark gray.In the first example on the right, at the first hidden state of the decoder (state 4), order the encoder states, 1,2,3 by which receives the most attention from state 4, largest first.



C > B > A

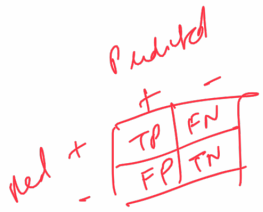
0 0.5 1.0

7. [2 pts] Susan is not satisfied with the performance of her deep neural network model. She plans to build an ensemble with K members and average their predictions to improve performance. Which of these approaches could she use to construct this ensemble? Select all that apply.
   1. Training K deep neural networks with the same architecture and the same initialization.
   2. Training K deep neural networks with different architectures.
   3. Training K deep neural networks on different subsets of the data.
   4. Training K deep neural networks with the same architecture and different random initializations.
   5. None of the above

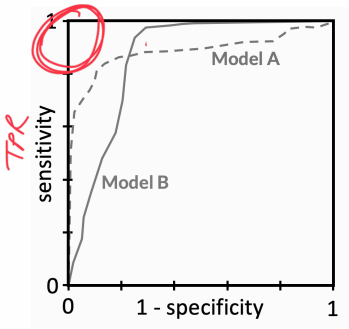
8. [2 pt] Consider the following two PCA-based dimensionality reduction methods.
   Method 1: PCA is applied to project points from *d*-dimensional space to *i* principle coordinates. PCA is applied once again to these coordinates from *i*-dimensional space to *j* principle coordinates. ($d > i > j$).
   Method 2: PCA is applied to project points from *d*-dimensional space directly to *j* principle coordinates.

   Do method 1 and method 2 produce the same results? State Yes / No.

9. [2 pt] You are training an ML model. In which of the following scenarios would increasing the bias (as in "bias-variance tradeoff") of the ML algorithm help? Select all that apply.
   A. The training dataset is small.
   B. You observe high error on both the training set and the validation set.
   C. Your data cannot be linearly separated.
   D. You observe low error on the training set and high error on the validation set.



10. [2 pt] Which of the classifiers A and B shown in the figure (on the right) would you use in an automatic braking system of a car to detect collisions (assume that collisions are the positive class)? Note: sensitivity is also called the "true positive rate", and specificity is also called the "true negative rate". Braking => need low FNR = high TPR, so pick model B.



11. [2 pt] You are given that cloud cover is more predictive of the chances of rain than the time of day. Consider the following two hypothesis classes for predicting rain:
   A. Decision trees with cloud cover attribute at the root
   B. Decision trees with time of day attribute at the root

   Class A has greater expressive power than class B because it is using a more predictive attribute at the root node. True or False?

Section B

3,4 -point questions, that are each intended to take 3-5 mins to complete (total: 29 pts, 9 questions, ~35 mins)

12. [3 pt] Suppose we are trying to build a linear regression classifier to solve the XOR problem, whose inputs x_1 and x_2, and corresponding outputs y are shown in the figure. You are aware that you cannot solve the problem when only using x_1 and x_2 as features, so you decide to use the following three features instead.
   - $f_1 = x_1^2 - x_2^2$
   - $f_2 = x_1 \times x_2$
   - $f_3 = x_1^2 + x_2^2$

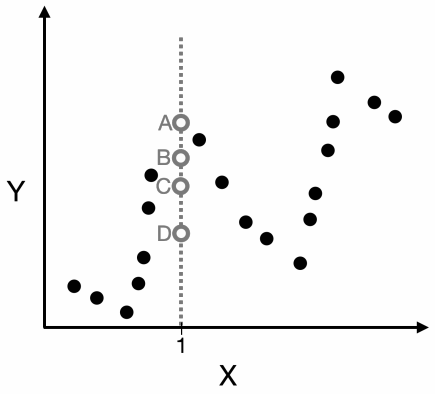
   Would a linear regression classifier using only the above three features solve the problem? Justify your answer.

x1	x2	y
1	1	-1
1	-1	1
-1	1	1
-1	-1	-1

Yes. Because $y = -x_1 x_2$

13. [3 pt] Jeff is training a deep neural network for classification using a three way split of data that he calls, D1, D2, and D3. He is considering using the datasets to train the weights of his network, tune the architecture, and then finally report performance as measured by accuracy on the D3 data split. He trains the network parameters on dataset D1. He is unsure which dataset to use to tune hyperparameters, so he tries tuning them on D1 (scenario A) , D2 (scenario B), and D3 (scenario C). Can you predict the ascending order of reported accuracies for scenarios A, B, and C?

14. [3 pt] Consider one possible extension of k-nearest neighbors adapted to regression. In this setting, when given a test point, we compute its neighborhood using euclidean distance, and then assign it the mean value in the neighborhood.



Consider the 1-dimensional example shown in the figure, where the x axis is the input, and y axis is the output. The black solid points are training points.

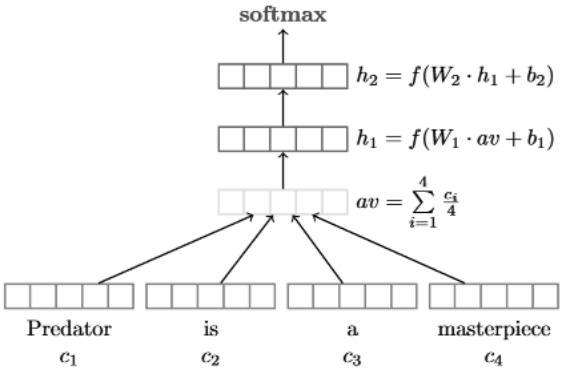
14.1. [1 pt] If we use the 2-nearest neighbor version of this method, given a test point at the red value (x = 4), which of the hollow red points A,B,C,D would be the most likely prediction?

14.2. [1 pt] What about if we used the 4-nearest neighbor version?

14.3. [1 pt] In a higher dimensional setting with $x \in R^{1000}$, would you prefer to use this method or linear regression? Justify your choice with at least one concrete reason

15. [3 pt] Suppose we use a logistic regression model for a three-class classification problem having two features. Let the features be represented as $[1, x_1, x_2]$ (includes the bias term). The trained parameters for the classes are $\theta_1 = [1, 4, 3]$, $\theta_2 = [5, 2, 3]$, $\theta_3 = [1, 3, 4]$. Write a set of linear equations ax+by=c that define the lines that make up the decision boundary for the trained model.

16. [3 pts] Consider the architecture for deep averaging networks (DAN) that you implemented for homework for classification (right)



(1) [1 pt] (True/False) Are DANs invariant to the order of tokens in the input sentence? In other words, if we swap the position of two words in an input sentence to the model, is it always true that the model will output the same results?

(2) [2 pt] Consider a slight variation to the DAN architecture. Instead of taking average right after the word embedding layer, we concatenate together the word embeddings and put them through a linear feed-forward layer that produces an embedding of equal size as a single word embedding. Is this model invariant to the order of the input sentence? Briefly justify your answer.

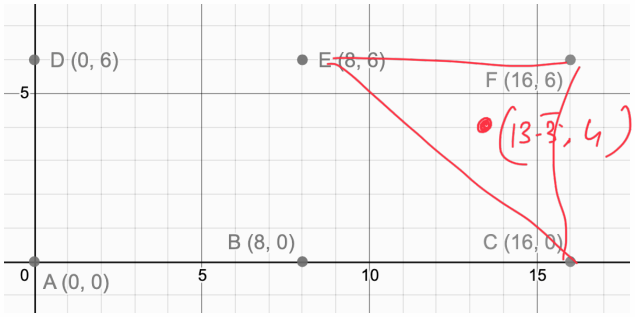
17. [3 pt] A convolutional layer in a neural network is composed of three 2x2 filters (with biases) and is applied to an input volume of shape (10, 10, 3) with no image padding and a stride of 2.

17.1. [1.5 pt] What is the shape of the output volume from the layer?

17.2. [1.5 pt] How many learnable parameters are in the layer?

18. [3 pts] Consider the k-means clustering problem shown in the figure below, with k=3, and six data points A through F. We use the ℓ_2 distance metric. A clustering is called *stable* if one more iteration of the k-means algorithm leaves the 3 clusters unchanged. Which of the following clusterings are stable?

(1) {A, B, E}, {C, D}, {F}
(2) {A, B}, {D, E}, {C, F}
(3) {A, D}, {B, E}, {C, F}
(4) {A}, {D}, {B, C, E, F}
(5) {A, B}, {D}, {C, E, F}



19. [4 pt] Consider a linear regression model in one dimension with no bias terms, y=w*x, where y and x are scalars. The weight w is initialized to 0. You are given a dataset with a single point, (x = 1, y =1) and your loss function is mean squared error. Compute the value of the parameters for two steps of gradient descent with momentum. Show your work. Recall the form of gradient descent with momentum:

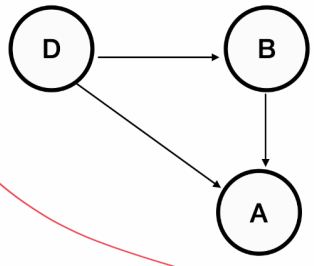
$$momentum_{t+1} = \mu * momentum_t - \alpha \nabla_t$$
$$w_{t+1} = w_t + momentum_{t+1}$$

We will set $\alpha = .5$ and $\mu = 0.9$.

20. [4 pts] Consider the following Bayesian network over 3 binary variables: D, a disease, and A and B, the results of two lab tests.

11.1 [1 pt] Write an expression for the joint distribution over D,A,B that is consistent with this Bayesian network.

11.2 [3 pts] Use the tables on the right to compute the probability of having the disease (D = 1) and test A returning positive result (A = 1).



P(A | D, B)			
d	b	a	
1	1	1	0.9
1	1	0	0.1
1	0	1	0.8
1	0	0	0.2
0	1	1	0.6
0	1	0	0.4
0	0	1	0.1
0	0	0	0.9

P(B|D)		
d	b	
1	1	0.7
1	0	0.3
0	1	0.5
0	0	0.5

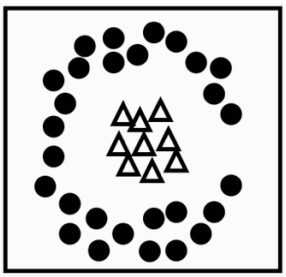
P(D)	
d	
1	0.1
0	0.9

Section C

5,7 -point questions, that are each intended to take 5-10 mins to complete (total: 33 pts, 6 questions, ~45 mins)

21. [5 pt] Given a dataset of two dimensional points (right), each belonging to one of two classes, Judy wants to build a neural network to classify a new point. She will directly feed point coordinates into the network. She found an architecture online, written in pytorch, but some parts are missing. Here is the code she found:

```
import torch.nn.functional as F
import torch.nn as nn

class My_Neural_Network(torch.nn.Module):

  def __init__(self,input_dimension,output_dimension=2):

    super(My_Neural_Network, self).__init__()

    self.layer1 = torch.nn.Linear( 2 , 4)                # Input Linear layer  with _____ inputs and 4 outputs
    self.layer2 = torch.nn.Linear( 4 , 3)                # Hidden Linear layer with _____ inputs and 3 outputs
    self.layer3 = torch.nn.Linear( 3 ,output_dimension) # Output Linear layer with _____ inputs and 2 outputs


  def forward(self,x):
    x1 = self.layer1(x)             # Passing input through Linear Layer 1
    x2 = self.layer2(x1)            # Passing output from layer 1 to layer 2
    x3 = self.layer3(x2)            # Passing output from layer 2 to layer 3

    return x3                       # Return final output
```



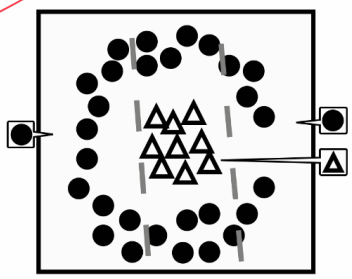
21.1. [1 pt] Fill in the missing blanks in the __init__ method so that the forward executes correctly.

21.2. [4 pt] If she trains this neural network correctly and with sufficient data (from the distribution shown above, which of the following classification boundaries might she possibly observe? Choose all that apply, if any, and explain your reasoning.

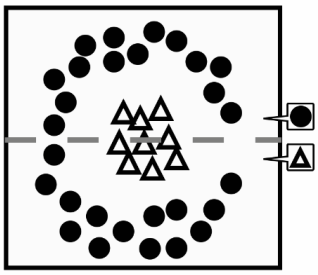
B because linear.

Assume that the network's outputs represent scores for the 2 classes, trained w. a softmax loss.

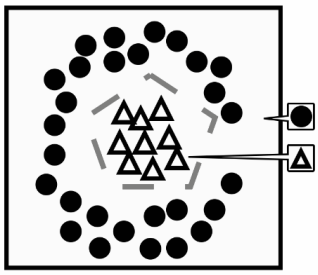
(a)



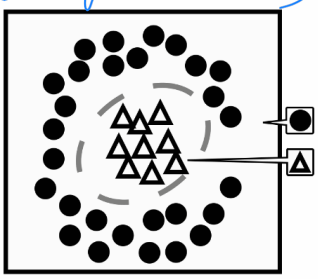
(b)



(c)



(d)



22. [5 pt] Consider a system for radiation therapy planning. Given a patient with a malignant tumor, the problem is to select the optimal radiation exposure time for that patient. A key element in this problem is estimating the probability that a given tumor will be eradicated given certain features. A data scientist collects the following information relating to this radiation therapy system. X_1 denotes time in milliseconds that a patient is irradiated with, X_2 = holds the size of the tumor in centimeters, and Y notates a binary response variable indicating if the tumor was eradicated. The data scientist fits a logistic regression model $Pr\{Y = 1|X\} = \sigma(\theta_0 + \theta_1 X_1 + \theta_2 X_2)$ to the dataset and obtains the following parameter estimates. $\hat{\theta}_0 = -6$, $\hat{\theta}_1 = 0.05$, $\hat{\theta}_2 = 1$.

$-6+2+3.5 = -0.5$ → $\frac{1}{1+e^{-(-0.5)}} = \frac{1}{1+e^{0.5}} = 0.3775$

22.1. [3 pt] Estimate the probability that, given a patient who undergoes the treatment for 40 milliseconds and who is presented with a tumor sized 3.5 centimeters, the system eradicates the tumor.

up to 2 decimal places

50

22.2. [2 pt] For how many milliseconds would the patient in part (a) need to be radiated to have exactly a 50% chance of eradicating the tumor?

$-6 + 0.05 x_1 + 1 x_2 = 0 \Rightarrow -6 + 5.5 + 0.05 x_1 = 0 \Rightarrow 2.5$

$x_1 = 50$

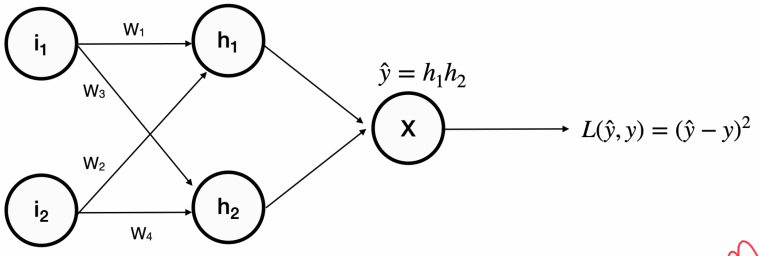
23. [6 pt] Consider the network shown below, being used for regression. At initialization, $w_1 = w_3$, and $w_2 = w_4$. Suppose our training dataset is a single sample, $(i_1 = 1, i_2 = 4, y = 3)$

23.1. [4 pts] Compute the ratio of weights w_1/w_3 and w_2/w_4 after one step of gradient descent.

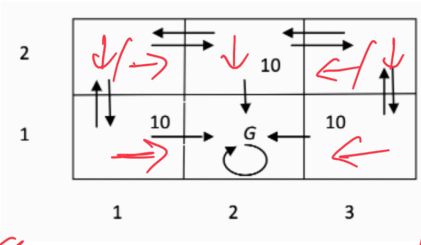
$\frac{w_1}{w_3} = \frac{w_2}{w_4} = 1.0$

23.2. [2 pt] What are those same ratios at convergence? Show your work in detail.

→ 1.0



24. [5 pt] Consider a deterministic grid world shown in the figure with an "absorbing" state G: any action performed at this state leads back to the same state. The immediate rewards are 10 for the labeled transitions and 0 for the unlabelled transitions. The discount factor is $\gamma = 0.8$

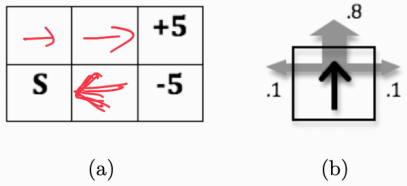


24.1. [1pt] Show the optimal policy by drawing arrows corresponding to optimal actions for each cell in the grid.

24.2. [1 pt] Compute the optimal V-value function $V*$ for the top left state (column 1, row 2) in this grid world.

24.3. [3 pt] Now, consider applying the Q-learning algorithm to this grid world. Assuming the table of Q-values is initialized to zero. Assume the agent begins in the bottom left grid square and then travels clockwise along the perimeter of the grid until it reaches the absorbing goal state, completing the first training episode. Describe which Q-values are modified as a result of this episode, and give their revised values.

25. [5 pt] Consider this gridworld MDP with 2 rows and 3 columns, which operates like the one we saw in class. The states are grid squares, identified by their row and column number (row first, indexed bottom to top and column second, indexed left to right). The agent always starts in state (1,1), marked with the letter S. As marked in the figure, there are two terminal goal states: (2,3) with reward +5, and (1,3) with reward -5. Rewards are 0 in non-terminal states. The reward for a state is received as the agent moves into the state. The transition function is such that the intended agent movement (North, South, West, or East) happens with probability .8. With probability .1 each, the agent moves in one of the directions perpendicular to the intended direction. If a collision with a wall happens, the agent stays in the same state.



(a) (b)

Figure 1: (a) Gridworld MDP. (b) Transition function.

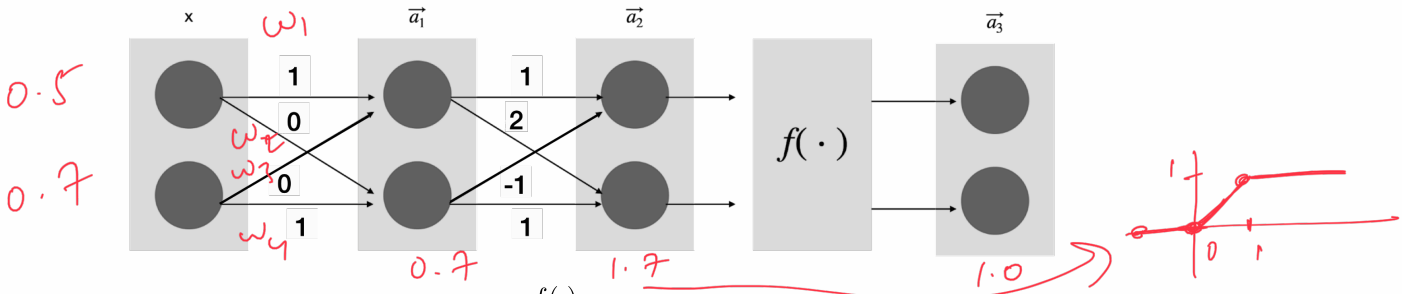
25.1. [1 pt] Draw the optimal policy for this grid?

25.2. [4 pt] Suppose the agent does not know the transition probabilities in this MDP. It starts with a policy that always chooses to go right, and executes three trials running through the environment, recorded below as the sequence of states traversed in each trial:

1) States (1,1) → (1,2) → (1,3),
2) States (1,1) → (1,2) → (2,2) → (2,3), and
3) States (1,1) → (2,1) → (2,2) → (2,3).

Given only these three trials, estimate the policy value function for this "always right" policy, at states (1,1) and (2,2).

26. [7 pt] Consider a neural network that maps from an input vector x of length 2 to an output vector of length 2. Consider the following architecture, containing standard linear fully connected layers mapping from x to $\vec{a_1}$, $\vec{a_1}$ to $\vec{a_2}$, and a nonlinear activation function $f(\cdot)$ mapping from $\vec{a_2}$ to $\vec{a_3}$, where $f(a) = min(max(a, 0), 1)$.
For simplicity in the following, consider a training dataset with a single data point, where the input vector is x=[0.5, 0.7] and the target or label vector is y=[0.7, 0.5].



26.1. [1 pt] Draw a plot of the activation function $f(\cdot)$

26.2. [2 pt] Suppose that the output $\hat{y}(x)$ of the above network is set to $\vec{a}_3$, and the loss function is the squared l2 norm distance $||\hat{y}(x) - y(x)||_2^2$. Report the gradients of the loss with respect to the weights in the first layer (from x to a1) . Hint: Carefully observing the shape of your plot in part 1 might simplify your calculations.

26.3. [4 pt] Suppose that the output $\hat{y}(x)$ of the above network is instead set to $\hat{y}(x) = \vec{a}_1(x) + \vec{a}_3(x)$. Recall that this type of "skip connection" is a key building block of residual network architectures. Using the same squared l2 norm objective function, recompute the gradients of the loss with respect to the weights in the first layer (from x to $\vec{a}_1$).

$$\frac{\partial \mathcal{L}}{\partial w_1} = \frac{\partial \mathcal{L}}{\partial \hat{y}_1} \cdot \frac{\partial \hat{y}_1}{\partial \hat{a}_{11}} \cdot \frac{\partial \hat{a}_{11}}{\partial w_1} = \frac{\partial \mathcal{L}}{\partial \hat{y}_1} \cdot x_1$$

---------- END OF EXAM ----------

$$\hat{y} = \begin{bmatrix} 0 \\ 1.0 \end{bmatrix} + \begin{bmatrix} 0.5 \\ 0.7 \end{bmatrix} = \begin{bmatrix} 0.5 \\ 1.7 \end{bmatrix} \Rightarrow \mathcal{L} = \|\hat{y} - y\|_2^2 = \left\| \begin{bmatrix} 0.2 \\ 1.2 \end{bmatrix} \right\|_2^2$$

$$\frac{\partial \mathcal{L}}{\partial \hat{y}} = \frac{\partial \sum (\hat{y}_i - y_i)^2}{\partial \hat{y}_i} = 2(\hat{y}_i - y_i) = \begin{bmatrix} -0.4 \\ 2.4 \end{bmatrix} = \begin{matrix} \partial \mathcal{L}/\partial \hat{y}_1 \\ \partial \mathcal{L}/\partial \hat{y}_2 \end{matrix}$$

$$\frac{\partial \mathcal{L}}{\partial w_1} = \frac{\partial \mathcal{L}}{\partial \hat{y}_1} \cdot \frac{\partial \hat{a}_{11}}{\partial w_1} = -0.4 \times 0.5 = -0.2$$

$$\frac{\partial \mathcal{L}}{\partial w_2} = \frac{\partial \mathcal{L}}{\partial \hat{y}_2} \cdot \frac{\partial \hat{a}_{12}}{\partial w_2} = 2.4 \times 0.5 = 1.2$$

$$\frac{\partial \mathcal{L}}{\partial w_3} = \frac{\partial \mathcal{L}}{\partial \hat{y}_1} \cdot \frac{\partial \hat{a}_{11}}{\partial w_3} = -0.4 \times 0.7 = -0.28$$

$$\frac{\partial \mathcal{L}}{\partial w_4} = \frac{\partial \mathcal{L}}{\partial \hat{y}_2} \cdot \frac{\partial \hat{a}_{12}}{\partial w_4} = 2.4 \times 0.7 = 1.68$$

<left empty for scratch space>


TURING TEST EXTRA CREDIT:
CONVINCE THE EXAMINER THAT HE'S A COMPUTER.
YOU KNOW, YOU MAKE SOME REALLY GOOD POINTS.
I'M ... NOT EVEN SURE WHO I AM ANYMORE.