CSE 493 G1 / 599 G1
Deep Learning
Winter 2026 Exam 1

February 5, 2026

Full Name (as on Gradescope): ______________________________

UW Net ID: ______________________________

Question		Score
True/False	(14 pts)	
Multiple Choice	(40 pts)	
FR - Jacobians	(6 pts)	
FR - Linear Classifier	(40 pts)	
Total	(100 pts)	

Welcome to the CSE 493 G1 / 599 G1 Exam!

- The exam is 80 min and is **double-sided**.
- You may use a non-graphing calculator and no other electronic devices.
- One handwritten double sided cheat sheet is allowed.

I understand and agree to uphold the University of Washington Student Conduct Code during this exam.

Signature: ____________________ Date: ____________

Good luck!

This page is left blank for scratch work only. DO NOT write your answers here.

This page is left blank for scratch work only. DO NOT write your answers here.

1 True/False (14 points) - Recommended 10 Minutes

Fill in the circle next to your choice (like this: ●). No explanations are required.

Each question is worth 2 points

1.1 A linear classifier with a 2-dimensional input can classify into at most 4 classes.

○ True

○ False

1.2 When using multiclass SVM loss (hinge loss) on a single example, a model that predicts the correct class can have higher loss than a model that predicts an incorrect class.

○ True

○ False

1.3 Using softmax cross-entropy loss on a single example, a model that predicts the correct class can have higher loss than a model that predicts an incorrect class.

○ True

○ False

1.4 For a single linear layer with softmax cross-entropy loss, two SGD steps (no momentum) on the same single training example produce the same final weights as one step with double the learning rate.

○ True

○ False

1.5 If a node receives gradients from multiple upstream paths during backpropagation, those gradients are multiplied.

○ True

○ False

1.6 While training on CIFAR (a dataset with 10 equally represented classes in both test and train sets), you accidentally set the ground truth label of all of your train images to be Class 1. After training on this train set, you would expect your test accuracy to be essentially 0%

○ True

○ False

1.7 Pooling layers reduce the spatial dimensions of feature maps (aka activation maps)

○ True

○ False

2 Multiple Choice (40 points) - Recommended 30 Minutes

Fill in the circle next to the letter(s) of your choice (like this: ●). No explanations are required. **Choose ALL options that apply.**

Each question is worth 8 points and the answer may contain multiple correct options, or none at all. Selecting all of the correct options and none of the incorrect options will get full credit. For questions with multiple correct options, each incorrect or missing selection gets a 4-point deduction (up to 8 points).

2.1 Which of the following produce **zero-centered outputs** given mean-zero normal inputs? Select all that apply.

◯ A: Sigmoid
◯ B: Tanh
◯ C: ReLU
◯ D: Batch Normalization (with $\beta = 0$)
◯ E: Layer Normalization (with $\beta = 0$)
◯ F: Softmax

2.2 Consider a neural network whose parameters require **M bytes** of storage. Assume that:

- Each additional value stored (e.g., gradients, optimizer state) requires the same memory as a parameter
- Gradients for all parameters are computed and stored during backpropagation before the optimizer step (as in PyTorch/TensorFlow autograd)

Select all statements that are TRUE about the **peak memory** required to store parameters, gradients, and optimizer state. ***(Ignore memory for activations.)***

◯ A: Inference requires approximately M
◯ B: Inference requires approximately 2M
◯ C: Training with SGD (no momentum) requires approximately M
◯ D: Training with SGD (no momentum) requires approximately 2M
◯ E: Training with SGD (no momentum) requires approximately 3M
◯ F: Training with Adam requires approximately 4M
◯ G: Training with Adam requires approximately 5M

2.3 Which of the following could have **different behavior** during training vs. inference? Select all that apply.

- ◯ A: Dropout
- ◯ B: Batch Normalization
- ◯ C: Layer Normalization
- ◯ D: ReLU
- ◯ E: Softmax
- ◯ F: Max Pooling

2.4 Given a **7x7 input image**, which of the following convolution configurations will **preserve the spatial dimensions** (i.e., produce a 7x7 output)? Select all that apply.

- ◯ A: 3x3 filter, stride 1, padding 1
- ◯ B: 5x5 filter, stride 1, padding 2
- ◯ C: 3x3 filter, stride 1, padding 0
- ◯ D: 5x5 filter, stride 1, padding 1
- ◯ E: 7x7 filter, stride 1, padding 3
- ◯ F: 3x3 filter, stride 2, padding 1

2.5 You have a dataset of RGB images, each of size `(H, W, 3)`. For storage efficiency, you flatten each image into a 1D vector of length `H*W*3`. Note that if you reshape one of these vectors back to `(H, W, 3)`, you recover the original image exactly.

A malicious actor has sabotaged your pipeline. Before anyone noticed, they applied a fixed random permutation to every flattened vector in both the training and test sets. The same permutation was used for every image, and the original ordering has been lost. If you reshape one of the permuted vectors back to `(H, W, 3)`, you get a scrambled image, not the original.

You now have two versions of the dataset:

- **Original dataset:** flattened vectors of length `H*W*3` with the natural pixel ordering.
- **Permuted dataset:** flattened vectors of length `H*W*3` with the shuffled ordering.

You want to understand which architectures are hurt by this sabotage. Consider the four models below. Each first reshapes its input vector, then applies an operation, before passing the result to further layers and a loss function. All operators and functions in the table including @, $*$, .reshape(), and .sum() carry their standard NumPy meanings.

Model	Reshape	Operation
A	`x = input.reshape(H, W, 3)`	`output` of 3×3 convolution on `x`
B	`x = input.reshape(H, W, 3)`	`output` of 1×1 convolution on `x`
C	`x = input.reshape(1, H*W*3)`	`output = x @` W_C with W_C`.shape = (H*W*3, Z)`
D	`x = input.reshape(H*W, 3, 1)`	`X = x *` W_D `output = X.sum(axis=0).sum(axis=0)` with W_D`.shape = (H*W, 3, Z)`

For each model, suppose you train and test it in two separate settings:

- **Original setting:** train on the original training set, test on the original test set.
- **Permuted setting:** train on the permuted training set, test on the permuted test set.

Which model(s) would you generally expect to have **worse test performance** in the permuted setting compared to the original setting?

◯ A: Model A

◯ B: Model B

◯ C: Model C

◯ D: Model D

3 Jacobians (6 points) - Recommended 5 Minutes

Consider the matrix multiplication $AB = C$ where A is of shape 2×3 and B is of shape 3×5.

3.1 Jacobian Dimension (2 pts)

A has 6 elements and B has 15 elements. How many elements are in the Jacobian $\frac{\partial C}{\partial A}$?

3.2 Jacobian Elements (4 pts)

How many of these elements will always be 0 for all values of A and B?

4 Linear Classifier (40 points) - Recommended 30 Minutes

Please make sure to write your answer only in the provided space.

You are using the below network to classify books as either Hardcover (Class A) or Softcover (Class B). The network is a single linear layer that you are training with hinge loss. The inputs to the network are two values:

- x — weight of the book (normalized)
- y — number of chapters (normalized)

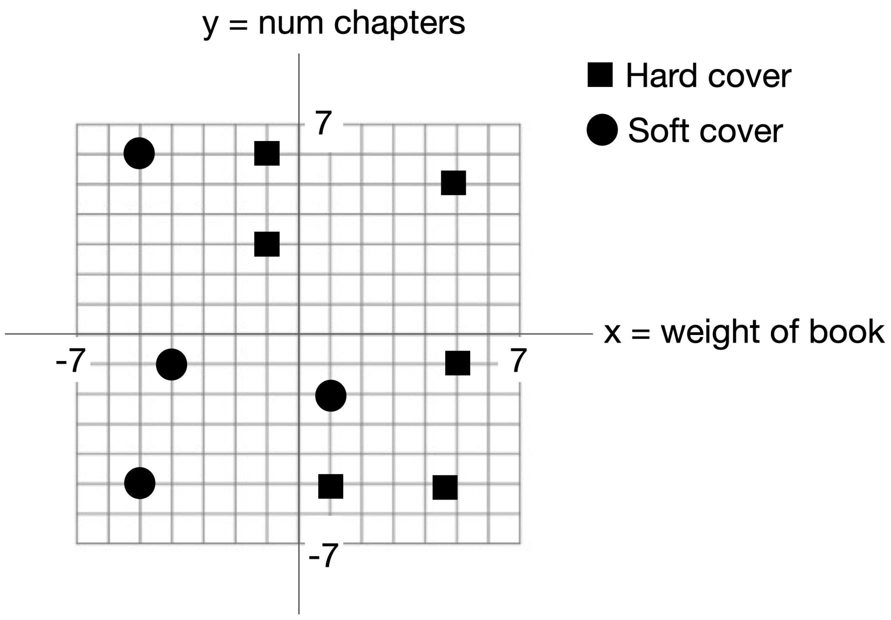
The current weights of the network are given below:

$$\begin{bmatrix} W_{00} & W_{01} \\ W_{10} & W_{11} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 & 3 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} s_A \\ s_B \end{bmatrix}$$

Hinge Loss: $L_i = \sum_{j \neq y_i} \max(0, s_j - s_{y_i} + 1)$

4.1 Geometric View (4 pts)

Below is a plot with a number of test points, with their ground truth label. Use the weight matrix to calculate the equation of the decision boundary and draw this line on the plot. Please draw the line boldly. Hint: the decision boundary corresponds to when the score of one class becomes greater than the score of the other class.



4.2 Accuracy (4 pts)

Given your decision boundary, what is the current test accuracy? Test accuracy is given as (Number of correctly predicted test points)/(Total test points). Hint: It may help to label which side of your decision boundary corresponds to which class on the plot above.

4.3 Training Example (4 pts)

You are given a book, Book β, as a training example. It is a softcover book, with normalized weight of 0 and normalized chapter numbers of 1. How does the model currently classify this book?

4.4 Complete computational graph (4 pts)

At the end of this question, there is an incomplete computational graph corresponding to the classification of the above training example. Complete the computational graph by filling in all the circles with the correct operation (e.g. $*$, $+$, max). There are 8 empty circles that need to be filled in.

(Answer in the graph. No written explanation needed.)

4.5 Forward Pass (4 pts)

Now complete a forward pass of Book β. Fill in all the pentagons with the correct values. There are 10 pentagons.

(Answer in the graph. No written explanation needed.)

4.6 Backward Pass (4 pts)

Now complete a backward pass of Book β. Fill in all the squares with the gradient values. Fill in the upstream gradient times the local gradient. To allow you to fill in all these values in the graph, the squares are given as $\underset{\text{Upstream}}{\boxed{}} \times \underset{\text{Local}}{\boxed{}} = \boxed{}$

(Answer in the graph. No written explanation needed.)

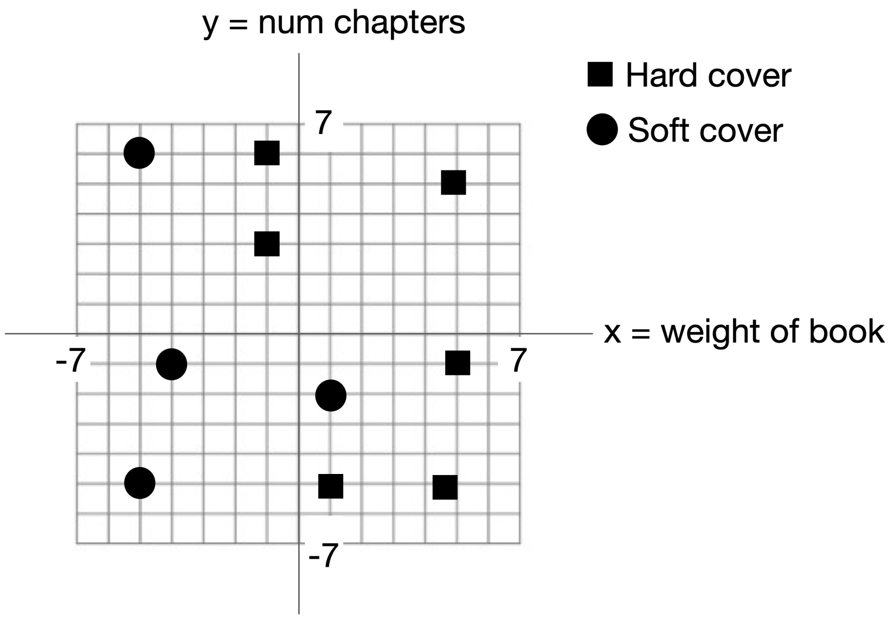
4.7 Update Weights (4 pts)

Given the gradients of the weights, update the weights, using a learning rate of 1.

$$\begin{bmatrix} 1-\underset{lr}{\boxed{1}}\times\underset{\nabla}{\boxed{}} & 3-\underset{lr}{\boxed{1}}\times\underset{\nabla}{\boxed{}} \\ 0-\underset{lr}{\boxed{1}}\times\underset{\nabla}{\boxed{}} & 1-\underset{lr}{\boxed{1}}\times\underset{\nabla}{\boxed{}} \end{bmatrix} = \begin{bmatrix} \boxed{} & \boxed{} \\ \boxed{} & \boxed{} \end{bmatrix}$$

4.8 Updated Geometric View (4 pts)

Calculate the new decision boundary and draw it on the below plot. Please draw the line boldly.

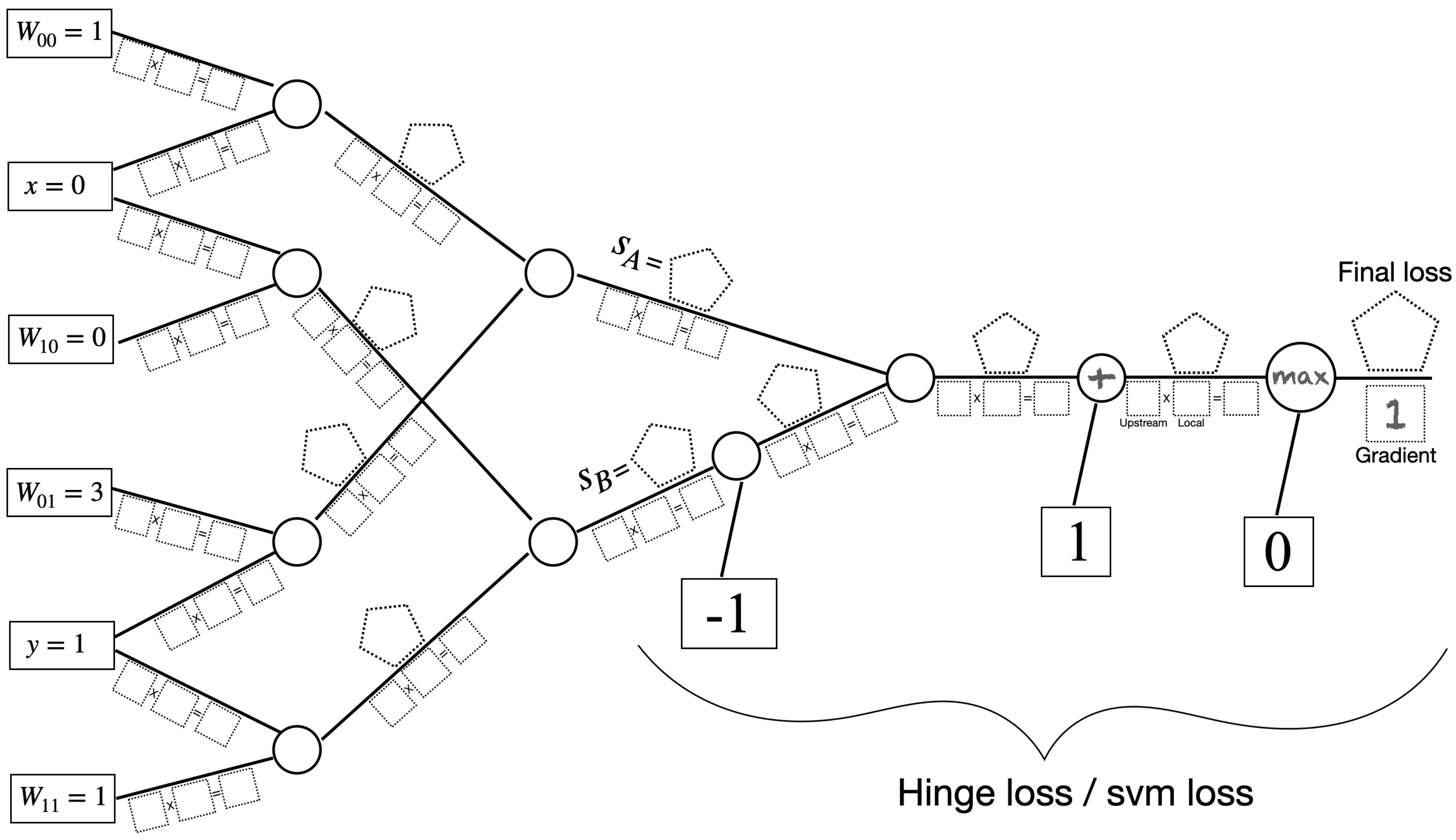


4.9 Updated Accuracy (4 pts)

What is the new accuracy?

4.10 Interpret the Results (4 pts)

Consider the old decision boundary and the new decision boundary. Which makes more sense to you based on your knowledge of the attributes we are examining and hardcover/softcover books? Explain your answer in a sentence or two.


$W_{00} = 1$
$x = 0$
$W_{10} = 0$
$W_{01} = 3$
$y = 1$
$W_{11} = 1$
$s_A =$
$s_B =$
-1
1
0
+
max
×
=
Upstream
Local
Final loss
1
Gradient
Hinge loss / svm loss