Midterm Exam

Midterm exam will be done individually. Use of partial or entire solutions obtained from others or online is strictly prohibited.

- There will be 8 pages in this exam (including this cover sheet)
- This is a **SEMI-OPEN BOOK** exam. You can only use our lecture slides. You **CANNOT** use other books or online materials.
- For **MULTIPLE CHOICE** questions, choose ALL options that apply. Selecting all of the correct options will get full credits. Partial credits will be given to correct but incomplete options. Answers containing any incorrect options will get 0 points.
- Work efficiently and independently.
- You have 150 minutes.
- Good luck!

Question	Topic	Max. score	Score
1	True/False	10	
2	Multiple Choice	20	
3	Vector Derivative	15	
4	Back-propagation	20	
5	CNN architecture	20	
6	Short Answer Questions	15	
Total		100	

1. **True/False Question** (10 points)

   (a) (2 pts) **True** or **False**: The set $\left\{\mathbf{x} \in R^2 \;\middle|\; x_1^2 + x_2^2 \geq 1\right\}$ is a convex set.

   (b) (2 pts) **True** or **False**: Consider the unconstrained optimization problem: $\min_w Q(w)$. The derivative of Q w.r.t. w at a is zero:

   $$\frac{\partial Q}{\partial w}\Big|_{w=a} = 0.$$

   Then the point $w = a$ is a local optimum or the global optimum.

   (c) (2 pts) **True** or **False**: A neural network with multiple hidden layers and ReLU nodes can form non-linear decision boundaries.

   (d) (2 pts) **True** or **False**: Consider a trained logistic regression. The weight vector is W and its test accuracy on a given data set is A. Assuming there is NO bias, dividing W by 10 won't change the test accuracy.

   (e) (2 pts) **True** or **False**: We are given a training set and a test set. We are asked to train a LASSO model:

   $$\min_{\mathbf{w}} \; \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2 + \lambda \|\mathbf{w}\|_1.$$

   Since $\lambda \geq 0$ is a tuning hyper-parameter, we need to find a good λ. So we can propose a grid, e.g., $\lambda \in \{10^{-6}, 10^{-5}, \cdots, 10^{6}\}$, and use the **test set** to choose the λ that **minimizes** the **test mean squared error (MSE)**.

2. **Multiple Choice** (20 points)

(a) (3 pts) Suppose we build a softmax classifier for multi-class classification. The number of classes is 1000. The inputs are 100-dimensional feature vectors. What is the number of trainable parameters?

A. $\approx 10^5$

B. $\approx 10^4$

C. $\approx 10^3$

D. $\approx 10^2$

(b) (3 pts) The quality of wine can be (1) outstanding, (2) very good, (3) good, (4) mediocre, or (5) not recommended. We want to predict the quality based on the measurements of chemicals, e.g., alcohol, malic acid, magnesium, etc. (Hint: "outstanding" is better than "very good", "very good" is better than "good', and so on.)

A. This is a classification task.

B. This is a regression task.

C. This is a clustering task.

D. This is a dimensionality reduction task.

(c) (3 pts) Suppose we want to train a CNN on the large scale ImageNet dataset. As we make the CNN deeper, both the training and test errors are getting worse. As the CNN net gets deeper, what makes the performance get worse?

A. Underfitting.

B. Lack of good regularization.

C. Overfitting.

D. Poor optimization.

(d) (3 pts) The input is a 20×20 matrix and the filter (aka kernel) is a 5×5 matrix. We set stride $= 1$ and no zero-padding. What is the shape of the output matrix (aka feature map)?

A. 1×1.

B. 4×4.

C. 5×5.

D. 15×15.

E. 16×16.

F. 20×20.

(e) (3 pts) We want to apply dropout to a dense layer (aka fully-connected layer):

$$\mathbf{x}^{\text{out}} \;=\; \mathsf{ReLU}\big(\mathbf{W}\mathbf{x}^{\text{in}} + \mathbf{b}\big).$$

We can randomly mask 50% entries of ____?_____ and scale the rest entries by 2.

A. the input vector $\mathbf{x}^{\text{in}}$

B. the output vector $\mathbf{x}^{\text{out}}$

C. the parameter matrix $\mathbf{W}$

D. both the parameter matrix $\mathbf{W}$ and intercept vector $\mathbf{b}$

(f) (5 pts) Which of the following are valid activation functions (elementwise non-linearities) you could use in a neural network? (That is, which functions could be effective when training a neural net in practice?)

A. $f(x) = \min(0, x)$

B. $f(x) = \max(0.25x, 0.75x)$

C. $f(x) = 0.5x$

D. $f(x) = \begin{cases} 1, & \text{if } x \ > 1 \\ -1, & \text{otherwise} \end{cases}$

Please also briefly **EXPLAIN** why the remaining options CANNOT be used as non-linear activation function in network training.

3. **Vector Derivative** (15 points) Let x_1, x_2 be scalar values and $\mathbf{x} \in \mathbf{R}^{2\times 1}$ be the concatenate of x_1 and x_2. Let $J = \frac{1}{2}(x_1^2 x_2 - y)^2$.

   (a) (2 pts) Calculate $\frac{\partial J}{\partial x_1}$.

   (b) (3 pts) Calculate second-order derivative $\frac{\partial^2 J}{\partial x_1 \partial x_2}$.

   (c) (5 pts) Calculate $\frac{\partial J}{\partial \mathbf{x}}$.

   (d) (5 pts) Calculate second-order derivative $\frac{\partial^2 J}{\partial \mathbf{x} \partial \mathbf{x}^T}$. (Hint: you should get 2×2 matrix)

4. **Back-propagation** (20 points) Suppose we train 2 layer fully connected neural network with the following structure for the regression.

   - Input layer: 2-dimensional input vector.
   - Hidden layer: 3 hidden units with ReLU non-linear activation.
   - Output layer: 1 scalar output without any non-linear activation.

   Noted that ReLU activation is defined as ReLU(x)=max$(0, x)$, we also use following notations:

   - $\mathbf{x}$ is the training input vector, y denotes the true target. $\hat{y}$ is the output of your neural network. All vectors are **column vectors**. Note that vector $\mathbf{x}$ has two elements, y and $\hat{y}$ has only one element (i.e., scalar).
   - We consider L_2 squared loss for regression, i.e., $L = \frac{1}{2}|y - \hat{y}|^2$.
   - Denote $\mathbf{g}$ is the vector of hidden unit values before non-linear activation functions are applied, $\mathbf{h}$ is the vector of hidden unit values after they are applied.
   - $\mathbf{V}$ is the weight matrix that map the input layer to the hidden layer (assume NO bias)
   - $\mathbf{W}$ is the weight matrix that map the hidden layer to the output layer (assume NO bias)

   (a) (3 pts) Write down the corresponding math expression for each layer using the notations defined above. (Hint: you may have 3 equations)

   (b) (6 pts) Calculate $\frac{\partial L}{\partial W}$

   (c) (6 pts) Calculate $\frac{\partial L}{\partial V}$

   (d) (5 pts) Suppose we have n training samples, i.e., $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$. Briefly describe how you would use **stochastic gradient decent** to update model parameters $\mathbf{V}$ and $\mathbf{W}$. Write down the update rule in each iteration. Assume the learning rate is η.

5. **CNN Architecture** (20 points) Consider the convolutional neural network defined by the layers in the left column below. Fill in the size of output feature map at each layer, and the number of trainable parameters at each layer. You can write your output shape as an array (e.g. $[150, 150, 3]$), and just list the multiplication expression for number of paramters.

   - CONV(N, K) denotes the 2D convolutional layer with N filters with spatial filter size $K \times K$. By default, stride is 1 and there is no padding involved.
   - POOL(K) denotes the 2D maximum pooling layer with pool size $K \times K$. By default, pool stride is 2
   - FC(N) denotes the fully connected layer (dense layer) with N output neurons. Noted that before FC layer, you might need to "flatten" the feature map.

Layer	Output shape	Number of parameters
Input	[32, 32, 3]	0
CONV(10, 5)		
POOL(2)		
CONV(5, 3)		
POOL(2)		
FC(10)		

6. **Short Answer Question** (15 points)

   (a) (3 pts) Describe the main difference between full-batch gradient descent, mini-batch gradient descent and stochastic gradient descent. Name one advantage of using mini-batch gradient descent over full-batch gradient descent.

   (b) (3 pts) Why do the deep neural network needs to include non-linear activation function?

   (c) (4 pts) Describe two techniques that could reduce the over-fitting of your neural network. Noted that you need to briefly explain those techniques.

   (d) (5 pts) How to tune the *hyper-parameters*? Describe the basic steps.