[redacted]

[redacted]

[redacted]

Question 1

1 / 1 pts

For this question, please read the paper: **Rumelhart, Hinton and Williams (1986 (http://www.cs.toronto.edu/~hinton/absps/naturebp.pdf)) (http://www.cs.toronto.edu/~hinton/absps/naturebp.pdf)** .

[Can be found at: http://www.cs.toronto.edu/~hinton/absps/naturebp.pdf]

One drawback of the learning procedure in the paper is that the error-surface may contain local minima so that gradient descent is not guaranteed to find a global minimum. This happens if the network has **more** than enough connections.

○ True

● False

"Adding a few more connections creates extra dimensions in weight-space and these dimensions provide paths around the barriers that create poor local minima in the lower dimensional subspaces" -p535

Answer key: Happens if the network has *just* enough connections. The question here asks if the network has “more than enough” connections, which in that case, it will be able to create a path to go around this barrier.

Question 2

1 / 1 pts

(Select all that apply) As discussed in lecture, which of the following is true for the backpropagation algorithm?

Hint: Lecture 5, starting at "training by gradient descent".

☑ It cannot be performed without first doing a feed-forward pass of the input(s) through the network

☑ It is used to compute derivatives that are required for the gradient descent algorithm that trains the network

☑ It can be used to compute the derivative of the divergence with respect to the input of the network

□ It computes the derivative of the average divergence for a batch of inputs

☑

It computes the derivative of the divergence between the true and desired outputs of the network for a training input

Question 3

1 / 1 pts

We are given a binary classification problem where the training data from both classes are linearly separable. We compare a perceptron, trained using the perceptron learning rule with a sigmoid-activation perceptron, trained using gradient descent that minimizes the L2 Loss. In both cases, we restrict the weights vector of the perceptron to have finite length. In all cases, we will say the algorithm has found a “correct” solution if the learned model is able to correctly classify the training data. Which of the following statements are true (select all that are true).

Hint: See slides 13-32, lecture 6

□ The gradient-descent algorithm will always find the correct solution.

□ We cannot make any statement about the truth of falsity of the other options provided, based only on the information provided.

☑ There are situations where the gradient-descent algorithm will not find the correct solution.

☑ The perceptron algorithm will always find the correct solution.

Question 4

1 / 1 pts

Consider a perceptron in a network that has the following vector activation:

$$y_j = \prod_{j \neq i} z_i$$

Where y_j is the j-th component of column vector y, and z_i is the i-th component of column vector z. Using the notation from lecture, which of the following is true of the derivative of y w.r.t. z? (select all that are true)

Hint: Vector calculus notes 1 (and beyond)

□ It is a column vector whose i-th component is given by $\prod_{j \neq i} z_j$

☑ It is a matrix whose (i, j)th component where $i \neq j$ is given by $\prod_{k \neq i, k \neq j} z_k$

□ It is a row vector whose i-th component is given by $\prod_{j \neq i} z_j$

☑ It will be a matrix whose diagonal entries are all 0.

□ It is a matrix whose (i,j)th component is given by $z_i z_j$

Question 5

1 / 1 pts

(Select all that apply) At any point, the gradient of a scalar function with multivariate inputs…

Hint: Lecture 4, "Gradient of a scalar function of a vector" and "properties of a gradient".

☑ Is the vector of local partial derivatives w.r.t. all the inputs

☑ Is in the direction of steepest ascent

□ Is parallel to equal-value contours of the function

□ Is in the direction of steepest descent

Question 6

1 / 1 pts

Which of the following are valid subgradients of a RELU, given by Relu(x), at x = 0? We will represent the subgradient as $\nabla_{subgrad} RELU\left(x\right)$

Hint: Lecture 5, slides 112-114.

☑ $\nabla_{subgrad} RELU\left(0\right) = 1$

□ $\nabla_{subgrad} RELU\left(0\right) = 1.5$

☑ $\nabla_{subgrad} RELU\left(0\right) = 0$

☑ $\nabla_{subgrad} RELU\left(0\right) = 0.5$

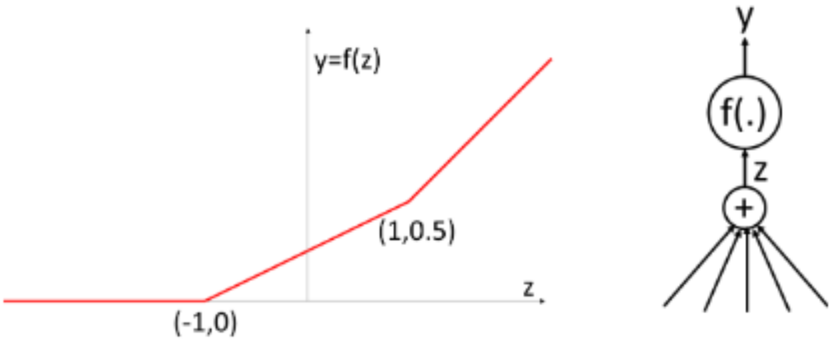
□ $\nabla_{subgrad} RELU\left(0\right) = -0.5$

Question 7

1 / 1 pts

The following piecewise linear function with “hinges” at (-1,0) and (1,0.5) is used as an activation for a neuron. The slope of the last segment is 40 degrees with respect to the *z* axis (going anti-clockwise). Our objective is to find a z that minimizes the divergence div(y,d). Which of the following update rules is a valid subgradient descent update rule at z=1? Here η is the step size and is a positive number.

The superscript on *z* represents the step index in an iterative estimate. The derivative $\frac{\partial div(y,\, d)}{\partial\ z}$ is computed at $z^k = 1$. The value of η must not factor into your answer (i.e. remember that η has only been included in the equations for completeness sake and do not argue with us that you can always adjust η to make any answer correct ☺)



Hint: Lecture 5, slides 112-114

□ $z^{k+1} = z^k + \eta \frac{\partial div(y,d)}{\partial y}$

☑ $z^{k+1} = z^k - \eta 0.75 \frac{\partial div(y,d)}{\partial y}$

☑ $z^{k+1} = z^k - \eta 0.25 \frac{\partial div(y,d)}{\partial y}$

□ $z^{k+1} = z^k - \eta \frac{\partial div(y,d)}{\partial y}$

□ $z^{k+1} = z^k - \eta 0.1 \frac{\partial div(y,d)}{\partial y}$

Question 8

1 / 1 pts

Gradient descent yields a solution that is not sensitive to how a network’s weights are initialized.

Hint: Basic gradient descent from lecture 5 - slide 5

○ True

● False

Question 9

1 / 1 pts

What are the challenges of using Newton’s method with neural networks?

Hint: Lecture 6, "Issues 1"

□ It cannot find the minimum in any quadratic function

□ It has a very large Jacobian

☑ It is difficult to compute the inverse of the Hessian

☑ It can produce unstable updates if the optimized function is not strictly convex

☑ Its memory usage scales with the square of the number of weights

Question 10

1 / 1 pts

Let $f(.)$ be a scalar-valued function with multivariate input and $x = [x_1, x_2]$ be a two-component vector such that $y = f(x)$. y is being minimized using RProp from lecture. In the k-th iteration, the derivative of y with respect to x_1 is $\frac{dy}{dx_1} = 2$, the derivative of y with respect to x_2 is $\frac{dy}{dx_2} = -1$. As a result, x_1 has a step size of

$\Delta x_1^{(k)} = 1$ and x_2 has a step size of $\Delta x_2^{(k)} = 1$. At the (*k*+1)-th iteration, the derivative of y with respect to x_1 is $\frac{dy}{dx_1} = 0.5$ and the derivative of y with respect to x_2 is $\frac{dy}{dx_2} = 1$. Which of the following is true about the step size at the (*k*+1)-th iteration?

Hint: Lecture 6, RProp

○ $\Delta x_1^{(k+1)} < 1$ and $\Delta x_2^{(k+1)} > 1$

○ $\Delta x_1^{(k+1)} < 1$ and $\Delta x_2^{(k+1)} < 1$

● $\Delta x_1^{(k+1)} > 1$ and $\Delta x_2^{(k+1)} < 1$

○ $\Delta x_1^{(k+1)} > 1$ and $\Delta x_2^{(k+1)} > 1$

[redacted]