[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)** **(http://www.cs.toronto.edu/~hinton/absps/naturebp.pdf)** .

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

One version of gradient descent changes each weight by an amount proportional to the accumulated $\delta E/\delta w$.

$$\Delta w = -\epsilon \frac{\delta E}{\delta w}$$

Select all that are true about this method:

- [x] It's simpler than methods that use second derivatives.

> "This method does not converge as rapidly as methods which make use of the second derivatives, but it is much simpler [...]" p535

- [x] It can be improved without sacrificing simplicity and locality.

> "It can be significantly improved, without sacrificing the simplicity and locality, [...]" p535

- [ ] This method converges as rapidly as methods that make use of second derivatives.

- [ ] It cannot be implemented by local computations in parallel hardware.

Question 2

1 / 1 pts

(**Select all that apply**) Which of the following is true of the vector and scalar versions of backpropagation?

Hint: Lecture 5

- [ ] Both scalar backpropagation and vector backpropagation are optimization algorithms that are used to find parameters that minimize a loss function
- [x] Scalar backpropagation and vector backpropagation only differ in their arithmetic notation and the implementation of their underlying arithmetic
- [x] Scalar backpropagation rules explicitly loop over the neurons in a layer to compute derivatives, while vector backpropagation computes derivative terms for all of them in a single matrix operation
- [ ] Scalar backpropagation is required for scalar activation functions, while vector backpropagation is essential for vector activation functions

Question 3

1 / 1 pts

Backpropagation can be applied to any differentiable activation function.

- (•) True
- ( ) False

You can propagate derivatives backward through any differentiable activation function, or even activations that only have a finite subgradient.

Incorrect

Question 4

0 / 1 pts

Which of the following is true given the Hessian of a scalar function with multivariate inputs?

Hint: Lec 4 "Unconstrained minimization of a function". Also note that an eigen value of 0 indicates that the function is flat (to within the second derivative) along the direction of the corresponding Hessian Eigenvector.

- [ ] The eigenvalues are all strictly negative at a local maximum.
- [ ] The eigenvalues are all non-negative at local minima.
- [x] The eigenvalues are all strictly positive at global minima, but not at local minima.
- [ ] The eigenvalues are all strictly positive at a local minimum.

Question 5

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)

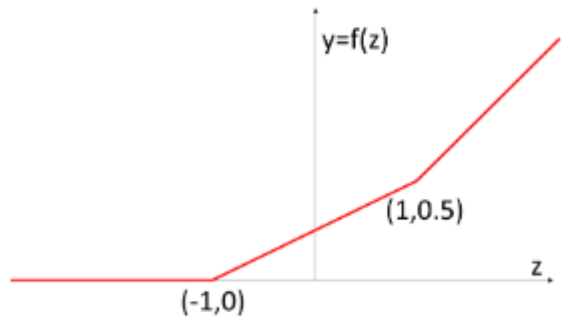
Hint: Vector calculus notes 1 (and beyond)

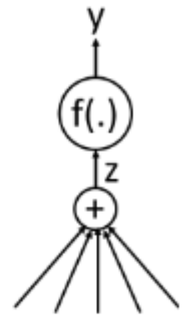
- [x] It is a matrix whose (i, j)th component where $i \neq j$ is given by $\prod_{k \neq i, k \neq j} z_k$
- [ ] It is a row vector whose i-th component is given by $\prod_{j \neq i} z_j$
- [x] It will be a matrix whose diagonal entries are all 0.
- [ ] It is a matrix whose (i,j)th component is given by $z_i z_j$
- [ ] It is a column vector whose i-th component is given by $\prod_{j \neq i} z_j$

Question 6

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

- [ ] $z^{k+1} = z^k - \eta \frac{\partial div(y,d)}{\partial y}$
- [x] $z^{k+1} = z^k - \eta 0.25 \frac{\partial div(y,d)}{\partial y}$
- [x] $z^{k+1} = z^k - \eta 0.75 \frac{\partial div(y,d)}{\partial y}$
- [ ] $z^{k+1} = z^k - \eta 0.1 \frac{\partial div(y,d)}{\partial y}$
- [ ] $z^{k+1} = z^k + \eta \frac{\partial div(y,d)}{\partial y}$

Question 7

1 / 1 pts

The KL divergence between the output of a multi-class network with softmax output $y = [y_1 \dots y_K]$ and *desired* output $d = [d_1 \dots d_K]$ is defined as $KL = \sum_i d_i \log d_i - \sum_i d_i \log y_i$. The first term on the right hand side is the entropy of d , and the second term is the *Cross-entropy* between d and y , which we will represent as $Xent(y, d)$. Minimizing the KL divergence is strictly equivalent to minimizing the cross entropy, since $\sum_i d_i \log d_i$ is not a parameter of network parameters. When we do this, we refer to $Xent(y, d)$ as the cross-entropy loss.

Defined in this manner, which of the following is true of the cross-entropy loss $Xent(y, d)$? Recall that in this setting both y and d may be viewed as probabilities (i.e. they satisfy the properties of a probability distribution).

- [x] It is always non-negative

- [ ] It's derivative with respect to y goes to zero at the minimum (when y is exactly equal to d)

- [ ] It goes to 0 when y equals d

- [ ] It only depends on the output value of the network for the correct class

> If d is not one hot (e.g. when we use label smoothing), the cross entropy may not be 0 when d = y.
>
> For one-hot d, we saw in class that the KL divergence is equal to the cross entropy. Also, in this case, at d=y, the gradient of the DL divergence (and therefore Xent(y,d)) is not 0.

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

Which of the following update rules explicitly computes second-order derivatives or their approximations? (select all that apply)

Hint: Lecture 6

- [ ] RProp
- [x] Quickprop
- [ ] Gradient descent
- [x] Newton’s Method

Question 10

1 / 1 pts

Let f be a quadratic function such that at $x = 1$, $f(x) = 10$, $f'(x) = -4$, and $f''(x) = 1$. The minimum has a value of x = 5 and a value of $f(x)$ = 2 . (Truncate your answer to 1 digit after the decimal point i.e. enter your answer in the format x.x, e.g. 4.5)

Hint: Lecture 6 "Convergence for quadratic surfaces"

Answer 1:

5

Answer 2:

2

Quiz Score: **9** out of 10