Mid-term Exam

CSCE 5013-002

Start: 2:00 PM Oct 24, 2019

End: 2:00PM Oct 25, 2019

Submit your answer to blackboard

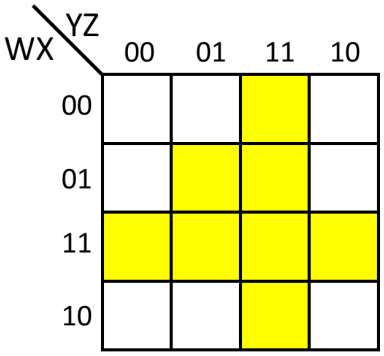
Note: (no late day is applied)

- **Instruction: Write your answer(s) in the answer sheet as the below. Write all your answer(s) if you have multiple choices. Please member to enter your name and ID. The answer will be submitted to blackboard. https://wiki.umbc.edu/pages/viewpage.action?pageId=24477796**
- **If you have N correct answers (among X questions), you will get: $P = \frac{N\ x\ 100}{X}$. If a question has n choices and you got m correct choices, you will get : $P = \frac{100\ x\ m}{X\ x\ n}$ for that question. Each completely correct answer receive $P = \frac{100}{X}$ points.**
- **Requirement: Work individually, no discussing or sharing**
- **You have 1 hour (from 2:00PM – 3:15PM) to ask if you don't understand the questions. All question will be emailed (thile@uark.edu)**

Good luck!

1. Which of the following statements are true about MLPs?
    a. Deeper networks may require far fewer neurons than shallower networks to express the same function
    b. To compose arbitrarily complex decision boundaries MLPs need multiple hidden layers depending on the complexity of the boundaries.
    c. The VC dimension of a MLP is bounded by the square of the number of weights in the network.
    d. A network comprising exactly one layer is a Universal Boolean Machine

2. How many neurons will the smallest (in terms of neurons) network that implements the truth table shown by the following Karnaugh map need?



3. How does the number of weights in a XOR network with 1 hidden layer grow with the number of inputs to the network?
    a. Exponential or faster
    b. Between polynomial and exponential
    c. Polynomial but faster than linear
    d. Linear

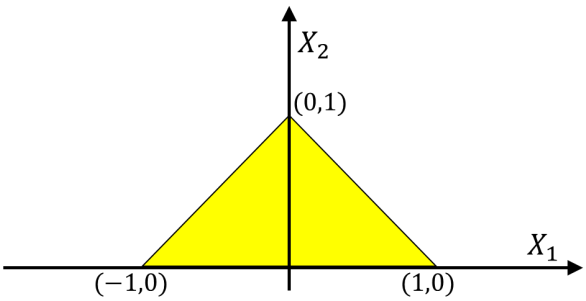
4. How does the number of weights in a XOR network with 2 hidden layers grow with the number of inputs to the network?
    e. Faster than exponential
    f. Between polynomial and exponential
    g. Polynomial but faster than linear
    h. Linear

5. How does the number of weights in a XOR network with arbitrarily many hidden layers grow with the number of inputs to the network?
    i. Faster than exponential
    j. Between polynomial and exponential
    k. Polynomial but faster than linear
    l. Linear

6. Which of the following are impossible in theory? Assume all networks are finite in size, though they can be as large as needed.
    m. Using a threshold network with one hidden layer to perfectly classify all digits in the MNIST dataset.
    n. Using a threshold network with one hidden layer to determine if an arbitrary 2D input lies within the unit circle.
    o. Using a threshold network, as deep as you need, to calculate the L1 distance from a point to the origin.

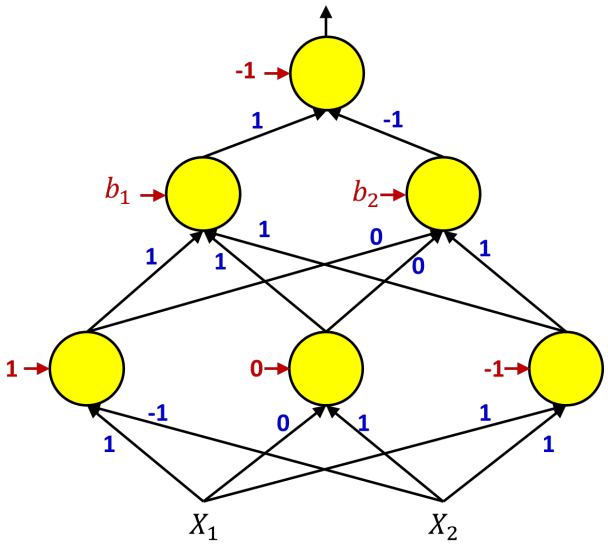
p. Using a threshold network, as deep as you need, to determine if an arbitrary 2D input lies within the square with vertices {(1, 0), (-1, 0), (0, 1), (0, -1)}.

7. We want to build an MLP that composes the decision boundary shown in the figure below. The output of the MLP must 1 in the yellow regions and 0 elsewhere



The MLP we design is suboptimal and has the structure and weights shown in the figure below, where each perceptron computes the function.

$$y = \begin{cases} 1 & if \sum_i weight_i . input_i + bias_i \geq 0 \\ 0 & else \end{cases}$$



The weights of the connections are shown against the corresponding arrows, in black. The *biases* are shown in red for all but two of the perceptrons. What must the biases b_1 and b_2 be for the network to compute our target function perfectly? We require the biases to be *integer* values. Please give the value of first b_1 and b_2 second in the spaces provided below.

8. The *true* objective of learning network parameters, when we want it to represent a specific function, is to (pick all that apply)
   a. Minimize the average (expected) divergence between the output of the network and the actual function being approximated, over the entire domain of the input
   b. Minimize the average error over all training points
   c. Minimize the *expected* empirical risk over a training set
   None of the above

9. In order for our NN to represent a specific function, in practice we will actually try to:

d. Minimize the weights and biases
e. Minimize the empirical error on the training data
f. Maximize the network's parameters
g. Adjust network parameters such that the network's output matches the desired output as closely as possible, on the training instances.

10. The perceptron learning algorithm is always guaranteed to converge in a finite number of steps. Please answer True or False to the answer sheet

11. Networks of perceptrons with threshold activations are hard to train because (select all that apply)
    a. The training data usually only provides labels for the entire network, and not for individual neurons in the network.
    b. The computational complexity of identifying the appropriate labels for each of the training instances for each of the hidden perceptrons may be exponential in the number of training instances.
    c. We cannot generally get any indication of whether increasing any particular parameter will increase or decrease the overall error.
    d. Threshold activations are inadequate to approximate most functions.

12. You are performing gradient descent on the function $f(x) = x^2$ (squared). Currently, x = 5. Your step size is 0.1. What is the value of x after your next step?

13. We find the minimum point of a function *f*(*x*), that is twice differentiable and defined over the reals by:
    a. Computing the derivative and solving for zero
    b. Computing the second derivative and solving for zero
    c. Computing the second derivative *f''*(*x*) and find a positive *x* when *f'*(*x*) = 0
    d. Computing the second derivative *f''*(*x*) and find an *x* where *f''*(*x*) is positive and *f'*(*x*) = 0

14. In order to determine whether a point is a critical point, you should consider:
    a. The Hessian
    b. The function value
    c. The Hamiltonian
    d. The first derivative

15. The Hessian of the following function:
    $f(x_1, x_2, x_3) = x_1^2 x_2 + x_2^2 x_3 + x_3^3 + 2x_1 x_3 + x_2 + 6$ at (1, 1, 1) is
    a. Positive definite
    b. Positive semidefinite
    c. Negative definite
    d. Negative semidefinite
    e. Indefinite

16. Which of the following is true of the gradient of a function, computed at any point
    a. The gradient is a vector that points in the direction of fastest increase of the function at that point.
    b. The gradient is a vector that points in the direction of fastest decrease of the function at that point.
    c. The length of the gradient is indicative of the actual rate of increase or decrease of the function, in the direction of the gradient
    d. The gradient is a vector composed of the partial derivatives of the scalar output of a function with respect to the components of its vector input.
    e. You can always compute the gradient of any function at any location.

17. When we use gradient descent to find the minimum of a function we do so by
    a. Starting at some initial location and iteratively moving this initial location in the direction of the gradient, until a minimum is arrived at.
    b. Starting at some initial location and iteratively moving this initial location opposite the direction of the gradient, until a minimum is arrived at.
    c. Explicitly solving for the location where the gradient is minimum.
    d. Solving for the location of the minimum using the Hessian of the function
    .
18. Gradient descent steps will always result in a decrease in the loss function we are minimizing. Please answer true or false

19. It's possible to use backpropagation with any arbitrary function as an activation function, as long as it is differentiable. Please answer true or false

20. If the data are linearly separable, then a lone perceptron with the sigmoid (a.k.a. logistic) activation trained with gradient descent and an appropriate learning rate with the L2 loss will always learn the weights needed to correctly classify all the data. Please answer true or false

21. Which of the following is true of the Hessian of a scalar function of a multivariate inputs
    a. The Eigen values are all strictly positive at a local minimum
    b. The Eigen values are all strictly negative at a local maximum
    c. The Eigen values are all strictly positive at global minima, but not at local minima
    d. The Eigen values are all non-negative at local minima

22. The gradient of a multivariate scalar function with respect to its inputs at any point
    a. Is the direction of steepest ascent
    b. Is the direction of steepest descent
    c. Is the vector of local derivatives w.r.t. all the inputs
    d. Is parallel to equal-value contours of the function

23. Which of the following is true of neural network training
    a. Its objective is to minimize the expected divergence between the true output of the network and the desired output
    b. It minimizes an empirical estimate of the expected divergence between the true and desired outputs of the network on a training set
    c. If the network architecture has sufficient capacity, minimizing the empirical risk to zero will result in an exact fit to the target function
    d. We employ gradient descent to train the network

24. Which of the following is true of the backpropagation algorithm (as it was explained in class)
    a. It computes the derivative of the divergence (between true and desired outputs of the network) for a single training input
    b. It computes the derivative of the average divergence for a batch of inputs
    c. It cannot be performed without first performing a feed-forward pass of the input(s) through the network
    d. It can be used to compute the derivative of the divergence with respect to the input of the network
    e. Backpropagation is used to compute derivatives that are required for the gradient descent algorithm that trains the network

25. Assume a function you are modelling is a quadratic. At the current location, $x = 1, f(x) = 10, f'(x) = -4, and\ f''(x) = 1$. At what value of x should the minimum occur?. What is the value of y at that minimum?
    Please answer in format:
    x =
    y =

26. Assume a function you are modelling is linear. At the current location, $x = 3, f(x) = 7, f'(x) = 2$. You are performing gradient descent with learning rate = 0.1. What is your x after a single step? What is $f(x)$ after a single step?
    Please answer in format:
    x =
    f(x) =

27. The solution that gradient descent finds is not sensitive to the initialization of the weights in the network. Please answer true or false

28. Which of the following is true of the step sizes in gradient descent algorithm for minimizing a function? All statements refer to the general class of twice differentiable functions, unless specified otherwise.
    a. It must be at least twice the inverse double derivative of the function at the current location for the algorithm to converge quickly
    b. If the step size is more than twice the optimal step size for a quadratic approximation at the current location, it can cause the algorithm to diverge
    c. The inverse double derivative is the optimal step size for a quadratic approximation of the function at the current estimate
    d. If the step size is greater than the inverse double derivative, the algorithm will converge to the optimum in an oscillatory manner
    e. If the step size is less than the inverse double derivative, the algorithm is likely to converge to the optimum monotonically, without oscillation

29. In gradient descent, which of the following is the best strategy for step sizes, in order to maximize the possibility of escaping local minima and finding a global minimum?
    a. Start with large, divergent step sizes (e.g. greater than twice the optimal step size for a quadratic approximation at the initial location) and gradually decrease it over iterations
    b. Start with large, but non-divergent step sizes (e.g. less than twice the optimal step size for a quadratic approximation at the initial location) and gradually decrease it over iterations
    c. Maintain step sizes consistently close to the optimal step size (e.g. close to the inverse second derivative at the current estimate)
    d. Keep the step sizes low throughout to prevent divergence into local minima

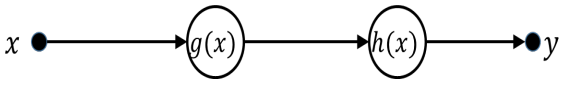
30. Which of the following is true of the cross-entropy loss $Xent(y, d)$, where *y* is the (multi-class) output of a network with softmax output layer and d is the desired target output?
    a. Its derivative with respect to y goes to zero at the minimum (when *y* is exactly equal to *d*)
    b. It is always non-negative
    c. It only depends on the output value of the network for the correct class
    d. (a), (b) and (c) are all true
    e. None of the above

31. We are given the following relationship

$$y = f(x, g(x), h(x, g(x)))$$

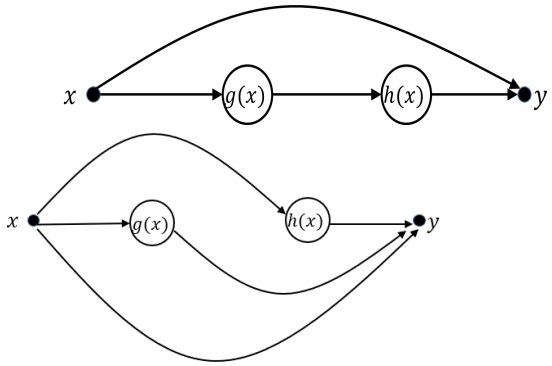
Which of the following figures is the influence diagram for *y* as a function of *x*.

a.

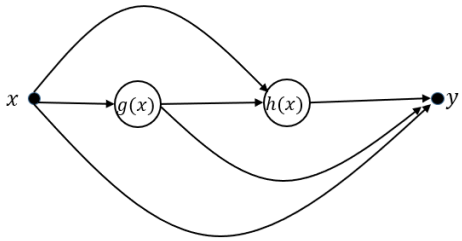


b.

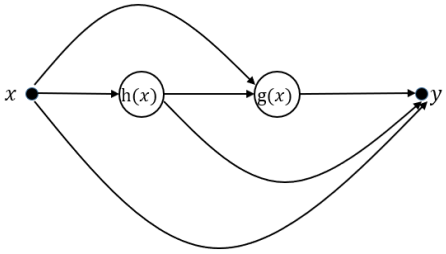
c.



d.



e.



32. You are trying to minimize the cross-entropy loss of the logistic function $y = 1/(1 + \exp(0.5w))$ with respect to parameter w, when the target output is 1.0. Note that this corresponds to optimizing the logistic function $y = 1/(1 + \exp(-wx))$, given only the training input $(x, d(x)) = (-0.5, 1)$. You are using momentum updates. Your current estimate (in the k-th step) is $w^{(k)} = 0$. The last step you took was $\Delta w^{(k)} = 0.5$. Using the notation from class, you use $\beta = 0.9$ and $\eta = 0.1$. What is the value of $w^{(k+1)}$ when using momentum update? Truncate the answer to three decimals (do not round up).

33. Why is momentum learning rule an improvement over gradient descent?
    a. Momentum learning forces the network to find global minima as opposed to local minima.
    b. Momentum learning allows us to forget about the learning rate since the learning rule will automatically adjust the step size for us.

c. Momentum learning smoothens noisy gradients, reduces oscillations, and encourages updates in directions with smooth convergence behaviors.
d. The magnitude of the changes to the parameters can increase without bound if the gradients don't change very much across iterations.

34. Compared to batch gradient descent, SGD is often faster because :
    a. it needs more iterations to converge, but we get many more updates in a single pass through the training data
    b. its iterations take longer to compute, but it converges in much fewer iterations
    c. it is not faster
    d. it needs about the same number of iterations, but they are faster to compute

35. What could happen if incremental gradient descent was not stochastic (i.e. if we selected the training points in a constant order) ?
    a. The loss value would converge very quickly
    b. Overfitting
    c. The behavior would not change
    d. A function that swings around instead of converging

36. Using the optimal learning rate with SGD, we will get an optimal solution:
    a. Always
    b. Only in the strongly convex case
    c. In the convex case
    d. That is never guaranteed

37. Using the optimal learning rate with SGD, we achieve a convergence rate of 1/T
    a. for strongly convex functions, if we use Polynomial decay averaging
    b. or strongly convex functions
    c. for convex functions
    d. for strongly convex functions, if we use loss smoothing

38. Consider a 2D convolutional layer in a neural network, where the kernel size (filter size) is 5x5, the input size is 28x28, the number of input channels (input depth) is 31, the number of output channels (output depth) is 37, the stride is 2 in both the width and height directions, and we are not using any padding. How many parameters must be learned in this layer?

39. You can implement EXACTLY the operations of a convolutional layer using a single fully connected layer. Please answer true or false

40. While backpropagating through a max pooling layer, the divergence derivative at the output of a max pool filter is

    a. equally distributed over the input pool
    b. assigned to the input location of the maximum value, when there is a unique maximum value in the pool
    c. randomly assigned to one of the input locations
    d. assigned such that more weightage is given to the location of the maximum value and the remaining locations get equal parts of the derivative

41. While backpropagating through a max pooling layer, the divergence derivative at the input locations of the non maximum values is

a. 0
b. Identical to the derivative at the location of the maximum value
c. NaN (not a number)
d. given a small, nonzero value

42. While backpropagating through a mean pooling layer, the divergence derivative at the output of a mean pool filter is
    a. equally distributed over the input pool
    b. assigned to the input location of the maximum value
    c. 0
    d. distributed over the inputs in proportion to their values

43. While backpropagating through a mean pooling layer, the divergence derivative at the input location of the non maximum values is
    a. 0
    b. Identical to the derivative at the input location of the maximum value
    c. Proportional to the value of the input at that location
    d. -1

44. As explained in class, the following pseudo code performs the forward computation for one layer (layer l) of a CNN.

```
for j = 1:D_l
    for x = 1:W-K+1
        for y = 1:H-K+1
            z(l,j,x,y) = 0
            for i = 1:D_l-1
                for x' = 1:K_l
                    for y' = 1:K_l
                        z(l,j,x,y) += w(l,j,i,x',y')Y(l-1,i,x+x'-1,y+y'-1)
            Y(l,j,x,y) = activation(z(l,j,x,y))
```

Which of the following is true?

e. K_l is the width of the filter and D_{l-1} is the number of channels in the $l-1$ layer and D_l is the number of channels in the $l^{\{th\}}$ layer.
f. K_l is the height of the input map and D_{l-1} is the number of channels in the $l-1$ layer and D_l is the number of channels in the $l^{\{th\}}$ layer.
g. K_l is the height of the filter and D_{l-1} is the filter width in the $l-1$ layer and D_l is the filter width in the $l^{\{th\}}$ layer.

45. K_l is the total number of layers and D_{l-1} is the number of channels in the $l-1$ layer and D_l is the number of channels in the $l^{\{th\}}$ layer.
As explained in class, the following pseudo code performs the forward computation for one layer (layer l) of a CNN.

```
for j = 1:D_l
    for x = 1:W-K+1
        for y = 1:H-K+1
```

```
z(l,j,x,y) = 0
for i = 1:D_{l-1}
        for x' = 1:K_l
                for y' = 1:K_l
                        z(l,j,x,y) += w(l,j,i,x',y') Y(l-1,i,x+x'-1,y+y'-1)
Y(l,j,x,y) = activation(z(l,j,x,y))
```

Which of the following is true?

h. The variable i goes over the number of the channels in the $l-1$ th layer. The variable x' goes over the width of the filter and variable y' goes over the height of the filter.
i. The variable i goes over the number of the channels in the l th layer. The variable x' goes over the height of the filter and variable y' goes over the width of the filter.
j. The variable i goes over the number of the channels in the $l-1$ th layer. The variable x' goes over the width of the input and variable y' goes over the height of the input.
k. The variable i goes over the instances in a training batch. The variable x' goes over the width of the filter and variable y' goes over the height of the filter.
a.

46. True or False: Having a convolution layer with stride equal to 1 and linear activation followed by a mean pooling layer with the stride of 2 can also be achieved with convolutional layer with larger filter and a stride of 2.

47. True or False: CNNs are naturally invariant to rotational transform

48. True/False: In a CNN that has been explicitly designed for invariance to a set of transforms T. During backpropagation every transformed version of every filter is independently learned.

49. Assume we want to train a CNN to identify the object in a given image and simultaneously generate the bounding box coordinates of the object in the image. Which of the following statements are true for such a model?
    a. The process has to be done serially, first finding the bounding box, and subsequently classifying the object in the box.
    b. The model requires two different loss criterions to learn to perform the task.
    c. Each loss function should be backpropagated independently
    d. The overall loss function should be a linear combination of the multiple loss functions used for the objectives.

50. Assume we want to train a CNN to identify the object in a given image and simultaneously generate the bounding box coordinates of the object in the image. Which of the following statements are true for such a model?
    a. The process has to be done serially, first finding the bounding box, and subsequently classifying the object in the box
    b. The model requires two different loss criterions to learn to perform the task.
    c. Each loss function should be backpropagated independently

d. The overall loss function should be a linear combination of the multiple loss functions used for the objectives.

51. In a 2D CNN the filters of the l^{th} layer are convolved with the outputs of the (l-1)th layer with a stride greater than 1. Select all of the following statements that are true.

    a. The outputs of the $(l-1)^{th}$ layer are never recoverable from the filters and affine combination maps at the l^{th} layer
    b. The outputs of the $(l-1)^{th}$ layer are recoverable from the filters and affine combination maps at the l^{th} layer if there are a sufficiently large number of filters
    c. If all information must be retained even after convolution with stride greater than 1, then a sufficiently large number of filters must be used.
    d. The output of such a convolution will be always the same size as the input

52. Select all true statements
    a. In a CNN the relationship between the output maps of the (l-1)th layer and the affine maps of the l^{th} layer is given by a set of simultaneous equations
    b. The set of affine maps at the l^{th} layer can always be used to recover the output maps of the (l-1)th layer exactly given the filters.
    c. As the stride of the convolution increases, the number of filters must also be increased, if we want the output of the convolution to retain all information about the input (such that the input maps can be recovered from the output affine maps).
    d. In general, the information loss from downsampling can be compensated by increasing the number of filters in subsequent layers.

53. Which of the following is true when the size of the output map in the l^{th} layer of a 2D CNN is greater than the size of the maps in the (l-1)th layer.
    a. This is equivalent to scanning the input with an MLP where the number of neurons in the l^{th} layer is greater than the number of neurons in the (l-1)th layer
    b. Every location in the output map is computed from the same number of location in the input map
    c. This can be equivalently interpreted as one layer of an MLP where the weight matrix is taller than it is broad.
    d. All input maps contribute to each location of the output map.

54. The feature maps in lower layers (layers closer to the input) of a convolutional neural network compared to higher layers (layers closer to the output) of a convolutional neural network tend to:
    a. Have more localized features (like edges), with smaller spatial extents of the image represented
    b. Have more global features (like shapes), with smaller spatial extents of the image represented
    c. Have more global features (like shapes), with larger spatial extents of the image represented
    d. Have more localized features (like edges), with larger spatial extents of the image represented