

Q1-1: Which is NOT one of the game characteristics we considered?

- A. Zero-sum
- B. Fair
- C. Discrete
- D. Deterministic

Q1-1: Which is NOT one of the game characteristics we considered?

A. Zero-sum

B. Fair



C. Discrete

D. Deterministic

Q1-2: Which is true about the kind of games we focus on in our lectures?

- A. Players can make decisions simultaneously
- B. Rolling a die belongs to this kind of games
- C. There is a finite number of states and decisions
- D. Zero-sum ensures fairness

Q1-2: Which is true about the kind of games we focus on in our lectures?

- A. Players can make decisions simultaneously
- B. Rolling a die belongs to this kind of games
- C. There is a finite number of states and decisions
- D. Zero-sum ensures fairness



Q1-3: Which belongs to the kind of games we focus on in our lectures?

- A. Football
- B. Rock-paper-scissors
- C. 2-player checkers
- D. Monopoly

Q1-3: Which belongs to the kind of games we focus on in our lectures?

A. Football

B. Rock-paper-scissors

C. 2-player checkers



D. Monopoly

Q2-1: Which one is true about the game trees for our focused kind of games?

- A. The tree can have infinite different states.
- B. There is no need to expand the tree to terminal nodes.
- C. The game score at the terminal node is the score of the first player.
- D. There can be a node where both players move.

Q2-1: Which one is true about the game trees for our focused kind of games?

- A. The tree can have infinite different states.
- B. There is no need to expand the tree to terminal nodes.
- C. The game score at the terminal node is the score of the first player.
- D. There can be a node where both players move.



Q2-2: Which one is true about the game tree for II-Nim?

- A. Different nodes have different game states
- B. The longest trajectory has 5 moves
- C. Both A and B
- D. None of the above

Q2-2: Which one is true about the game tree for II-Nim?

A. Different nodes
have different
game states

B. The longest
trajectory has 5
moves

C. Both A and B

D. None of the above

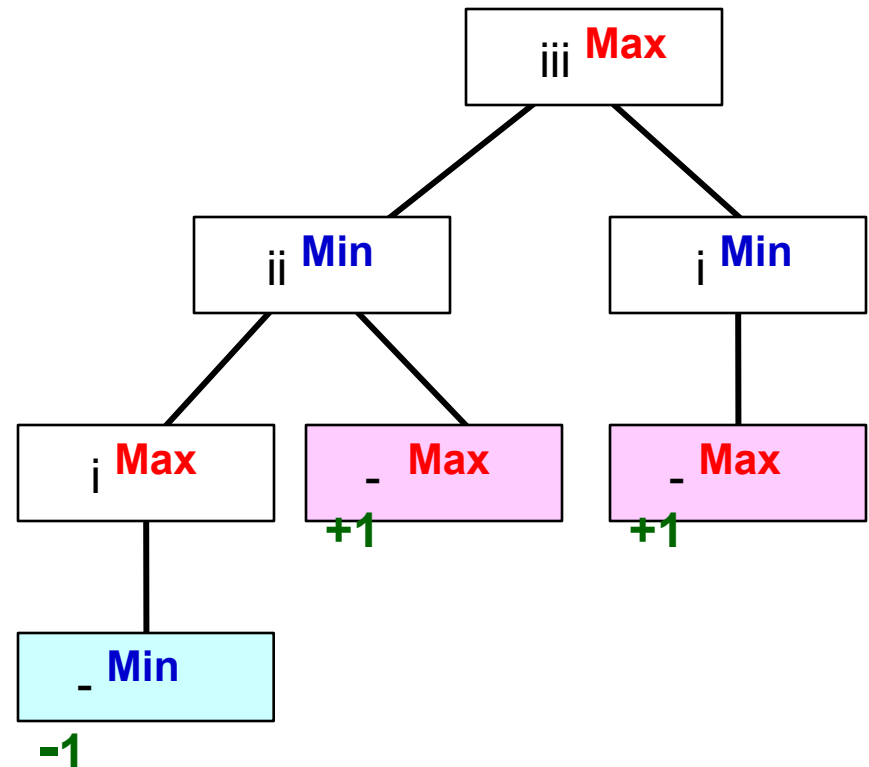


Q2-3: Consider a variant of the Nim game. There is only 1 pile of 3 sticks. And the player takes 1 or 2 sticks from a pile. Which is true about the game tree?

- A. Max always wins along all possible trajectories
- B. The longest trajectory has 3 moves
- C. There are 4 possible trajectories
- D. None of the above

Q2-3: Consider a variant of the Nim game. There is only 1 pile of 3 sticks. And the player takes 1 or 2 sticks from a pile. Which is true about the game tree?

- A. Max always wins along all possible trajectories
- B. The longest trajectory has 3 moves
- C. There are 4 possible trajectories
- D. None of the above



Q3-1: Let b be the max number of legal moves at any point, and m the maximum tree depth. Which is true?

A. Time complexity

$O(bm)$, space

$O(bm)$

B. Time complexity

$O(bm)$, space

$O(b^m)$

C. Time complexity

$O(b^m)$, space

$O(bm)$

D. Time complexity

$O(b^m)$, space

$O(b^m)$

Q3-1: Let b be the max number of legal moves at any point, and m the maximum tree depth. Which is true?

A. Time complexity

$O(bm)$, space

$O(bm)$

B. Time complexity

$O(bm)$, space

$O(b^m)$

C. Time complexity

$O(b^m)$, space

$O(bm)$



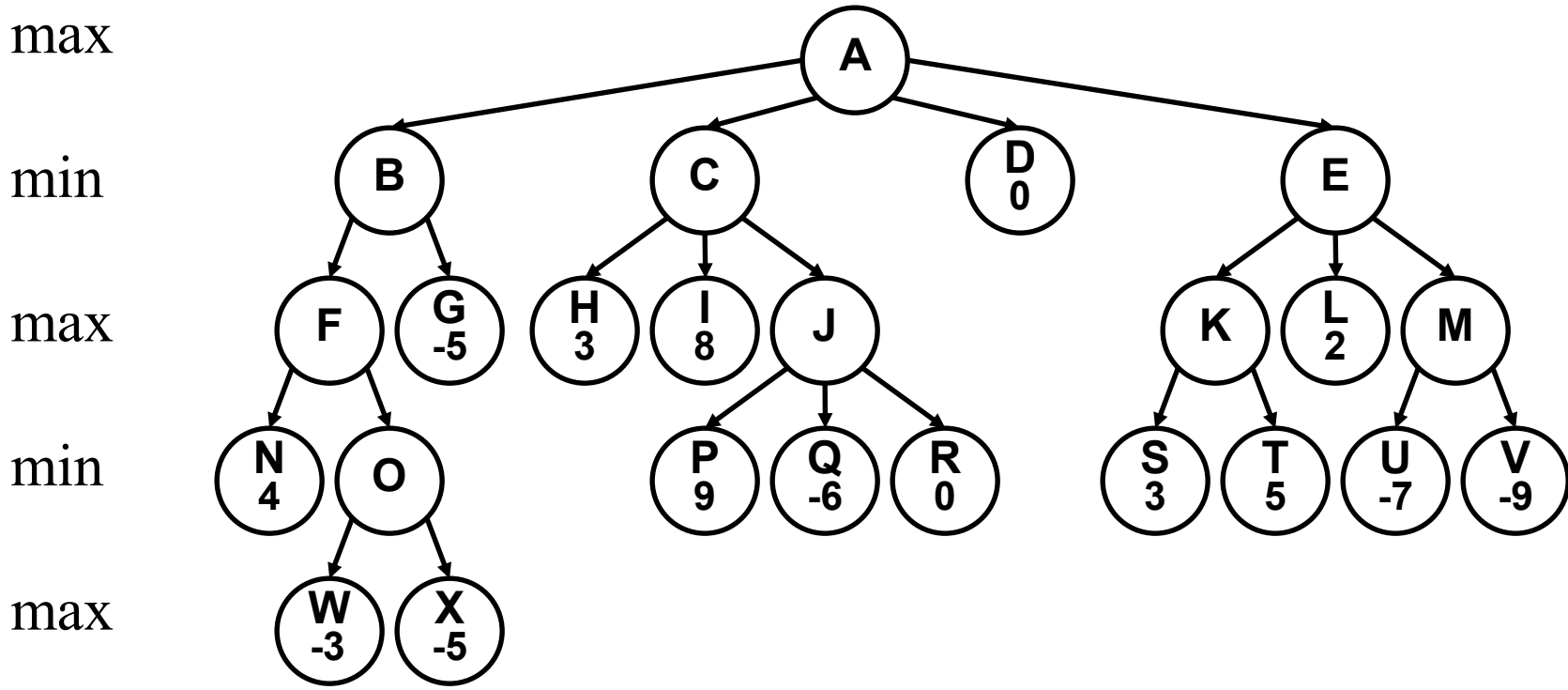
D. Time complexity

$O(b^m)$, space

$O(b^m)$

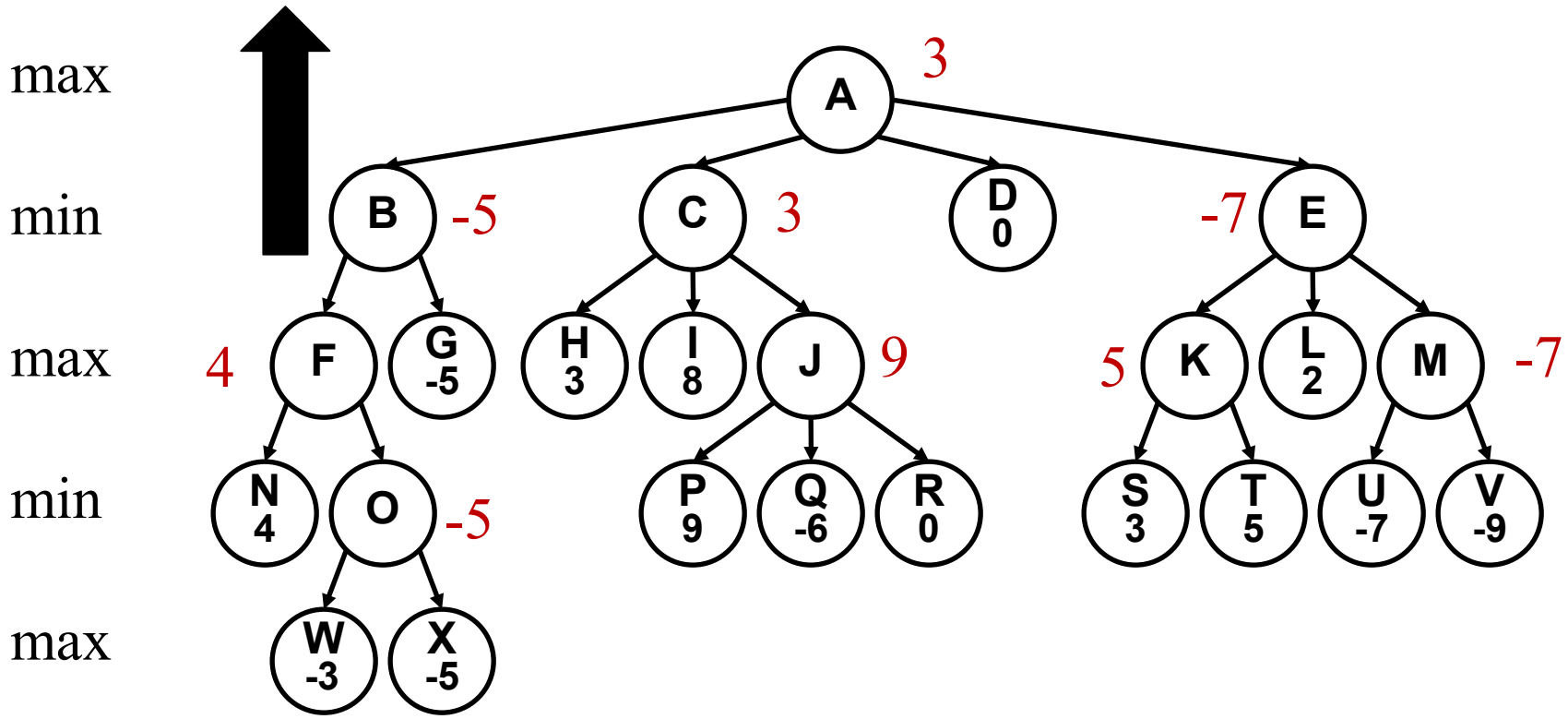
Q3-2: What's the game theoretic value of node A?

- A. 4 B. 3 C. -7 D. 0



Q3-2: What's the game theoretic value of node A?

- A. 4 B. 3 C. -7 D. 0

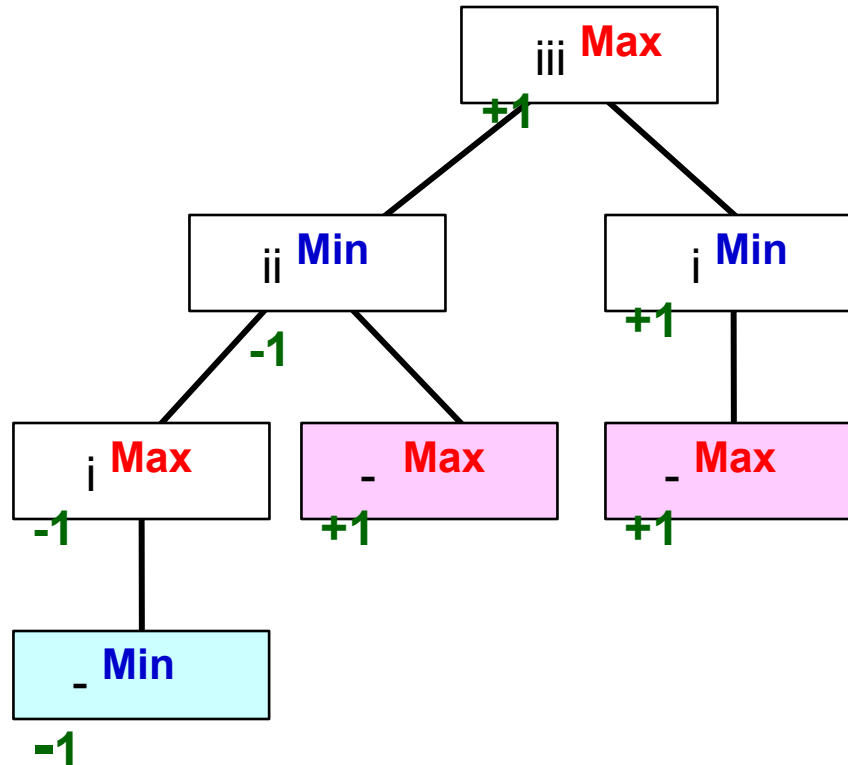


Q3-3: Consider a variant of the Nim game. There is only one pile with 3 sticks. And the player takes 1 or 2 sticks from a pile. What's the game theoretic value of the initial state?

- A. +1
- B. -1
- C. 0
- D. None of the above

Q3-3: Consider a variant of the Nim game. There is only one pile with 3 sticks. And the player takes 1 or 2 sticks from a pile. What's the game theoretic value of the initial state?

- A. +1
- B. -1
- C. 0
- D. None of the above



Q1-1: Which is true about the two approaches to compute the value on the initial node of a game tree?

1. The DFS implementation of minimax search has better time complexity than the bottom up approach
2. The DFS implementation of minimax search has better space complexity than the bottom up approach
3. Both 1 and 2
4. None of the above

Q1-1: Which is true about the two approaches to compute the value on the initial node of a game tree?


1. The DFS implementation of minimax search has better time complexity than the bottom up approach
2. The DFS implementation of minimax search has better space complexity than the bottom up approach
3. Both 1 and 2
4. None of the above



Q1-2: Which is true about the DFS implementation of minimax search? Suppose it evaluates the children from left to right.

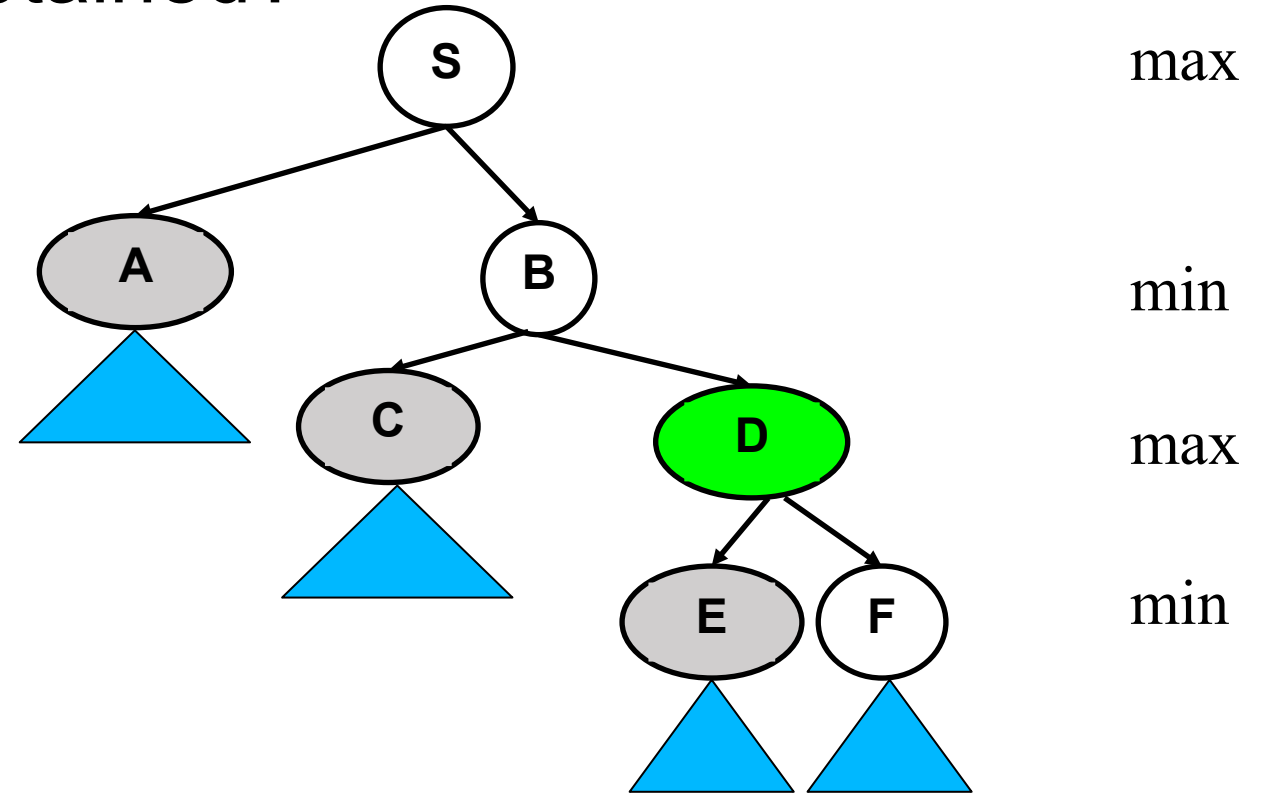
1. It will visit the leaves in the subtree of a left child before visiting a right child
2. It will finish computing the value of a left child before visiting a right child
3. Both 1 and 2
4. None of the above

Q1-2: Which is true about the DFS implementation of minimax search? Suppose it evaluates the children from left to right.

1. It will visit the leaves in the subtree of a left child before visiting a right child
2. It will finish computing the value of a left child before visiting a right child
3. Both 1 and 2 
4. None of the above

Q1-3: Suppose the minimax search evaluates the children from left to right. It has computed the value of E and returned to D but hasn't visited F. Up to now, the best value Max can make sure is X (no matter what subtree of F looks like, Max has a way to get a score $\geq X$). Where can X be obtained?

1. X can be the value of A or E
2. X can be the value of C
3. X can be the value of B or D
4. Both 1 and 2

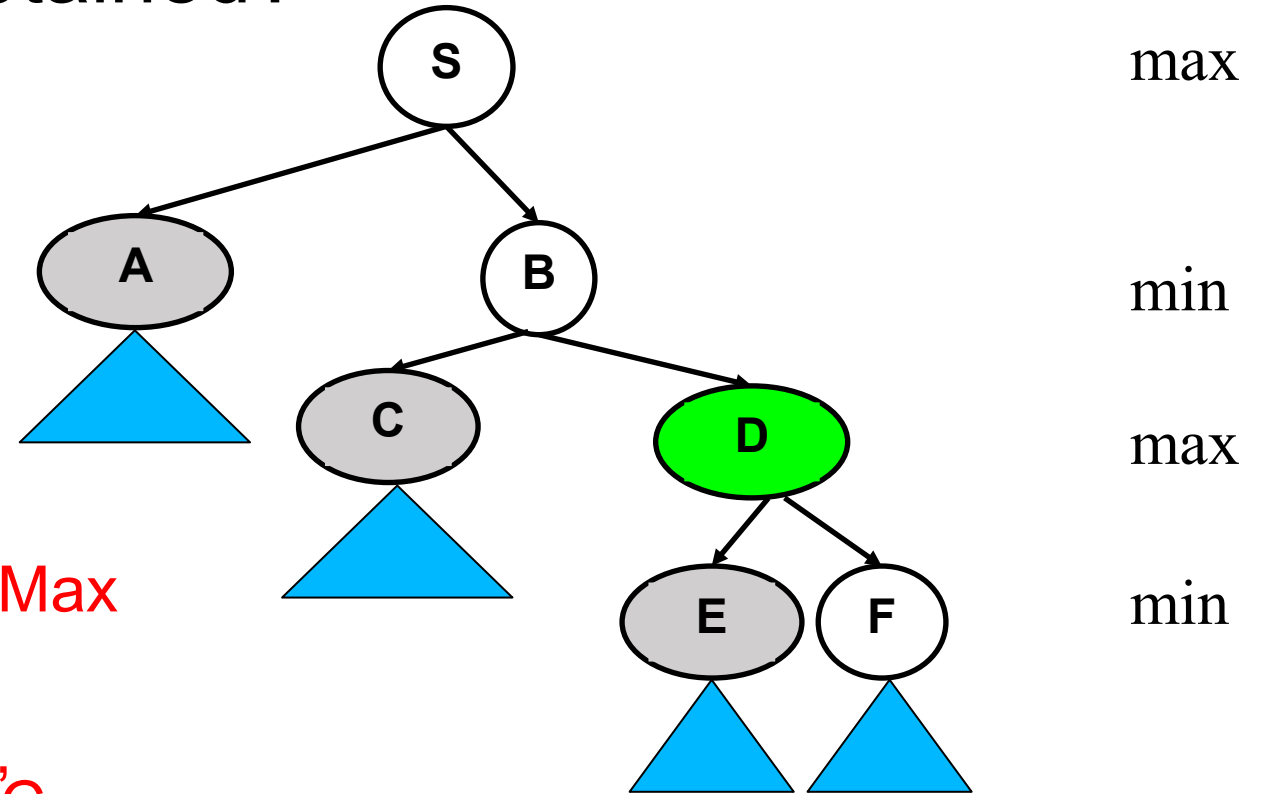


Q1-3: Suppose the minimax search evaluates the children from left to right. It has computed the value of E and returned to D but hasn't visited F. Up to now, the best value Max can make sure is X (no matter what subtree of F looks like, Max has a way to get a score $\geq X$). Where can X be obtained?


1. X can be the value of A or E
2. X can be the value of C
3. X can be the value of B or D
4. Both 1 and 2



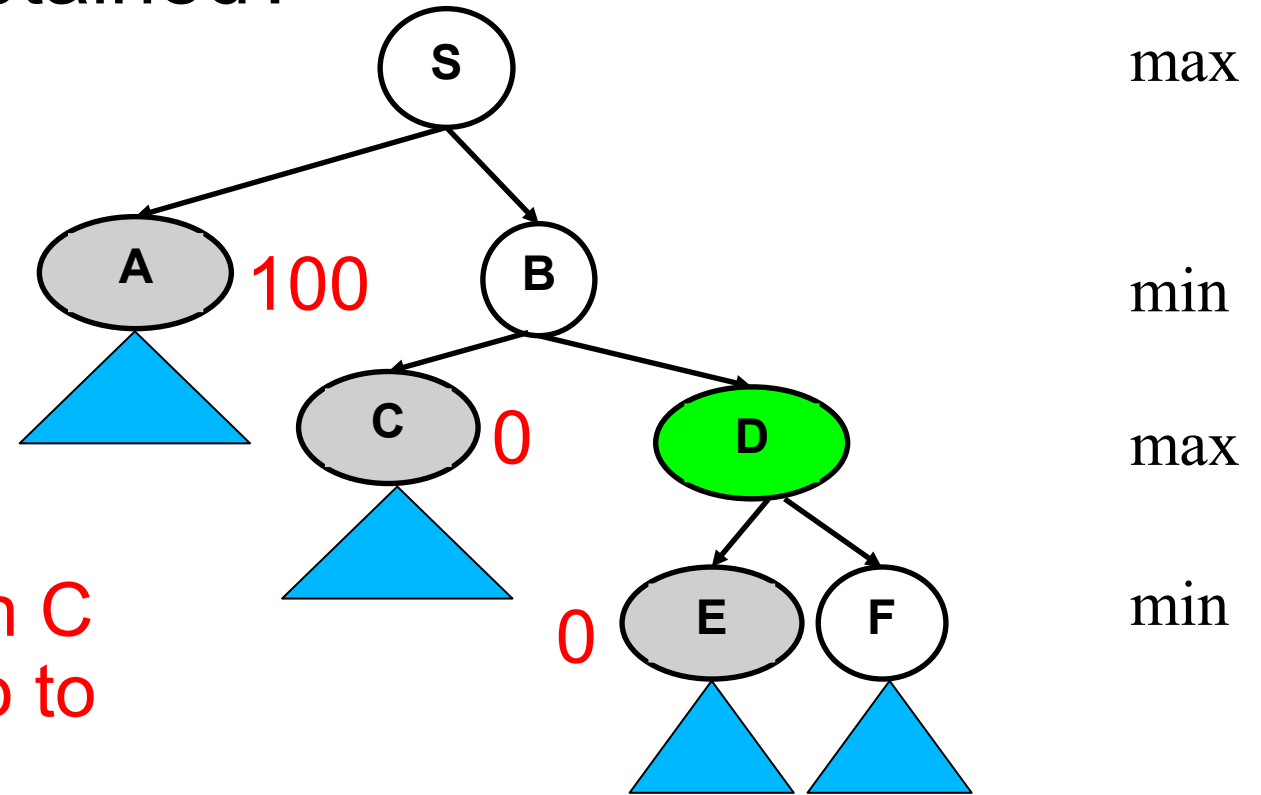
If A's value is larger than C and E, then Max can choose to go to A. If the values are $C > E > A$, Max can choose to go to B and guarantees at least E's value. If $E > C > A$, then Max will go to B and min will go to C, so X is obtained on C. The value of B or D has not been computed yet.




Q1-3: Suppose the minimax search evaluates the children from left to right. It has computed the value of E and returned to D but hasn't visited F. Up to now, the best value Max can make sure is X (no matter what subtree of F looks like, Max has a way to get a score $\geq X$). Where can X be obtained?

1. X can be the value of A or E
2. X can be the value of C
3. X can be the value of B or D
4. Both 1 and 2 

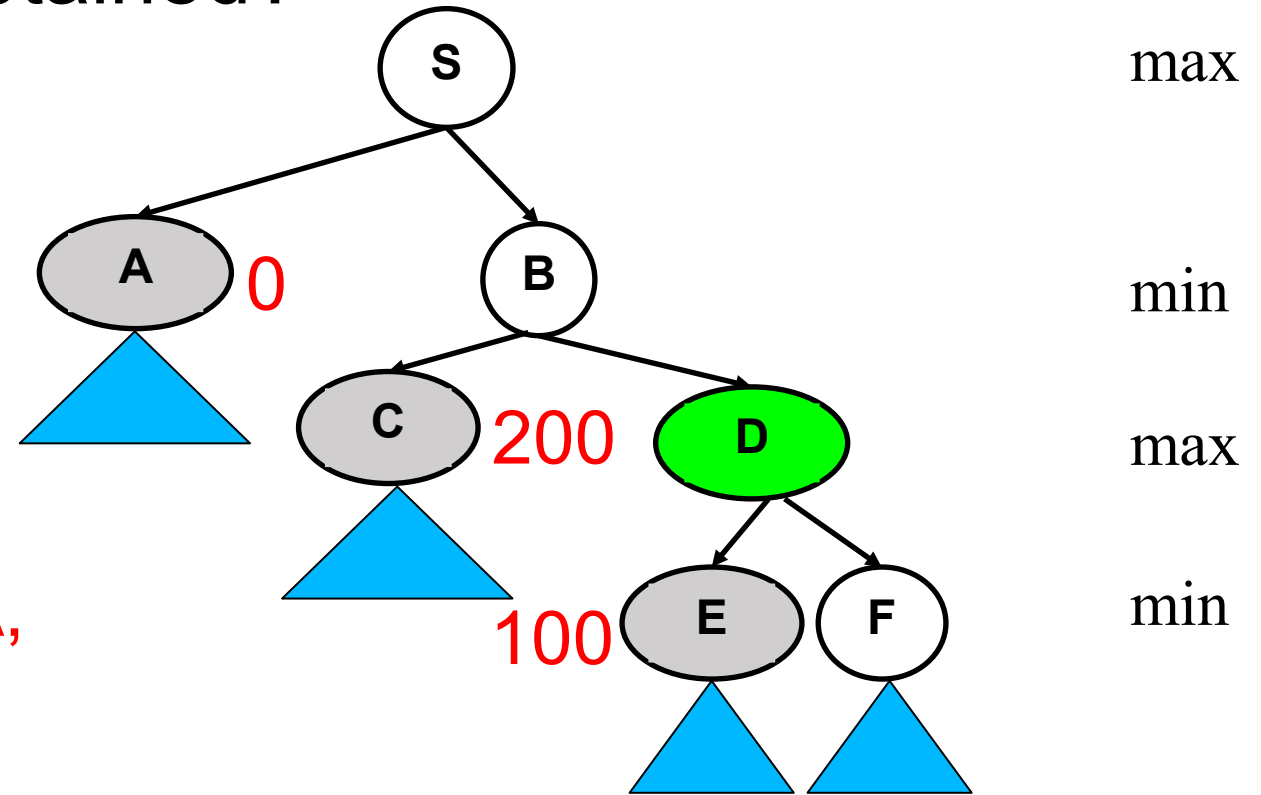
Example: If A's value is larger than C and E, then Max can choose to go to A.




Q1-3: Suppose the minimax search evaluates the children from left to right. It has computed the value of E and returned to D but hasn't visited F. Up to now, the best value Max can make sure is X (no matter what subtree of F looks like, Max has a way to get a score $\geq X$). Where can X be obtained?

1. X can be the value of A or E
2. X can be the value of C
3. X can be the value of B or D
4. Both 1 and 2 

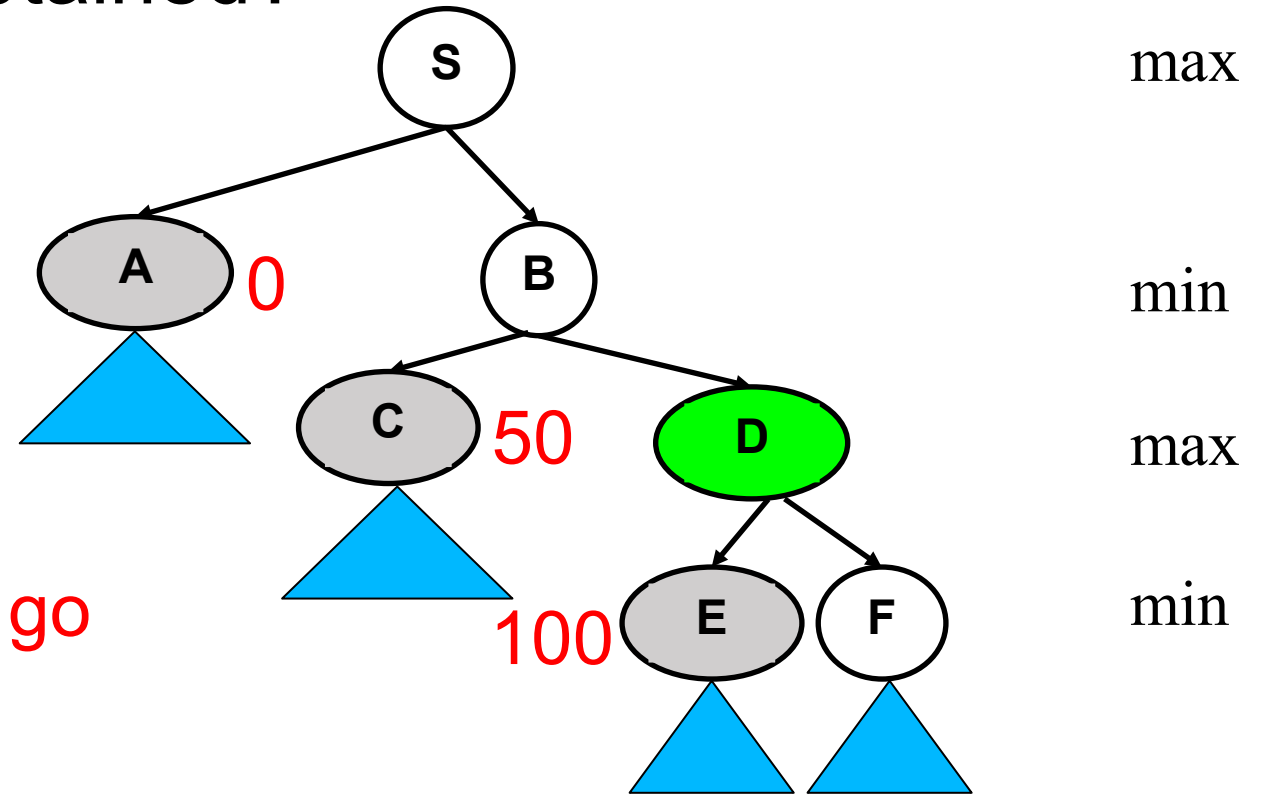
Example: If the values are $C > E > A$, Max can choose to go to B and guarantees at least E's value.



Q1-3: Suppose the minimax search evaluates the children from left to right. It has computed the value of E and returned to D but hasn't visited F. Up to now, the best value Max can make sure is X (no matter what subtree of F looks like, Max has a way to get a score $\geq X$). Where can X be obtained?

1. X can be the value of A or E
2. X can be the value of C
3. X can be the value of B or D
4. Both 1 and 2 

Example: If $E > C > A$, then Max will go to B and min will go to C, so X is obtained on C.



Q2-1: Under which of the circumstance can the alpha on a max node or the beta value on a min node be determined (i.e., not infinity)?

- A. all leaves under that node must have been evaluated
- B. all subtree under that node must have been evaluated
- C. at least a leave under that node have been evaluated
- D. at least a subtree under that node have been evaluated

Q2-1: Under which of the circumstance can the alpha on a max node or the beta value on a min node be determined (i.e., not infinity)?



- A. all leaves under that node must have been evaluated
- B. all subtree under that node must have been evaluated
- C. at least a leave under that node have been evaluated
- D. at least a subtree under that node have been evaluated



Q2-2: In which of the following situations, we can prune some subtree? (multiple correct answers)

- A. On a max node, its alpha is larger than its parent's beta
- B. On a min node, its beta goes below its parent's alpha
- C. On a max node, its alpha is larger than its parent's alpha
- D. On a min node, its beta goes below its parent's beta

Q2-2: In which of the following situations, we can prune some subtree? (multiple correct answers)

- A. On a max node, its alpha is larger than its parent's beta 
- B. On a min node, its beta goes below its parent's alpha 
- C. On a max node, its alpha is larger than its parent's alpha
- D. On a min node, its beta goes below its parent's beta

Q2-3: When on a node v , which of the following is correct regarding the alpha value on that node?

- A. Alpha is the maximum value over all the leaves we've seen so far
- B. Alpha is the maximum value over all the evaluated children of the nodes from root to v (regardless of max nodes or min nodes)
- C. Alpha is the maximum value over all the evaluated children of the max nodes from root to v
- D. Alpha is the maximum value over all the evaluated children of the min nodes from root to v

Q2-3: When on a node v , which of the following is correct regarding the alpha value on that node?

- A. Alpha is the maximum value over all the leaves we've seen so far
- B. Alpha is the maximum value over all the evaluated children of the nodes from root to v (regardless of max nodes or min nodes)
- C. Alpha is the maximum value over all the evaluated children of the max nodes from root to v
- D. Alpha is the maximum value over all the evaluated children of the min nodes from root to v

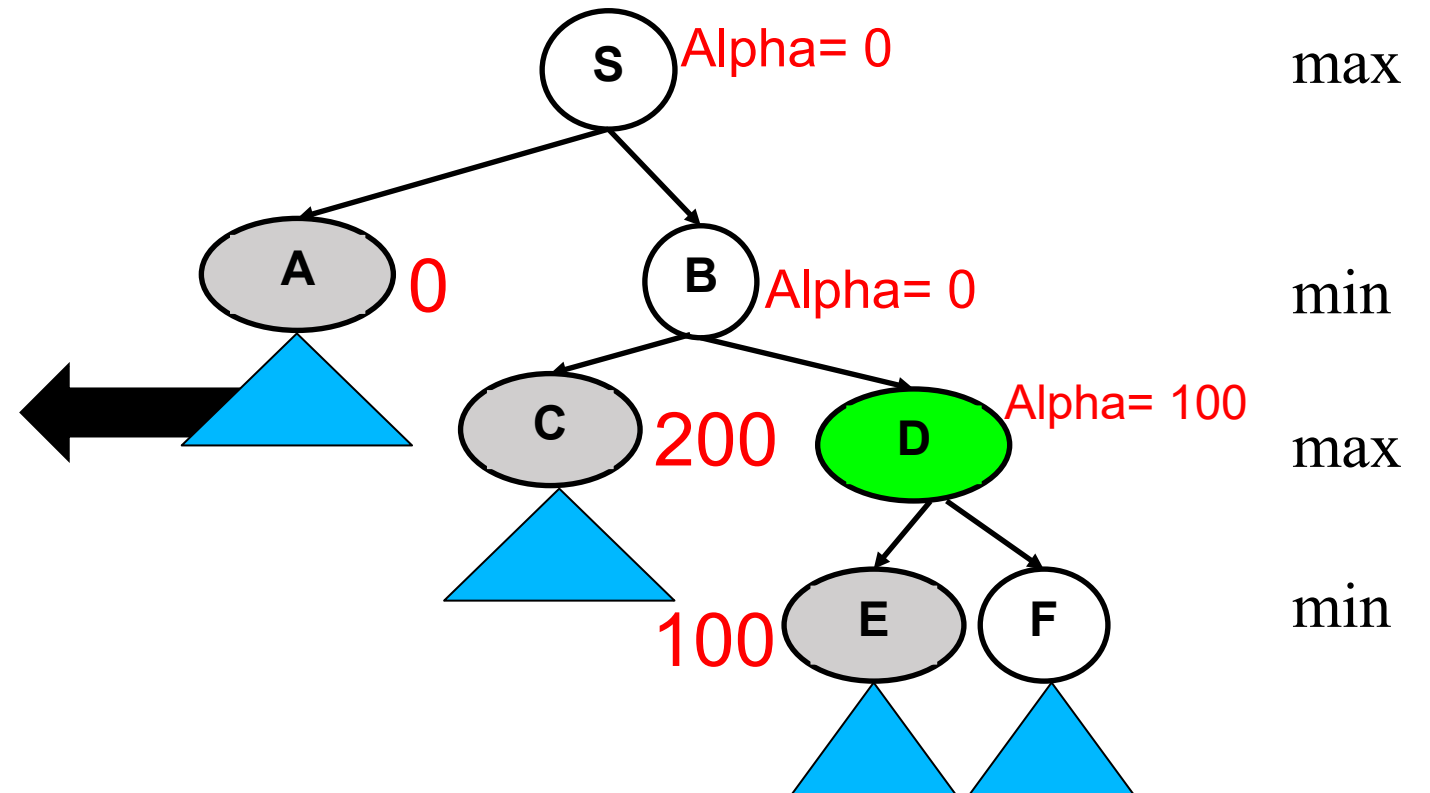
Alpha is inherited from the parent, and only get updated on max nodes using their children's values. The updates can only increase the value.



Q2-3: When on a node v , which of the following is correct regarding the alpha value on that node?

- A. Alpha is the maximum value over all the leaves we've seen so far
- B. Alpha is the maximum value over all the evaluated children of the nodes from root to v (regardless of max nodes or min nodes)
- C. Alpha is the maximum value over all the evaluated children of the max nodes from root to v
- D. Alpha is the maximum value over all the evaluated children of the min nodes from root to v

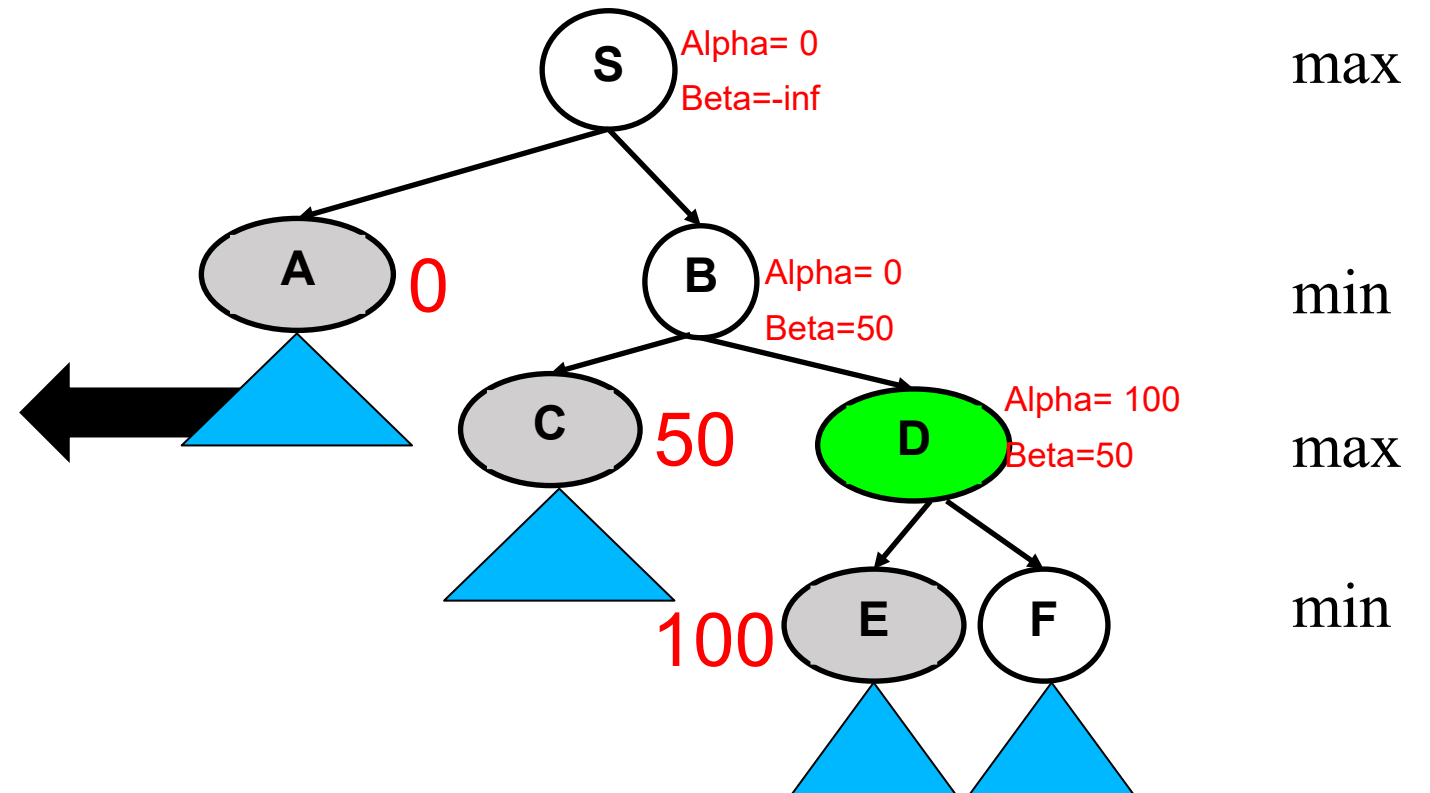
This example shows why B and D are wrong: consider C. It also shows why A is wrong: consider when C is a leaf.



Q2-3: When on a node v , which of the following is correct regarding the alpha value on that node?

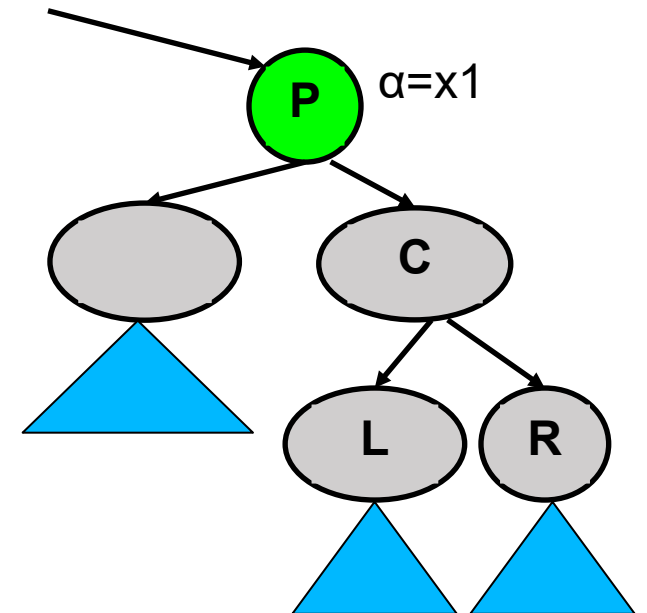
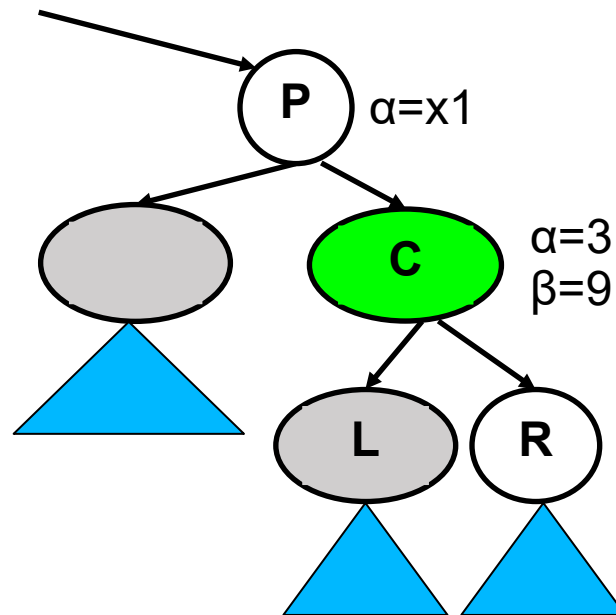
- A. Alpha is the maximum value over all the leaves we've seen so far
- B. Alpha is the maximum value over all the evaluated children of the nodes from root to v (regardless of max nodes or min nodes)
- C. Alpha is the maximum value over all the evaluated children of the max nodes from root to v
- D. Alpha is the maximum value over all the evaluated children of the min nodes from root to v

Consider another example. At this point, alpha is still the max value on the max nodes A and E. But alpha is not the best value Max can make sure, since at this point $\alpha > \beta$ on D so Min won't choose to go to D.




Q3-1: We have $\beta=9$, $\alpha=3$ on the current node C after checking L but not R. Suppose after checking R and returning to the parent node P, the α on P is not updated. Which value of the node R guarantees that this happens?

- A. 2
- B. 4
- C. 6
- D. 8

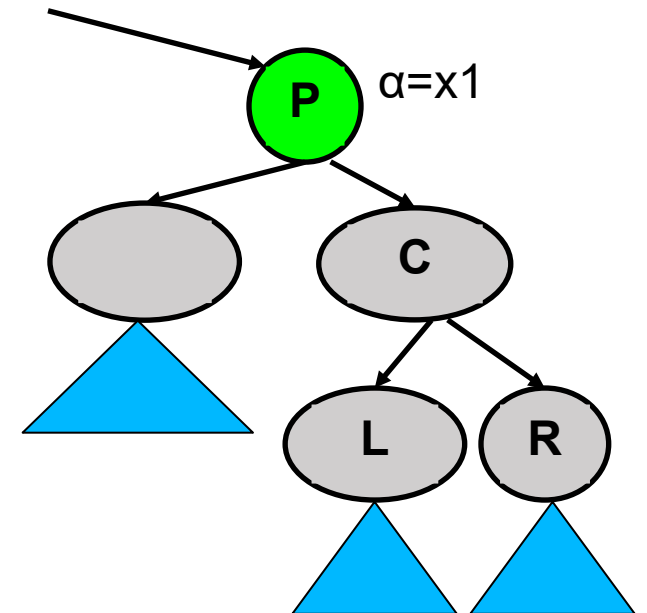
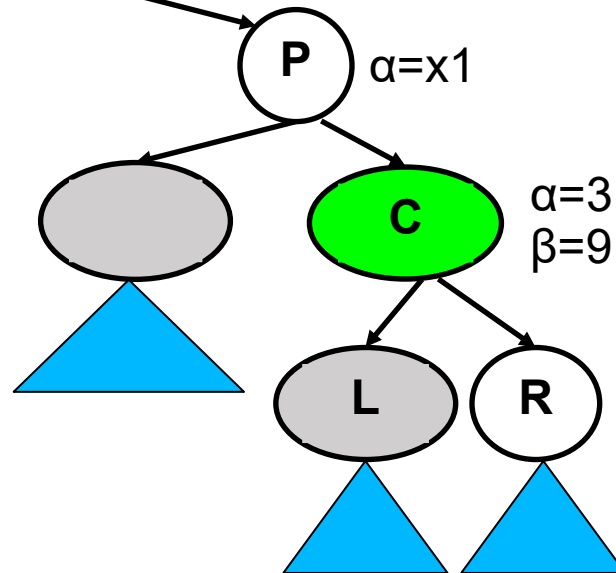


Q3-1: We have $\beta=9$, $\alpha=3$ on the current node C after checking L but not R. Suppose after checking R and returning to the parent node P, the α on P is not updated. Which value of the node R guarantees that this happens?

- A. 2 
- B. 4
- C. 6
- D. 8

Think about the execution of alpha-beta pruning.

1. If the current node is a max node where α is updated. Then P is a min node and only updates its β value.
2. If the current node is a min node where β is updated. Then x_1 must be 3. Also, β on C is updated to 2, and we return 3 to the parent P.




Q3-2: We have enough computation resource to evaluate a tree with depth m without pruning. In the **worst** case, what is the depth of the tree we can evaluate with alpha-beta pruning?

- A. $2m$
- B. m
- C. m^2
- D. $\ln(m)$

Q3-2: We have enough computation resource to evaluate a tree with depth m without pruning. In the **worst** case, what is the depth of the tree we can evaluate with alpha-beta pruning?

A. $2m$

B. m 


C. m^2

D. $\ln(m)$

Q3-3: We have enough computation resource to evaluate a tree with depth m without pruning. In the **best** case, what is the depth of the tree we can evaluate with alpha-beta pruning?

- A. $2m$
- B. m
- C. m^2
- D. $\ln(m)$

Q3-3: We have enough computation resource to evaluate a tree with depth m without pruning. In the **best** case, what is the depth of the tree we can evaluate with alpha-beta pruning?

A. $2m$ 

B. m

C. m^2

D. $\ln(m)$

Q1-1: Consider we are working on an image classification problem. Which of the following could be considered as unlabeled data?

- A. Vehicle images with the type of the vehicle
- B. Fruit images with the height and width
- C. Digit images with the class of the digit (0-9)
- D. Furniture images with the name of the Furniture

Q1-1: Consider we are working on an image classification problem. Which of the following could be considered as unlabeled data?

- A. Vehicle images with the type of the vehicle
- B. Fruit images with the height and width
- C. Digit images with the class of the digit (0-9)
- D. Furniture images with the name of the Furniture




The height and width of the fruit images are the features, not labels.

Q1-2: Which is true about machine learning?

- A. The process doesn't involve human inputs
- B. The machine is given the training and test data for learning
- C. In clustering, the training data also have labels for learning
- D. Supervised learning involves labeled data

Q1-2: Which is true about machine learning?

- A. The process doesn't involve human inputs
- B. The machine is given the training and test data for learning
- C. In clustering, the training data also have labels for learning
- D. Supervised learning involves labeled data 

- A. The labels are human inputs
- B. The machine should not have test data for learning
- C. No labels available for clustering

Q1-3: Which is true about feature vectors?

- A. Feature vectors can have at most 10 dimensions
- B. Feature vectors have only numeric values
- C. The raw image can also be used as the feature vector
- D. Text data don't have feature vectors

Q1-3: Which is true about feature vectors?

- A. Feature vectors can have at most 10 dimensions
- B. Feature vectors have only numeric values
- C. The raw image can also be used as the feature vector
- D. Text data don't have feature vectors



- A. Feature vectors can be in high dimen.
- B. Some feature vectors can have other types of values like strings
- D. Bag-of-words is a type of feature vector for text

Q2-1: Which of the following is not a common task of unsupervised learning?

- A. Clustering
- B. Anomaly detection
- C. Dimensionality reduction
- D. Classification

Q2-1: Which of the following is not a common task of unsupervised learning?


- A. Clustering
- B. Anomaly detection
- C. Dimensionality reduction
- D. Classification



Q2-1: Which is true about the unsupervised learning tasks?

- A. There are only 3 types of unsupervised learning tasks
- B. Anomaly detection doesn't have test data
- C. PCA is a type of dimensionality reduction
- D. Kmeans clustering is a type of hierarchical clustering

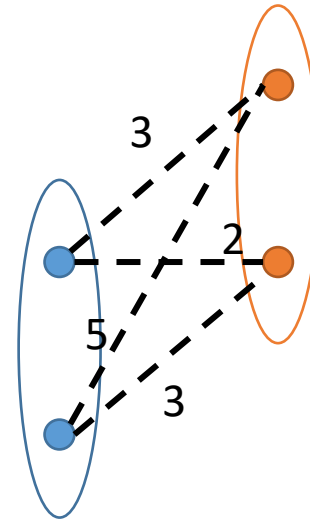
Q2-1: Which is true about the unsupervised learning tasks?

- A. There are only 3 types of unsupervised learning tasks
- B. Anomaly detection doesn't have test data
- C. PCA is a type of dimensionality reduction 
- D. Kmeans clustering is a type of hierarchical clustering

Notice that Anomaly detection also has test data

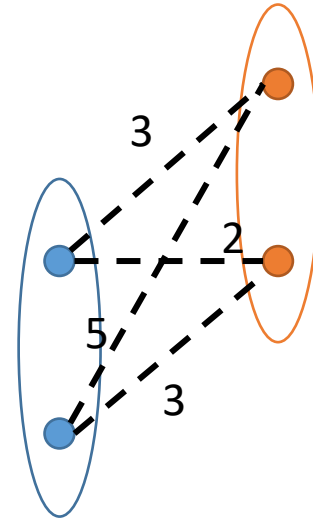
Q3-1: If we use **single linkage** to measure the distance from two clusters, what is the distance of these two clusters in the following example?

- A. 2
- B. 3
- C. 5
- D. 2.5



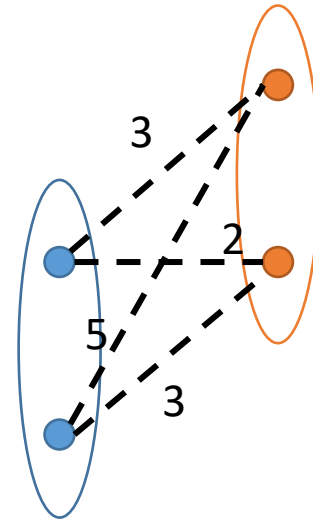
Q3-1: If we use **single linkage** to measure the distance from two clusters, what is the distance of these two clusters in the following example?

- A. 2
- B. 3
- C. 5
- D. 2.5



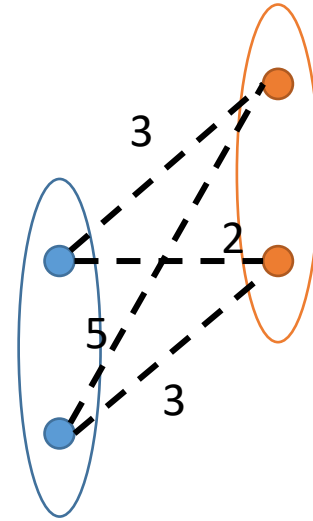
Q3-2: If we use **complete linkage** to measure the distance from two clusters, what is the distance of these two clusters in the following example?

- A. 2
- B. 3
- C. 5
- D. 2.5



Q3-2: If we use **complete linkage** to measure the distance from two clusters, what is the distance of these two clusters in the following example?

- A. 2
- B. 3
- C. 5
- D. 2.5



Q3-3: Consider the dataset in 1-dimension below. Now we have 3 clusters $C1=\{0,2\}$, $C2=\{4,5\}$, $C3=\{7.5,8.5\}$.

(1) Single-linkage will merge C1 and C2.

(2) Complete-linkage will merge C1 and C2.

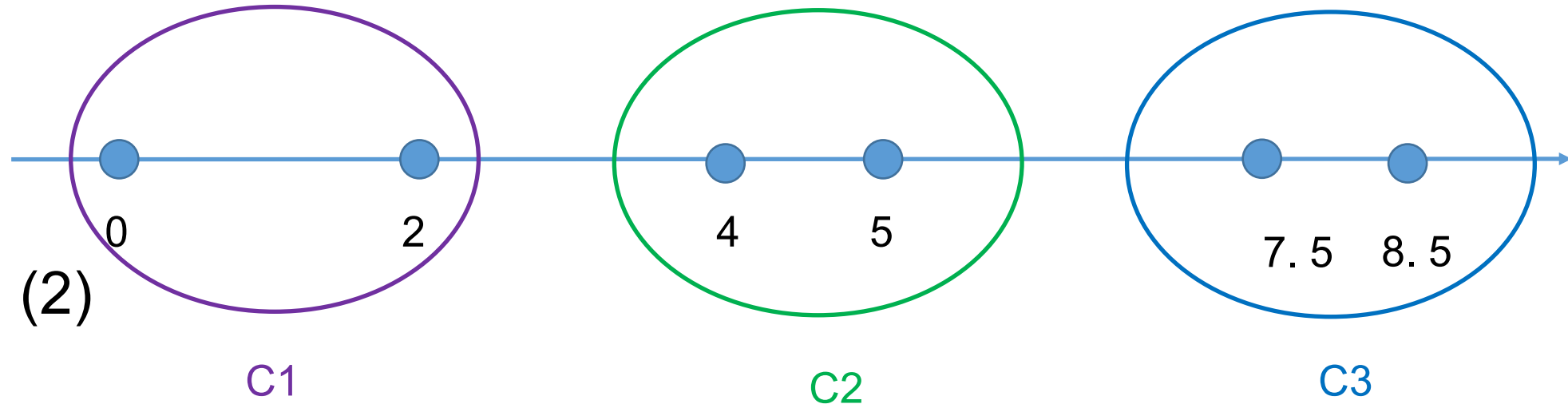
Which statement is true?

A. Only (1)

B. Only (2)

C. None

D. Both (1) and (2)



Q3-3: Consider the dataset in 1-dimension below. Now we have 3 clusters $C1=\{0,2\}$, $C2=\{4,5\}$, $C3=\{7.5,8.5\}$.

(1) Single-linkage will merge $C1$ and $C2$.

(2) Complete-linkage will merge $C1$ and $C2$.

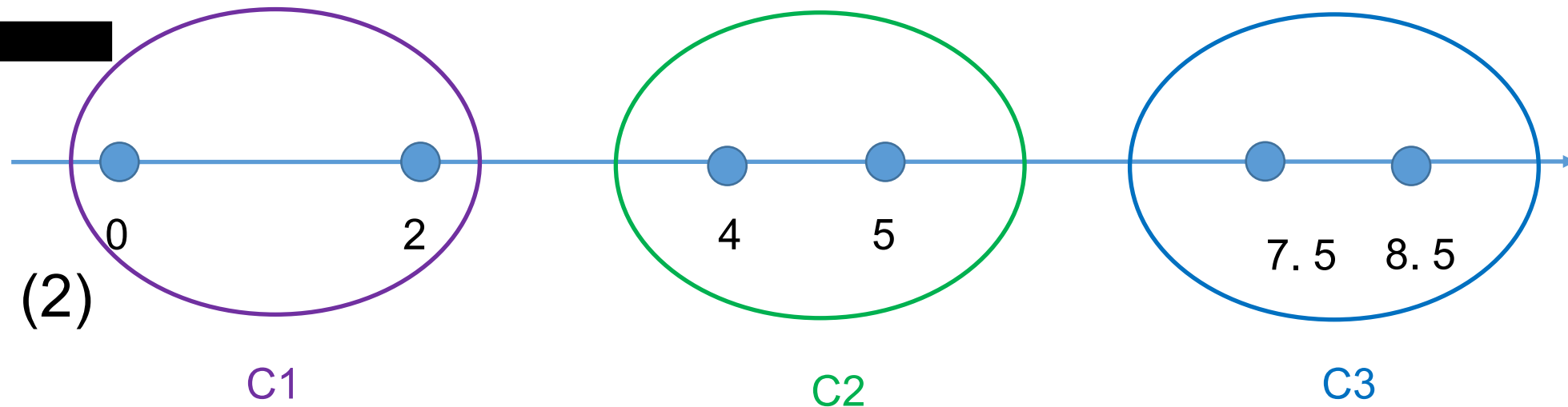
Which statement is true?

A. Only (1) ←

B. Only (2)

C. None

D. Both (1) and (2)



Single linkage: $d(C1, C2) = d(2, 4) = 2$, $d(C2, C3) = d(5, 7.5) = 2.5$

Complete linkage: $d(C1, C2) = d(0, 5) = 5$, $d(C2, C3) = d(4, 8.5) = 4.5$

Q1-1: Given that you are using K-means clustering algorithm to obtain 3 clusters from 7 data points in 2-dim. In the first iteration, clusters C1, C2 and C3 are assigned data points as below.

C1: $\{(2,2),(4,4),(6,6)\}$, C2: $\{(0,4),(4,0)\}$, C3: $\{(5,5),(9,9)\}$


What will be the cluster centroids at the start of second iteration?

1. C1: (4,4), C2: (2,2), C3: (7,7)
2. C1: (6,6), C2: (4,4), C3: (9,9)
3. C1: (2,2), C2: (0,0), C3: (5,5)
4. C1: (2,6), C2: (0,4), C3: (5,9)

Q1-1: Given that you are using K-means clustering algorithm to obtain 3 clusters from 7 data points in 2-dim. In the first iteration, clusters C1, C2 and C3 are assigned data points as below.

C1: $\{(2,2),(4,4),(6,6)\}$, C2: $\{(0,4),(4,0)\}$, C3: $\{(5,5),(9,9)\}$

What will be the cluster centroids at the start of second iteration?

1. C1: (4,4), C2: (2,2), C3: (7,7) 
2. C1: (6,6), C2: (4,4), C3: (9,9)
3. C1: (2,2), C2: (0,0), C3: (5,5)
4. C1: (2,6), C2: (0,4), C3: (5,9)

Q1-2: Consider the K-means algorithm with $K = 3$. After current iteration, we have 3 centers $C1: (0,1)$, $C2: (2,1)$, $C3: (-1,2)$.

Which cluster assignment is possible for the points $A: (1,1)$ and $B: (-1,1)$ respectively? Assume ties are broken arbitrarily.

- (i) $C1, C1$
- (ii) $C2, C3$
- (iii) $C1, C3$

1. Only (i)
2. Only (ii) and (iii)
3. Only (i) and (iii)
4. All of them

Q1-2: Consider the K-means algorithm with $K = 3$. After current iteration, we have 3 centers $C1: (0,1)$, $C2: (2,1)$, $C3: (-1,2)$.

Which cluster assignment is possible for the points $A: (1,1)$ and $B: (-1,1)$ respectively? Assume ties are broken arbitrarily.

- (i) $C1, C1$
- (ii) $C2, C3$
- (iii) $C1, C3$

1. Only (i)
2. Only (ii) and (iii)
3. Only (i) and (iii)
4. All of them



Squared Euclidean distance
between A and centers: 1, 1, 5

For B: 1, 9, 1

So A can be assigned to $C1$ and $C2$, B can be to $C1$ and $C3$

Q1-3: Given the following points in 1D: $x_1 = -1$, $x_2 = 0$, $x_3 = 1$, $x_4 = 8$, $x_5 = 9$, $x_6 = 10$, what are the locations of cluster centers at convergence assuming $K=2$? Assume we start with cluster centers $c_1 = 2$ and $c_2 = 8$.

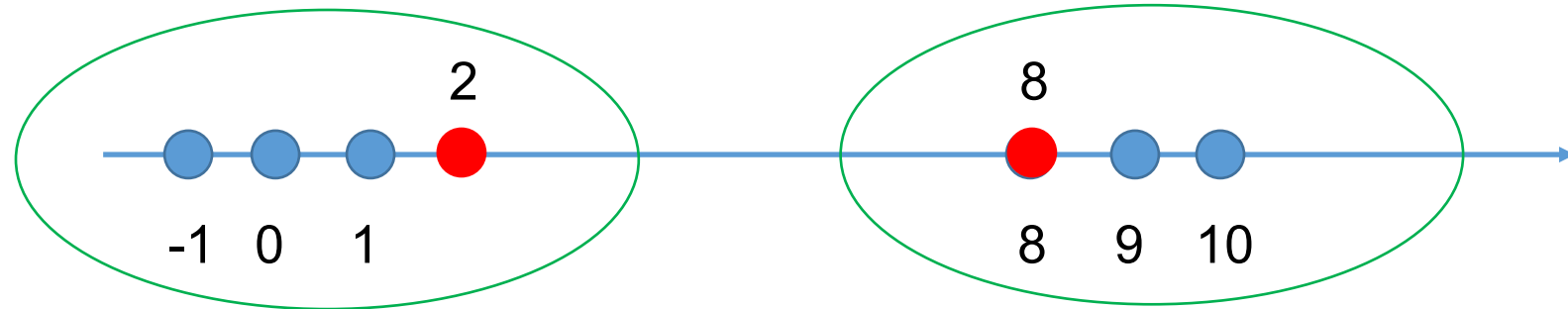
1. $c_1 = 2$, $c_2 = 8$
2. $c_1 = 0$, $c_2 = 9$
3. $c_1 = -1$, $c_2 = 10$
4. $c_1 = 0$, $c_2 = 0$



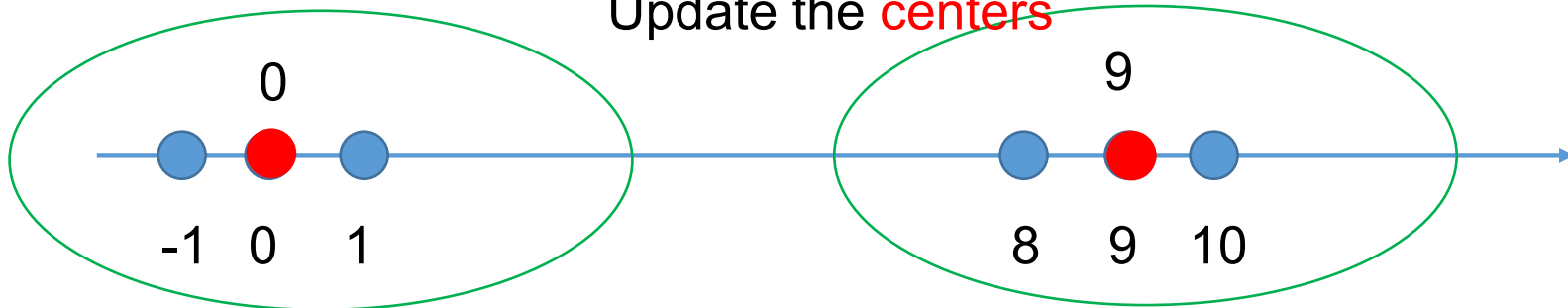
Q1-3: Given the following points in 1D: $x_1 = -1$, $x_2 = 0$, $x_3 = 1$, $x_4 = 8$, $x_5 = 9$, $x_6 = 10$, what are the locations of cluster centers at convergence assuming $K=2$? Assume we start with cluster centers $c_1 = 2$ and $c_2 = 8$.

1. $c_1 = 2$, $c_2 = 8$
2. $c_1 = 0$, $c_2 = 9$ ←
3. $c_1 = -1$, $c_2 = 10$
4. $c_1 = 0$, $c_2 = 0$

Assign the points to centers



Update the centers



Q2-1: Consider the K-means algorithm from the slides. Which step changes cluster centers to minimize distortion?

1. Step 1
2. Step 2

Q2-1: Consider the K-means algorithm from the slides. Which step changes cluster centers to minimize distortion?

1. Step 1


2. Step 2



Q2-2: Consider the K-means algorithm from the slides. Which step assigns each x to its closest cluster center $y(x)$ to minimize the distortion?

1. Step 1
2. Step 2

Q2-2: Consider the K-means algorithm from the slides. Which step assigns each x to its closest cluster center $y(x)$ to minimize the distortion?

1. Step 1 
2. Step 2

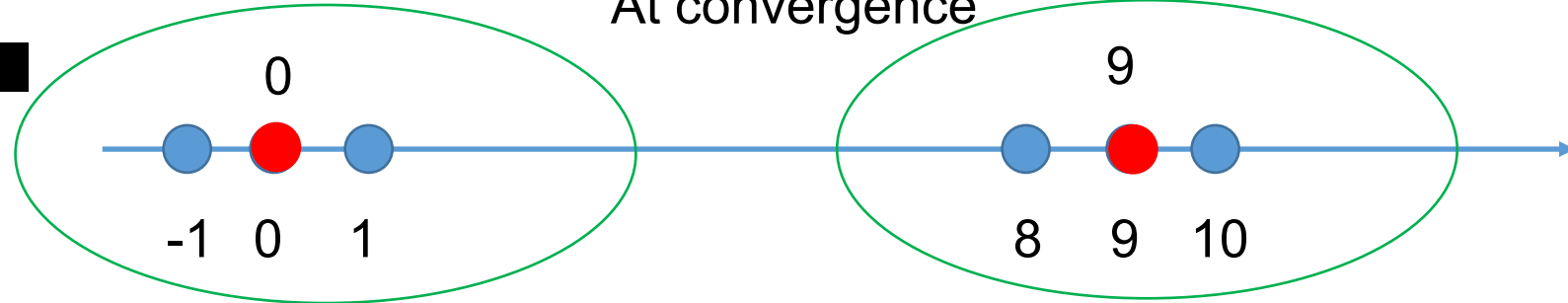
Q2-3: Given the following data points in 1D: $x_1 = -1$, $x_2 = 0$, $x_3 = 1$, $x_4 = 8$, $x_5 = 9$, $x_6 = 10$, what is the distortion of x_6 and the whole dataset respectively **at convergence**? Assume $K=2$ and we start with cluster centers $c_1 = 2$ and $c_2 = 8$.

1. 1, 0
2. 2, 2
3. 1, 4
4. 2, 4



Q2-3: Given the following data points in 1D: $x_1 = -1$, $x_2 = 0$, $x_3 = 1$, $x_4 = 8$, $x_5 = 9$, $x_6 = 10$, what is the distortion of x_6 and the whole dataset respectively **at convergence**? Assume $K=2$ and we start with cluster centers $c_1 = 2$ and $c_2 = 8$.


1. 1, 0
2. 2, 2
3. 1, 4
4. 2, 4



Q2-4: If we choose number of clusters equal to number of data points, i.e. $K = n$, what will be the distortion of the dataset at convergence? Assume the starting cluster centers are same as the data points.

1. 0
2. n
3. 1
4. $n-1$

Q2-4: If we choose number of clusters equal to number of data points, i.e. $K = n$, what will be the distortion of the dataset at convergence? Assume the starting cluster centers are same as the data points.

- 1. 0 
- 2. n
- 3. 1
- 4. $n-1$

Q3-1: If we run K-means clustering twice with random starting cluster centers, are we guaranteed to get same clustering results?

1. Yes
2. No

Q3-1: If we run K-means clustering twice with random starting cluster centers, are we guaranteed to get same clustering results?

1. Yes

2. No



Q3-2: Is it guaranteed that K-means will always terminate? Does K-means always lead to global optimum?

1. Yes, Yes
2. No, Yes
3. Yes, No
4. No, No

Q3-2: Is it guaranteed that K-means will always terminate? Does K-means always lead to global optimum?

1. Yes, Yes
2. No, Yes
3. Yes, No
4. No, No



Q3-3: Which of the following could help for K-means to find a global optimum?

- i) Run K-means only for a fixed number of iterations
- ii) Run K-means multiple times with different starting cluster centers.
- iii) Pick the starting cluster centers intelligently.

1. only (i)
2. (i) and (ii)
3. (i) and (iii)
4. (ii) and (iii)

Q3-3: Which of the following could help for K-means to find a global optimum?

- i) Run K-means only for a fixed number of iterations
- ii) Run K-means multiple times with different starting cluster centers.
- iii) Pick the starting cluster centers intelligently.

1. only (i)

2. (i) and (ii)

3. (i) and (iii)

4. (ii) and (iii)



Q1-1: The parameters to be estimated in the Linear Regression model $y = \beta_0 + \beta_1 x$ are

1. β_0, β_1
2. y
3. β_0, y
4. β_1, y

Q1-1: The parameters to be estimated in the Linear Regression model $y = \beta_0 + \beta_1 x$ are

1. β_0, β_1



2. y


3. β_0, y

4. β_1, y

Q1-2: In the regression model $y = \beta_0 + \beta_1 x$, the change in y for a one unit increase in x is:

1. Will always be the same amount, β_0
2. Will always be the same amount, β_1
3. Will depend on both β_0 and β_1
4. None of above

Q1-2: In the regression model $y = \beta_0 + \beta_1 x$, the change in y for a one unit increase in x is:



1. Will always be the same amount, β_0
2. Will always be the same amount, β_1 
3. Will depend on both β_0 and β_1
4. None of above

If $y = \beta_0 + \beta_1 x$, $x' = x + 1$, then we have $y' = \beta_0 + \beta_1 x' = \beta_0 + \beta_1(x + 1) = \beta_0 + \beta_1 x + \beta_1 = y + \beta_1$.

Q1-3: Suppose that the value of r^2 for an estimated regression model is exactly zero. Which are true? (Multiple answers)

1. The slope coefficient estimate will be zero
2. The fitted line will be horizontal
3. The fitted line will be vertical
4. The intercept coefficient estimate will be zero

Q1-3: Suppose that the value of r^2 for an estimated regression model is exactly zero. Which are true? (Multiple answers)



- 1. The slope coefficient estimate will be zero 
- 2. The fitted line will be horizontal 
- 3. The fitted line will be vertical
- 4. The intercept coefficient estimate will be zero

If $r^2 = 0$, then the linear function has exactly the same error as that of a constant. So it is just the function $y = \bar{y}$

Q2-1: In general, the Least Squares Regression approach finds the equation: (multiple answers)

1. that includes the best set of predictor variables
2. of the best fitting straight line/hyperplane through a set of points
3. with the lowest r^2 , after comparing all possible models
4. that has the smallest sum of squared errors

Q2-1: In general, the Least Squares Regression approach finds the equation: (multiple answers)

1. that includes the best set of predictor variables
2. of the best fitting straight line/hyperplane through a set of points 
3. with the lowest r^2 , after comparing all possible models
4. that has the smallest sum of squared errors 

For 1: there can be noise in the labels, so may not find the best set of predictor variables.

For 3: actually the objective is to get the lowest sum of squared errors, so is to get the highest r^2

Q2-2: Suppose you train two linear regression models on the same dataset, one with 0 regularization, one use large positive λ for regularization. You get the following two vectors of coefficients.

$$\theta_1 = [55, 66, 77, 88]$$

$$\theta_2 = [5, 6, 7, 8]$$

Which linear model has utilized regularization during training?


1. Model 1
2. Model 2
3. Need more information to tell

Q2-2: Suppose you train two linear regression models on the same dataset, one with 0 regularization, one use large positive λ for regularization. You get the following two vectors of coefficients.

$$\theta_1 = [55, 66, 77, 88]$$

$$\theta_2 = [5, 6, 7, 8]$$

Which linear model has utilized regularization during training?

1. Model 1
2. Model 2 
3. Need more information to tell

Regularization will penalize the norm of the parameter vector, so it will lead to smaller norm solutions.

Q2-3: Consider the regression problem

$$\min_{\beta} ||\mathbf{y} - \mathbf{X}\beta||^2 + \lambda ||\beta||^2$$

Which of the following is appropriate if we want to further penalize the flexibility of the model?

1. Increase λ
2. Decrease λ
3. Set $\lambda = 1$
4. Set $\lambda < 0$

Q2-3: Consider the regression problem

$$\min_{\beta} ||\mathbf{y} - \mathbf{X}\beta||^2 + \lambda ||\beta||^2$$

Which of the following is appropriate if we want to further penalize the flexibility of the model?

1. Increase λ
2. Decrease λ
3. Set $\lambda = 1$
4. Set $\lambda < 0$



Q3-1: Is logistic regression an appropriate substitute for linear regression?

1. Yes
2. No

Q3-1: Is logistic regression an appropriate substitute for linear regression?

- 1. Yes
- 2. No



Logistic regression is for linear classification (though it's called regression for historical reasons).

Q3-2: Given the training data

$(x, y): (0, +), (1, -), (2, +), (3, -)$

Is this true: A logistic regression model can be trained to classify the data points with zero training error?

1. True
2. False

Q3-2: Given the training data

$(x, y): (0, +), (1, -), (2, +), (3, -)$

Is this true: A logistic regression model can be trained to classify the data points with zero training error?

- 1. True
- 2. False



The decision boundary between + and – by a logistic regression model must be a linear hyperplane. It is a threshold in 1-dim space. Then it cannot get zero classification errors on the data since the labels are interweaving.

Q3-3: If a dataset is linearly separable, which of the following training methods is more suitable to train a logistic regression classifier?

1. MLE
2. MAP

Q3-3: If a dataset is linearly separable, which of the following training methods is more suitable to train a logistic regression classifier?

1. MLE

2. MAP



Q1-1: K-NN algorithms can be used for:

1. Only classification
2. Only regression
3. Both

Q1-1: K-NN algorithms can be used for:

1. Only classification
2. Only regression
3. Both



Q1-2: Which of the following distance measure do we use in case of categorical variables in k-NN?

1. Hamming Distance
2. Euclidean Distance
3. Manhattan Distance

Q1-2: Which of the following distance measure do we use in case of categorical variables in k-NN?



1. Hamming Distance
2. Euclidean Distance
3. Manhattan Distance



Q1-3: Consider binary classification in 2D where the intended label of a point $x = (x_1, x_2)$ is positive if $x_1 > x_2$ and negative otherwise. Let the training set be all points of the form $x = [4a, 3b]$ where a, b are integers. Each training item has the correct label that follows the rule above. With a 1NN classifier (Euclidean distance), which ones of the following points are labeled positive? Multiple answers.

1. $[5.52, 2.41]$
2. $[8.47, 5.84]$
3. $[7, 8.17]$
4. $[6.7, 8.88]$

Q1-3: Consider binary classification in 2D where the intended label of a point $x = (x_1, x_2)$ is positive if $x_1 > x_2$ and negative otherwise. Let the training set be all points of the form $x = [4a, 3b]$ where a, b are integers. Each training item has the correct label that follows the rule above. With a 1NN classifier (Euclidean distance), which ones of the following points are labeled positive? Multiple answers.

- 1. $[5.52, 2.41]$ 
- 2. $[8.47, 5.84]$ 
- 3. $[7, 8.17]$
- 4. $[6.7, 8.88]$

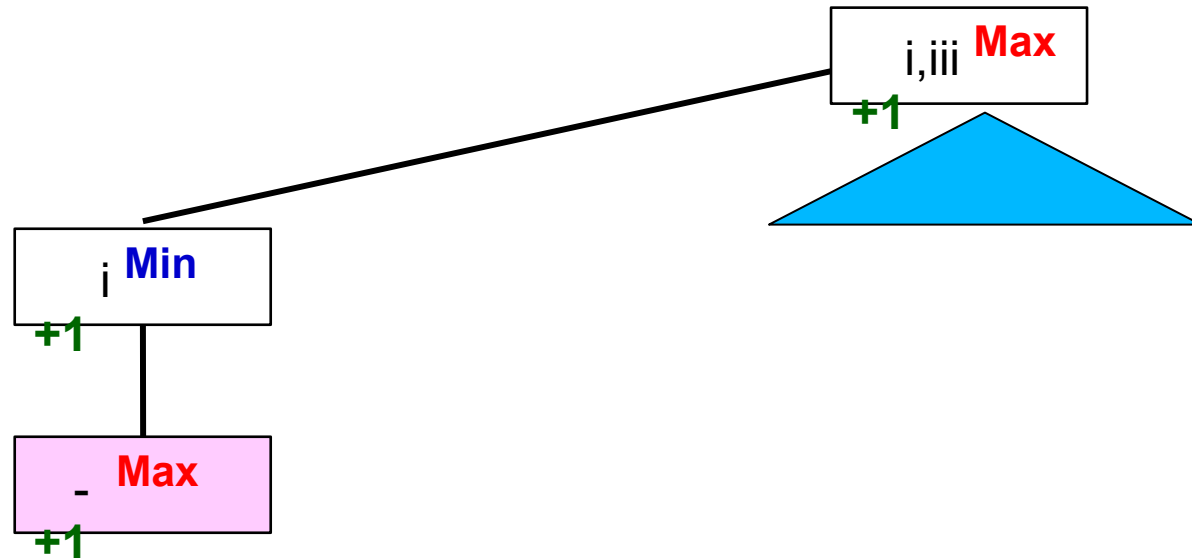
Nearest neighbors are
 $[4, 3] \Rightarrow$ positive
 $[8, 6] \Rightarrow$ positive
 $[8, 9] \Rightarrow$ negative
 $[8, 9] \Rightarrow$ negative
Individually.

Q2-1: Consider a variant of II Nim game where there are 2 piles, with 1 and 3 sticks, respectively. Each time one player takes some stick(s) from only one pile (can take 1 or 2 or 3 sticks). What's the game theoretic value of the initial state?

- A. 1
- B. 0
- C. -1

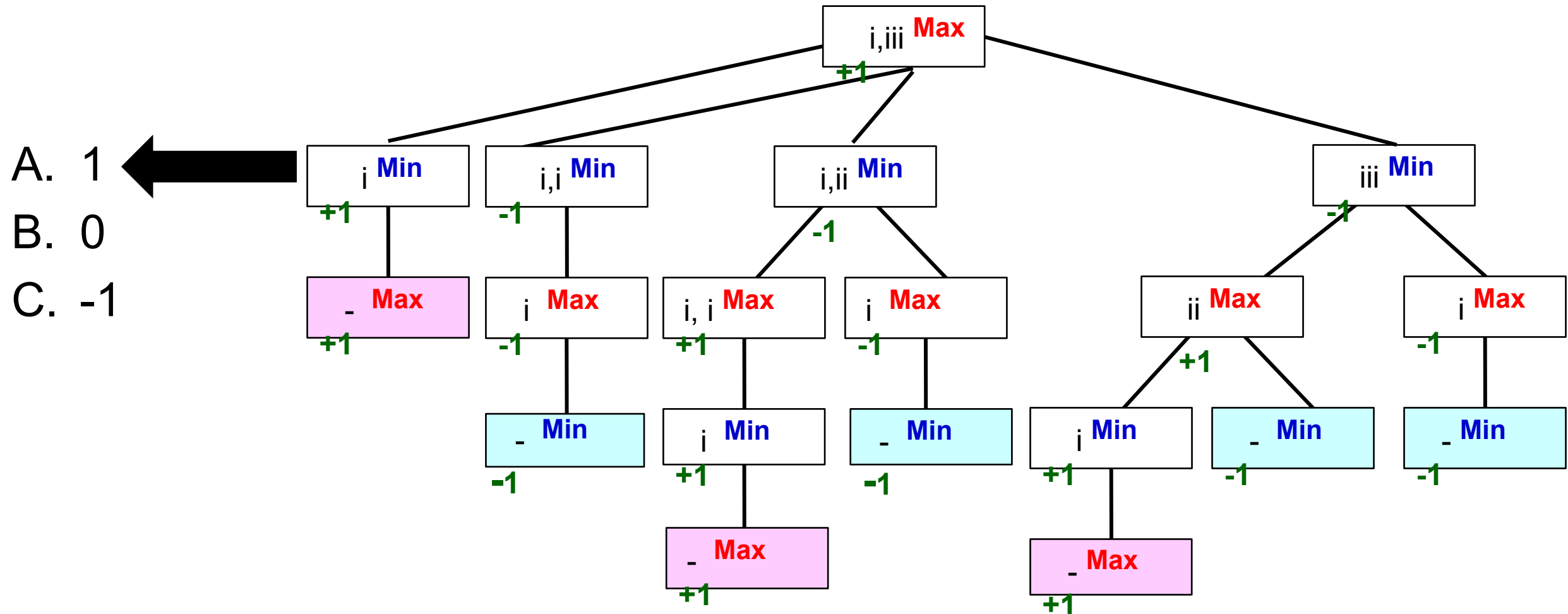
Q2-1: Consider a variant of II Nim game where there are 2 piles, with 1 and 3 sticks, respectively. Each time one player takes some stick(s) from only one pile (can take 1 or 2 or 3 sticks). What's the game theoretic value of the initial state?

- A. 1 ←
- B. 0
- C. -1



The first player could always take 3 sticks from the pile with 3 sticks, which guarantees the game value +1. No need to check the other branches.

Q2-1: Consider a variant of II Nim game where there are 2 piles, with 1 and 3 sticks, respectively. Each time one player takes some stick(s) from only one pile (can take 1 or 2 or 3 sticks). What's the game theoretic value of the initial state?

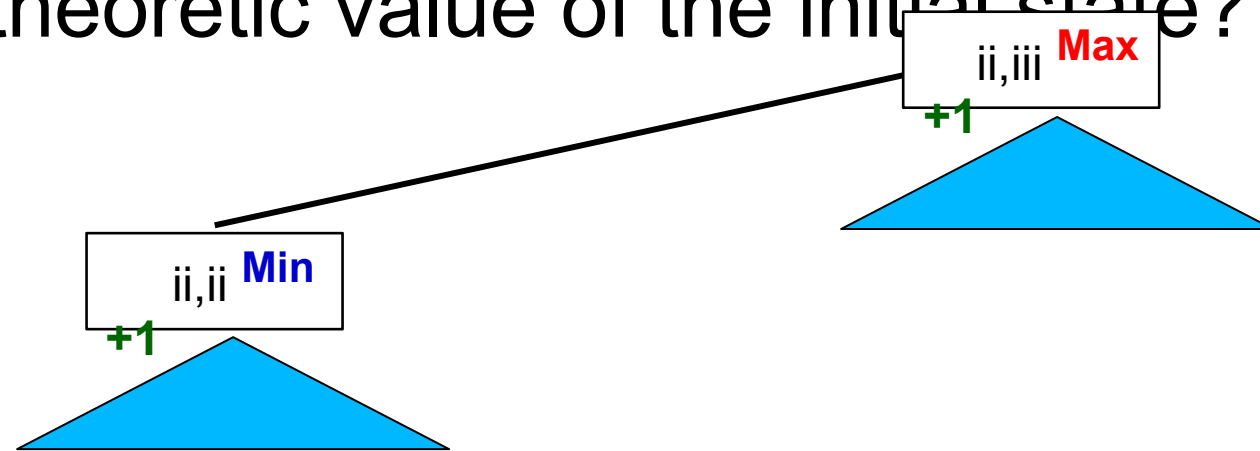


Q2-2: We know that the game theoretic value of the initial state is -1 in II Nim game where there are 2 piles, each with 2 sticks. Now consider a variant of II Nim game where there are 2 piles, with 2 and 3 sticks, respectively. Each time one player takes some stick(s) from only one pile (can take 1 or 2 or 3 sticks). What's the game theoretic value of the initial state?

- A. 1
- B. 0
- C. -1

Q2-2: We know that the game theoretic value of the initial state is -1 in II Nim game where there are 2 piles, each with 2 sticks. Now consider a variant of II Nim game where there are 2 piles, with 2 and 3 sticks, respectively. Each time one player takes some stick(s) from only one pile (can take 1 or 2 or 3 sticks). What's the game theoretic value of the initial state?

- A. 1
- B. 0
- C. -1



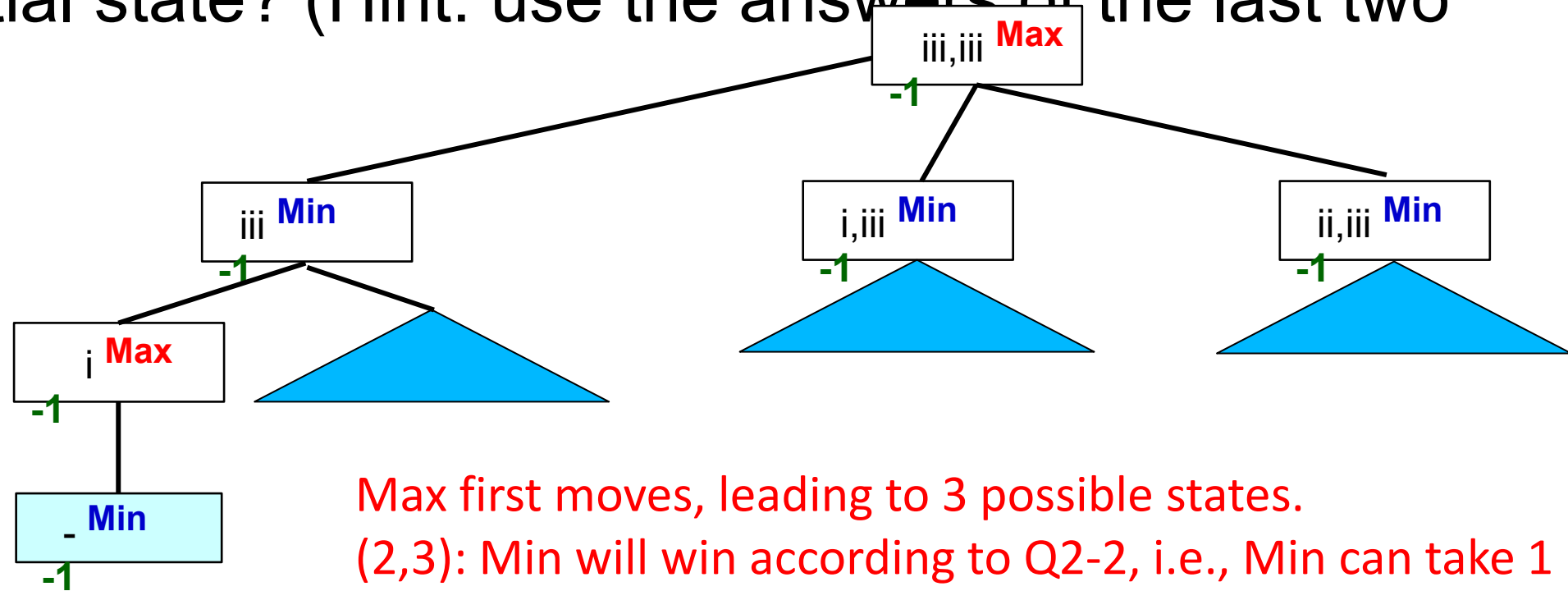
The first player could always take 1 stick from the pile with 3 sticks and make it a (2,2) state, in which case the second player will always lose.

Q2-3: Consider a variant of II Nim game with 2 piles, both with 3 sticks. Each time one player takes some stick(s) from only one pile (can take 1 or 2 or 3 sticks). What's the game theoretic value of the initial state? (Hint: use the answers of the last two questions)

- A. 1
- B. 0
- C. -1

Q2-3: Consider a variant of II Nim game with 2 piles, both with 3 sticks. Each time one player takes some stick(s) from only one pile (can take 1 or 2 or 3 sticks). What's the game theoretic value of the initial state? (Hint: use the answers of the last two questions)

- A. 1
- B. 0
- C. -1



Max first moves, leading to 3 possible states.

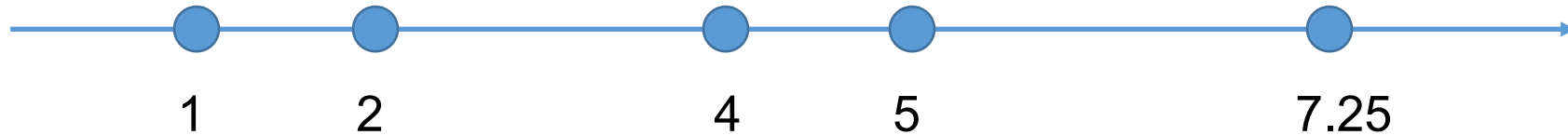
(2,3): Min will win according to Q2-2, i.e., Min can take 1 stick from the pile with 3 sticks and make it (2,2).

(1,3): Min will win according to Q2-1, i.e., Min can take all the second pile, and make it (1,-).

(0,3): Min can take 2 sticks and will win.

Q3-1: Suppose we run average linkage on the following data set to get two clusters. What is the result?

- A. $\{1\}, \{2, 4, 5, 7.25\}$
- B. $\{1, 2\}, \{4, 5, 7.25\}$
- C. $\{1, 2, 4\}, \{5, 7.25\}$
- D. $\{1, 2, 4, 5\}, \{7.25\}$



Q3-1: Suppose we run average linkage on the following data set to get two clusters. What is the result?

A. $\{1\}, \{2, 4, 5, 7.25\}$

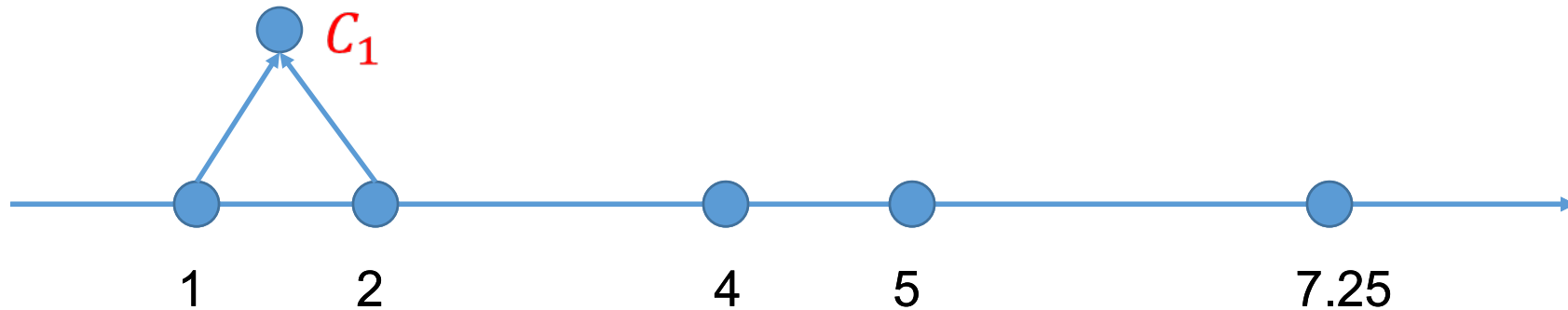
B. $\{1, 2\}, \{4, 5, 7.25\}$

C. $\{1, 2, 4\}, \{5, 7.25\}$

D. $\{1, 2, 4, 5\}, \{7.25\}$



$$d(C_1, \{4\}) = \frac{3 + 2}{2} = 2.5,$$
$$d(\{4\}, \{5\}) = 1$$



Q3-1: Suppose we run average linkage on the following data set to get two clusters. What is the result?

A. $\{1\}, \{2,4,5,7.25\}$

B. $\{1,2\}, \{4, 5, 7.25\}$

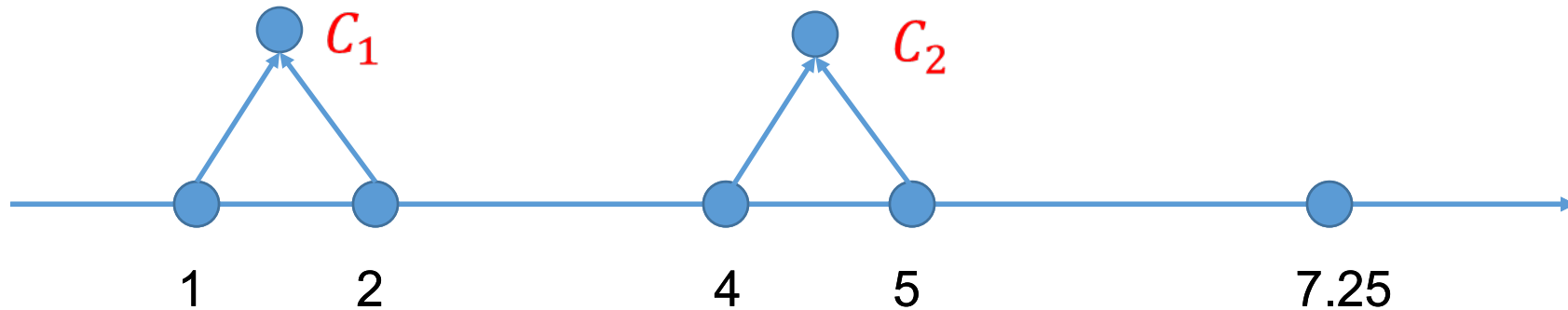
C. $\{1,2,4\}, \{5, 7.25\}$

D. $\{1,2,4,5\}, \{7.25\}$



$$d(C_1, C_2) = \frac{3 + 2 + 4 + 3}{4} = 3,$$

$$d(C_2, \{7.25\}) = \frac{3.25 + 2.25}{2} = 2.75$$



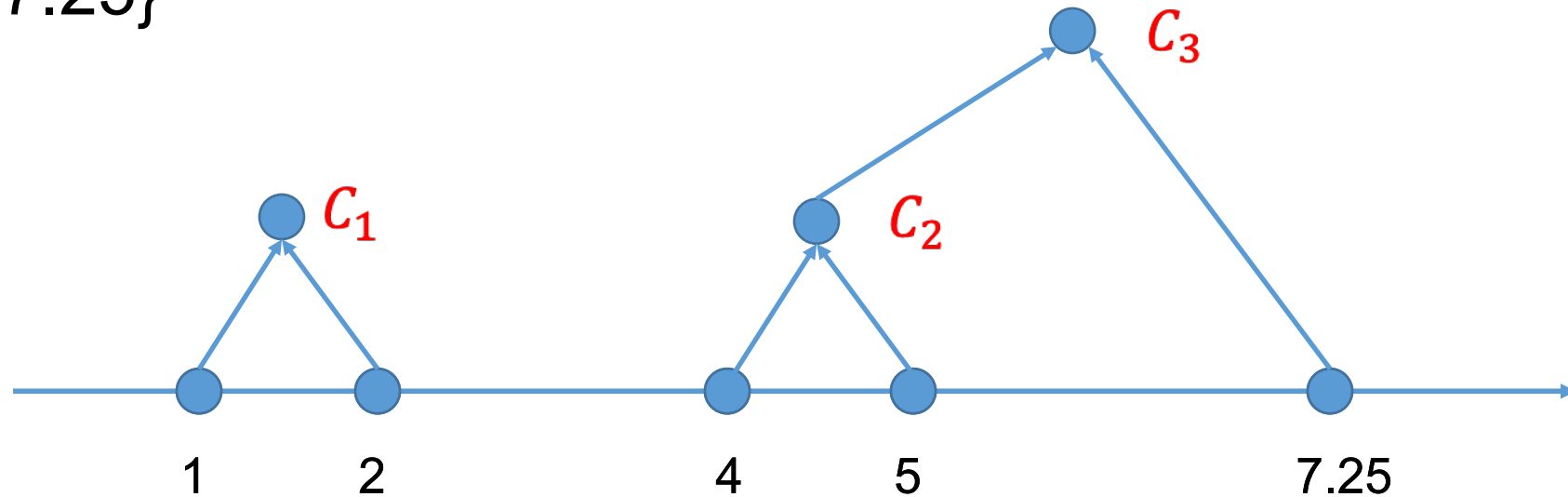
Q3-1: Suppose we run average linkage on the following data set to get two clusters. What is the result?

A. $\{1\}, \{2,4,5,7.25\}$

B. $\{1,2\}, \{4, 5, 7.25\}$ ←

C. $\{1,2,4\}, \{5, 7.25\}$

D. $\{1,2,4,5\}, \{7.25\}$



Q3-2: Assume, you want to cluster 7 points in 2-dim into 3 clusters using K-Means clustering algorithm. After the first iteration clusters, C1, C2, C3 has the following points:

C1: $\{(1,1), (3,3), (5,5)\}$

C2: $\{(0,4), (-2,4)\}$

C3: $\{(7,7), (9,9)\}$

What will be the cluster centroids if you want to proceed for the second iteration?

A. C1: (3,3), C2: (-1,4), C3: (8,8)

B. C1: (3,3), C2: (0,4), C3: (8,8)

C. C1: (3,3), C2: (-1,4), C3: (7,8)

D. C1: (2,2), C2: (-1,4), C3: (7,8)

Q3-2: Assume, you want to cluster 7 points in 2-dim into 3 clusters using K-Means clustering algorithm. After first iteration clusters, C1, C2, C3 has following points:

C1: $\{(1,1), (3,3), (5,5)\}$

C2: $\{(0,4), (-2,4)\}$

C3: $\{(7,7), (9,9)\}$

What will be the cluster centroids if you want to proceed for second iteration?

A. C1: (3,3), C2: (-1,4), C3: (8,8)

B. C1: (3,3), C2: (0,4), C3: (8,8)

C. C1: (3,3), C2: (-1,4), C3: (7,8)

D. C1: (2,2), C2: (-1,4), C3: (7,8)



Compute the average of the data points in each cluster

Q3-3: Which are true about linear regression?

1. When $\lambda \rightarrow +\infty$, ridge regression reduces to OLS.
2. The regression function must be linear in the original input features.
3. Gradient descent can be used to solve OLS.

A. 1,2

B. 1,3

C. 2,3

D. None of the above

Q3-3: Which are true about linear regression?

1. When $\lambda \rightarrow +\infty$, ridge regression reduces to OLS.
2. The regression function must be linear in the original input features.
3. Gradient descent can be used to solve OLS.

A. 1,2

B. 1,3

C. 2,3

D. None of the above



1: No. when $\lambda \rightarrow 0$, ridge regression reduces to OLS
2: No. It only needs to be linear in the parameter
3: Yes. Gradient descent is a general method for optimization.

Q3-4: Which are true about logistic regression?

1. When $\theta^T x_i = 0$, the model will predict label $+1$ with probability close to 1
2. There is ground-truth θ^* that can achieve 0 classification error on the training set
3. Gradient descent can be used to solve the regularized logistic regression problem

A. 1,2

B. 1,3

C. 2,3

D. None of the above

Q3-4: Which are true about logistic regression?

1. When $\theta^T x_i = 0$, the model will predict label $+1$ with probability close to 1
2. There is ground-truth θ^* that can achieve 0 classification error on the training set
3. Gradient descent can be used to solve the regularized logistic regression problem

A. 1,2

B. 1,3

C. 2,3

D. None of the above



1: No. predicts $\frac{1}{2}$ for $+1$, $\frac{1}{2}$ for -1 .


2: No. Not 0 classification error. There is randomness in the labels.

3: Yes. Gradient descent is a general method for optimization.

Q3-5: Suppose we have $\theta = [0.5, 0.6, 1]$ for logistic regression. What label will the model predict for $x = [1, -5, 2]$?

- A. -1
- B. +1
- C. Equal probabilities for -1 and +1
- D. None of the above

Q3-5: Suppose we have $\theta = [0.5, 0.6, 1]$ for logistic regression. What label will the model predict for $x = [1, -5, 2]$?

- A. -1 
- B. +1
- C. Equal probabilities for -1 and +1
- D. None of the above

The $\theta^T x = -0.5$, so $p[y=+1 | x] < 0.5$, so we will predict -1

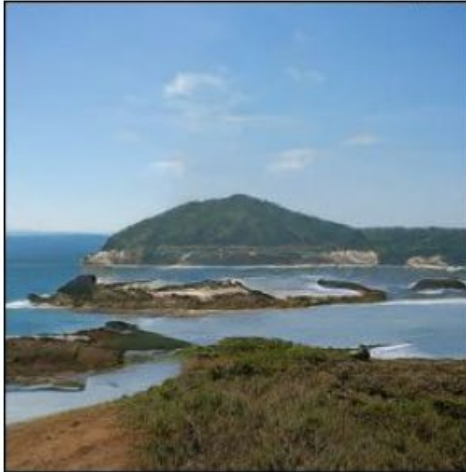
A Fun Quiz

Let us look at the following images. Some of them are generated by an AI model. Which one do you think is a *real image*?

(1)



(2)



(3)



(4)



- A. (1)
- B. (2)
- C. (3)
- D. (4)

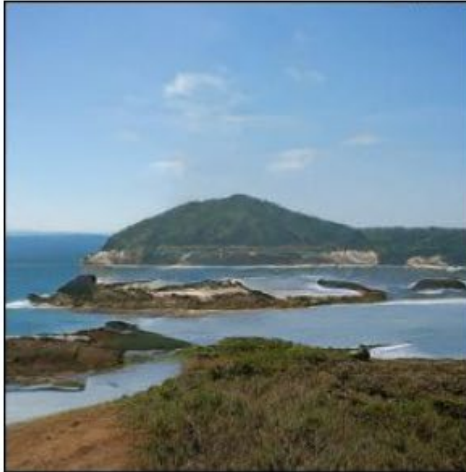
A Fun Quiz

Let us look at the following images. Some of them are generated by an AI model. Which one do you think is a *real image*?

(1)



(2)



(3)



(4)



- A. (1)
- B. (2)
- C. (3)
- D. (4)

Answer: None. They are all generated by AI ;)

Gradient Descent

Consider the function $y = (x^2 + 1) \cdot w$. What's the derivative/gradient for x ?

- A. $w \cdot 2x$
- B. $w \cdot x$
- C. $2x$
- D. w

Gradient Descent

Consider the function $y = (x^2 + 1) \cdot w$. What's the derivative/gradient for x ?

- A. $w \cdot 2x$
- B. $w \cdot x$
- C. $2x$
- D. w

Answer: A.

Let $y = h(x) \cdot w = (x^2 + 1) \cdot w$ where $h(x) = (x^2 + 1)$. According to the chain rule,

$$\frac{\partial y}{\partial x} = \frac{\partial y}{\partial h} \cdot \frac{\partial h}{\partial x} = w \cdot 2x.$$

Linear Perceptron

Consider the linear perceptron with x as the input. Which function can the linear perceptron compute?

(1) $y = ax + b$

(2) $y = ax^2 + bx + c$

A. (1)

B. (2)

C. (1)(2)

D. None of the above

Linear Perceptron

Consider the linear perceptron with x as the input. Which function can the linear perceptron compute?

(1) $y = ax + b$

(2) $y = ax^2 + bx + c$

A. (1)

B. (2)

C. (1)(2)

D. None of the above

Answer: A. All units in a linear perceptron are linear. Thus, the model can not present non-linear functions.

Linear Perceptron: Learning

Consider using a linear perceptron for regression given a training dataset. If we use gradient descent for learning the weights of the model and start from the same initial weights, what will happen if we increase the learning rate (within a reasonable range)?

- A. The model will always take less steps to converge.
- B. The model might not converge at all.
- C. The model will always converge, but might converge to different solutions.
- D. The model will always converge to the same solution.

Linear Perceptron: Learning

Consider using a linear perceptron for regression given a training dataset. If we use gradient descent for learning the weights of the model and start from the same initial weights, what will happen if we increase the learning rate (within a reasonable range)?

- A. The model will always take less steps to converge.
- B. The model might not converge at all.
- C. The model will always converge, but might converge to different solutions.
- D. The model will always converge to the same solution.

Answer: D. This is the same as linear regression (a convex optimization problem)

Perceptron

Perceptron can be used for:

- A. classification
- B. regression
- C. both classification and regression

Perceptron

Perceptron can be used for:

- A. classification
- B. regression
- C. both classification and regression

Answer: C. Perceptron can be used in both tasks by using different activation functions.

Q1. Consider a small dataset with four points, where each point is in 2D, and y is their classification label. Can we classify this dataset perfectly using a single nonlinear perceptron?

x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	0

a) Yes

b) No

Q1. Consider a small dataset with four points, where each point is in 2D, and y is their classification label. Can we classify this dataset perfectly using a single nonlinear perceptron?

x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	0

a) Yes

b) No 

Solution:

XOR is not linearly separable, so we cannot use a single neuron (perceptron) to classify this problem.

Q1. Consider a three-layer network with **linear Perceptrons** for binary classification. The hidden layer has 3 neurons. Can the network represent a XOR problem?

- a) Yes
- b) No

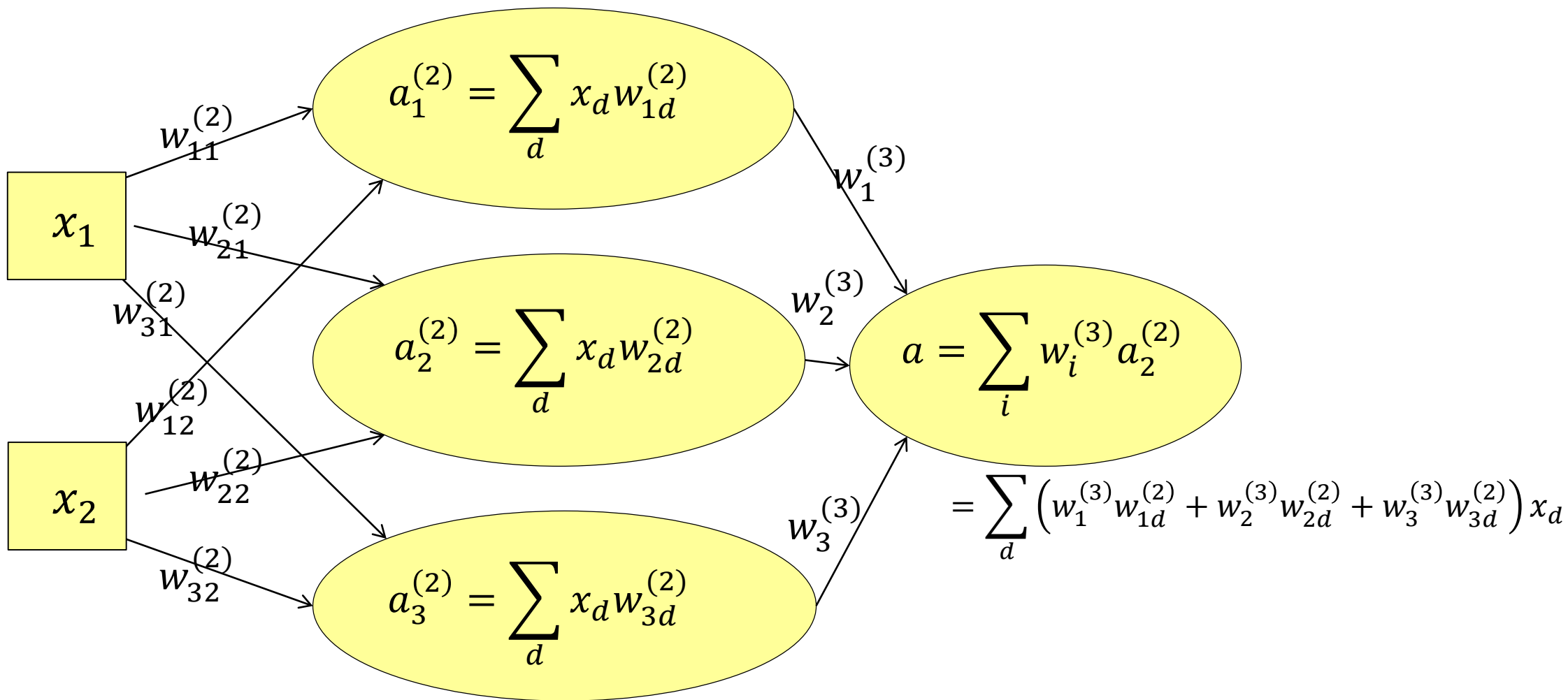
Q1. Consider a three-layer network with **linear Perceptrons** for binary classification. The hidden layer has 3 neurons. Can the network represent a XOR problem?

a) Yes

b) No 

Solution:

A combination of linear Perceptrons are still a linear function.



Q5. Gradient descent in neural networks computes the _____ of a loss function w.r.t. the model _____ until convergence.

- a) gradients, parameters
- b) parameter, gradients
- c) loss, parameters
- d) parameters, loss

Q5. Gradient descent in neural networks computes the _____ of a loss function w.r.t. the model _____ until convergence.

- a) gradients, parameters ←
- b) parameter, gradients
- c) loss, parameters
- d) parameters, loss

Consider a hidden layer of a neural network. The input of the layer is a 4-D vector. The layer has 16 neurons. What is the size of the weight matrix \mathbf{W} for this layer? Assume we have $\mathbf{a} = g(\mathbf{W}^T \mathbf{x} + \mathbf{b})$.

- A. 4 x 4
- B. 4 x 16
- C. 16 x 16
- D. 16 x 4

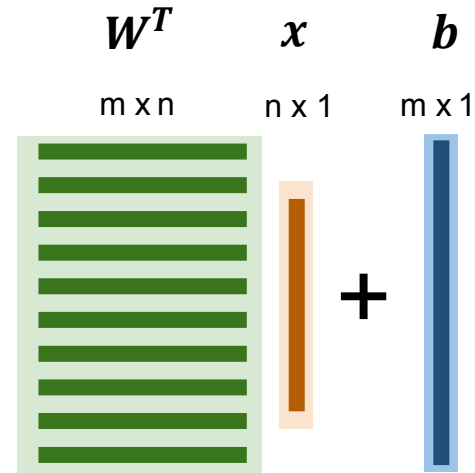
Consider a hidden layer of a neural network. The input of the layer is a 4-D vector. The layer has 16 neurons. What is the size of the weight matrix \mathbf{W} for this layer? Assume we have $\mathbf{a} = g(\mathbf{W}^T \mathbf{x} + \mathbf{b})$.

A. 4 x 4

B. 4 x 16

C. 16 x 16

D. 16 x 4



Let $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$. Which of following functions is NOT an element-wise operation that can be used as an activation function?

A. $f(\mathbf{x}) = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

B. $f(\mathbf{x}) = \begin{bmatrix} \max(0, x_1) \\ \max(0, x_2) \end{bmatrix}$

C. $f(\mathbf{x}) = \begin{bmatrix} \exp(x_1) \\ \exp(x_2) \end{bmatrix}$

D. $f(\mathbf{x}) = \begin{bmatrix} \exp(x_1 + x_2) \\ \exp(x_2) \end{bmatrix}$

Let $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$. Which of following functions is NOT an element-wise operation that can be used as an activation function?

A. $f(\mathbf{x}) = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

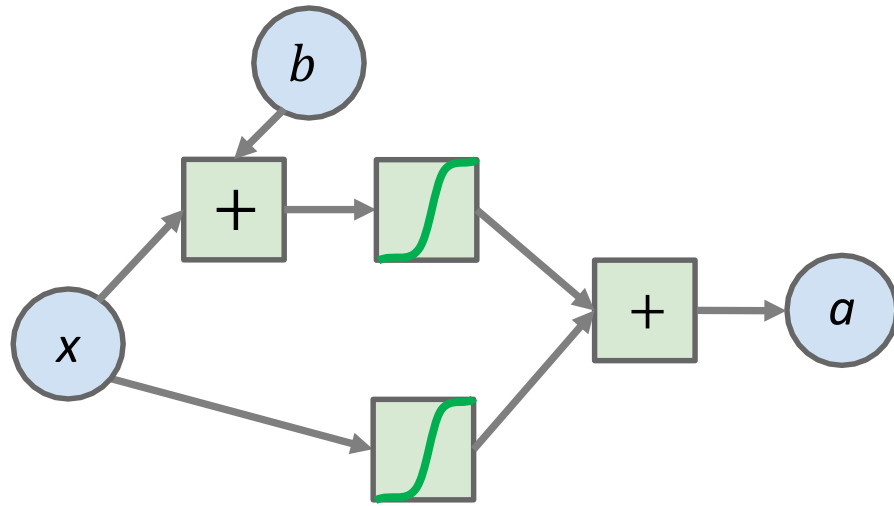
B. $f(\mathbf{x}) = \begin{bmatrix} \max(0, x_1) \\ \max(0, x_2) \end{bmatrix}$

C. $f(\mathbf{x}) = \begin{bmatrix} \exp(x_1) \\ \exp(x_2) \end{bmatrix}$

D. $f(\mathbf{x}) = \begin{bmatrix} \exp(x_1 + x_2) \\ \exp(x_2) \end{bmatrix}$

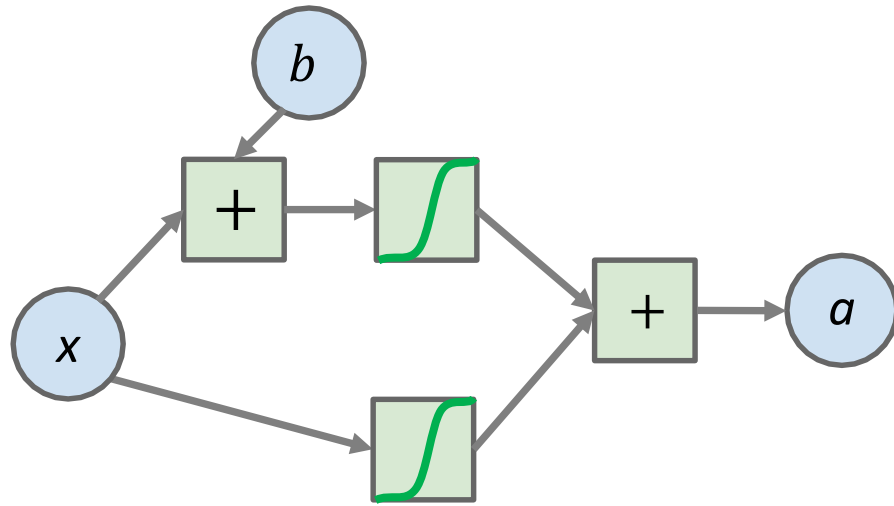
This is not an element-wise operation as the first output depends on both input values.

Consider the following computational graph. Which function does it represent? Assuming a sigmoid activation function.



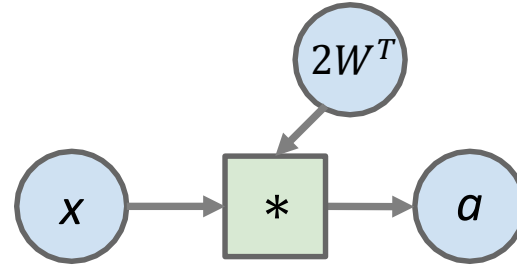
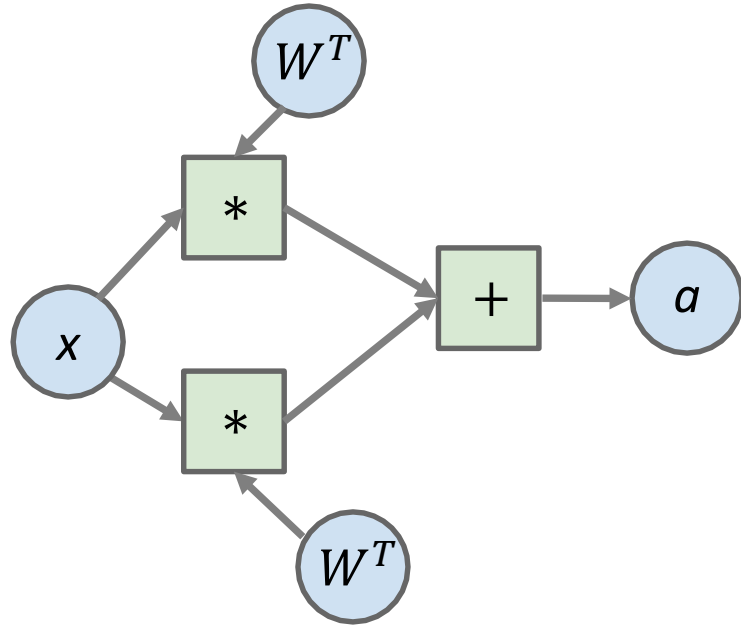
- A. $\text{sigmoid}(x + b)$
- B. $\text{sigmoid}(x)$
- C. $\text{sigmoid}(x + b) + \text{sigmoid}(x)$
- D. $x + b$

Consider the following computational graph. Which function does it represent? Assuming a sigmoid activation function.



- A. $\text{sigmoid}(x + b)$
- B. $\text{sigmoid}(x)$
- C. $\text{sigmoid}(x + b) + \text{sigmoid}(x)$**
- D. $x + b$

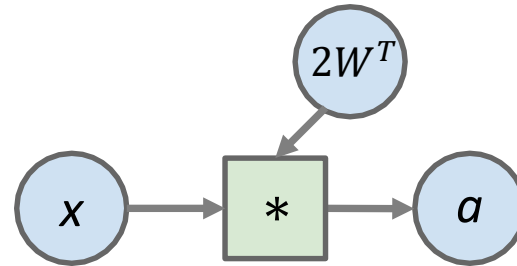
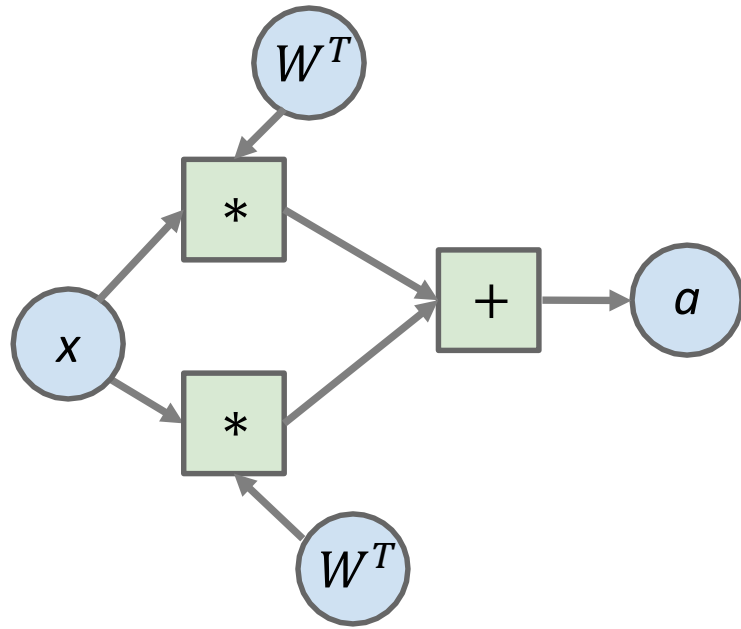
Consider the following two computational graphs. Do they represent the same function?



Yes

No

Consider the following two computational graphs. Do they represent the same function?



Yes. The first graph is $W^T x + W^T x$ and the second graph is $2W^T x$.

Let $f(x) = \begin{cases} -1 & x < 0.5 \\ 1 & x \geq 0.5 \end{cases}$. Can we use this function as an operation on a computational graph that supports backward propagation?

Yes

No

Let $f(x) = \begin{cases} -1 & x < 0.5 \\ 1 & x \geq 0.5 \end{cases}$. Can we use this function as an operation on a computational graph that supports backward propagation?

No. The function is not continuous and not differentiable when $x=0.5$.

Let $f(x) = \begin{cases} 0 & x < 0 \\ x & x \geq 0 \end{cases}$. Can we use this function as an operation on a computational graph that supports backward propagation? Assume that we define the “gradient” $f'(0) = 0$.

Yes

No

Let $f(x) = \begin{cases} 0 & x < 0 \\ x & x \geq 0 \end{cases}$. Can we use this function as an operation on a computational graph that supports backward propagation? Assume that we define the “gradient” $f'(0) = 0$.

Yes. The function is continuous but not differentiable at 0. With the patch, we can compute a “gradient” (known as sub-gradient) for this function and thus use this function as an operation on the graph.

Consider a comparison between a sigmoid function and a rectified linear unit (ReLU). Which of following statement is NOT true?

- Sigmoid function is more expensive to compute
- ReLU has non-zero gradient everywhere
- Sigmoid has a large zone that has zero gradient
- It is possible to compute the “gradient” of ReLU

Consider a comparison between a sigmoid function and a rectified linear unit (ReLU). Which of following statement is NOT true?

- Sigmoid function is more expensive to compute
- ReLU has non-zero gradient everywhere
- Sigmoid has a large zone that has zero gradient
- It is possible to compute the “gradient” of ReLU

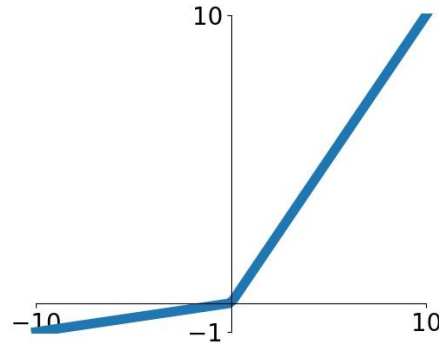
A Leaky ReLU is defined as $f(x)=\max(0.1x, x)$. Does it have non-zero gradient everywhere?

- Yes
- No

A Leaky ReLU is defined as $f(x) = \max(0.1x, x)$. Does it have non-zero gradient everywhere?

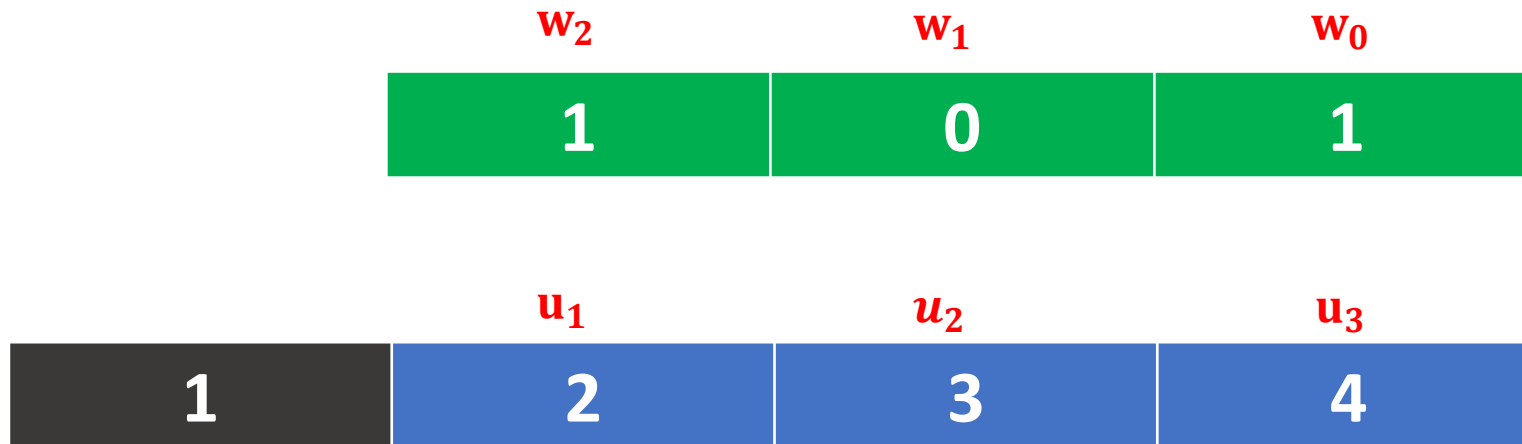
- Yes

- No



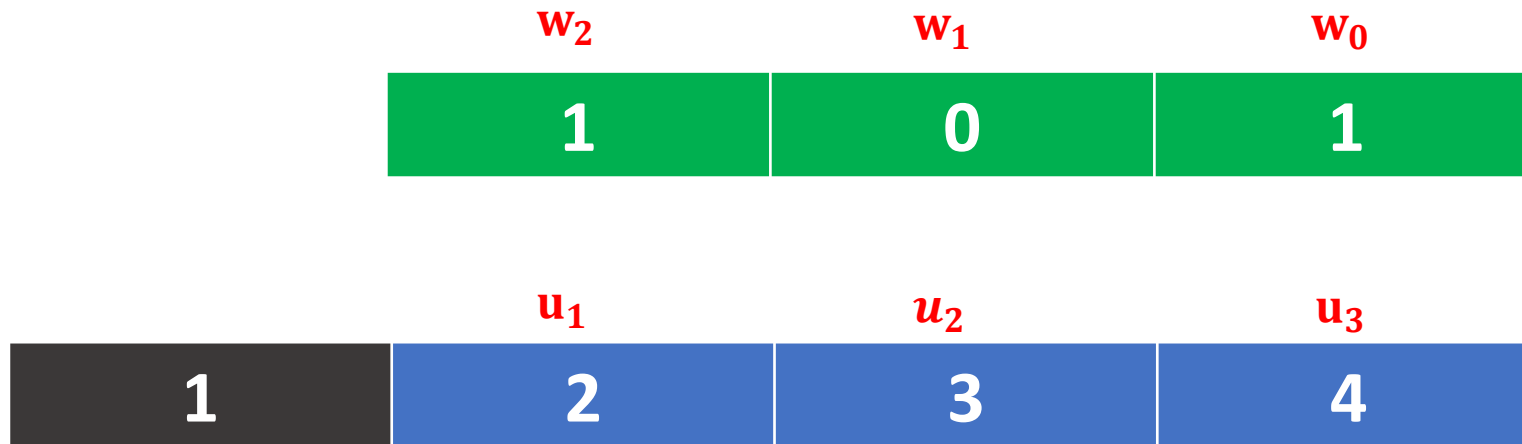
What is the output value of the given convolution operation at the current step?

- 6
- 4
- 9
- 2



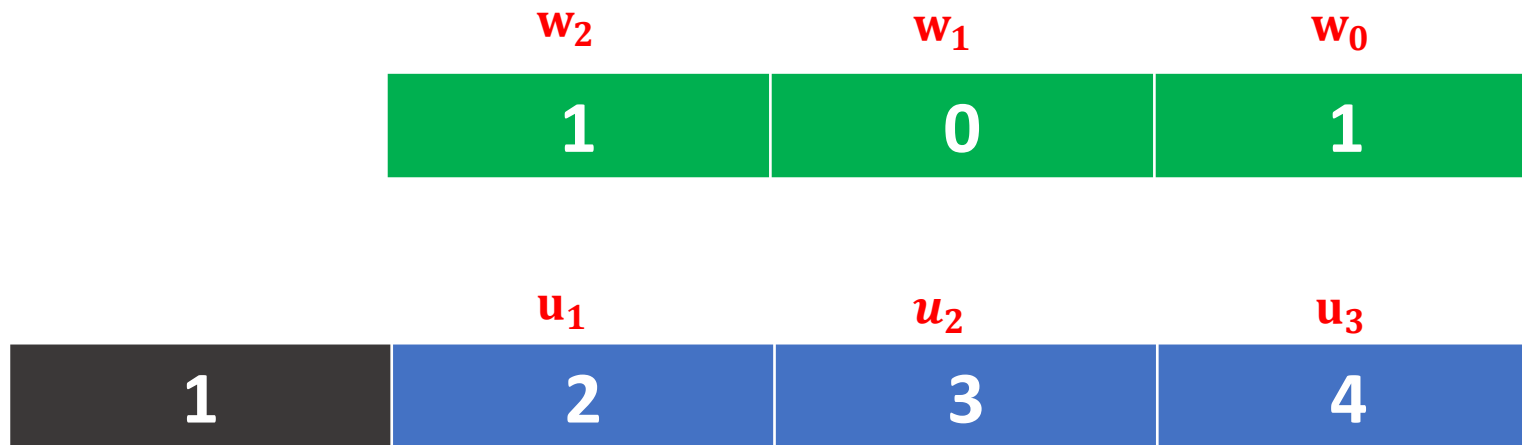
What is the output value of the given convolution operation at the current step?

- 6
- 4
- 9
- 2



What is the output value of the given convolution operation at the **next** step?

- 5
- 4
- 3
- 2



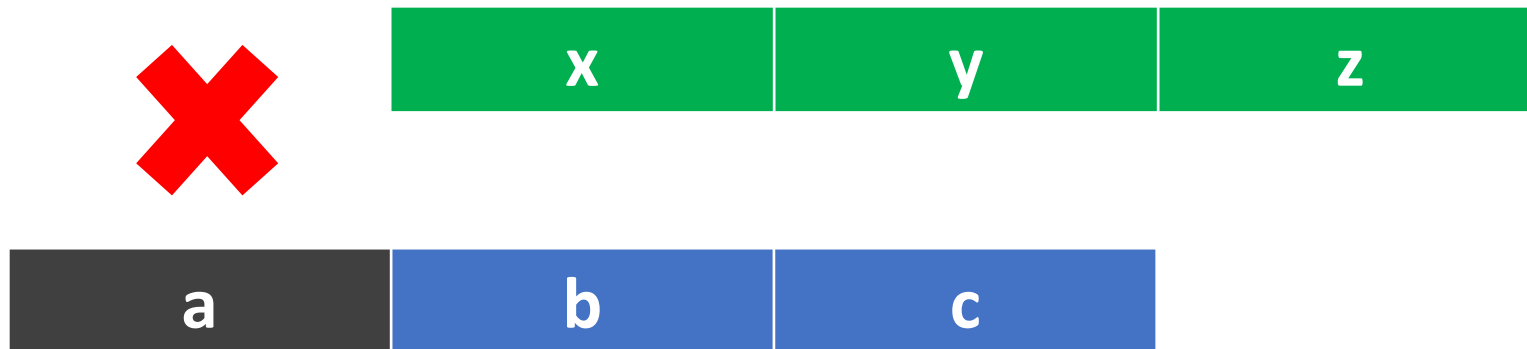
What is the output value of the given convolution operation at the next step?

- 5
- 4
- 3
- 2



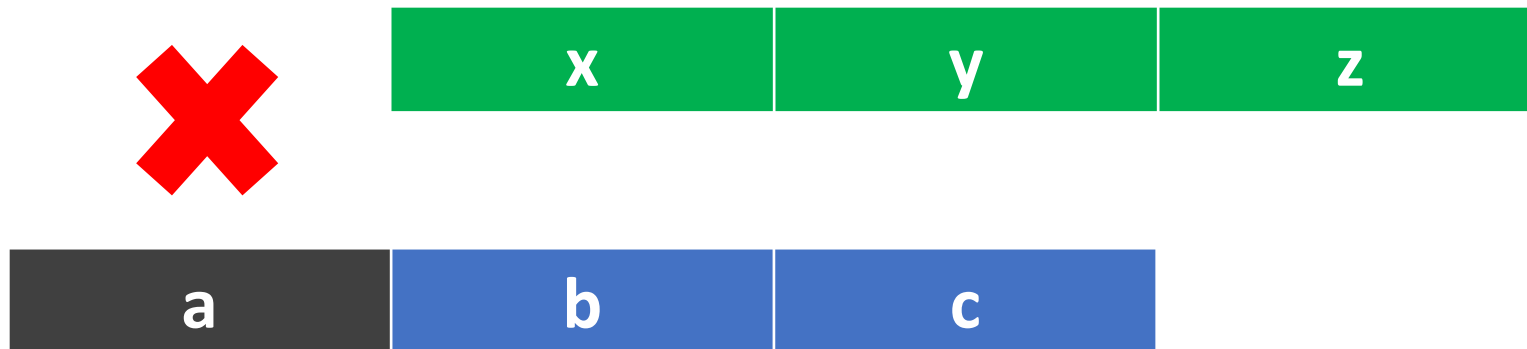
Given an input 1D array of size 7, a convolutional kernel of size 3 with stride 1. If we don't allow the kernel to partly fall outside of the input, what is the output size?

- 5
- 4
- 7
- 6



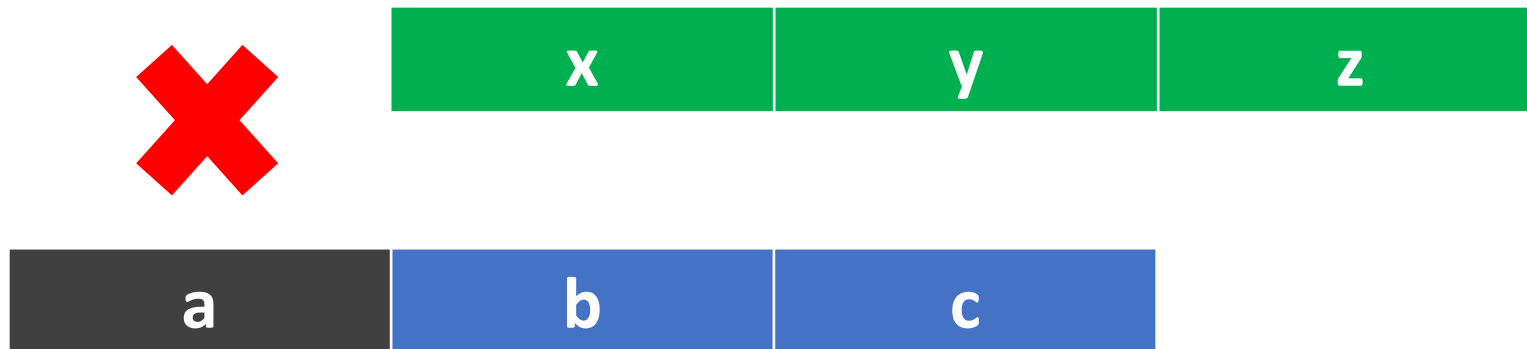
Given an input 1D array of size 7, a convolutional kernel of size 3 with stride 1. If we don't allow the kernel to partly fall outside of the input, what is the output size?

- 5
- 4
- 7
- 6



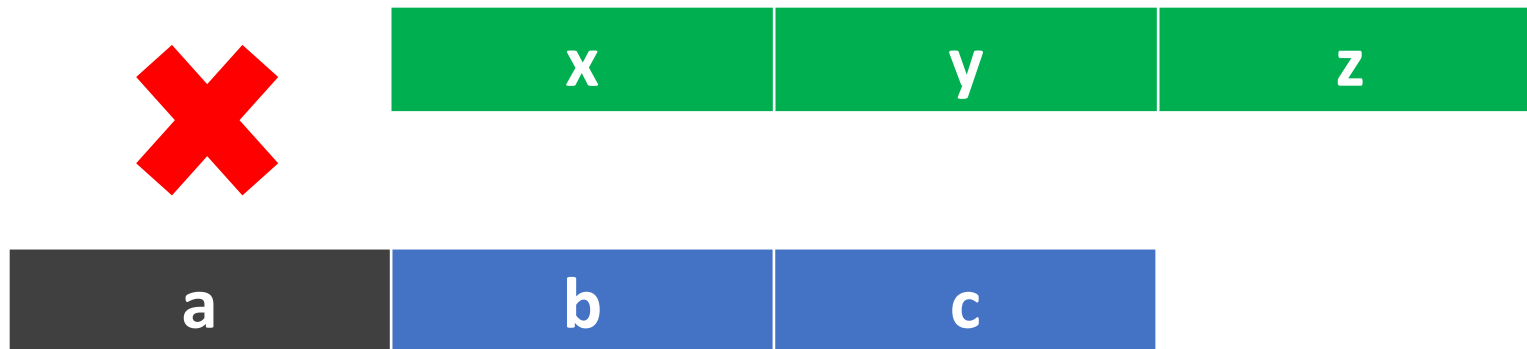
Given an input 1D array of size 7, a convolutional kernel of size 3 with stride 2. If we don't allow the kernel to partly fall outside of the input, what is the output size?

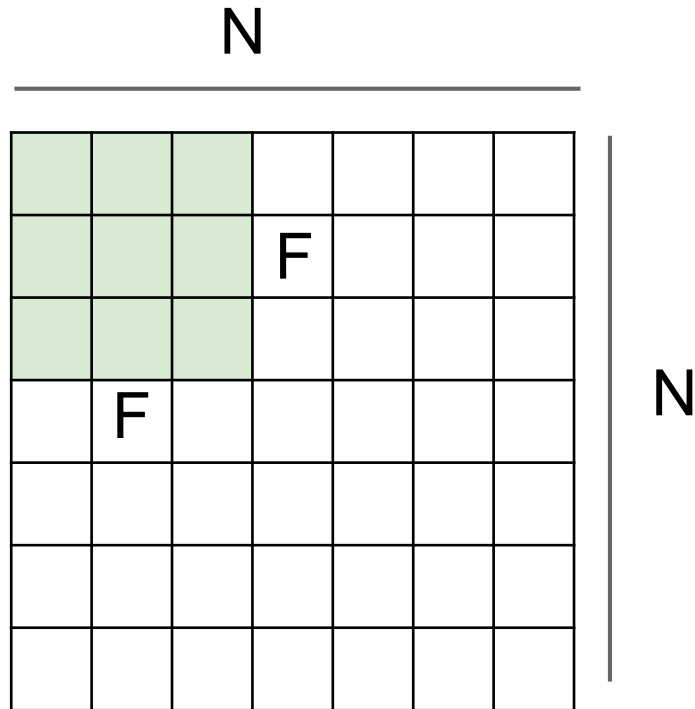
- 6
- 5
- 4
- 3



Given an input 1D array of size 7, a convolutional kernel of size 3 with stride 2. If we don't allow the kernel to partly fall outside of the input, what is the output size?

- 6
- 5
- 4
- 3





Valid Output size:
 $(N - F) // \text{stride} + 1$

e.g. $N = 7, F = 3$:

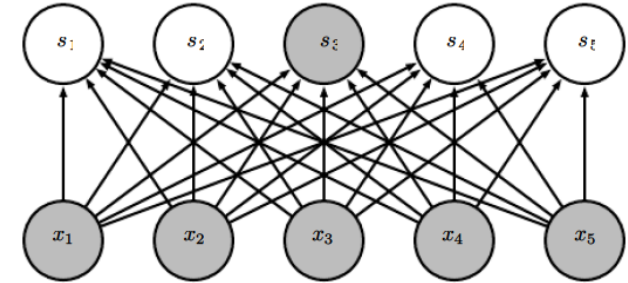
stride 1 $\Rightarrow (7 - 3) // 1 + 1 = 5$

stride 2 $\Rightarrow (7 - 3) // 2 + 1 = 3$

stride 3 $\Rightarrow (7 - 3) // 3 + 1 = 2$

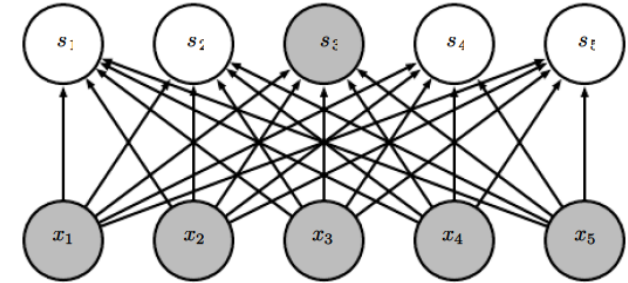
Let us compare a convolutional layer vs. a standard fully connected layer. Which of the following is TRUE?

- Convolution layer has more parameters
- Fully connected layer can be used to represent the convolution
- Convolution layer can be used to represent fully connected layer
- Fully connected layer is more efficient



Let us compare a convolutional layer vs. a standard fully connected layer. Which of the following is TRUE?

- Convolution layer has more parameters
- Fully connected layer can be used to represent the convolution
- Convolution layer can be used to represent fully connected layer
- Fully connected layer is more efficient



y	z					a
x	y	z				b
	x	y	z			c
		x	y	z		d
			x	y	z	e
				x	y	f

Consider a convolution $s = u * w$ with the following inputs: $u=[3, 2, 0, 1]$, $w=[-1, 2, 1]$ and the output $s=[4, 7, 1, 2]$. What is the size of the gradient $\frac{\partial s}{\partial w}$?

- 3 x 3
- 4 x 4
- 3 x 4
- 4 x 3

Consider a convolution $s = u * w$ with the following inputs: $u=[3, 2, 0, 1]$, $w=[-1, 2, 1]$ and the output $s=[4, 7, 1, 2]$. What is the size of the gradient $\frac{\partial s}{\partial w}$?

- 3 x 3
- 4 x 4
- 3 x 4
- 4 x 3

Consider a convolution $s = u * w$ with the following inputs: $u=[3, 2, 0, 1]$, $w=[-1, 2, 1]$ and the output $s=[4, 7, 1, 2]$. Fill in the missing value for the gradient $\frac{\partial s}{\partial w}$.

$$\begin{bmatrix} 2 & 3 & 0 \\ 0 & ? & 3 \\ 1 & 0 & 2 \\ 0 & 1 & 0 \end{bmatrix} \begin{array}{l} \bullet 3 \\ \bullet 2 \\ \bullet 1 \\ \bullet 0 \end{array}$$

Consider a convolution $s = u * w$ with the following inputs: $u=[3, 2, 0, 1]$, $w=[-1, 2, 1]$ and the output $s=[4, 7, 1, 2]$. Fill in the missing value for the gradient $\frac{\partial s}{\partial w}$.

$\begin{bmatrix} 2 & 3 & 0 \\ 0 & ? & 3 \\ 1 & 0 & 2 \\ 0 & 1 & 0 \end{bmatrix}$	• 3
	• 2
	• 1
	• 0

Consider a 1D max pooling with filter size 3, stride 2. Given the following input sequence, what is the output after pooling?

0.5	1	0.7	0.1	0.2
-----	---	-----	-----	-----

- 1, 1, 0.7
- 1, 0.7
- 0.5, 0.7
- 0.5, 1, 0.7

Consider a 1D max pooling with filter size 3, stride 2. Given the following input sequence, what is the output after pooling?

0.5	1	0.7	0.1	0.2
-----	---	-----	-----	-----

- 1, 1, 0.7
- 1, 0.7
- 0.5, 0.7
- 0.5, 1, 0.7

Define a 1D mean pooling operation that takes the average value (instead of max value) within a local window. With filter size 3, stride 2, and the following input sequence, what is the output after pooling?

0.5	1	0.6	0.1	0.2
-----	---	-----	-----	-----

- 0.7, 0.3
- 0.5, 0.7
- 0.5, 0.3
- 0.7, 0.7

Define a 1D mean pooling operation that takes the average value (instead of max value) within a local window. With filter size 3, stride 2, and the following input sequence, what is the output after pooling?

0.5	1	0.6	0.1	0.2
-----	---	-----	-----	-----

- 0.7, 0.3
- 0.5, 0.7
- 0.5, 0.3
- 0.7, 0.7

For a multi-class classification problem, which output normalization is often considered?

- Sigmoid function
- Rectified linear unit (ReLU)
- Softmax function
- No normalization is needed

For a multi-class classification problem, which output normalization is often considered?

- Sigmoid function
- Rectified linear unit (ReLU)
- **Softmax function**
- No normalization is needed

Softmax function normalizes the output to a multinomial distribution.

Consider a convolutional network with a single convolutional layer and one fully connected layer for a 5-way classification problem. The feature map after convolution is of size $3 \times 3 \times 10$. What is the size of the weight in the fully connected layer?

- 90×5
- 9×5
- 30×5
- 10×5

Consider a convolutional network with a single convolutional layer and one fully connected layer for a 5-way classification problem. The feature map after convolution is of size $3 \times 3 \times 10$. What is the size of the weight in the fully connected layer?

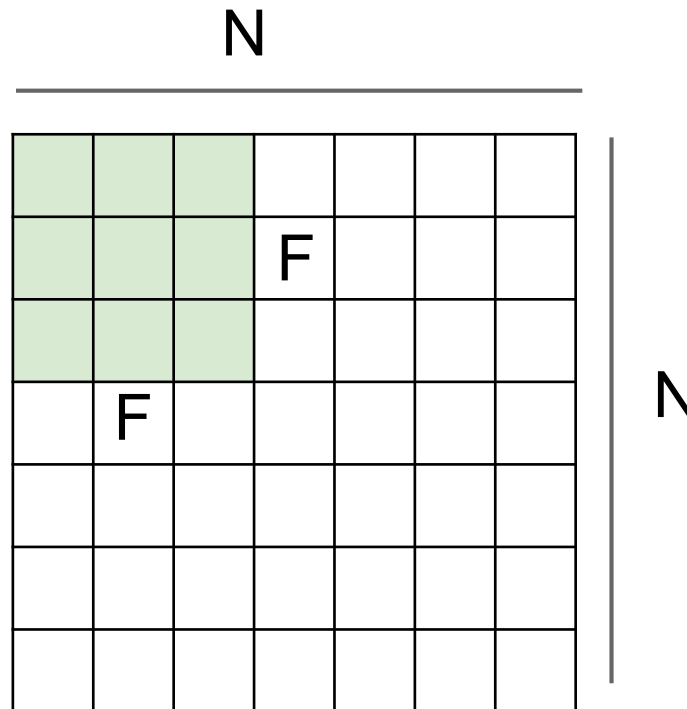
- 90 x 5
- 9 x 5
- 30 x 5
- 10 x 5

Consider a convolution layer with 16 filters. Each filter has a size of $11 \times 11 \times 3$, a stride of 2×2 . Given an input image of size $22 \times 22 \times 3$, if we don't allow a filter to fall outside of the input, what is the output size?

- $11 \times 11 \times 16$
- $6 \times 6 \times 16$
- $7 \times 7 \times 16$
- $5 \times 5 \times 16$

Consider a convolution layer with 16 filters. Each filter has a size of $11 \times 11 \times 3$, a stride of 2×2 . Given an input image of size $22 \times 22 \times 3$, if we don't allow a filter to fall outside of the input, what is the output size?

- $11 \times 11 \times 16$
- $6 \times 6 \times 16$
- $7 \times 7 \times 16$
- $5 \times 5 \times 16$



Valid Output size:
 $(N - F) // \text{stride} + 1$

Consider a convolutional network with 4 operations in a sequential order:

Conv1 + max pooling + Conv2 + sigmoid + FC.

Which of the following statement is NOT true?

- A nonlinear activation function is usually added between Conv1 and Conv2
- ReLU can be used to replace sigmoid
- A pooling operation is required after Conv2
- The input to FC needs to be a vector

Consider a convolutional network with 4 operations in a sequential order:
Conv1 + max pooling + Conv2 + sigmoid + FC.
Which of the following statement is NOT true?

- A nonlinear activation function is usually added between Conv1 and Conv2
- ReLU can be used to replace sigmoid
- A pooling operation is required after Conv2
- The input to FC needs to be a vector

Which of the following statement is True for the success of deep models?

- Better design of the neural networks
- Large scale training dataset
- Available computing power
- All of the above

Which of the following statement is True for the success of deep models?

- Better design of the neural networks
- Large scale training dataset
- Available computing power
- All of the above

Simply stacking more convolutional layers in a deep convolutional network will always lead to better performance.

- True
- False

Simply stacking more convolutional layers in a deep convolutional network will always lead to better performance.

- True
- False

For an MDP, the optimal policy is guaranteed to remain the same if we only change the reward function

1. True
2. False

For an MDP, the optimal policy is guaranteed to remain the same if we only change the reward function

1. True
2. False

Consider an MDP with 2 states A, B and 2 actions: “stay” stays at the current state and “move” moves to the other state. Let r be the reward function such that $r(A) = 1$, $r(B) = 0$. Let A be the start state and γ be the discounting factor.

Consider the “always move” policy π : $\pi(A) = \pi(B) = \text{move}$ and an infinite sequence of A, B, A, B, ... from this policy. What is the utility (i.e., the expected sum of discounted reward) of this sequence?

1. 0
2. $1 / (1 - \gamma)$
3. $1 / (1 - \gamma^2)$
4. 1

Consider an MDP with 2 states A, B and 2 actions: “stay” stays at the current state and “move” moves to the other state. Let r be the reward function such that $r(A) = 1$, $r(B) = 0$. Let A be the start state and γ be the discounting factor.

Consider the “always move” policy π : $\pi(A) = \pi(B) = \text{move}$ and an infinite sequence of A, B, A, B, ... from this policy. What is the utility (i.e., the expected sum of discounted reward) of this sequence?

1. 0
2. $1 / (1 - \gamma)$
3. $1 / (1 - \gamma^2)$
4. 1

Sequence: A, B, A, B, A, B,

Discounted rewards: $1, 0, \gamma^2, 0, \gamma^4, 0, \gamma^6, \dots$

Sum of discounted rewards: $1 + \gamma^2 + \gamma^4 + \gamma^6 + \dots = 1/(1 - \gamma^2)$

In the above MDP, what is the optimal policy π^* ? Assume A as the start state.

1. $\pi(A) = \pi(B) = \text{move}$
2. $\pi(A) = \pi(B) = \text{stay}$
3. $\pi(A) = \text{stay}, \pi(B) = \text{move}$
4. $\pi(A) = \text{move}, \pi(B) = \text{stay}$

In the above MDP, what is the optimal policy π^* ? Assume A as the start state.

1. $\pi(A) = \pi(B) = \text{move}$
2. $\pi(A) = \pi(B) = \text{stay}$
3. $\pi(A) = \text{stay}, \pi(B) = \text{move}$
4. $\pi(A) = \text{move}, \pi(B) = \text{stay}$

Value iteration is guaranteed to converge if the discount factor (γ) satisfies $0 < \gamma < 1$.

1. True
2. False

Value iteration is guaranteed to converge if the discount factor (γ) satisfies $0 < \gamma < 1$.

1. True
2. False

Perceptron

Consider a nonlinear perceptron $a = \text{sigmoid}(\sum_d x_d w_d)$, what is the gradient of $\frac{\partial a}{\partial w_d}$?

- x_d
- ax_d
- $(1 - a)x_d$
- $a(1 - a)x_d$

Perceptron

Consider a nonlinear perceptron $a = \text{sigmoid}(\sum_d x_d w_d)$, what is the gradient of $\frac{\partial a}{\partial w_d}$?

- x_d
- $a x_d$
- $(1 - a) x_d$
- $a(1 - a) x_d$

$$\frac{\partial a}{\partial w_d} = \frac{\partial a}{\partial \sum x_d w_d} \frac{\partial \sum x_d w_d}{\partial w_d} = a(1 - a) x_d$$

Neural Network

Consider one layer in a neural network $\mathbf{a} = \text{sigmoid}(\mathbf{W}^T \mathbf{x} + \mathbf{b})$,

where $\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}$, $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, $\mathbf{W}^T = \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \end{bmatrix}$ and $\mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$.

What is the gradient of $\frac{\partial a_1}{\partial w_{11}}$ and $\frac{\partial a_1}{\partial w_{21}}$

- $a_1(1 - a_1)x_1, 0$
- $a_1(1 - a_1)x_1, a_1(1 - a_1)x_2$
- $0, a_1(1 - a_1)x_2$
- $a_1(1 - a_1)x_1 + b_1, a_1(1 - a_1)x_2 + b_2$

Neural Network

Consider one layer in a neural network $\mathbf{a} = \text{sigmoid}(\mathbf{W}^T \mathbf{x} + \mathbf{b})$,

where $\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}$, $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, $\mathbf{W}^T = \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \end{bmatrix}$ and $\mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$.

What is the gradient of $\frac{\partial a_1}{\partial w_{11}}$ and $\frac{\partial a_1}{\partial w_{21}}$

- $a_1(1 - a_1)x_1, 0$
- $a_1(1 - a_1)x_1, a_1(1 - a_1)x_2$
- $0, a_1(1 - a_1)x_2$
- $a_1(1 - a_1)x_1 + b_1, a_1(1 - a_1)x_2 + b_2$

Note that we have

$$a_1 = \text{sigmoid}(w_{11}x_1 + w_{12}x_2 + b_1)$$
$$\rightarrow \frac{\partial a_1}{\partial w_{11}} = a_1(1 - a_1)x_1$$

a_1 is not related to w_{21} , thus $\frac{\partial a_1}{\partial w_{21}} = 0$

Convolutional Neural Network

Consider a convolutional neural network that has three layers and outputs a scalar value. The convolutions do not allow values out of bounds.

$$z_1 = \text{ReLU}(w_1 * x) \text{ (conv with one kernel)}$$

$$z_2 = \text{ReLU}(w_2 * z_1 - 1) \text{ (conv with one kernel)}$$

$$a = \text{sigmoid}(w^T z_2) \text{ (fully connected)}$$

If $x = [1, 0, 1, 0, 1]^T$, $w_1 = w_2 = [1, 0, 1]^T$, compute z_2

- $[1, -1, 1]^T$
- $[2, 0, 2]^T$
- 4
- 3

Convolutional Neural Network

Consider a convolutional neural network that has three layers and outputs a scalar value. The convolutions do not allow values out of bounds.

$$z_1 = \text{ReLU}(w_1 * x) \text{ (conv with one kernel)}$$

$$z_2 = \text{ReLU}(w_2 * z_1 - 1) \text{ (conv with one kernel)}$$

$$a = \text{sigmoid}(w^T z_2) \text{ (fully connected)}$$

If $x = [1, 0, 1, 0, 1]^T$, $w_1 = w_2 = [1, 0, 1]^T$, compute z_2

- $[1, -1, 1]^T$
- $[2, 0, 2]^T$
- 4
- 3

$$z_1 = \text{ReLU}(w_1 * x) = [2, 0, 2]^T$$

$$z_2 = \text{ReLU}(w_2 * z_1 - 1) = \text{ReLU}(4 - 1) = 3$$

Convolutional Neural Network

Consider a convolutional neural network that has three layers and outputs a scalar value. The convolutions does not allow values out of bounds.

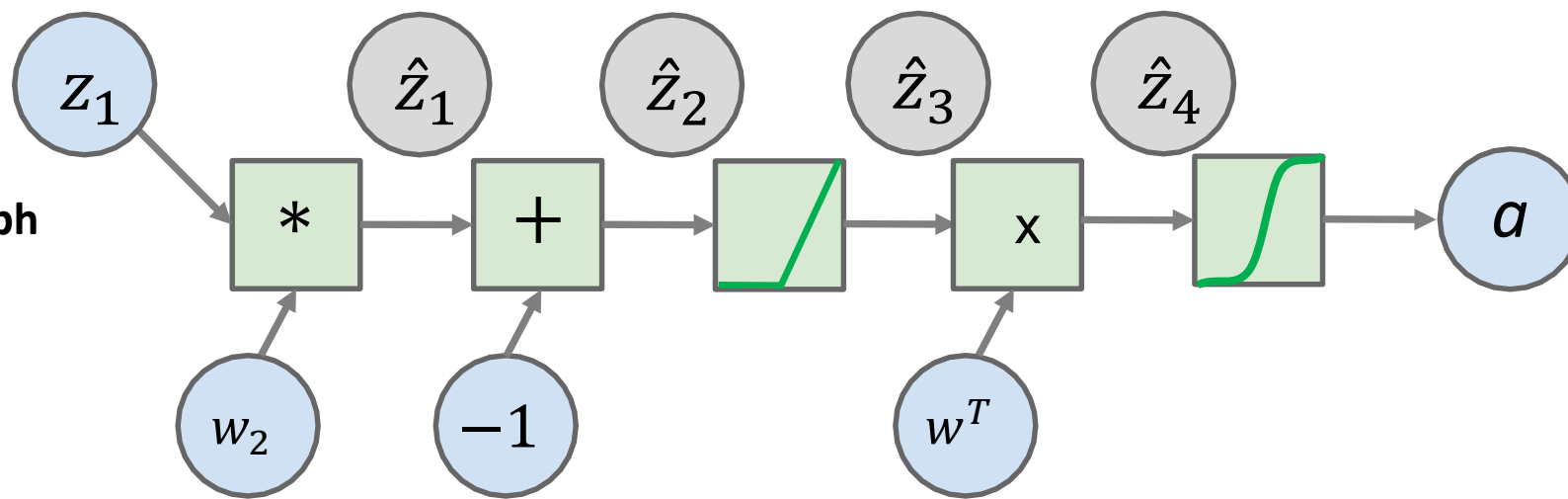
$$z_1 = \text{ReLU}(w_1 * x) \text{ (conv with one kernel)}$$

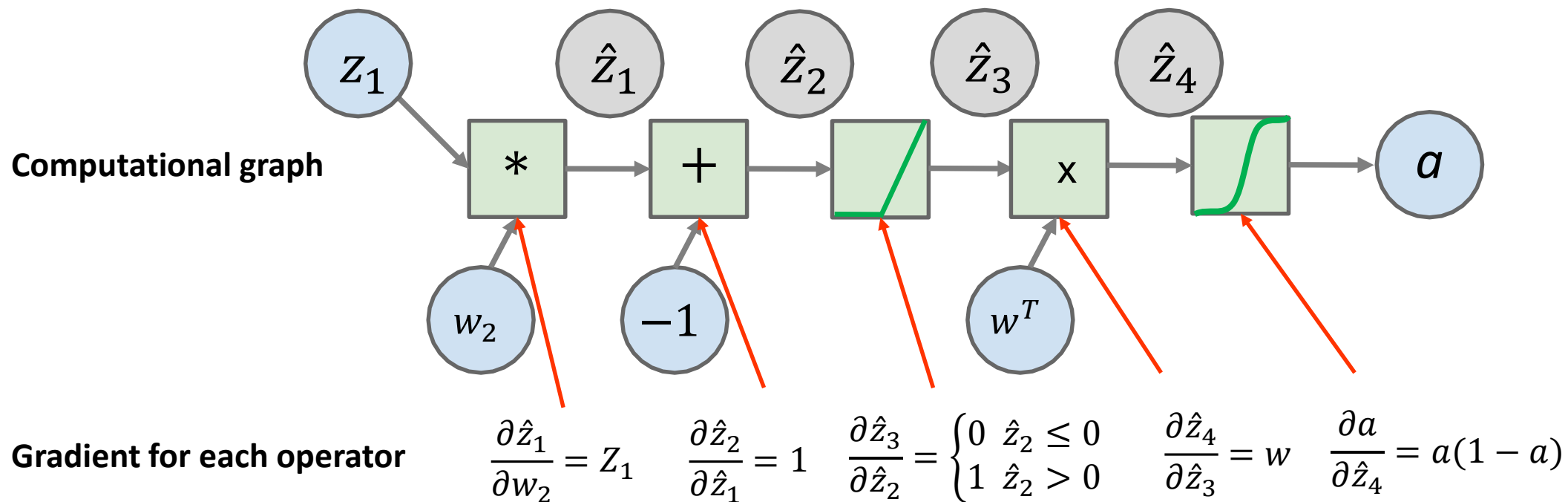
$$z_2 = \text{ReLU}(w_2 * z_1 - 1) \text{ (conv with one kernel)}$$

$$a = \text{sigmoid}(w^T z_2) \text{ (fully connected)}$$

Assume that we have a loss function E , how can we compute $\frac{\partial E}{\partial w_2}$?

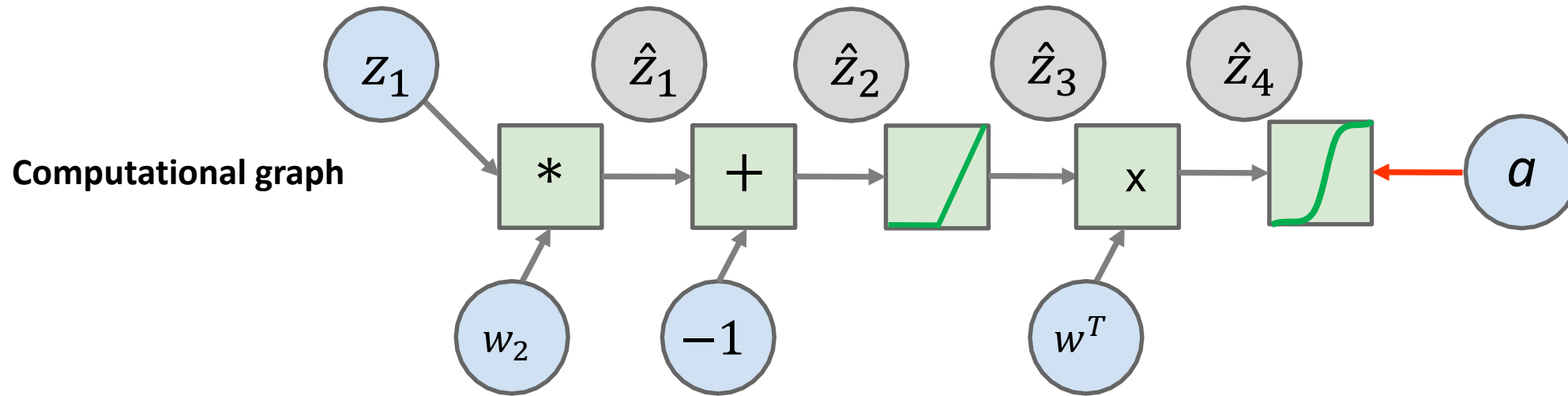
Computational graph





z_1 is the weight matrix
produced by the
convolutional kernel z_1

$$\frac{\partial E}{\partial \hat{z}_4} = \frac{\partial E}{\partial a} \frac{\partial a}{\partial \hat{z}_4}$$

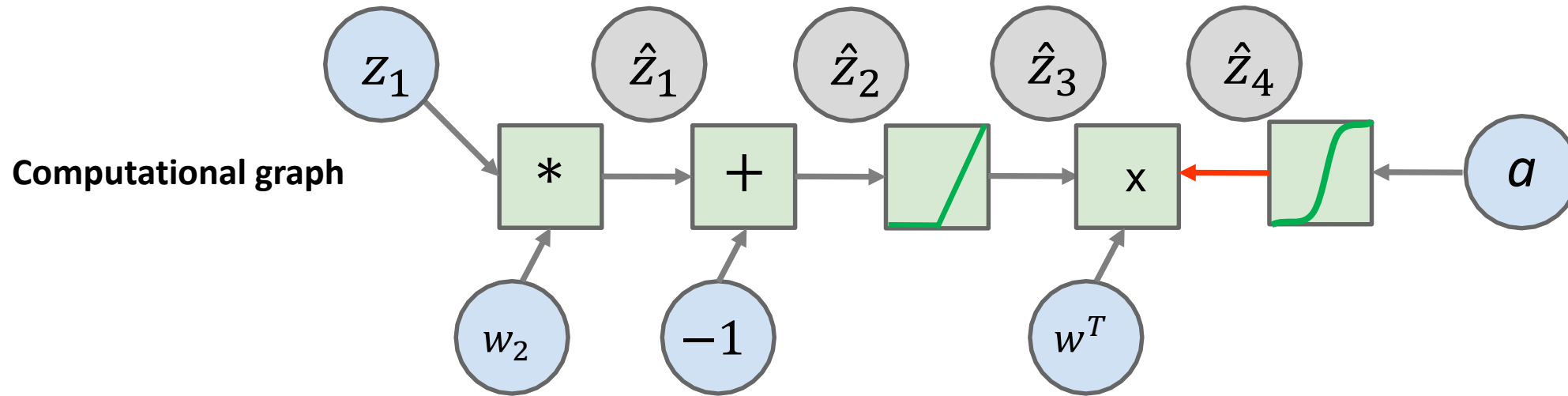


Gradient for each operator

$$\frac{\partial \hat{z}_1}{\partial w_2} = z_1 \quad \frac{\partial \hat{z}_2}{\partial \hat{z}_1} = 1 \quad \frac{\partial \hat{z}_3}{\partial \hat{z}_2} = \begin{cases} 0 & \hat{z}_2 \leq 0 \\ 1 & \hat{z}_2 > 0 \end{cases} \quad \frac{\partial \hat{z}_4}{\partial \hat{z}_3} = w \quad \frac{\partial a}{\partial \hat{z}_4} = a(1 - a)$$

z_1 is the weight matrix
produced by the
convolutional kernel z_1

$$\frac{\partial E}{\partial \hat{z}_3} = \frac{\partial E}{\partial \hat{z}_4} \frac{\partial \hat{z}_4}{\partial \hat{z}_3} \quad \frac{\partial E}{\partial \hat{z}_4} = \frac{\partial E}{\partial a} \frac{\partial a}{\partial \hat{z}_4}$$

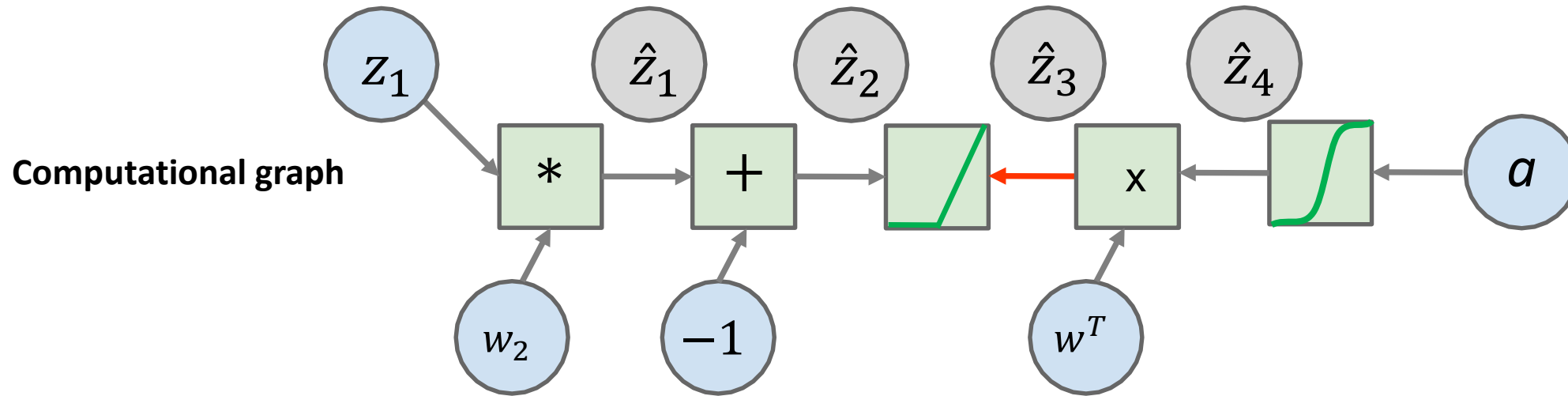


Gradient for each operator

$$\frac{\partial \hat{z}_1}{\partial w_2} = z_1 \quad \frac{\partial \hat{z}_2}{\partial \hat{z}_1} = 1 \quad \frac{\partial \hat{z}_3}{\partial \hat{z}_2} = \begin{cases} 0 & \hat{z}_2 \leq 0 \\ 1 & \hat{z}_2 > 0 \end{cases} \quad \frac{\partial \hat{z}_4}{\partial \hat{z}_3} = w \quad \frac{\partial a}{\partial \hat{z}_4} = a(1 - a)$$

z_1 is the weight matrix
produced by the
convolutional kernel z_1

$$\frac{\partial E}{\partial \hat{z}_2} = \frac{\partial E}{\partial \hat{z}_3} \frac{\partial \hat{z}_3}{\partial \hat{z}_2} \quad \frac{\partial E}{\partial \hat{z}_3} = \frac{\partial E}{\partial \hat{z}_4} \frac{\partial \hat{z}_4}{\partial \hat{z}_3} \quad \frac{\partial E}{\partial \hat{z}_4} = \frac{\partial E}{\partial a} \frac{\partial a}{\partial \hat{z}_4}$$

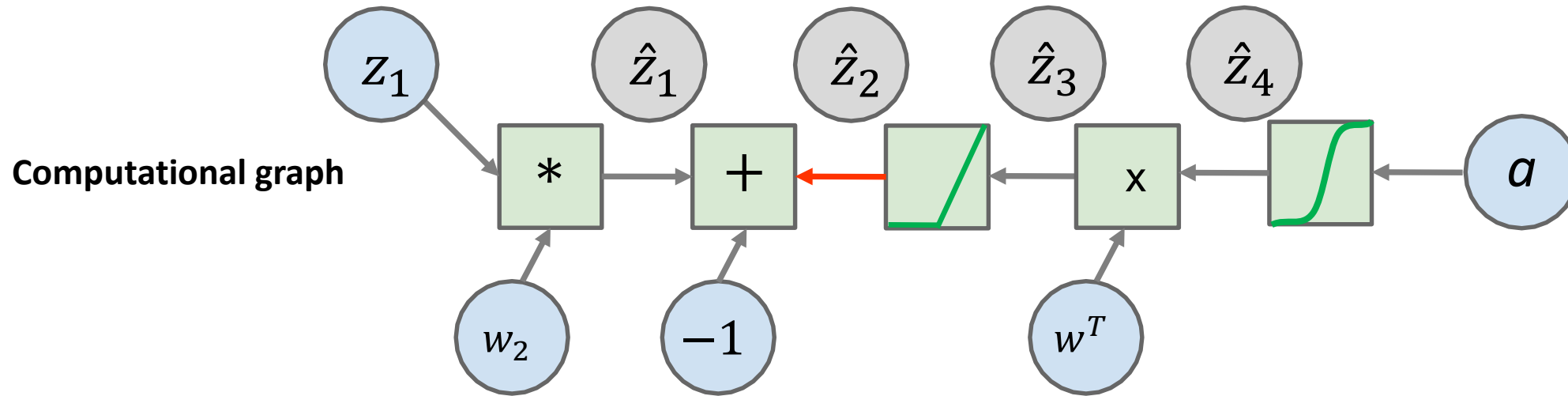


Gradient for each operator

$$\frac{\partial \hat{z}_1}{\partial w_2} = z_1 \quad \frac{\partial \hat{z}_2}{\partial \hat{z}_1} = 1 \quad \frac{\partial \hat{z}_3}{\partial \hat{z}_2} = \begin{cases} 0 & \hat{z}_2 \leq 0 \\ 1 & \hat{z}_2 > 0 \end{cases} \quad \frac{\partial \hat{z}_4}{\partial \hat{z}_3} = w \quad \frac{\partial a}{\partial \hat{z}_4} = a(1 - a)$$

z_1 is the weight matrix
produced by the
convolutional kernel z_1

$$\frac{\partial E}{\partial \hat{z}_1} = \frac{\partial E}{\partial \hat{z}_2} \frac{\partial \hat{z}_2}{\partial \hat{z}_1} \quad \frac{\partial E}{\partial \hat{z}_2} = \frac{\partial E}{\partial \hat{z}_3} \frac{\partial \hat{z}_3}{\partial \hat{z}_2} \quad \frac{\partial E}{\partial \hat{z}_3} = \frac{\partial E}{\partial \hat{z}_4} \frac{\partial \hat{z}_4}{\partial \hat{z}_3} \quad \frac{\partial E}{\partial \hat{z}_4} = \frac{\partial E}{\partial a} \frac{\partial a}{\partial \hat{z}_4}$$

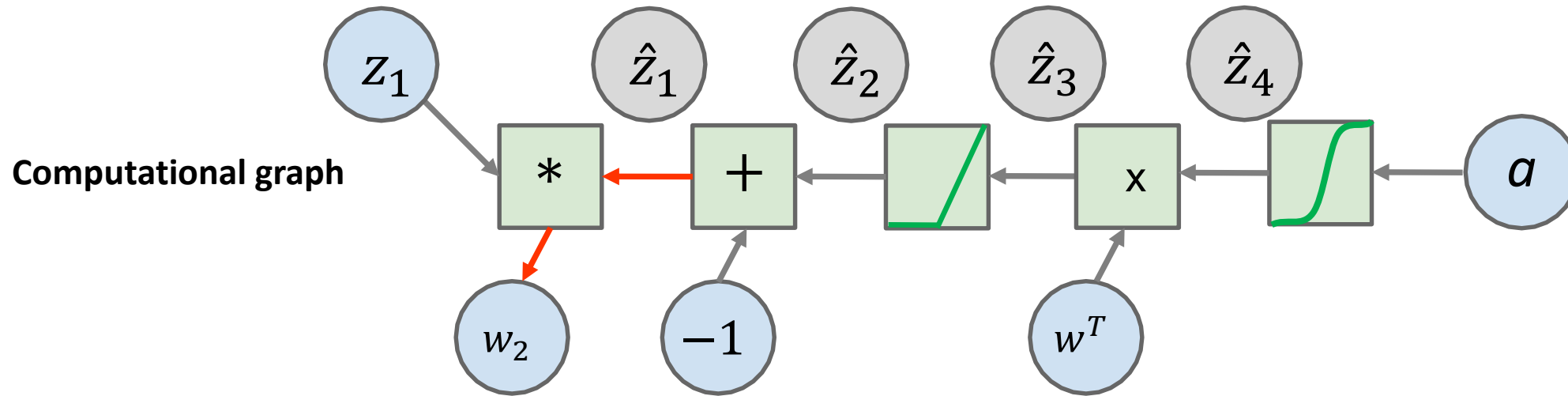


Gradient for each operator

$$\frac{\partial \hat{z}_1}{\partial w_2} = Z_1 \quad \frac{\partial \hat{z}_2}{\partial \hat{z}_1} = 1 \quad \frac{\partial \hat{z}_3}{\partial \hat{z}_2} = \begin{cases} 0 & \hat{z}_2 \leq 0 \\ 1 & \hat{z}_2 > 0 \end{cases} \quad \frac{\partial \hat{z}_4}{\partial \hat{z}_3} = w \quad \frac{\partial a}{\partial \hat{z}_4} = a(1 - a)$$

Z_1 is the weight matrix
produced by the
convolutional kernel z_1

$$\frac{\partial E}{\partial w_2} = \frac{\partial E}{\partial \hat{z}_1} \frac{\partial \hat{z}_1}{\partial w_2} \quad \frac{\partial E}{\partial \hat{z}_1} = \frac{\partial E}{\partial \hat{z}_2} \frac{\partial \hat{z}_2}{\partial \hat{z}_1} \quad \frac{\partial E}{\partial \hat{z}_2} = \frac{\partial E}{\partial \hat{z}_3} \frac{\partial \hat{z}_3}{\partial \hat{z}_2} \quad \frac{\partial E}{\partial \hat{z}_3} = \frac{\partial E}{\partial \hat{z}_4} \frac{\partial \hat{z}_4}{\partial \hat{z}_3} \quad \frac{\partial E}{\partial \hat{z}_4} = \frac{\partial E}{\partial a} \frac{\partial a}{\partial \hat{z}_4}$$



Gradient for each operator

$$\frac{\partial \hat{z}_1}{\partial w_2} = z_1 \quad \frac{\partial \hat{z}_2}{\partial \hat{z}_1} = 1 \quad \frac{\partial \hat{z}_3}{\partial \hat{z}_2} = \begin{cases} 0 & \hat{z}_2 \leq 0 \\ 1 & \hat{z}_2 > 0 \end{cases} \quad \frac{\partial \hat{z}_4}{\partial \hat{z}_3} = w \quad \frac{\partial a}{\partial \hat{z}_4} = a(1 - a)$$

z_1 is the weight matrix
produced by the
convolutional kernel z_1

Convolutional Neural Network

Consider a convolutional neural network that has three layers and outputs a scalar value *for binary classification*

$$z_1 = \text{ReLU}(w_1 * x) \text{ (conv with one kernel)}$$

$$z_2 = \text{ReLU}(w_2 * z_1 - 1) \text{ (conv with one kernel)}$$

$$a = \text{sigmoid}(w^T z_2) \text{ (fully connected)}$$

How can we improve the design of this network?

- Adding more filters to convolutional layers
- Make the network deeper (more convolutional and FC layers)
- Adding pooling operations
- All of the above

Convolutional Neural Network

Consider a convolutional neural network that has three layers and outputs a scalar value *for binary classification*

$$z_1 = \text{ReLU}(w_1 * x) \text{ (conv with one kernel)}$$

$$z_2 = \text{ReLU}(w_2 * z_1 - 1) \text{ (conv with one kernel)}$$

$$a = \text{sigmoid}(w^T z_2) \text{ (fully connected)}$$

How can we improve the design of this network?

- Adding more filters to convolutional layers
- Make the network deeper (more convolutional and FC layers)
- Adding pooling operations
- All of the above

Markov Decision Processes (MDPs)

Which of the following statement about MDP is NOT True?

- The reward function must output a scalar value
- The policy maps from states to actions
- The probability of next state can depend on current and previous states
- The solution of MDP is to find a policy that maximizes the cumulative rewards

Markov Decision Processes (MDPs)

Which of the following statement about MDP is NOT True?

- The reward function must output a scalar value
- The policy maps from states to actions
- The probability of next state can depend on current and previous states (*This violates the Markov property*)
- The solution of MDP is to find a policy that maximizes the cumulative rewards

Value Function

Consider an MDP with 2 states A, B and 2 actions: “stay” stays at the current state and “move” moves to the other state. Let r be the reward function such that $r(A) = 1$, $r(B) = 0$. Let γ be the discounting factor.

Let π : $\pi(A) = \pi(B) = \text{move}$ (“always move” policy). What is the value of $V^\pi(A)$ (the value function)

- 0
- $1 / (1 - \gamma)$
- $1 / (1 - \gamma^2)$
- 1

Value Function

Consider an MDP with 2 states A, B and 2 actions: “stay” stays at the current state and “move” moves to the other state. Let r be the reward function such that $r(A) = 1$, $r(B) = 0$. Let γ be the discounting factor.

Let π : $\pi(A) = \pi(B) = \text{move}$ (“always move” policy). What is the value of $V^\pi(A)$ (the value function)

- 0
 - $1 / (1 - \gamma)$
 - $1 / (1 - \gamma^2)$
 - 1
- Sequence: A, B, A, B, A, B,
Discounted rewards: $1, 0, \gamma^2, 0, \gamma^4, 0, \gamma^6, \dots$
Sum of discounted rewards: $1 + \gamma^2 + \gamma^4 + \gamma^6 + \dots = 1/(1 - \gamma^2)$
 $P(\text{sequence}) = 1$, $U(\text{sequence}) = 1/(1 - \gamma^2)$
 $V^\pi(A) = \sum P(\text{sequence}) U(\text{sequence}) = 1/(1 - \gamma^2)$

Value Iteration

Consider a grid world example with 2x2 grids, initial state s_0 and a goal state shown on the right.

The agent can move to top, bottom, left and right grid (if it exists). The move has a probability of 0.8 to reach the correct grid (incorrect move probability 0.2).

Assume we have a discount factor of 0.9, a reward of +1 at the goal state and a reward of -0.1 at all other states. What is the estimated utility of the top left grid after the second iteration?

$V_2?$	Goal
s_0	

- 0.8
- 0.72
- 0.702
- 0.602

Value Iteration

Consider a grid world example with 2x2 grids, initial state s_0 and a goal state shown on the right.

The agent can move to top, bottom, left and right grid (if it exists). The move has a probability of 0.8 to reach the correct grid (incorrect move probability 0.2).

Assume we have a discount factor of 0.9, a reward of +1 at the goal state and a reward of -0.1 at all other states. What is the estimated utility of the top left grid after the second iteration?

$V_2?$	Goal
s_0	

- 0.8
- 0.72
- 0.702
- 0.602 (details on next slide)



Setup

$V_2?$	Goal
s_0	

Initialization

0	0
0	0

1st iteration
(the rewards)

-0.1 	1.0
-0.1	 -0.1

2nd iteration

0.602	+1
-0.19	0.602

$$V_{i+1}(s) = r(s) + \gamma \max_a \sum_{s'} P(s'|s, a) V_i(s')$$

$$= -0.1 + 0.9 \max_a \sum_{s'} P(s'|s, a) V_i(s')$$

$$= -0.1 + 0.9 * (0.8 * 1.0 + 0.2 * (-0.1))$$

$$= 0.602$$

Best action is to move to right

Q1-1:

What is a key reason to bias in AI:

- A. Coincidence, there is no bias
- B. Added by human deliberately
- C. Training data are biased

Q1-1:

What is a key reason to bias in AI:

- A. Coincidence, there is no bias
- B. Added by human deliberately
- C. Training data are biased



Q1-2:

How can we solve the fairness problem?

- A. Remove bias from training data
- B. Design fair learning methods
- C. Both of the above

Q1-2:

How can we solve the fairness problem?

- A. Remove bias from training data
- B. Design fair learning methods
- C. Both of the above




Q1-3:

Which of the following is **wrong** about AI fairness?

- A. There could be any kind of bias in data, e.g., gender, race, etc.
- B. We can add fairness constraints to the learning model to impose fairness.
- C. There are only 2 kinds of constraints, depending on definitions of fairness.

Q1-3:

Which of the following is **wrong** about AI fairness?

- A. There could be any kind of bias in data, e.g., gender, race, etc.
- B. We can add fairness constraints to the learning model to impose fairness.
- C. There are only 2 kinds of constraints, depending on definitions of fairness. 


Q2-1:

In class, we've seen a video of Obama. Which is true about the video?

- A. It's a video of BBC interview.
- B. It's a private video of Obama leaked by hackers.
- C. It's a fake video.

Q2-1:

In class, we've seen a video of Obama. Which is true about the video?

- A. It's a video of BBC interview.
- B. It's a private video of Obama leaked by hackers.
- C. It's a fake video. 


Q2-2:

Which of the following is right?

- A. Fake images can have drawbacks, so a person can detect a fake image easily.
- B. Fake news detection is hard but not impossible.
- C. Fake things make life happier so we should generate as many as possible.

Q2-2:

Which of the following is right?

- A. Fake images can have drawbacks, so a person can detect a fake image easily.
- B. Fake news detection is hard but not impossible. 
- C. Fake things make life happier so we should generate as many as possible.

Q2-3:

How can we detect fake content?

- A. Each individual needs to be responsible
- B. Social media needs to be responsible
- C. Both of the above

Q2-3:

How can we detect fake content?

- A. Each individual needs to be responsible
- B. Social media needs to be responsible
- C. Both of the above




Q3-1:

Which of the following is correct about privacy?

- A. Privacy is a great concern in current big data era.
- B. Big tech companies can always protect individual privacy well enough.
- C. Both of above.

Q3-1:

Which of the following is correct about privacy?

- A. Privacy is a great concern in current big data era. 
- B. Big tech companies can always protect individual privacy well enough.
- C. Both of above.


Q3-2:

Which is right about protecting privacy?

- A. Simply deleting some data features from the whole dataset is sufficient.
- B. Individuals should have the right to be forgotten.
- C. Adding some noise to some data features is sufficient.

Q3-2:

Which is right about protecting privacy?

- A. Simply deleting some data features from the whole dataset is sufficient.
- B. Individuals should have the right to be forgotten. 
- C. Adding some noise to some data features is sufficient.

Q3-3:

Which of the following is **wrong** about privacy?

- A. We can use the popular framework of differential privacy.
- B. The data to be forgotten need to be unlearned if used for training.
- C. If we've done the above, there will be no concern about privacy.

Q3-3:

Which of the following is **wrong** about privacy?

- A. We can use the popular framework of differential privacy.
- B. The data to be forgotten need to be unlearned if used for training.
- C. If we've done the above, there will be no concern about privacy.

