Question 2: For a binary classification task with a deep neural network (containing at least one hidden layer) equipped with linear activation functions, which of the following loss functions guarantees a convex optimization problem? Justify your answer with a formal proof or a clear argument. (a) CE (b) MSE (c) Both (A) and (B) (d) None

Answer: For a binary classification task with a deep neural network containing at least one hidden layer and equipped with linear activation functions, neither cross-entropy (CE) loss nor mean squared error (MSE) guarantees a convex optimization problem.

Here is the reasoning:

1. Cross-Entropy Loss (CE): - In a binary classification task, binary cross-entropy loss (BCE) is commonly used. Although BCE is convex when used with sigmoid activation functions (which produce probabilities between 0 and 1), it is not guaranteed to be convex when used with linear activation functions. When linear activation functions are used in the output layer, BCE reduces to a form similar to MSE. However, the convexity of the resulting loss function depends on the specific problem setup and the data distribution. Therefore, CE does not guarantee convexity when linear activation functions are used.

2. Mean Squared Error (MSE): - Mean squared error is often used in regression tasks, but it is not suitable for binary classification tasks. In a binary classification context, MSE is not convex, especially when used with linear activation functions. MSE measures the squared difference between actual and predicted values, and it does not penalize misclassifications as effectively as cross-entropy loss does for binary classification tasks. Therefore, MSE does not guarantee convexity either.

Let's approach this problem methodically for both cross-entropy (CE) loss and mean squared error (MSE) loss functions:

1. Cross-Entropy Loss (CE): - For binary classification tasks, binary cross-entropy (BCE) loss is commonly used. The binary cross-entropy loss function is defined as:

$$L(y, \hat{y}) = -\left(y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})\right)$$

Here, y is the true binary label (0 or 1) and $\hat{y}$ is the predicted probability of belonging to the positive class. When linear activation functions are used in the output layer, the predicted probability $\hat{y}$ is a linear function of the input features. Substituting $\hat{y} = \mathbf{w}^T\mathbf{x} + b$ (where $\mathbf{w}$ is the weight vector and b is the bias) into the BCE loss formula results in a non-convex loss function. The non-convexity arises from the logarithm terms and the product $y \log(\hat{y})$ and $(1 - y) \log(1 - \hat{y})$. Therefore, binary cross-entropy loss with linear activation functions does not guarantee convex optimization.

2. Mean Squared Error (MSE): - The mean squared error loss function is defined as:

$$L(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

Here, y_i is the true target, and $\hat{y}_i$ is the predicted target. When linear activation functions are used in the output layer, the predicted target $\hat{y}_i$ is a linear function of the input features. Substituting $\hat{y}_i = \mathbf{w}^T\mathbf{x}_i + b$ into the MSE loss formula results in a non-convex loss function. The non-convexity arises from the squared term $(y_i - \hat{y}_i)^2$. Therefore, mean squared error loss with linear activation functions also does not guarantee convex optimization.

In both cases, the non-convexity arises from the non-linear terms in the loss functions, which are introduced by the logarithm in BCE loss and the square in MSE loss. Thus, neither BCE nor MSE loss functions guarantee convex optimization when used with linear activation functions in a deep neural network for binary classification.

Hence, the correct answer is (d) None.