**Assignment - 14**

1. Is it okay to initialize all the weights to the same value as long as that value is selected randomly using He initialization?

Ans: No, it is not recommended to initialize all the weights to the same value, even if that value is selected randomly using He initialization. Initializing all weights to the same value can lead to symmetry breaking issues, where neurons in the same layer learn identical features, reducing the capacity and expressiveness of the model.

1. Is it okay to initialize the bias terms to 0?

Ans: Yes, it is generally okay to initialize the bias terms to 0. Biases are used to shift the activation function, and initializing them to 0 is a common practice. However, in some cases, initializing biases to non-zero values might be beneficial to improve the learning dynamics, especially when dealing with dead neurons or vanishing gradients.

1. Name three advantages of the ELU activation function over ReLU.

Ans: ELU has non-zero gradients for negative inputs, which helps alleviate the vanishing gradient problem.

ELU can handle negative inputs gracefully, preventing the dying ReLU problem.

ELU has a smoother derivative compared to ReLU, which can lead to faster convergence and improved training stability.

1. In which cases would you want to use each of the following activation functions: ELU, leaky ReLU (and its variants), ReLU, tanh, logistic, and softmax?

Ans: ELU: Suitable when dealing with deep neural networks to alleviate the vanishing gradient problem and prevent dead neurons.

Leaky ReLU (and its variants): Useful when you want to prevent dying ReLU units and improve the robustness of the network to negative inputs.

ReLU: Widely used as a default activation function due to its simplicity and effectiveness, especially in shallow networks or CNNs.

Tanh: Appropriate for hidden layers in recurrent neural networks (RNNs) or when you need outputs in the range [-1, 1].

Logistic (Sigmoid): Suitable for binary classification tasks at the output layer, where you need probabilities between 0 and 1.

Softmax: Ideal for multi-class classification tasks at the output layer, where you need normalized class probabilities.

1. What may happen if you set the momentum hyperparameter too close to 1 (e.g., 0.99999) when using a MomentumOptimizer?

Ans: Setting the momentum hyperparameter too close to 1 can lead to unstable training behavior and oscillations in the optimization process. It may cause the optimizer to overshoot the minimum and oscillate around it, slowing down convergence or preventing it from converging altogether.

1. Name three ways you can produce a sparse model.

Ans: L1 Regularization: Introduce a penalty term in the loss function that encourages sparsity in the weights.

Dropout: Randomly set a fraction of the activations to zero during training, effectively creating a sparse representation.

Pruning: Identify and remove connections or weights with low magnitudes based on certain criteria, such as magnitude thresholding or iterative pruning techniques.

1. Does dropout slow down training? Does it slow down inference (i.e., making predictions on new instances)?

Ans: Dropout can slow down training because it introduces stochasticity and requires more iterations for convergence. However, it can prevent overfitting and improve generalization performance.

During inference, dropout is typically turned off, so it does not slow down inference. In fact, inference may be faster because dropout is not applied, and the model can make predictions more efficiently.