Lesson 8

Task 1: Implement a Basic CNN for CIFAR10 Classification
-Implement a Convolutional Neural Network using TensorFlow and Keras to classify handwritten digits from the CIFAR10 dataset. The network should include convolutional layers, activation functions, pooling layers, and at least one fully connected layer.
- Load and preprocess the CIFAR10 dataset.
- Build a CNN model with the following layers:
  - At least two convolutional layers with ReLU activation functions.
  - Pooling layers to reduce the spatial dimensions of the feature maps.
  - A Flatten layer before the fully connected layers.
  - Fully connected (Dense) layers, with a dropout layer to prevent overfitting.
  - An output layer with a softmax activation function for classification.
- Train the model on the training data and validate its performance on a validation set.
- Evaluate the model's performance on the test set and report the accuracy.

Task 2: Experiment with Different Activation Functions
-Explore the impact of different activation functions on the performance of a CNN trained on the MNIST dataset. Specifically, compare the ReLU activation function with at least one other (e.g., sigmoid, tanh).
- Utilize the CNN architecture from Task 1 but vary the activation functions in the convolutional and fully connected layers.
- Train each version of the model and monitor the training and validation accuracy and loss.
- Evaluate and compare the final performance of the models using the test set.
- Prepare a brief report discussing the observed differences in performance and training dynamics.

Task 3: Analyze the Effect of Pooling Layers
- Investigate how different pooling strategies affect the CNN's ability to classify images from the MNIST dataset. Compare the effects of max pooling versus average pooling.
- Using the original CNN architecture from Task 1, implement two versions of the model: one with max pooling layers and the other with average pooling layers.
- Train both models and record their training and validation performance across epochs.
- Evaluate both models on the test set, focusing on accuracy and computational efficiency (e.g., training time, number of parameters).
- Summarize the findings, highlighting how the choice of pooling layer impacts the model's performance and efficiency.

Interview Questions

Q1. Consider a CNN tasked with classifying scenes into categories such as beaches, forests, and cities. Discuss the role of pooling layers in this context. Would you choose max pooling or average pooling, and why? How does the choice of pooling strategy affect the network's ability

to generalize from training data to new, unseen images? What are the implications of this choice for the spatial resolution and the computational efficiency of the network?

Q2. You are optimizing a CNN that categorizes x-ray images into normal and various types of pathological findings. The network currently uses ReLU activation functions. However, you notice that some neurons are becoming inactive and not learning during training—a problem often referred to as "dying ReLU." How would you address this issue? Would you consider switching to another activation function or modifying the network architecture? Explain your reasoning and the expected impact on the network's learning capability and performance.