AI Homework

Ong Jung Yi (1006655)

1. *Code has included in the Python Jupyter Notebook*

**b) Experimenting with Activation Functions and Optimizers**

To evaluate the impact of activation functions and optimizers on the model's performance, the following experiments were conducted:

* **Three Activation Functions:** ReLU, Sigmoid, and Tanh.
* **Two Optimizers:** Stochastic Gradient Descent (SGD) and Adam.

|  |  |  |  |
| --- | --- | --- | --- |
| **Activation** | **Optimizer** | **Test Accuracy (%)** | **Rank** |
| ReLU | SGD | 79.26 | 5 |
| ReLU | Adam | 87.08 | 1 |
| Sigmoid | SGD | 10.00 | 6 |
| Sigmoid | Adam | 86.27 | 3 |
| Tanh | SGD | 81.83 | 4 |
| Tanh | Adam | 87.06 | 2 |

**Observations:**

* The Sigmoid activation function performed poorly, especially with SGD.
  + This is likely because Sigmoid activation function suffers from vanishing gradients, limiting weight updates and leading to ineffective learning.
  + When paired with SGD the training can stagnate resulting in low test accuracy
* Adam consistently outperformed SGD across all activation functions.

The best performance was from the **ReLU + Adam**, activation-optimizer configuration which was selected for further improvements in Part (c)

The worst-performing combination was Sigmoid + SGD (10.00%), confirming that Sigmoid is unsuitable for deep networks.

**c) Results of more epochs with Learning rate scheduler**

After identifying ReLU + Adam as the best combination, two strategies were explored to further enhance the model’s performance:

* Increase the Number of Training Epochs
* Implement a Learning Rate Scheduler

A StepLR scheduler to reduce the learning rate by half every 5 epochs. This prevents the model from overshooting minima and helps fine-tune weights in later epochs.

|  |  |  |  |
| --- | --- | --- | --- |
| **Activation** | **Optimizer** | **Test Accuracy (%)** | **Epoch** |
| ReLU | SGD | 88.15 | 5 |
| ReLU | SGD | 89.53 | 15 |

A graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of

AI-generated content may be incorrect.

The improved model achieved **89.53% accuracy**, surpassing the previous best of **87.75%**. These improvements were likely because more epochs allowed the model to learn deeper feature representation while the learning rate scheduling ensured stable training while updating the weights.

By increasing epochs and adding a learning rate scheduler, we improved test accuracy from 87.75% to 89.53%.