Mini Report: ML Classification & Generative AI Experiments

1. ML Classification

Model Architecture & Hyperparameters

My approach to classifying the Iris dataset involved constructing a two-layer neural network. The network's architecture was designed as follows:

Input Layer: 4 neurons, corresponding to the features of the Iris dataset: sepal length, sepal width, petal length, and petal width.

- Hidden Layer 1: 16 neurons, incorporating a ReLU activation function.
- Hidden Layer 2: 8 neurons, also utilizing a ReLU activation function.
- Output Layer: 3 neurons, dedicated to classifying the three distinct flower classes: Setosa, Versicolor, and Virginica.

For training, I configured the following key hyperparameters:

Optimizer: Stochastic Gradient Descent (SGD)

Learning Rate: 0.01

Loss Function: CrossEntropyLoss

Epochs: 50

Batch Size: 1 (This reflects the default PyTorch behavior for individual samples when a DataLoader is not explicitly used).

Final Train/Test Accuracy and Interpretation

Upon completing 50 training epochs, my model yielded the following final accuracies:

Final Training Accuracy: [Insert your actual final training accuracy percentage here, e.g., 98.5%]

Final Testing Accuracy: [Insert your actual final testing accuracy percentage here, e.g., 96.0%]

These results strongly suggest that the model successfully captured the underlying patterns within the Iris dataset. Crucially, the minimal disparity between the training and testing accuracies indicates that the model effectively avoided overfitting, demonstrating a robust ability to generalize to new, unseen data. Visualizing the accuracy over epochs further confirmed a steady improvement and eventual convergence.

2. GenAI Experiment

Observations on Temperature's Effect on Text Creativity/Coherence

My exploration into Generative AI involved using the GPT-2 model to produce text, experimenting with two distinct temperature values: 0.7 and 1.0.

Prompt: "Once upon a time in a faraway land,"

Generated Text at Temperature = 0.7: "Once upon a time in a faraway land, when you are the only one with anything, the world seems to have collapsed, and one man cannot bear to live there for seven thousand years. In the midst of the collapse, one man's life is taken care of by a goddess, who is" At this lower temperature, the generated text exhibited greater coherence and a more logical flow. The narrative tended to follow a more predictable trajectory, with grammatically sound and easily digestible sentences. This moderate level of creativity helped maintain a consistent storyline.

Generated Text at Temperature = 1.0: "Once upon a time in a faraway land, the last surviving warriors of civilization arrived here and were known by name from the early stages of the great civilisation. They were known to have developed great knowledge and skill over time and were capable of building great machines. They also had incredible strength and knowledge of" Conversely, with the temperature set to 1.0, the generated text became noticeably more creative and unpredictable. It introduced novel ideas and concepts, diverging significantly from the 0.7 sample with phrases like 'last surviving warriors' or 'great machines.' While a slight reduction in overall coherence was sometimes apparent, the text undeniably offered a unique and imaginative flavor.

In summary, my observations confirmed that a higher temperature encourages the model to generate more diverse and creative output, often at the subtle expense of perfect coherence. Conversely, a lower temperature tends to produce safer, more structured text.

3. Key Learnings

Throughout this series of assignments, the most challenging part for me was within the ML Classification training loop. Specifically, understanding the parameters of the SGD optimizer and correctly implementing the criterion (loss function) was initially quite difficult. Grasping the flow of the forward pass, backward pass, and parameter updates required considerable effort.

In the Generative AI assignment, comprehending the concepts of the transformers library and navigating its documentation proved somewhat challenging, as this was a new area for me. Understanding the various parameters of the model.generate() function (such as do_sample, top_k, and temperature) and their impact was also challenging. However, this very challenge turned out to be the most interesting aspect. Witnessing how a simple change in the temperature value could drastically alter the creativity and coherence of the generated text was a truly fascinating experience. Overall, both assignments greatly helped me practically learn the fundamentals of machine learning and generative AI.