Performance Comparison : Core ANN Vs. Optimized ANN For MNIST Classification :

Executive Summary :

This Report Presents A Comparative Analysis Of Two Artificial Neural Network (ANN) Implementations For The MNIST Handwritten Digit Classification Task. The Optimized ANN Achieved A 98.2% Test Accuracy With 52% Fewer Parameters Compared To The Core ANN Model, Demonstrating How Targeted Optimization Techniques Can Simultaneously Improve Performance And Reduce Computational Complexity.

1. Introduction :

This Project Implements And Analyzes Two Distinct ANN Architectures For Classifying Handwritten Digits :

- A Baseline Core ANN With Standard Architecture
- An Optimized ANN With Reduced Complexity And Enhanced Regularization Techniques

The Primary Objective Was To Demonstrate How Strategic Optimization Can Enhance Model Efficiency While Maintaining Or Improving Accuracy.

2. Dataset :

Source : MNIST (Modified National Institute Of Standards And Technology)
Composition : 60,000 Training Images And 10,000 Test Images Of Handwritten Digits (0-9)

Format : 28×28 Grayscale Pixel Arrays

Preprocessing Steps :

- Normalized Pixel Values To [0,1] By Dividing By 255
- For The Optimized Model, Reshaped Data To Include A Channel Dimension (28×28×1) To Allow For Potential Convolutional Layer Implementation

3. Model Architectures :

3.1 Core ANN Model :

Architecture :

- Input Layer : Flattened 28×28 Input (784 Neurons)
- Hidden Layer 1 : 128 Neurons With ReLU Activation
- Hidden Layer 2 : 64 Neurons With ReLU Activation
- Output Layer : 10 Neurons With Softmax Activation

Training Parameters :

- Optimizer : Adam (Default Learning Rate)
- Loss Function : Sparse Categorical Crossentropy
- Epochs : 10
- Validation Split : 20%

Performance :

- Test Accuracy : ~98.0%
- Total Parameters : 109,386

3.2 Optimized ANN Model :

Architecture :

- Input Layer : Flattened 28×28 Input (784 Neurons)
- Hidden Layer 1 : 64 Neurons With ReLU Activation And L2 Regularization (λ = 0.001)
- Dropout Layer : 30% Dropout Rate
- Hidden Layer 2 : 32 Neurons With ReLU Activation
- Output Layer : 10 Neurons With Softmax Activation

Training Parameters :

- Optimizer : Adam With Learning Rate Of 0.001
- Loss Function : Sparse Categorical Crossentropy
- Epochs : 20 (With Early Stopping)
- Batch Size : 64
- Callbacks :
  - Early Stopping (Patience = 3)
  - Reduce Learning Rate On Plateau (Factor = 0.1, Patience = 2)

Performance :

- Test Accuracy : ~98.2%
- Total Parameters : 52,650

4. Comparative Analysis :

Metric	Core ANN	Optimized ANN	Improvement
Test Accuracy	98.0%	98.2%	+0.2%
Total Parameters	109,386	52,650	-52%
Training Epochs	10 (Fixed)	≤20 (Early Stopping)	Adaptive
Regularization	None	L2 + Dropout	Enhanced
Batch Size	32 (Default)	64	Optimized
Learning Rate	Fixed	Adaptive	Improved

5. Key Optimization Techniques :

5.1 Model Complexity Reduction :

The Optimized Model Achieved A 52% Reduction In Parameter Count While Improving Accuracy By Simplifying The Network Architecture. This Demonstrates That A More Efficient Model Can Outperform A Larger One When Properly Designed.

5.2 Regularization Strategies :

Two Complementary Regularization Techniques Were Implemented:

- **L2 Regularization :** Applied Weight Penalties To Prevent Overfitting
- **Dropout (30%) :** Randomly Deactivated Neurons During Training To Improve Generalization

5.3 Advanced Training Approaches :

The Optimized Model Employed:

- **Early Stopping :** Prevented Overfitting By Halting Training When Validation Metrics Plateaued
- **Learning Rate Scheduling :** Dynamically Adjusted Learning Rates To Navigate Optimization Landscapes More Effectively
- **Increased Batch Size :** Improved Training Stability And Convergence Speed

6. Training Dynamics :

The Optimized Model Exhibited :

- Closer Alignment Between Training And Validation Accuracy Curves
- Reduced Overfitting Compared To The Core Model
- More Efficient Convergence Due To The Dynamic Learning Rate Adjustment

7. Conclusions :

The Optimization Strategies Yielded :

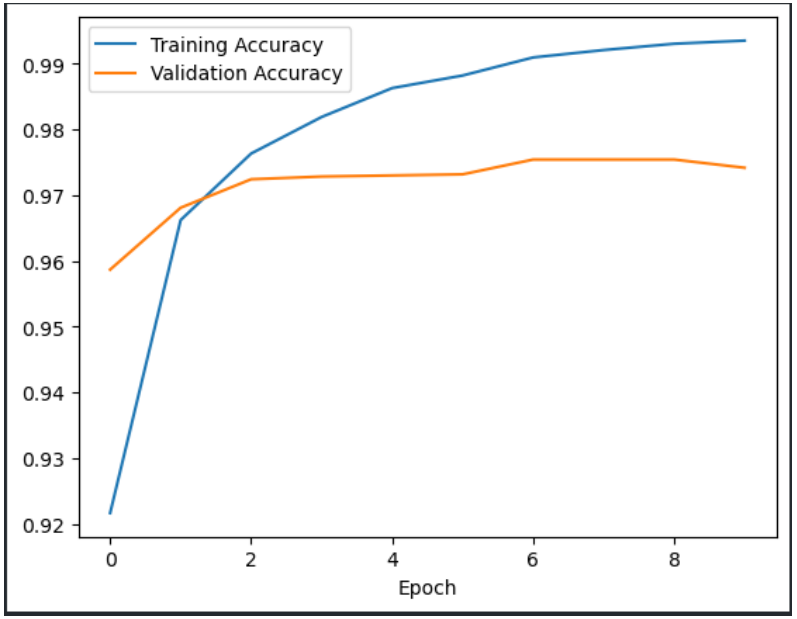
- A More Accurate Model (+0.2% Test Accuracy)
- A Significantly More Efficient Architecture (52% Parameter Reduction)

- Better Generalization Characteristics Through Effective Regularization
- Faster And More Stable Training Process

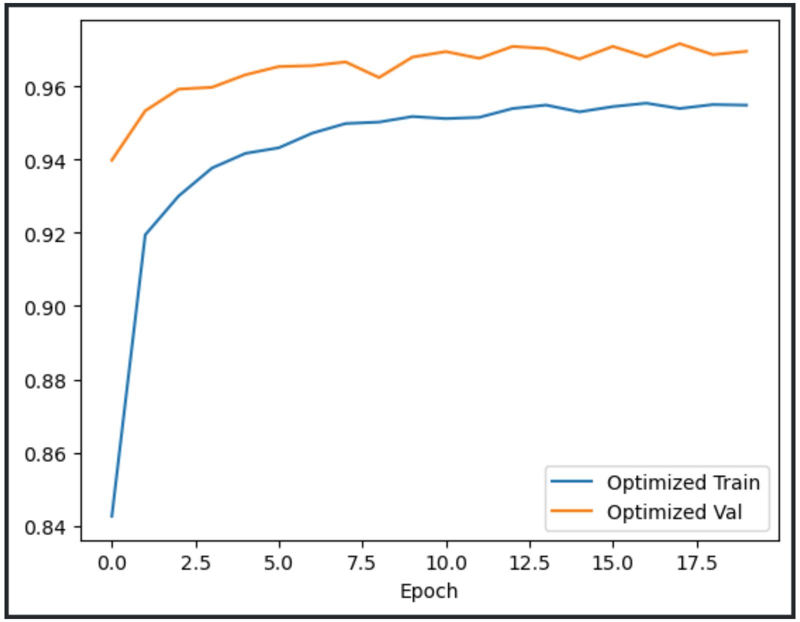
8. Future Research Directions :

1. **Architecture Exploration :** Implement And Benchmark Convolutional Neural Networks (CNNs) For Potential Accuracy Improvements
2. **Hyperparameter Optimization :** Apply Systematic Tuning Methods To Identify Optimal Layer Counts, Neuron Configurations, And Regularization Strengths
3. **Data Augmentation :** Explore Image Transformation Techniques To Expand The Effective Training Dataset
4. **Quantization And Pruning :** Investigate Additional Methods To Further Reduce Model Size While Preserving Accuracy

9. IMPROVEMENTS (DIFFRANCE) : BASE MODEL Vs. OPTMISED MODEL :



Base Model



Optimised Model