Fashion MNIST Classification Using a Neural Network Implemented From Scratch (NumPy Only)

Project Report

February 15, 2026

Abstract

This report presents the implementation and evaluation of a fully connected neural network trained entirely from scratch using NumPy for Fashion MNIST classification. The model includes manual forward propagation, backpropagation, dropout, L2 regularization, and the Adam optimizer. The final model achieves approximately 89% test accuracy without using convolutional layers or deep learning frameworks.

1 Introduction

The objective of this project is to classify Fashion MNIST images using a multi-layer perceptron (MLP) implemented purely in NumPy. Unlike framework-based implementations, all mathematical components including gradients and optimization are manually derived and coded.

Fashion MNIST is more challenging than digit MNIST because many classes are visually similar (e.g., Shirt vs T-shirt/top), making generalization harder for fully connected networks without spatial feature extraction.

2 Dataset and Preprocessing

2.1 Dataset

- **Name:** Fashion MNIST
- **Source:** torchvision dataset
- **Total samples:** 70,000
- **Image size:** 28×28 grayscale
- **Flattened dimension:** 784

2.2 Train / Validation / Test Split

- Training set: 50,000 samples
- Validation set: 10,000 samples
- Test set: 10,000 samples

Random seed used: 42.

2.3 Preprocessing

- Images flattened to 784-dimensional vectors
- Normalization using mean 0.286 and standard deviation 0.353
- No data augmentation applied

3 Model Architecture

3.1 Network Structure

$$784 \rightarrow 512 \rightarrow 256 \rightarrow 128 \rightarrow 10$$

- Hidden activations: ReLU
- Output activation: Stable softmax
- Dropout rate: 0.3 (hidden layers)

3.2 Initialization

- He initialization for hidden layers
- Xavier initialization for output layer

3.3 Mathematical Formulation

For each layer l:

$$Z^{(l)} = W^{(l)} A^{(l-1)} + b^{(l)}$$

$$A^{(l)} = \mathrm{ReLU}(Z^{(l)})$$

Output probabilities:

$$\hat{Y} = \mathrm{softmax}(Z^{(L)})$$

Loss function (cross-entropy):

$$\mathcal{L} = -\sum_{i=1}^{N} y_i \log(\hat{y}_i)$$

L2 regularization term:

$$\mathcal{L}_{\text{total}} = \mathcal{L} + \lambda \sum_{l} \|W^{(l)}\|_2^2$$

where $\lambda = 10^{-4}$.

4 Training Procedure

4.1 Optimization

- Optimizer: Adam
- Learning rate: 0.001
- $\beta_1 = 0.9$, $\beta_2 = 0.999$
- Batch size: 128
- Maximum epochs: 50
- Early stopping patience: 10 (based on validation accuracy)

Early stopping did not trigger. The best validation accuracy occurred at epoch 50.

4.2 Training Results

- Final training loss: ≈ 0.220
- Final training accuracy: $\approx 94.6\%$
- Final validation loss: ≈ 0.291
- Final validation accuracy: $\approx 90.1\%$
- Best checkpoint: Epoch 50 (validation accuracy 90.08%)
- Training time: approximately 11–12 minutes (CPU)

Training loss decreases steadily, while validation loss stabilizes around 0.29–0.31, indicating moderate but controlled overfitting due to dropout and L2 regularization.

5 Test Set Evaluation

5.1 Overall Performance

- Test Accuracy: **89.01%**
- Test Loss: **0.328**

The model generalizes to unseen data with approximately 89% classification accuracy.

5.2 Per-Class Metrics

Class	Precision	Recall	F1 Score
T-shirt/top	0.841	0.845	0.843
Trouser	0.995	0.966	0.980
Pullover	0.831	0.800	0.815
Dress	0.873	0.901	0.887
Coat	0.793	0.845	0.818
Sandal	0.982	0.953	0.968
Shirt	0.735	0.698	0.716
Sneaker	0.931	0.959	0.945
Bag	0.970	0.970	0.970
Ankle boot	0.952	0.964	0.958

Table 1: Per-Class Precision, Recall, and F1 Scores (Test Set)

5.3 Observations

- Strongest classes: Trouser, Sandal, Bag, Ankle boot (F1 $\approx$ 0.95–0.98)
- Weakest class: Shirt (F1 = 0.716)
- Major confusions:
  - Shirt $\leftrightarrow$ T-shirt/top
  - Shirt $\leftrightarrow$ Pullover / Coat
  - Coat $\leftrightarrow$ Pullover
  - Sandal $\leftrightarrow$ Sneaker

The MLP struggles most on visually similar apparel categories due to lack of spatial feature extraction.

6 Conclusion

A fully connected neural network implemented entirely in NumPy achieves approximately 89% test accuracy on Fashion MNIST. While performance is strong for a non-convolutional architecture, errors primarily occur among semantically and visually similar clothing categories. The results demonstrate that even without convolutional layers or automatic differentiation frameworks, carefully implemented MLPs can achieve solid generalization on moderately complex image classification tasks.