

Báo Cáo Tóm Tắt: Mô Hình DNN Phân Loại MNIST

Đề Bài

Xây dựng một mô hình mạng nơ-ron sâu (DNN) từ đầu bằng Python và NumPy, không sử dụng thư viện học sâu như TensorFlow hay PyTorch, để phân loại ảnh chữ số viết tay từ tập dữ liệu MNIST. Yêu cầu:

- Mạng gồm 3 lớp: lớp đầu vào, 1 lớp ẩn, và lớp đầu ra.
- Huấn luyện và đánh giá trên ít nhất 5 bộ siêu tham số khác nhau (batch_size, learning_rate, số nơ-ron lớp ẩn, hàm kích hoạt).
- Mỗi bộ siêu tham số được thử nghiệm ít nhất 5 lần, tính trung bình và độ lệch chuẩn của độ chính xác trên tập test.

Mô Hình

Mô hình DNN được thiết kế với cấu trúc đơn giản:

- **Lớp đầu vào:** 784 nơ-ron (tương ứng 28x28 pixel của ảnh MNIST, được chuẩn hóa về [0,1]).
- **Lớp ẩn:** Số nơ-ron thay đổi theo siêu tham số (16, 32, 64, 128, 256), sử dụng hàm kích hoạt ReLU hoặc Sigmoid.
- **Lớp đầu ra:** 10 nơ-ron (tương ứng 10 chữ số), sử dụng hàm softmax.
- Thuật toán huấn luyện: Gradient Descent với mini-batch, lan truyền xuôi (forward) và ngược (backward) được triển khai thủ công.

Siêu Tham Số

Năm bộ siêu tham số được thử nghiệm:

1. (32, 0.100, 64, relu)
2. (64, 0.010, 128, sigmoid)
3. (128, 0.001, 256, relu)
4. (64, 0.010, 128, sigmoid)
5. (64, 0.030, 128, relu)

Trong đó:

- batch_size: [32, 64, 128]

- learning_rate: [0.001, 0.010, 0.030, 0.100]
- hidden_neurons: [64, 128, 256]
- activation: [relu, sigmoid]

Mỗi bộ được huấn luyện trong 10 epoch, thử nghiệm 5 lần để tính độ chính xác trung bình và độ lệch chuẩn.

Kết Quả

Index	Batch Size	Learning Rate	Hidden Neurons	Activation	Mean Accuracy	Std Accuracy
0	32	0.100	64	relu	0.971914	0.000464
1	64	0.010	128	sigmoid	0.890086	0.000984
2	128	0.001	256	relu	0.835971	0.001356
3	64	0.010	128	sigmoid	0.891100	0.000576
4	64	0.030	128	relu	0.955257	0.001045

Bộ siêu tham số tốt nhất:

- Batch Size: 32
- Learning Rate: 0.100
- Hidden Neurons: 64
- Activation: relu
- Mean Accuracy: 0.9719

Nhận Xét

1. **Hiệu suất tốt nhất:** Bộ siêu tham số (32, 0.100, 64, relu) đạt độ chính xác trung bình cao nhất (0.9719) với độ lệch chuẩn thấp nhất (0.0005), cho thấy mô hình ổn định và hiệu quả.
2. **Ảnh hưởng của hàm kích hoạt:** ReLU thường cho kết quả tốt hơn Sigmoid (0.9719 và 0.9553 so với 0.8911 và 0.8901), có thể do khả năng giảm thiểu vanishing gradient.

3. **Learning rate:** Giá trị lớn hơn (0.100, 0.030) cho kết quả tốt hơn so với giá trị nhỏ (0.001), nhưng cần kết hợp với batch size phù hợp để tránh dao động.
4. **Số nơ-ron lớp ẩn:** Số nơ-ron ít (64) với cấu hình phù hợp đạt hiệu suất cao hơn số nơ-ron nhiều (256), có thể do mô hình đơn giản tránh được overfitting trên tập MNIST.
5. **Batch size:** Batch size nhỏ (32) hoạt động tốt hơn batch size lớn (128), có thể do cập nhật gradient thường xuyên hơn giúp hội tụ nhanh.

Kết luận: Mô hình đơn giản này đạt độ chính xác khá cao (97.19%) với siêu tham số tối ưu, chứng minh tính hiệu quả của DNN cơ bản trên bài toán MNIST, dù không sử dụng các thư viện tối ưu hóa.