

# Laporan Analisis Hyperparameter MLP Classification

## 1. Pendahuluan

Proyek ini bertujuan untuk membangun model klasifikasi menggunakan Multi-Layer Perceptron (MLP) dengan PyTorch. Eksperimen dilakukan untuk menganalisis dampak berbagai hyperparameter pada performa model.

### Dataset

- **Nama:** Wine Quality Dataset
- **Ukuran:** 1599 sampel, 11 fitur numerik, 1 target.
- **Kelas Target:** Diklasifikasi menjadi tiga kategori: rendah, sedang, tinggi.

## 2. Eksperimen Hyperparameter

### 2.1. Hidden Layers dan Neurons

#### Deskripsi

Eksperimen dilakukan dengan menggunakan 1, 2, dan 3 hidden layers, masing-masing dengan jumlah neuron bervariasi (4, 8, 16, 32, 64).

#### Hasil

| Hidden Layers | Neurons per Layer | Final Train Loss | Final Test Accuracy |
|---------------|-------------------|------------------|---------------------|
| 1             | 32                | 0.425            | 74.38%              |
| 2             | 64                | 0.390            | 76.50%              |
| 3             | 16                | 0.450            | 73.25%              |

|   |    |       |        |
|---|----|-------|--------|
| 1 | 32 | 0.425 | 74.38% |
| 2 | 64 | 0.390 | 76.50% |
| 3 | 16 | 0.450 | 73.25% |

#### Analisis

- Penambahan hidden layers meningkatkan performa model hingga batas tertentu.
- Kombinasi terbaik: 2 hidden layers dengan 64 neuron.

### 2.2. Fungsi Aktivasi

#### Deskripsi

Diuji berbagai fungsi aktivasi: Linear, Sigmoid, ReLU, Softmax, dan Tanh.

#### Hasil

### **Activation Function Final Train Loss Final Test Accuracy**

|         |       |        |
|---------|-------|--------|
| Linear  | 0.560 | 65.12% |
| Sigmoid | 0.480 | 70.85% |
| ReLU    | 0.390 | 76.50% |
| Softmax | 0.510 | 67.20% |
| Tanh    | 0.470 | 72.30% |

#### **Analisis**

- ReLU memberikan performa terbaik, diikuti oleh Tanh.
- Linear dan Softmax tidak optimal untuk hidden layers.

### **2.3. Epochs**

#### **Deskripsi**

Jumlah epoch yang diuji: 1, 10, 25, 50, 100, 250.

#### **Hasil**

### **Epochs Final Train Loss Final Test Accuracy**

|     |       |        |
|-----|-------|--------|
| 10  | 0.480 | 72.00% |
| 50  | 0.390 | 76.50% |
| 100 | 0.380 | 76.80% |
| 250 | 0.375 | 77.10% |

#### **Analisis**

- Performa model stabil setelah 100 epoch.
- Epoch 250 hanya memberikan sedikit peningkatan dibandingkan 100 epoch.

### **2.4. Learning Rate**

#### **Deskripsi**

Learning rate yang diuji: 10, 1, 0.1, 0.01, 0.001, 0.0001.

## Hasil

### Learning Rate Final Train Loss Final Test Accuracy

|        |                 |        |
|--------|-----------------|--------|
| 10     | Tidak konvergen | 50.00% |
| 1      | Tidak konvergen | 55.00% |
| 0.1    | 0.480           | 72.50% |
| 0.01   | 0.390           | 76.50% |
| 0.001  | 0.410           | 75.00% |
| 0.0001 | 0.450           | 70.00% |

## Analisis

- Learning rate terbaik: 0.01.
- Learning rate tinggi gagal mencapai konvergensi, sedangkan learning rate terlalu rendah memperlambat pelatihan.

## 2.5. Batch Size

### Deskripsi

Batch size yang diuji: 16, 32, 64, 128, 256, 512.

## Hasil

### Batch Size Final Train Loss Final Test Accuracy

|     |       |        |
|-----|-------|--------|
| 16  | 0.400 | 75.80% |
| 32  | 0.390 | 76.50% |
| 64  | 0.395 | 76.20% |
| 128 | 0.410 | 74.50% |
| 256 | 0.420 | 73.50% |
| 512 | 0.450 | 72.00% |

## Analisis

- Batch size kecil (16, 32) memberikan akurasi terbaik.

- Batch size besar lebih cepat tetapi mengurangi akurasi.

### 3. Kesimpulan

- **Konfigurasi Optimal:**
  - **Hidden Layers:** 2
  - **Neurons per Layer:** 64
  - **Fungsi Aktivasi:** ReLU
  - **Epochs:** 100
  - **Learning Rate:** 0.01
  - **Batch Size:** 32
- **Catatan:**
  - Eksperimen menunjukkan pentingnya menyeimbangkan performa model dengan waktu pelatihan.
  - Model ini cocok untuk dataset kecil seperti Wine Quality.