

Nama : Muhamad Hilmi Haidar

NIM : 1103213005

Berikut adalah analisis hasil eksperimen berdasarkan hyperparameter yang diuji:

1. Analisis Epochs

Jumlah epochs secara langsung memengaruhi kemampuan model untuk mencapai konvergensi. Pada epochs rendah (misalnya, 1 atau 10), rata-rata error cukup tinggi karena model tidak memiliki waktu yang cukup untuk belajar pola data. Dengan peningkatan jumlah epochs ke nilai menengah seperti 50, rata-rata error menurun signifikan. Namun, pada jumlah epochs yang lebih tinggi (100 atau lebih), penurunan rata-rata error mulai melambat, menunjukkan bahwa model telah mencapai konvergensi.

Poin-poin utama:

- **Epochs Rendah (1, 10):** Error tinggi karena model tidak cukup belajar.
- **Epochs Menengah (50):** Performa terbaik dengan error signifikan lebih rendah.
- **Epochs Tinggi (100, 250):** Error stabil tanpa peningkatan berarti.

Rekomendasi: Gunakan jumlah epochs antara **50–100** untuk mencapai hasil optimal tanpa menghabiskan waktu pelatihan secara berlebihan.

2. Analisis Hidden Layers

Jumlah dan ukuran hidden layers menentukan kapasitas model untuk menangkap pola kompleks. Dari hasil eksperimen, model dengan 16 hingga 32 neuron per layer memberikan performa terbaik, di mana rata-rata error cukup rendah. Ketika jumlah neuron terlalu kecil (4 atau 8), model mengalami underfitting, sedangkan penambahan neuron hingga 64 atau 128 menunjukkan diminishing returns, dengan penurunan error yang tidak signifikan tetapi waktu pelatihan meningkat.

Poin-poin utama:

- **Neuron Rendah (4, 8):** Model underfitting dengan error tinggi.
- **Neuron Menengah (16, 32):** Error terendah dan performa optimal.
- **Neuron Tinggi (64, 128):** Peningkatan kapasitas tidak memberikan pengurangan error yang signifikan.

Rekomendasi: Gunakan hidden layers dengan **16–32 neuron** untuk hasil terbaik.

3. Analisis Learning Rates

Learning rate menentukan kecepatan model dalam mencapai solusi optimal. Hasil eksperimen menunjukkan bahwa learning rate yang terlalu tinggi (10 atau 1) menyebabkan model gagal mencapai konvergensi, menghasilkan error tinggi. Sebaliknya, learning rate yang terlalu rendah (0.001 atau 0.0001) membuat model belajar terlalu lambat, sehingga tidak efisien. Nilai learning rate menengah, seperti 0.01 atau 0.1, memberikan hasil terbaik dengan error yang rendah dan stabilitas pelatihan yang baik.

Poin-poin utama:

- **Learning Rate Tinggi (10, 1):** Model tidak stabil dengan error tinggi.
- **Learning Rate Menengah (0.01, 0.1):** Error terendah dan pelatihan stabil.
- **Learning Rate Rendah (0.001, 0.0001):** Pelatihan terlalu lambat dengan error lebih tinggi.

Rekomendasi: Gunakan learning rate **0.01** untuk hasil optimal.

4. Analisis Activation Functions

Fungsi aktivasi memengaruhi bagaimana model menangkap hubungan non-linear. Dari hasil eksperimen, fungsi ReLU memberikan rata-rata error terendah, menunjukkan performa terbaik. Tanh menunjukkan performa yang baik, tetapi masih di bawah ReLU. Sebaliknya, Sigmoid memiliki error yang lebih tinggi, yang disebabkan oleh masalah vanishing gradient.

Poin-poin utama:

- **ReLU:** Memberikan error terendah dan performa terbaik.
- **Tanh:** Performa cukup baik tetapi tidak lebih baik dari ReLU.
- **Sigmoid:** Error tertinggi karena masalah vanishing gradient.

Rekomendasi: Gunakan fungsi aktivasi **ReLU** untuk hidden layers.

5. Analisis Batch Sizes

Ukuran batch memengaruhi stabilitas pembaruan gradien dan waktu pelatihan. Hasil menunjukkan bahwa batch size kecil (16 atau 32) memberikan rata-rata error lebih rendah karena pembaruan gradien dilakukan lebih sering. Namun, batch size besar (256 atau 512) mempercepat waktu pelatihan tetapi meningkatkan error karena model kehilangan granularitas dalam pembelajaran.

Poin-poin utama:

- **Batch Size Kecil (16, 32):** Error terendah dengan waktu pelatihan lebih lama.
- **Batch Size Menengah (64, 128):** Kompromi terbaik antara error dan waktu pelatihan.
- **Batch Size Besar (256, 512):** Waktu pelatihan cepat tetapi error meningkat.

Rekomendasi: Gunakan batch size **32 atau 64** untuk keseimbangan terbaik.

Kesimpulan

Konfigurasi optimal untuk MLP regresi pada dataset ini adalah:

- **Epochs:** 50–100
- **Hidden Layers:** 2–3 layers dengan 16–32 neuron per layer
- **Learning Rate:** 0.01
- **Activation Function:** ReLU
- **Batch Size:** 32 atau 64

Dengan konfigurasi ini, model mencapai error yang rendah tanpa mengorbankan efisiensi waktu pelatihan.