

Nama : Muhammad Rafindha Aslam

NIM : 1103213080

### **Laporan Analisis Hyperparameter Tuning**

ini berisi hasil eksperimen dengan berbagai kombinasi hyperparameter yang digunakan untuk melatih model. Hyperparameter yang diuji antara lain adalah jumlah lapisan tersembunyi (hidden\_layers), fungsi aktivasi (activation), laju pembelajaran (learning\_rate), dan ukuran batch (batch\_size). Tujuan dari eksperimen ini adalah untuk mengetahui bagaimana setiap kombinasi pengaturan hyperparameter mempengaruhi kerugian pada pelatihan (final\_train\_loss) dan pengujian (final\_test\_loss).

Kolom-kolom dalam dataset Hyperparameter Tuning:

1. hidden\_layers:

menunjukkan konfigurasi jumlah lapisan tersembunyi dalam jaringan. Sebagai contoh, nilai [4] berarti ada satu lapisan tersembunyi dengan 4 neuron. Tujuannya, menentukan kedalaman jaringan. Lebih banyak lapisan tersembunyi memungkinkan model untuk belajar representasi yang lebih kompleks.

2. activation:

Menunjukkan jenis fungsi aktivasi yang digunakan dalam lapisan-lapisan tersembunyi. Fungsi aktivasi seperti ReLU, Tanh, atau Sigmoid digunakan untuk menambah non-linearitas dalam model. Tujuannya untuk mengaktifkan neuron-neuron dalam lapisan tersembunyi berdasarkan input mereka. Ini memungkinkan model untuk belajar pola yang lebih kompleks.

3. Learning\_rate:

ini menunjukkan nilai laju pembelajaran (learning rate) yang digunakan dalam proses optimisasi. Nilai laju pembelajaran yang lebih tinggi akan menyebabkan pembaruan parameter yang lebih besar.

Beberapa nilai yang umum untuk learning rate:

- 0.1: Learning rate sedang, sering digunakan untuk model yang lebih sederhana.
- 0.01: Learning rate kecil, lebih sering digunakan untuk model yang lebih kompleks.
- 0.001: Learning rate sangat kecil, digunakan ketika model sudah cukup matang dan hanya membutuhkan sedikit penyesuaian.
- dan lebih tinggi: Learning rate yang tinggi, bisa menyebabkan model tidak stabil dan kesulitan dalam menemukan titik minimum.

4. `batch_size`:

Ukuran batch yang digunakan selama pelatihan. Ini menunjukkan berapa banyak data yang diproses dalam satu langkah pembaruan bobot (step). Dalam hyperparameter ini menggunakan jenis pelatihan Mini-batch Gradient Descent, yang menggunakan beberapa sampel dalam satu batch (misalnya, 32, 64, 128).

5. `final_train_loss`:

Nilai kerugian (loss) yang dihitung pada data pelatihan setelah proses pelatihan selesai. Nilai loss yang lebih rendah menunjukkan bahwa model lebih baik dalam mempelajari data pelatihan. Untuk Mengukur seberapa baik model dalam mempelajari data pelatihan.

6. `final_test_loss`:

Nilai kerugian yang dihitung pada data pengujian setelah pelatihan selesai. Ini menunjukkan seberapa baik model dalam menggeneralisasi data yang belum pernah dilihat. Tujuannya Mengukur performa model pada data yang tidak dilihat selama pelatihan.

**Best Hyperparameter Combination:**

- 1) `hidden_layers`     [32, 32]
- 2) `activation`        Sigmoid
- 3) `learning_rate`       0.1
- 4) `batch_size`         16
- 5) `final_train_loss`   0.007517
- 6) `final_test_loss`    0.002402

Name: 204, dtype: object