

Nama: Andi Cleopatra Maryam Jamila

Nim: 1103213071

Analisis Week 10

Bertujuan untuk mengevaluasi performa model Multi-Layer Perceptron (MLP) dalam mengklasifikasikan dataset Iris dengan menguji berbagai kombinasi hyperparameter, termasuk konfigurasi hidden layer, fungsi aktivasi, learning rate, jumlah epoch, dan batch size. Dengan pendekatan ini, diharapkan dapat ditemukan konfigurasi optimal yang memberikan akurasi terbaik pada data pengujian.

Dataset Iris digunakan sebagai data input, di mana fitur-fitur distandarisasi menggunakan StandardScaler dan target diolah dengan OneHotEncoder. Model MLP dikembangkan dengan kemampuan untuk mengubah jumlah hidden layer dan neuron, serta dapat dipilih fungsi aktivasi seperti Linear, ReLU, Sigmoid, dan Tanh. Eksperimen dilakukan dengan memvariasikan hyperparameter menggunakan kombinasi total 2592 konfigurasi, yang meliputi berbagai pilihan untuk learning rate (10, 1, 0.1, 0.01, 0.001, 0.0001), jumlah epoch (1, 10, 25, 50, 100, 250), dan batch size (16, 32, 64, 128, 256, 512).

Hasil menunjukkan bahwa konfigurasi dengan hidden layer [8, 16] dan fungsi aktivasi Linear mencapai performa terbaik dengan nilai test loss sangat rendah (6.06×10^{-14}). Learning rate 0.01 dan jumlah epoch 100 juga memberikan performa yang optimal. Batch size yang lebih kecil (16) berkontribusi pada test loss yang lebih rendah dibandingkan dengan ukuran batch yang lebih besar, yang menunjukkan bahwa pembelajaran yang lebih terperinci per batch membantu model memahami data dengan lebih baik.

Kesimpulan Model MLP sederhana dengan konfigurasi hidden layer [8, 16] dan fungsi aktivasi Linear efektif untuk mengklasifikasikan dataset Iris, dengan learning rate 0.01, 100 epoch, dan batch size 16 menunjukkan hasil terbaik. Pendekatan ini menunjukkan bahwa untuk dataset sederhana seperti Iris, konfigurasi dengan hidden layer yang lebih sederhana dan fungsi aktivasi Linear sudah cukup untuk mencapai performa yang sangat baik. Untuk dataset yang lebih kompleks, disarankan untuk mengeksplorasi fungsi aktivasi non-linear seperti ReLU atau menambahkan regularisasi seperti dropout untuk menghindari overfitting.