

Nama : Eko Putra Nugraha

NIM : 1103213212

Regression Model: MLP Regression

1. Hidden Layers

Pada eksperimen dengan berbagai konfigurasi *hidden layers*, model dengan lapisan tersembunyi sederhana, seperti [4] atau [8], menunjukkan performa yang kurang optimal. Hal ini terjadi karena model tidak memiliki kapasitas yang cukup untuk menangkap pola kompleks dalam data. Sebaliknya, konfigurasi dengan lapisan tersembunyi lebih besar, seperti [16, 16] atau [32, 32, 32], mampu memberikan hasil yang lebih baik. Namun, peningkatan jumlah lapisan dan neuron tidak selalu memberikan hasil signifikan setelah titik tertentu, menunjukkan bahwa terlalu banyak kompleksitas dapat menyebabkan overfitting.

2. Fungsi Aktivasi

Fungsi aktivasi yang diuji termasuk *ReLU*, *Sigmoid*, *Tanh*, dan *Softmax*. Fungsi *ReLU* mendominasi hasil terbaik karena kemampuannya dalam menghindari masalah *vanishing gradient*. Fungsi *Sigmoid* dan *Tanh* menghasilkan performa yang cukup baik tetapi membutuhkan waktu pelatihan lebih lama. *Softmax* tidak cocok untuk tugas regresi, karena dirancang untuk klasifikasi, sehingga menghasilkan performa terburuk. Secara keseluruhan, *ReLU* terbukti menjadi fungsi aktivasi yang paling efektif untuk model ini.

3. Epochs

Eksperimen menunjukkan bahwa jumlah epoch yang rendah, seperti 1 atau 10, menghasilkan model yang underfitting dengan MSE yang tinggi. Jumlah epoch sedang, seperti 50 hingga 100, memberikan hasil yang signifikan karena model memiliki waktu pelatihan yang cukup untuk mempelajari pola data. Pelatihan dengan 250 epoch menghasilkan sedikit peningkatan, tetapi dengan risiko overfitting, terutama pada konfigurasi model yang sangat kompleks.

4. Learning Rate

Learning rate besar, seperti 10 atau 1, menunjukkan ketidakstabilan model selama pelatihan, menghasilkan konvergensi yang buruk. Nilai *learning rate* sedang, seperti 0.01 atau 0.001, memberikan performa terbaik, karena memastikan pelatihan yang stabil dan konvergensi cepat. Sebaliknya, nilai sangat kecil, seperti 0.0001, menyebabkan pelatihan berjalan lambat tanpa perbaikan signifikan pada hasil akhir.

5. Batch Size

Ukuran batch yang kecil, seperti 16 atau 32, memungkinkan pelatihan lebih granular, tetapi memakan waktu lebih lama. Sebaliknya, batch besar, seperti 256 atau 512, mempercepat pelatihan tetapi kehilangan beberapa detail penting dalam data. Ukuran batch sedang, seperti 64 atau 128, memberikan keseimbangan terbaik antara waktu pelatihan dan akurasi model.

Kesimpulan

Hasil ini menunjukkan bahwa optimalisasi hyperparameter yang cermat sangat penting untuk meningkatkan performa model, terutama pada dataset yang kompleks. Visualisasi hasil menunjukkan bahwa model dapat memprediksi dengan baik nilai target, dengan MSE yang rendah pada data uji.

Classification

Dalam eksperimen ini, tujuan utama adalah untuk mengevaluasi pengaruh berbagai kombinasi hyperparameter terhadap performa model Multi-Layer Perceptron (MLP) yang diterapkan pada dataset Abalone. Kami menguji beberapa hyperparameter utama, yaitu jumlah hidden layers, fungsi aktivasi, learning rate, batch size, dan jumlah epoch. Untuk mempercepat pencarian, eksperimen dilakukan dengan pendekatan randomized search, di mana kami memilih kombinasi acak dari berbagai nilai hyperparameter dan mengevaluasi hasilnya berdasarkan akurasi model.

Berdasarkan hasil eksperimen, pengaruh jumlah hidden layers terhadap performa model sangat signifikan. Model yang menggunakan lebih dari satu hidden layer, seperti kombinasi [16, 32, 64], menunjukkan performa yang lebih baik daripada model dengan satu hidden layer seperti [4] atau [8]. Hal ini menunjukkan bahwa model dengan lebih banyak layer tersembunyi memiliki kapasitas lebih besar untuk menangkap pola-pola yang lebih kompleks dalam data. Sebaliknya, model dengan sedikit layer tersembunyi cenderung memiliki akurasi yang lebih rendah karena keterbatasan dalam menangani kompleksitas data.

Selain jumlah hidden layers, fungsi aktivasi juga berperan penting dalam menentukan performa model. Fungsi aktivasi ReLU terbukti memberikan hasil yang paling optimal di antara semua yang diuji. ReLU membantu menghindari masalah vanishing gradient yang sering terjadi pada fungsi aktivasi lain seperti Sigmoid atau Tanh, yang membuat pelatihan model dengan kedalaman lebih besar menjadi lebih lambat dan sulit konvergen. Fungsi aktivasi Identity tidak memberikan perubahan signifikan pada output neuron, sehingga model dengan fungsi ini cenderung menghasilkan akurasi yang lebih rendah karena tidak memberikan non-linearity yang dibutuhkan untuk pembelajaran yang efektif.

Pengaruh learning rate terhadap performa model juga cukup jelas. Nilai learning rate yang lebih kecil, seperti 0.001 atau 0.0001, cenderung menghasilkan akurasi yang lebih baik, terutama pada model yang dilatih dalam jumlah epoch yang lebih banyak. Learning rate yang lebih kecil memungkinkan model untuk melakukan konvergensi dengan lebih stabil, menghindari lonjakan besar dalam pembaruan bobot yang dapat menyebabkan model tidak stabil atau gagal menemukan solusi yang optimal. Sebaliknya, learning rate yang besar, seperti 10 atau 1, sering kali mengarah pada konvergensi yang cepat, tetapi tidak mencapai akurasi terbaik karena pembaruan yang terlalu besar menyebabkan model kehilangan titik optimal.

Batch size juga mempengaruhi kinerja model, dengan batch size yang lebih besar seperti 128, 256, atau 512 cenderung menghasilkan hasil yang lebih baik. Penggunaan batch size yang besar memberi model lebih banyak data untuk diproses dalam setiap iterasi, yang membantu mempercepat konvergensi dan meningkatkan kestabilan pelatihan. Sebaliknya, batch size yang kecil seperti 16 atau 32 dapat memperlambat proses pelatihan karena model perlu melakukan lebih banyak pembaruan bobot untuk setiap epoch, yang mungkin menyebabkan fluktuasi dalam gradien dan mengurangi akurasi.

Terakhir, jumlah epoch berpengaruh besar terhadap performa model. Model dengan jumlah epoch yang lebih tinggi, seperti 100 atau 250, cenderung menghasilkan akurasi yang lebih tinggi. Hal ini menunjukkan bahwa model membutuhkan lebih banyak waktu untuk belajar dan mengoptimalkan bobotnya agar dapat mencapai performa terbaik. Model yang hanya dilatih selama 1 atau 10 epoch biasanya memiliki akurasi yang lebih rendah karena belum cukup waktu untuk mempelajari pola dalam data secara mendalam.

Secara keseluruhan, kombinasi hidden layers yang lebih banyak (seperti [16, 32, 64]), fungsi aktivasi ReLU, learning rate yang lebih kecil (seperti 0.001), batch size besar (seperti 128 atau 256), dan epoch lebih banyak (seperti 100 atau 250) cenderung memberikan hasil yang lebih baik dalam hal akurasi model. Namun, perlu dicatat bahwa pencarian hyperparameter ini masih bisa diperbaiki lebih lanjut dengan mencoba variasi yang lebih halus pada setiap parameter, seperti memilih nilai learning rate yang lebih spesifik atau melakukan regularisasi untuk mencegah overfitting.

Berdasarkan temuan ini, untuk implementasi selanjutnya, disarankan untuk melakukan fine-tuning lebih lanjut pada hyperparameter seperti learning rate dan jumlah hidden layers, serta bereksperimen dengan teknik regularisasi atau early stopping untuk mencegah model dari overfitting, terutama jika model diuji dengan dataset yang lebih besar atau lebih kompleks. Pencarian hyperparameter yang lebih mendalam menggunakan pendekatan seperti Bayesian Optimization atau Hyperband juga bisa dipertimbangkan untuk meningkatkan performa secara lebih efisien.