

## ***Analisa***

---

1. Jika menggunakan model MLP dengan 3 hidden layer (256-128-64) menghasilkan underfitting pada dataset ini, modifikasi apa yang akan dilakukan pada arsitektur? Jelaskan alasan setiap perubahan dengan mempertimbangkan bias-variance tradeoff!

Jika model MLP dengan 3 hidden layers (256-128-64) mengalami underfitting, Anda bisa:

- Menambah jumlah neuron atau lapisan tersembunyi untuk meningkatkan kapasitas model dalam menangkap pola data. Misalnya, bisa mencoba lapisan seperti 512-256-128 untuk memberi model lebih banyak kapasitas.
- Mengurangi regularisasi jika terlalu kuat, karena regularisasi yang berlebihan bisa membatasi model untuk belajar dari data.

Perubahan ini akan menurunkan bias model dan memberikan model lebih banyak kemampuan untuk belajar, meskipun perlu memantau agar tidak menyebabkan overfitting.

2. Selain MSE, loss function apa yang mungkin cocok untuk dataset ini? Bandingkan kelebihan dan kekurangannya, serta situasi spesifik di mana alternatif tersebut lebih unggul daripada MSE!

Selain MSE, dua alternatif loss function yang bisa digunakan adalah MAE dan Huber Loss:

- MAE (Mean Absolute Error) lebih tahan terhadap outliers karena tidak mengubah kesalahan besar secara kuadrat. Namun, MAE cenderung lebih sulit dikendalikan karena menghasilkan gradien lebih besar untuk kesalahan kecil.
- Huber Loss menggabungkan keuntungan dari MSE dan MAE, memberikan penalti yang lebih ringan terhadap outliers namun tetap menjaga kestabilan untuk kesalahan kecil. Cocok ketika ada beberapa outliers tetapi juga ingin mempertahankan sensitivitas model terhadap kesalahan kecil.

3. Jika salah satu fitur memiliki range nilai 0-1, sedangkan fitur lain 100-1000, bagaimana ini memengaruhi pelatihan MLP? Jelaskan mekanisme matematis (e.g., gradien, weight update) yang terdampak!

fitur memiliki rentang nilai yang sangat berbeda (misalnya 0-1 dan 100-1000), maka saat backpropagation, fitur dengan nilai besar (100-1000) akan memiliki gradien yang lebih besar dan dominan selama pembaruan bobot.

- Hal ini akan menyebabkan ketidakstabilan dalam pelatihan, karena pembaruan bobot untuk fitur besar akan lebih besar daripada untuk fitur kecil.
- Solusinya adalah dengan normalisasi atau standarisasi data, agar semua fitur berada dalam skala yang sama, yang akan membuat pembaruan bobot lebih stabil.

4. Tanpa mengetahui nama fitur, bagaimana Anda mengukur kontribusi relatif setiap fitur terhadap prediksi model? Jelaskan metode teknikal (e.g., permutation importance, weight analysis) dan keterbatasannya!

mengukur kontribusi relatif setiap fitur meskipun tanpa mengetahui nama fitur, Anda dapat menggunakan Permutation Importance atau SHAP:

- Permutation Importance mengukur penurunan performa model ketika nilai suatu fitur diacak. Fitur yang paling mempengaruhi model akan menyebabkan penurunan performa yang signifikan.
- SHAP memberikan penjelasan lebih detil dan adil dengan mengukur kontribusi setiap fitur dalam konteks model secara keseluruhan, meskipun lebih kompleks dan membutuhkan lebih banyak komputasi.

Keterbatasannya yaitu Permutation Importance membutuhkan waktu komputasi yang cukup lama, terutama jika dataset besar. SHAP lebih mahal dari segi komputasi dan lebih rumit.

5. Bagaimana Anda mendesain eksperimen untuk memilih learning rate dan batch size secara optimal? Sertakan analisis tradeoff antara komputasi dan stabilitas pelatihan!

Dalam memilih learning rate dan batch size yang optimal, lakukan eksperimen dengan grid search atau random search:

- Learning rate yang terlalu tinggi bisa menyebabkan pelatihan tidak stabil dan model melompat-lompat, sementara yang terlalu rendah akan membuat pelatihan sangat lambat.
- Batch size kecil memberi pembaruan bobot yang lebih sering dan stabil tetapi lebih lambat dalam komputasi. Batch size besar mempercepat komputasi, namun bisa menyebabkan overfitting karena model tidak sering memperbarui bobot.
- Tradeoff: Batch size kecil lebih baik untuk stabilitas dan generalisasi, tetapi memperlambat pelatihan. Sedangkan batch size besar mempercepat pelatihan tetapi bisa mengurangi kemampuan model dalam generalisasi.