Nama : Faiz Hibatullah
Kelas : TK-45-G05
NIM : 1103210172

Analisa Regresi Model

1. Jika model linear regression atau decision tree mengalami underfitting pada dataset ini, strategi apa yang akan digunakan untuk meningkatkan performanya? Bandingkan setidaknya dua pendekatan (misal: transformasi fitur, penambahan fitur, atau perubahan model ke algoritma yang lebih kompleks), dan jelaskan bagaimana setiap Solusi mengaruhi bias-vatiance tradeoff!
2. Selain MSE, jelaskan dua alternatif loss function untuk masalah regresi (misal: MAE, Huber loss) dan bandingkan keunggulan serta kelemahannya. Dalam scenario apa setiap loss function lebih cocok digunakan? (Contoh: data dengan outlier, distribusi target non-Gaussian, atau kebutuhan interpretasi model).
3. Tanpa mengetahui nama fitur, metode apa yang dpat digunakan untuk mengukur pentingnya setiap fitur dalam model? Jelaskan prinsip teknikal di balik metode tersebut (misal: koefisien regresi, feature importance berdasarkan impurity reduction) serta keterbatasannya!
4. Bagaimana mendesain eksperimen untuk memilih hyperparameter optimal (misal: learning rate untuk SGDRegressor, max_depth untuk Decision Tree) pada dataset ini? Sertakan analisis trade off antara komputasi, stabilitas pelatihan , dan generalisasi model!
5. Jika menggunakan model linear regression dan residual plot menunjukkan pola non-linear serta heteroskedastisitas, Langkah-langkah apa yang akan diambil? (contohnya: Transformasi data/ubah model yang akan diapakai)

Jawab:

1. Saat model linear regression atau decision tree mengalami underfitting, model tersebut terlalu sederhana untuk menangkap pola dalam data. Ada dua strategi utama untuk mengatasinya:
    - Rekayasa Fitur: Teknik ini meningkatkan kapasitas model tanpa mengubah algoritma dasarnya. Untuk linear regression, kita dapat menambahkan fitur polinomial (x^2, x^3), membuat term interaksi antar fitur, atau melakukan transformasi non-linear. Pendekatan ini mempertahankan interpretabilitas model sambil mengurangi bias, meski berisiko meningkatkan variance jika tidak diregularisasi dengan baik.
    - Peningkatan Kompleksitas Model: Untuk linear regression, ini berarti beralih ke model seperti regresi polinomial, SVR dengan kernel non-linear, atau ensemble methods. Untuk decision tree, kita bisa meningkatkan max_depth, mengurangi min_samples_split, atau menurunkan threshold impurity. Pendekatan ini secara langsung mengurangi bias tetapi berisiko overfitting jika tidak divalidasi dengan tepat.

    Dari perspektif bias-variance tradeoff, rekayasa fitur umumnya lebih baik untuk mempertahankan interpretabilitas sambil meningkatkan performa. Namun, jika hubungan sangat kompleks, peningkatan kompleksitas model mungkin diperlukan meski dengan konsekuensi model lebih sulit diinterpretasi.

2. Mean Absolute Error (MAE): Dihitung sebagai rata-rata nilai absolut selisih antara nilai prediksi dan nilai sebenarnya. MAE lebih tahan terhadap outlier dibanding MSE karena tidak menghukum kesalahan besar secara kuadratik. Loss function ini mengoptimalkan nilai median dan memberikan metrik dalam unit yang sama dengan variabel target, sehingga mudah diinterpretasi. Kelemahannya adalah tidak diferensiabel di titik nol, yang bisa menghambat algoritma optimasi berbasis gradien. MAE ideal untuk data dengan outlier signifikan atau distribusi error non-Gaussian.
   Huber Loss: Menggabungkan karakteristik MSE dan MAE melalui parameter δ. Untuk error kecil ($|y-\hat{y}| \leq \delta$), berfungsi seperti MSE, dan untuk error besar, seperti MAE. Ini memberikan keseimbangan antara diferensiabilitas MSE dan ketahanan MAE terhadap outlier. Huber Loss memerlukan penyetelan parameter tambahan dan komputasi yang lebih kompleks, namun sangat berguna untuk dataset dengan beberapa outlier atau saat kita membutuhkan optimasi berbasis gradien yang stabil dengan ketahanan terhadap outlier.

3. Permutation Importance: Metode ini mengukur pentingnya fitur dengan mengacak nilai masing-masing fitur dan mengamati dampaknya pada performa model. Fitur yang pengacakannya menyebabkan penurunan performa terbesar dianggap paling penting. Keunggulannya adalah universalitas—dapat diterapkan pada hampir semua model machine learning. Kelemahannya termasuk komputasi intensif dan potensi underestimasi pentingnya fitur yang berkorelasi tinggi.
   Feature Importance dari Struktur Model: Untuk linear regression, koefisien terstandarisasi memberikan ukuran langsung pentingnya fitur. Untuk decision tree, pentingnya fitur diukur berdasarkan pengurangan impurity yang dihasilkan saat fitur digunakan untuk splitting. Metode ini bergantung pada jenis model dan memiliki keterbatasan seperti bias terhadap fitur kardinalitas tinggi (pada tree) atau sensitivitas terhadap multikolinearitas (pada model linear).

4. Optimasi hyperparameter efektif melibatkan tiga tahap utama:
   - Grid Search: Evaluasi sistematis semua kombinasi dari serangkaian nilai diskrit untuk setiap hyperparameter. Metode ini memberikan pemahaman komprehensif tentang landscape parameter tetapi dengan biaya komputasi tinggi.
   - Randomized Search: Mengambil sampel acak dari distribusi parameter, lebih efisien daripada Grid Search, terutama untuk ruang dimensi tinggi. Dengan iterasi yang sama, metode ini sering menemukan parameter yang lebih baik karena mengalokasikan lebih banyak sumber daya untuk parameter yang lebih berpengaruh.
   - Bayesian Optimization: Menggunakan informasi dari evaluasi sebelumnya untuk membangun model tentang hubungan antara hyperparameter dan performa, kemudian memilih titik evaluasi berikutnya yang menjanjikan. Paling efisien untuk menemukan parameter optimal.

5. Ketika residual plot menunjukkan pola non-linear dan heteroskedastisitas, beberapa pendekatan dapat diterapkan:
   - Transformasi Variabel Target: Mengubah skala variabel dependen (seperti log, akar kuadrat, atau Box-Cox) dapat membuat hubungan lebih linear dan menstabilkan varians error. Transformasi logaritmik efektif untuk distribusi miring kanan, namun memerlukan transformasi balik untuk interpretasi hasil.
   - Transformasi Fitur: Memodifikasi variabel independen seperti penerapan transformasi logaritmik untuk hubungan eksponensial atau transformasi polinomial

untuk hubungan melengkung. Pendekatan ini mempertahankan skala asli variabel target tetapi memerlukan pemahaman domain yang baik.

- Model Linear yang Lebih Fleksibel: Generalized Linear Models dengan fungsi link yang sesuai atau Weighted Least Squares untuk mengatasi heteroskedastisitas. Pendekatan ini mempertahankan interpretabilitas model linear sambil menangani pelanggaran asumsi.
- Model Non-Linear: Beralih ke algoritma seperti Random Forest, Gradient Boosted Trees, atau SVR dengan kernel non-linear untuk hubungan yang sangat kompleks. Model-model ini biasanya memberikan performa prediktif lebih baik dengan tradeoff interpretabilitas yang lebih rendah.