

Nama : Ardi Kamal Karima
NIM : 301230023
Kelas : 5C/IF
Program Studi : S1 Teknik Informatika
Mata Kuliah : Praktikum Machine Learning
Dosen Pengajar : Mohammad Bayu Anggara, S.Kom., M.Kom.
Tugas : Tugas Pendahuluan (TP) 14 Perancangan Studi Kasus Ensemble Learning dengan XGBoost

A. KONSEP INTI

Tonton video atau baca materi singkat mengenai **XGBoost** dan **Ensemble Learning** (Kata kunci: "*Ensemble Learning Explained*", "*Bagging vs Boosting*", atau "*XGBoost Mathematical Intuition*"). Setelah itu, tuliskan rangkuman singkat (4-6 kalimat) dengan bahasa Anda sendiri untuk menjawab pertanyaan berikut:

1. Bagging vs. Boosting: Jelaskan perbedaan mendasar antara kedua teknik ensemble ini.
 - Hint: Random Forest membangun pohon secara paralel (bersamaan), sedangkan XGBoost membangun pohon secara sekuensial (berurutan). Apa dampak perbedaan ini terhadap cara mereka mengurangi error?
2. Belajar dari Kesalahan (Residuals): Dalam XGBoost, pohon kedua tidak dilatih pada target asli, melainkan pada residual dari pohon pertama. Jelaskan dengan bahasa sederhana apa maksudnya? (Gunakan analogi murid yang belajar memperbaiki jawaban ujian yang salah).
3. Mengapa "Extreme"?: Apa yang membuat XGBoost lebih spesial dibandingkan algoritma Gradient Boosting biasa (GBM)? Sebutkan minimal 2 fitur unggulannya (misal: Regularisasi atau Kecepatan).

JAWABAN

Setelah menonton video terkait materi *Ensemble (Boosting, Bagging, and Stacking) in Machine Learning: Easy Explanation for Data Scientists* di salah-satu channel di Youtube dengan link video: <https://youtu.be/sN5ZcJLDMaE?si=axu8J4JBkIBB1LPQ>

1. Bagging (misalnya Random Forest) melatih banyak pohon keputusan secara paralel, lalu menggabungkan prediksi mereka (misalnya dengan rata-rata atau voting) untuk mengurangi variansi model. Sebaliknya, Boosting (misalnya XGBoost) melatih pohon secara berurutan, di mana setiap pohon baru berfokus memperbaiki kesalahan (residual) dari model sebelumnya. Perbedaan ini membuat bagging lebih efektif dalam mengurangi variansi, sedangkan boosting secara bertahap menurunkan bias model dengan memperbaiki kesalahan satu per satu.
2. Dalam XGBoost, pohon kedua dilatih bukan pada target asli, melainkan pada *residual* (selisih antara prediksi pohon pertama dan label sebenarnya). Artinya, pohon kedua berfokus memperbaiki kesalahan yang dibuat oleh pohon pertama. Analogi sederhananya, seperti seorang siswa yang memeriksa kembali jawaban ujian yang salah lalu mempelajari kesalahan tersebut agar di kesempatan berikutnya bisa menjawab dengan benar.
3. XGBoost disebut “Extreme” karena menambahkan fitur unggulan di atas GBM biasa. Misalnya, ia menggunakan regularisasi eksplisit (L1/L2) untuk mengendalikan kompleksitas model dan mencegah overfitting. Selain itu, XGBoost dioptimalkan untuk kecepatan dan efisiensi komputasi (misalnya dengan pelatihan paralel multithreading dan optimasi algoritmik), sehingga lebih cepat dilatih dan umumnya menghasilkan generalisasi yang lebih baik daripada GBM standar. Fitur-fitur ini membuat XGBoost lebih kuat dan fleksibel dibandingkan implementasi GBM konvensional.

B. FORMULIR PERANCANGAN

FORMULIR PERANCANGAN STUDIKASUS XGBOOST (TA-14)	JAWABAN
Judul Studi Kasus	Prediksi Kepuasan Layanan Pelanggan E-Commerce Menggunakan XGBoost
Tipe Masalah	Klasifikasi Biner
Definisi Data:	
1. Fitur (X)	<p>Dataset <i>E-commerce Customer Service Satisfaction</i> dari Kaggle berisi data interaksi pelanggan dengan layanan e-commerce. Fitur yang digunakan antara lain:</p> <ul style="list-style-type: none"> • Customer Age • Customer Gender • Product Category • Order Quantity • Order Value • Delivery Time • Payment Method • Customer Support Contact Count • Product Rating • Discount Applied <p>Link: https://www.kaggle.com/code/alastairplant/ecommerce-customer-service-satisfaction-analysis</p>
2. Target (y)	<p>0 : Pelanggan tidak puas terhadap layanan</p> <p>1 : Pelanggan puas terhadap layanan</p>
Alasan Memilih XGBoost	Data kepuasan layanan pelanggan memiliki karakteristik kompleks dengan banyak faktor yang saling berinteraksi, seperti waktu pengiriman, kualitas produk, dan respons layanan pelanggan. Algoritma sederhana seperti Naive Bayes tidak mampu menangkap hubungan non-linear antar fitur tersebut. XGBoost dipilih karena mampu mengolah data tabular dengan banyak fitur secara efektif, meningkatkan akurasi klasifikasi, serta lebih robust terhadap ketidakseimbangan data kelas.
Hyperparameter Utama	<i>learning_rate</i> Parameter ini mengatur laju pembelajaran model agar setiap pohon keputusan berkontribusi secara bertahap, sehingga membantu meningkatkan performa model dan mencegah overfitting.