Nama : Fadel Muhammad

NIM : 1103213062

1. Jika model Machine Learning menunjukkan AUC-ROC tinggi (0.92) tetapi Presisi sangat rendah (15%) pada dataset tersebut, jelaskan faktor penyebab utama ketidaksesuaian ini! Bagaimana strategi tuning hyperparameter dapat meningkatkan Presisi tanpa mengorbankan AUC-ROC secara signifikan? Mengapa Recall menjadi pertimbangan kritis dalam konteks ini, dan bagaimana hubungannya dengan cost false negative?

   AUC-ROC tinggi, Presisi rendah (15%)

- Penyebab: AUC-ROC mengukur kemampuan model membedakan kelas, tetapi Presisi dipengaruhi oleh threshold. Presisi rendah biasanya akibat threshold terlalu rendah, sehingga terlalu banyak prediksi positif salah (false positive).
- Tuning Hyperparameter: Optimalkan threshold (mis. lewat Precision-Recall Curve), ubah class_weight, atau gunakan penalti asimetris untuk menekan false positive.
- Recall penting: Jika false negative (FN) mahal (mis. deteksi penyakit/fraud), Recall tinggi menghindari kehilangan kasus positif penting.

2. Sebuah fitur kategorikal dengan 1000 nilai unik (high-cardinality) digunakan dalam model machine learning. Jelaskan dampaknya terhadap estimasi koefisien dan stabilitas Presisi! Mengapa target encoding berisiko menyebabkan data leakage dalam kasus dataset tersebut, dan alternatif encoding apa yang lebih aman untuk mempertahankan AUC-ROC?

   Fitur kategorikal high-cardinality (1000 kategori)

- Dampak: Sulit diestimasi (overfitting), noise tinggi, dan Presisi jadi tidak stabil karena sparsitas.
- Data Leakage dari Target Encoding: Encoding berdasarkan target rata-rata "membocorkan" informasi label ke fitur — fatal jika encoding dilakukan sebelum split data.

- Alternatif:
    - *Leave-one-out encoding* (dengan regularisasi),
    - *Frequency encoding*, atau
    - *Embedding* (dalam neural models) lebih aman dan stabil untuk AUC.

3. Setelah normalisasi Min-Max, model SVM linear mengalami peningkatan Presisi dari 40% ke 60% tetapi Recall turun 20%. Analisis dampak normalisasi terhadap decision boundary dan margin kelas minoritas! Mengapa scaling yang sama mungkin memiliki efek berlawanan jika diterapkan pada model Gradient Boosting?

   Min-Max Scaling naikkan Presisi tapi turunkan Recall (SVM)

- Dampak ke Decision Boundary: Scaling memperkecil jarak antar titik → margin bisa berubah drastis. Model bisa jadi overconfident pada mayoritas → minoritas terabaikan (Recall turun).
- Gradient Boosting: Tidak sensitif ke scaling karena pohon hanya melihat urutan/threshold → scaling tidak ubah struktur pohon, bahkan bisa memperburuk jika fitur menjadi terlalu homogen.

4. Eksperimen feature interaction dengan menggabungkan dua fitur melalui perkalian meningkatkan AUC-ROC dari 0.75 ke 0.82. Jelaskan mekanisme matematis di balik peningkatan ini dalam konteks decision boundary non-linear! Mengapa uji statistik seperti chi-square gagal mendeteksi interaksi semacam ini, dan metode domain knowledge apa yang dapat digunakan sebagai alternatif?

   Feature Interaction via perkalian naikkan AUC-ROC

- Mekanisme: Interaksi menghasilkan fitur non-linear → memungkinkan model menangkap pola kompleks yang linear model lewatkan.
- Chi-square gagal deteksi: Karena hanya uji ketergantungan 2 variabel → tidak mendeteksi interaksi non-linear multiplikatif.
- Alternatif Domain Knowledge:
    - Analisis korelasi domain (mis. rasio, produk fitur bisnis),
    - Decision tree feature importance,

- Partial dependence plot (PDP).

5. Dalam pipeline preprocessing, penggunaan oversampling sebelum pembagian train-test menyebabkan data leakage dengan AUC-ROC validasi 0.95 tetapi AUC-ROC testing 0.65. Jelaskan mengapa temporal split lebih aman untuk fraud detection, dan bagaimana stratified sampling dapat memperparah masalah ini! Bagaimana desain preprocessing yang benar untuk memastikan evaluasi metrik Presisi/Recall yang realistis?

   Oversampling sebelum split → data leakage

- Masalah: Model "melihat" variasi sintetis dari data uji saat pelatihan → validasi tampak bagus tapi test drop drastis.
- Temporal Split aman: Fraud sering bersifat time-dependent; split berdasarkan waktu menjaga urutan kejadian dan mencegah informasi masa depan bocor ke pelatihan.
- Stratified Sampling memperparah: Duplikasi minoritas ke seluruh dataset sebelum split → model overfit.
- Solusi Benar: Lakukan split dulu → baru oversample hanya pada data latih.