# Analisa Klasifikasi Model

**Nama:** Arif Al Imran

**Kelas:** TK-45-G05

**NIM:** 1103210193

## Soal:

1. Jika model Machine Learning menunjukan AUC-ROC tinggi (0.92) tetapi Presisi sangat rendah (15%) pada dataset tersebut, jelaskan factor 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?
2. Sebuah fitur kategorikal dengan 1000 nilai unik (high-cardinality) digunakan dalam machine learning. Jelaskan dampakanya 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?
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?
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 statistic seperti chi-square gagal mendeteksi interaksi semacam ini, dan metode domain knowledge apa yang dpata digunakan sebagai alternatif?
5. Dalam pipeline preprocessing, penggunaan oversampling sebelum pembagian train-test menyebabkam 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 realisitis?

## Jawab:

### 1. Ketidaksesuaian AUC-ROC Tinggi dengan Presisi Rendah

Ketidaksesuaian antara AUC-ROC tinggi (0.92) dan presisi rendah (15%) terutama disebabkan oleh:

**Faktor Penyebab Utama:**

- Imbalance dataset yang ekstrem (kelas positif sangat sedikit)

- Model berhasil membedakan kelas secara keseluruhan (AUC-ROC tinggi) namun gagal pada klasifikasi kasus positif spesifik
- Threshold prediksi default (0.5) tidak optimal untuk dataset tidak seimbang

**Strategi Tuning Hyperparameter:**

- Penyesuaian threshold prediksi berdasarkan kurva precision-recall
- Penerapan class_weight untuk memberikan penalti lebih besar pada misklasifikasi kelas minoritas
- Implementasi regularisasi yang tepat (L1/L2) untuk meningkatkan generalisasi model
- Teknik resampling selektif dan metode ensemble (bagging) untuk meningkatkan presisi

**Pentingnya Recall:**

- Recall berhubungan langsung dengan false negative (melewatkan kasus positif nyata)
- Dalam domain seperti fraud detection atau diagnosis medis, cost false negative jauh lebih tinggi daripada false positive
- Optimalisasi presisi-recall harus mempertimbangkan cost business aktual dari kedua jenis kesalahan klasifikasi

## 2. Dampak Fitur Kategorikal High-Cardinality

Fitur kategorikal dengan 1000 nilai unik menyebabkan:

**Dampak pada Estimasi Koefisien dan Presisi:**

- Menghasilkan matriks sangat sparse saat one-hot encoding
- Memicu curse of dimensionality yang menyebabkan overfitting
- Menghasilkan estimasi koefisien tidak stabil dan presisi rendah, terutama untuk kategori yang jarang muncul

**Risiko Target Encoding:**

- Target encoding mengganti kategori dengan rata-rata nilai target, berpotensi menyebabkan data leakage
- Informasi label testing secara tidak langsung masuk ke proses training
- Menghasilkan overestimasi performa model saat validasi

**Alternatif Encoding yang Lebih Aman:**

- Leave-one-out encoding: Mengurangi leakage dengan mengecualikan sampel saat ini
- Hash encoding: Mengelompokkan kategori yang mirip untuk mengurangi dimensi
- Weight of Evidence encoding dengan proper cross-validation
- Teknik regularisasi kategorikal untuk mengurangi overfitting sambil mempertahankan AUC-ROC

## 3. Dampak Normalisasi Min-Max pada SVM dan Gradient Boosting

**Dampak pada SVM Linear:**

- Peningkatan presisi (40% ke 60%) namun penurunan recall (20%) terjadi karena transformasi mengubah jarak relatif antar titik data
- Normalisasi mempengaruhi margin SVM dan lokasi decision boundary
- Beberapa titik data kelas minoritas yang sebelumnya terklasifikasi benar menjadi terklasifikasi salah
- SVM sangat sensitif terhadap skala fitur karena algoritmanya berbasis jarak

**Perbedaan dengan Gradient Boosting:**

- Gradient Boosting berbasis tree relatif kebal terhadap transformasi monoton
- Pemisahan pada pohon keputusan berdasarkan urutan/ranking nilai, bukan nilai absolut
- Transformasi Min-Max cenderung memiliki dampak minimal atau bahkan berlawanan pada model Gradient Boosting
- Karakteristik ini menjelaskan mengapa normalisasi yang sama dapat menghasilkan efek berbeda pada model yang berbeda

## 4. Mekanisme Feature Interaction dan Peningkatan AUC-ROC

**Mekanisme Matematis Feature Interaction:**

- Perkalian dua fitur ($X_1 \times X_2$) menciptakan representasi non-linear dari data
- Memperkenalkan komponen kuadratik ke decision boundary yang sebelumnya linear
- Model asli: $f(x) = w_1X_1 + w_2X_2 + b$ (linear)
- Dengan feature interaction: $f(x) = w_1X_1 + w_2X_2 + w_3(X_1 \times X_2) + b$ (non-linear)
- Decision boundary yang lebih fleksibel mampu menangkap pola kompleks yang tidak dapat direpresentasikan oleh model linear

**Keterbatasan Uji Chi-square:**

- Uji chi-square dirancang untuk menilai dependensi linear atau nominal sederhana
- Tidak mampu mendeteksi interaksi kompleks atau non-linear antar fitur
- Fokus pada hubungan marginal, bukan conditional dependencies

**Alternatif Berbasis Domain Knowledge:**

- Partial dependence plots untuk visualisasi interaksi fitur
- Analisis SHAP interaction values untuk mengukur kontribusi interaksi terhadap prediksi
- Pendekatan berbasis fisika/domain yang mempertimbangkan mekanisme kausal dalam data

## 5. Data Leakage dalam Pipeline Preprocessing dan Solusinya

**Penyebab Data Leakage dengan Oversampling:**

- Oversampling sebelum pembagian train-test menyebabkan kasus yang digandakan muncul di kedua set
- Model "menghafal" contoh spesifik alih-alih belajar generalisasi pola
- Menghasilkan kesenjangan besar antara AUC-ROC validasi (0.95) dan testing (0.65)

**Keunggulan Temporal Split untuk Fraud Detection:**

- Menghormati urutan kronologis data, mencegah model "melihat masa depan"
- Fraud patterns berevolusi seiring waktu, split berdasarkan waktu lebih realistis
- Lebih baik mengevaluasi kemampuan prediktif model terhadap pola fraud yang baru muncul

**Masalah dengan Stratified Sampling:**

- Memaksa distribusi kelas yang sama antara training dan testing
- Distribusi fraud sesungguhnya dapat berubah seiring waktu
- Menghasilkan evaluasi yang terlalu optimistis dan tidak mencerminkan realitas

**Desain Preprocessing yang Benar:**

1. Split data terlebih dahulu (sebaiknya temporal untuk fraud detection)
2. Lakukan normalisasi/transformasi fitur menggunakan statistik dari training set saja
3. Terapkan oversampling/undersampling pada training set saja
4. Evaluasi pada test set asli (tanpa perubahan) untuk mendapatkan estimasi performa yang realistis