

Analisa Classification Model

Nama : Atanasius Pradiptha Sampurno

Kelas : TK-45-02

NIM : 1103213036

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*?

Jawaban:

Ketidaksesuaian antara AUC-ROC yang tinggi dan Precision yang rendah sering kali disebabkan oleh model yang terlalu sensitif dalam memprediksi kelas positif, yang mengarah pada banyak *false positives*. Untuk meningkatkan Precision tanpa menurunkan AUC-ROC secara signifikan, salah satu strategi yang dapat diterapkan adalah mengubah *threshold* klasifikasi. Dengan menyesuaikan *threshold*, model akan lebih selektif dalam memprediksi kelas positif, mengurangi jumlah *false positives*. *Recall* menjadi pertimbangan kritis dalam konteks ini karena *false negatives* dapat membawa konsekuensi yang sangat mahal, seperti gagal mendeteksi potensi fraud atau penyakit.

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?

Jawaban:

Fitur kategorikal dengan 1000 nilai unik bisa menyebabkan *overfitting*, mengganggu estimasi koefisien model dan menurunkan stabilitas presisi. Target encoding berisiko menyebabkan *data leakage* karena menggunakan informasi target yang seharusnya tidak digunakan saat pelatihan. Sebagai alternatif yang lebih aman, *One-Hot Encoding* atau *CatBoost Encoding* dapat digunakan, yang lebih menjaga kestabilan model dan mempertahankan AUC-ROC tanpa memanfaatkan informasi target yang berisiko.

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*?

Jawaban:

Normalisasi Min-Max mempengaruhi model SVM dengan merubah skala fitur, yang dapat memperbaiki penghitungan *margin* dan *decision boundary*, terutama pada fitur dengan rentang yang berbeda. Peningkatan presisi menunjukkan bahwa model lebih akurat dalam memprediksi kelas positif, tetapi penurunan recall menunjukkan bahwa model cenderung melewati prediksi kelas minoritas (*false negatives*). Pada SVM, perubahan skala ini berpengaruh signifikan. Namun, untuk model seperti *Gradient Boosting*, *scaling* dapat membuat pembagian fitur menjadi kurang relevan, karena algoritma ini tidak bergantung pada jarak atau margin, melainkan pada pemisahan berbasis keputusan.

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?

Jawaban:

Menggabungkan dua fitur melalui perkalian menciptakan hubungan non-linear yang memungkinkan model untuk mengenali pola yang lebih kompleks, meningkatkan AUC-ROC dari 0.75 ke 0.82. Hal ini memengaruhi *decision boundary* dengan memungkinkan model memisahkan kelas dengan lebih akurat. Uji statistik seperti *chi-square* tidak efektif untuk mendeteksi interaksi non-linear karena hanya mengevaluasi hubungan linier. Sebagai alternatif, pengetahuan domain dapat digunakan untuk merancang fitur atau menerapkan teknik seperti *polynomial features* atau *neural networks* yang dapat menangkap interaksi kompleks antar fitur.

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?

Jawaban:

Oversampling yang dilakukan sebelum membagi data train-test dapat menyebabkan *data leakage*, karena informasi dari *test set* dapat masuk ke dalam pelatihan model, yang meningkatkan AUC-ROC pada validasi namun merendahkan hasil pada testing. Untuk *fraud detection*, *temporal split* lebih aman karena mempertahankan urutan waktu, mencegah model melihat data masa depan. *Stratified sampling* bisa memperburuk hal ini dengan memastikan distribusi kelas yang tidak realistis di test set. Solusi terbaik adalah memisahkan pembagian data sebelum melakukan *oversampling* untuk menghindari *data leakage*.