



FRAUD DETECTION

23 Juni 2025



Meet the Team



Yanto



Muhammad Sutisna



Vika Oktarina



Ryandito Mahendradani



A. Mambaus Sholihin



Mukhammad Fatkhur Rozi

Realitas Penipuan Keuangan

- **Kerugian Penipuan Global:** Bisnis dan individu di seluruh dunia kehilangan triliunan dolar setiap tahun karena berbagai bentuk penipuan keuangan.
- **Penipuan Identitas:** Pada tahun 2023 saja, penipuan identitas mengakibatkan 40 juta orang Amerika mengalami kerugian miliaran dolar.
- **Kejahatan Siber & Penipuan:** Sebagian besar penipuan keuangan sekarang terjadi secara online. Insiden penipuan siber meningkat lebih dari 50% dalam dua tahun terakhir, dengan phishing dan kompromi email bisnis (BEC) menjadi vektor serangan utama.
- **Dampak pada Konsumen:** Konsumen individu menghadapi kerugian rata-rata ratusan hingga ribuan dolar per insiden, sering kali mengalami tekanan emosional yang signifikan dan konsekuensi finansial jangka panjang.
- **Dampak Organisasi:** Selain kerugian finansial langsung, organisasi menderita kerusakan reputasi yang parah, penurunan kepercayaan pemangku kepentingan, denda peraturan, dan gangguan operasional yang signifikan karena penipuan.



Pentingnya Pencegahan Penipuan Keuangan

- **Melindungi Aset:** Menjaga sumber daya keuangan untuk individu, bisnis, dan ekonomi.
- **Menjaga Kepercayaan:** Mempertahankan kepercayaan pada sistem keuangan, institusi, dan transaksi digital. Memastikan Kepatuhan: Mematuhi persyaratan peraturan yang semakin ketat dan menghindari konsekuensi hukum.
- **Mendorong Stabilitas Ekonomi:** Mengurangi pengurasan sumber daya yang disebabkan oleh aktivitas ilegal, memungkinkan pertumbuhan dan investasi ekonomi yang lebih sehat.
- **Menjaga Reputasi:** Mencegah erosi kepercayaan publik dan investor yang sering menyertai insiden penipuan.



Machine Learning Dalam Penipuan Keuangan

Seiring dengan berkembangnya teknologi kecerdasan buatan, deteksi fraud tidak lagi hanya mengandalkan pemeriksaan manual, melainkan juga dapat didukung oleh sistem pintar yang mampu mengenali pola-pola anomali dalam data transaksi. Dengan menggunakan algoritma pembelajaran mesin seperti **Logistic Regression** dan **Random Forest**, sistem ini dapat secara adaptif dan otomatis mengidentifikasi kemungkinan terjadinya fraud dalam data.

Project ini bertujuan untuk **membangun sistem deteksi fraud berbasis AI** yang dapat menjadi solusi nyata terhadap ancaman kecurangan digital yang semakin kompleks.



Objectives & Scope

Objectives

- Mendeteksi transaksi tidak wajar menggunakan machine learning.
- Menerapkan dan membandingkan model Random Forest & Logistic Regression.
- Melakukan evaluasi performa model (Accuracy, Precision, Recall, F1, AUC).
- Melakukan hyperparameter tuning untuk optimasi

Scope

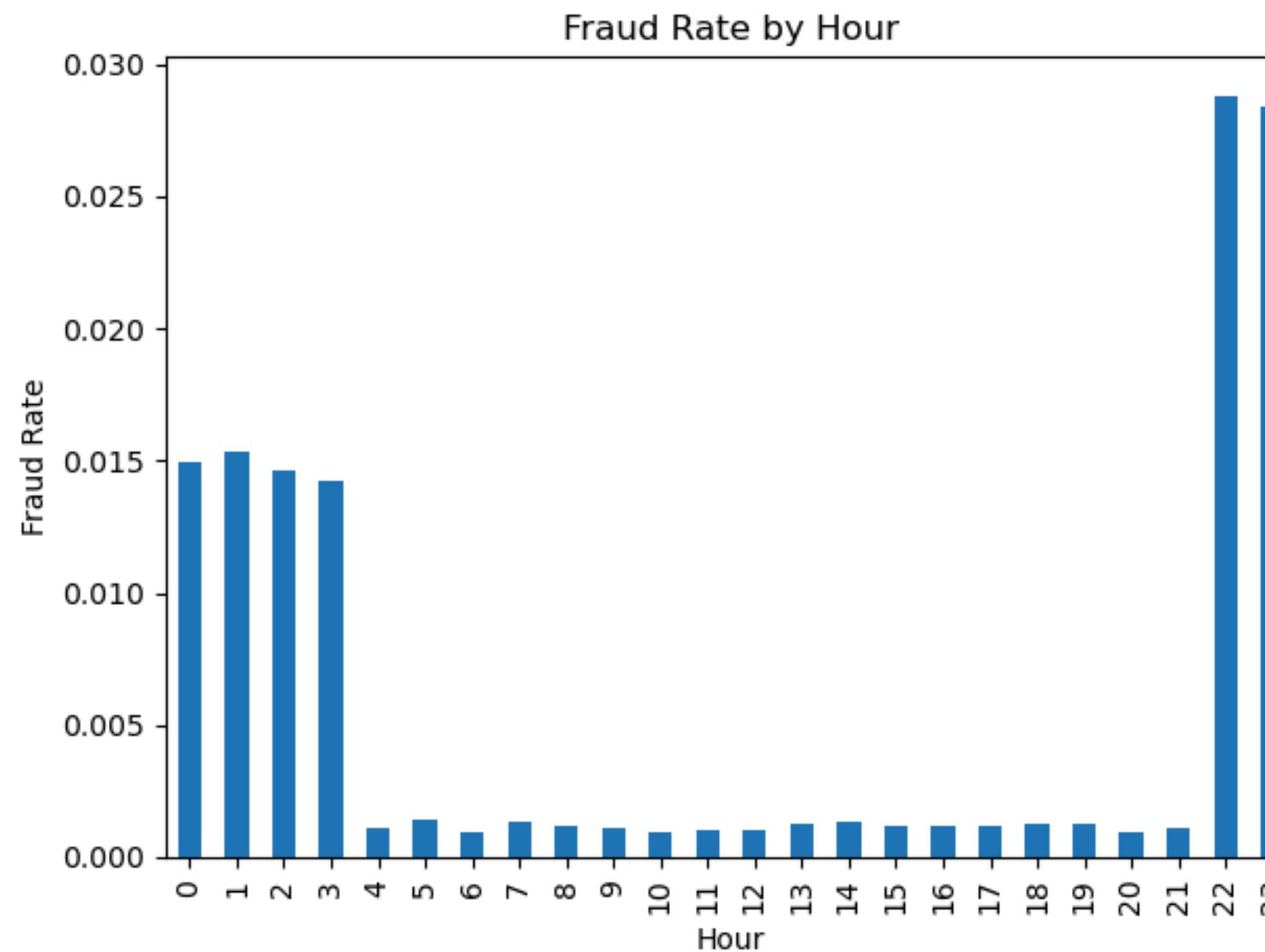
- Data cleaning & eksplorasi data (EDA).
- Feature engineering dan modeling.
- Evaluasi model & dokumentasi.



Data Collection & Preparation

Dataset berisi informasi transaksi: waktu, lokasi, merchant, kategori, jumlah, dan atribut pengguna (nama, umur, alamat).

- Dataset: 1 juta+ transaksi pengguna kartu kredit.
- Target: is_fraud (1: fraud, 0: non-fraud)
- Rasio fraud sangat kecil (~0.13%) = imbalanced dataset

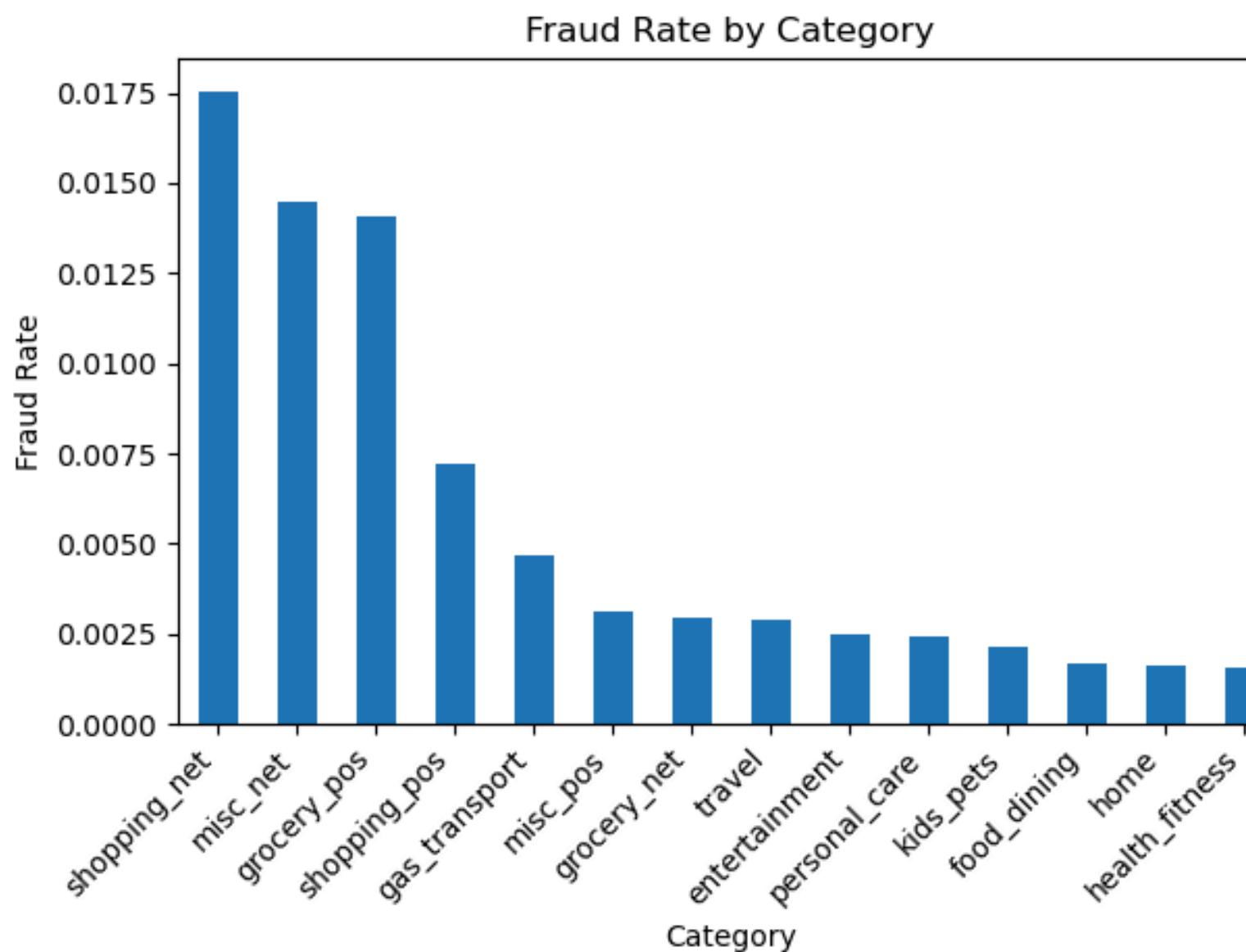


- Fraud paling sering terjadi malam hari (22:00–03:00). ini akan menjadi dasar pembuatan fitur engineering yaitu “Jam Risk” :
 - Jam 22-23 = 3
 - Jam 0-3 = 2
 - Jam 4-21 = 1

Data Collection & Preparation

TOP Fraud berdasarkan Category

- shopping_net merupakan jenis transaksi Belanja online (e-commerce, marketplace) yang rawan terjadi Fraud karena tanpa verifikasi fisik (PIN, tanda tangan, biometrik), bisa dilakukan dari mana saja (jarak jauh), rawan terkena phishing, malware, dan skimming.

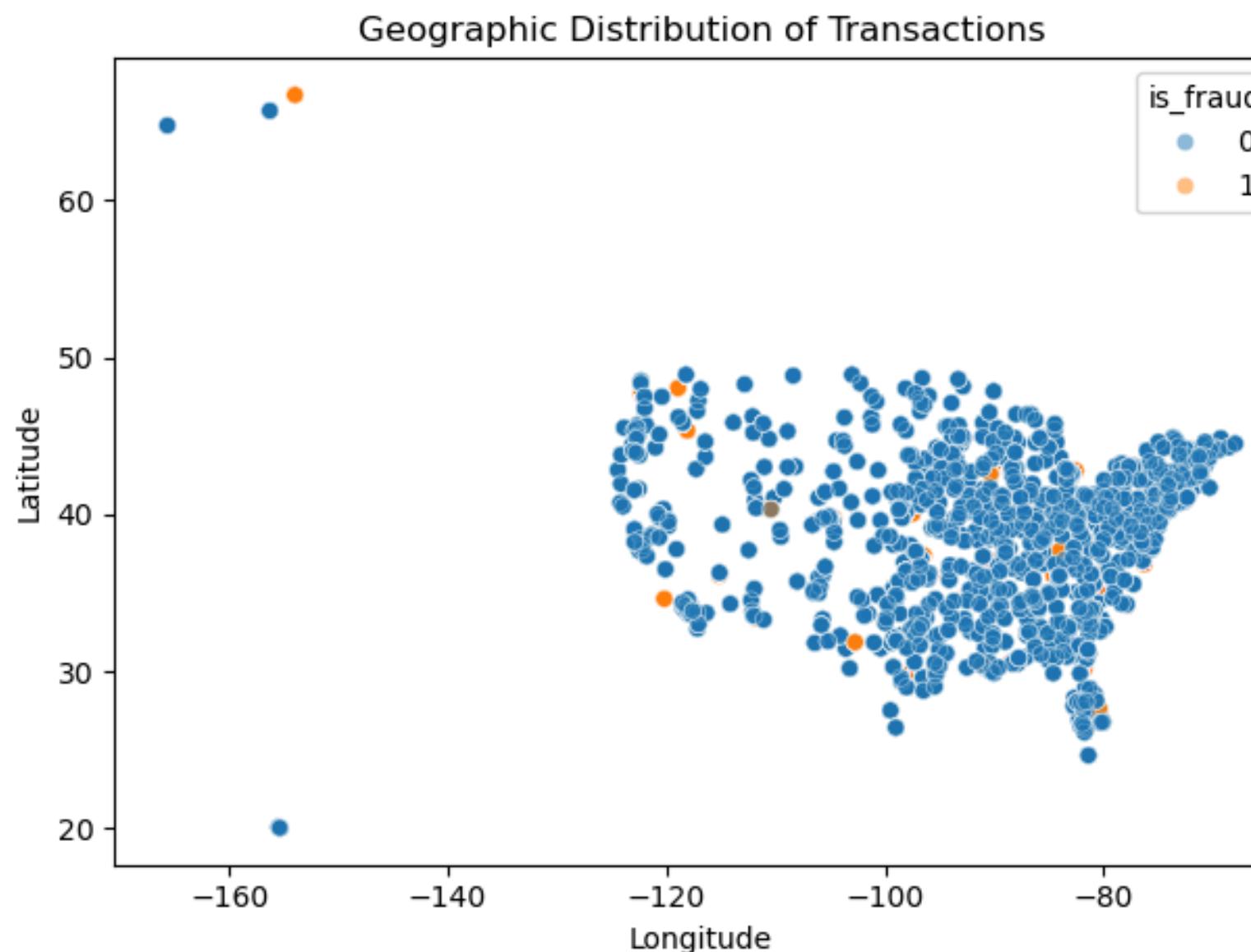


- fitur engineering yaitu “Category Encoded” :
 - 'shopping_net' = 4
 - 'misc_net', 'grocery_pos' = 3
 - 'shopping_pos', 'gas_transport' = 2
 - Else = 1

Data Collection & Preparation

Visualisasi sebaran geografis transaksi berdasarkan longitude & latitude :

- Warna biru = transaksi normal (`is_fraud = 0`)
- Warna oranye = transaksi fraud (`is_fraud = 1`)
- Mayoritas transaksi terjadi di daratan utama Amerika Serikat (sekitar -125 hingga -70 longitude). Beberapa titik fraud tampak muncul di lokasi yang jauh / tidak umum (misalnya dekat Alaska dan Hawaii) → indikasi potensi anomali. Transaksi fraud cenderung terjadi di lokasi yang tidak biasa (outlier).

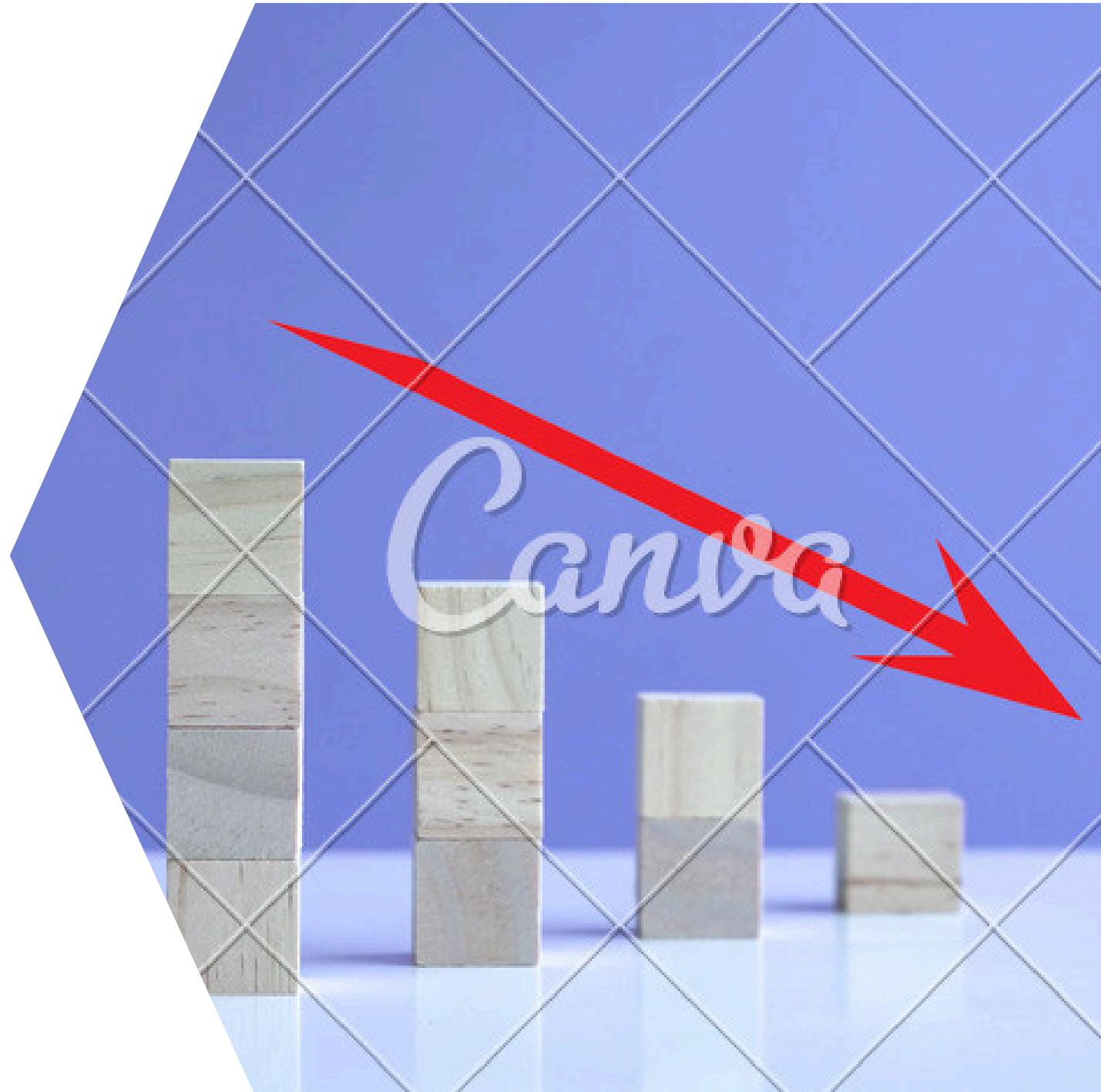


- Dari perhitungan Longitude dan Latitude bisa dibuat fitur engineering yaitu “Distance” :
 - Jarak antara transaksi dari titik koordinat dikonversikan menjadi Kilometer (Km)

Model Development

Logistic Regression

- Merupakan algoritma **klasifikasi biner** yang berbasis pada prinsip regresi logistik, dirancang untuk menghitung probabilitas suatu kejadian (misalnya, fraud atau non-fraud).
- Langkah-langkah Implementasi:
 - Pra-pemrosesan Data: Memuat dan membersihkan data.
 - Rekayasa Fitur: Membuat fitur baru (contoh: jam, hari, umur, jarak).
 - Inisialisasi Model: LogisticRegression (dengan class_weight='balanced').
 - Pelatihan Model: Melatih model dengan data X_train, y_train.
 - Prediksi & Evaluasi: Mengukur kinerja model:
 - Akurasi
 - Presisi
 - Recall
 - F1 Score
 - ROC AUC
 - Analisis Fitur Penting: Menginterpretasikan koefisien model.
 - Visualisasi Hasil: Memvisualisasikan temuan.
- Catatan Khusus: Untuk data tidak seimbang, digunakan penanganan seperti SMOTE dan ADASYN sebelum pelatihan model.



Model Development

Random Forest

- Merupakan algoritma **ensemble** yang membangun banyak **decision tree** (decision trees). Hasil akhir ditentukan melalui teknik voting dari prediksi setiap pohon.
- Keunggulan:
 - Mampu menangani volume data yang besar secara efisien.
 - Efektif untuk fitur kompleks dan hubungan non-linear.
 - Fleksibel dalam menangani data tidak seimbang (imbalance data), terutama dengan penyesuaian class_weight.
- Langkah Implementasi:
 - Muat & Bersihkan Data: Proses loading data dan pembersihan awal.
 - Rekayasa Fitur (Feature Engineering): Membuat fitur baru (tidak memerlukan scaling).
 - Encoding Kategorikal (One-Hot Encoding): Konversi variabel kategorikal menjadi format numerik.
 - Pembagian Data: Membagi data menjadi set pelatihan (train) dan pengujian (test).
 - Pelatihan Model (Fit Model): Melatih model Random Forest pada data pelatihan.
 - Prediksi & Evaluasi: Melakukan prediksi dan mengevaluasi kinerja model.
 - Ekstraksi Feature Importance: Mengambil nilai feature importance untuk interpretasi model.
 - Visualisasi: Membuat visualisasi untuk pemahaman lebih lanjut.



Training & Optimization

Pada tahap pelatihan data, pemilihan kolom fitur dilakukan dengan mengacu pada interpretasi visual dari grafik yang dihasilkan selama proses Rekayasa Fitur (Feature Engineering).

```
fitur = ['jam_risk', 'category_encoded', 'amt', 'distance', 'amt_outlier_flag', 'age_risk_score']
x = td[fitur]
y = td['is_fraud']

# Split data (stratify agar proporsi fraud tetap)
X_train, X_test, y_train, y_test = train_test_split(
    x, y, test_size=0.2, random_state=42, stratify=y
)
```



Results - Logistic Regression

Berikut hasil dari Model Logistic Regression secara manual

Akurasi tanpa Handling Imbalance : 0.8679275840129562				
	precision	recall	f1-score	support
0	1.00	0.87	0.93	257834
1	0.04	0.84	0.07	1501
accuracy			0.87	259335
macro avg	0.52	0.85	0.50	259335
weighted avg	0.99	0.87	0.92	259335

Akurasi dengan Handling Imbalance SMOTE : 0.884377349759963				
	precision	recall	f1-score	support
0	1.00	0.88	0.94	257834
1	0.04	0.82	0.08	1501
accuracy			0.88	259335
macro avg	0.52	0.85	0.51	259335
weighted avg	0.99	0.88	0.93	259335

Akurasi dengan Handling Imbalance ADASYN : 0.8541693176779069				
	precision	recall	f1-score	support
0	1.00	0.85	0.92	257834
1	0.03	0.84	0.06	1501
accuracy			0.85	259335
macro avg	0.52	0.85	0.49	259335
weighted avg	0.99	0.85	0.92	259335



Results - Logistic Regression

Berikut hasil dari Model Logistic Regression dengan Hyperparameter Tuning (Optional) bertujuan untuk menemukan kombinasi parameter terbaik untuk memaksimalkan performa model, dalam hal ini Logistic Regression.

Tanpa Handling Imbalance

```
Best Parameters : {'logreg_C': 0.1, 'logreg_class_weight': None, 'logreg_penalty': 'l1'}  
Best Score : 0.18147143019632186  
=====
```

Dengan Handling Imbalance SMOTE

```
Best Parameters : {'logreg_C': 0.1, 'logreg_class_weight': 'balanced', 'logreg_penalty': ''}  
Best Score : 0.8483337138091631  
=====
```

Dengan Handling Imbalance ADASYN

```
Best Parameters : {'logreg_C': 0.1, 'logreg_class_weight': 'balanced', 'logreg_penalty': ''}  
Best Score : 0.8004053558738149
```

Results - Logistic Regression (Hyperparameter Tuning)

Berikut hasil dari Model Logistic Regression dengan Hyperparameter Tuning (Optional) bertujuan untuk menemukan kombinasi parameter terbaik untuk memaksimalkan performa model, dalam hal ini Logistic Regression.

== Logistic Regression ==				
	precision	recall	f1-score	support
0	1.00	0.87	0.93	257834
1	0.04	0.84	0.07	1501
accuracy			0.87	259335
macro avg	0.52	0.85	0.50	259335
weighted avg	0.99	0.87	0.92	259335

== Logistic Regression dengan SMOTE ==				
	precision	recall	f1-score	support
0	1.00	0.88	0.94	257834
1	0.04	0.82	0.08	1501
accuracy			0.88	259335
macro avg	0.52	0.85	0.51	259335
weighted avg	0.99	0.88	0.93	259335

== Logistic Regression dengan ADASYN ==				
	precision	recall	f1-score	support
0	1.00	0.86	0.92	257834
1	0.03	0.84	0.06	1501
accuracy			0.86	259335
macro avg	0.52	0.85	0.49	259335
weighted avg	0.99	0.86	0.92	259335



Results - Random Forest

Berikut hasil dari Random Forest

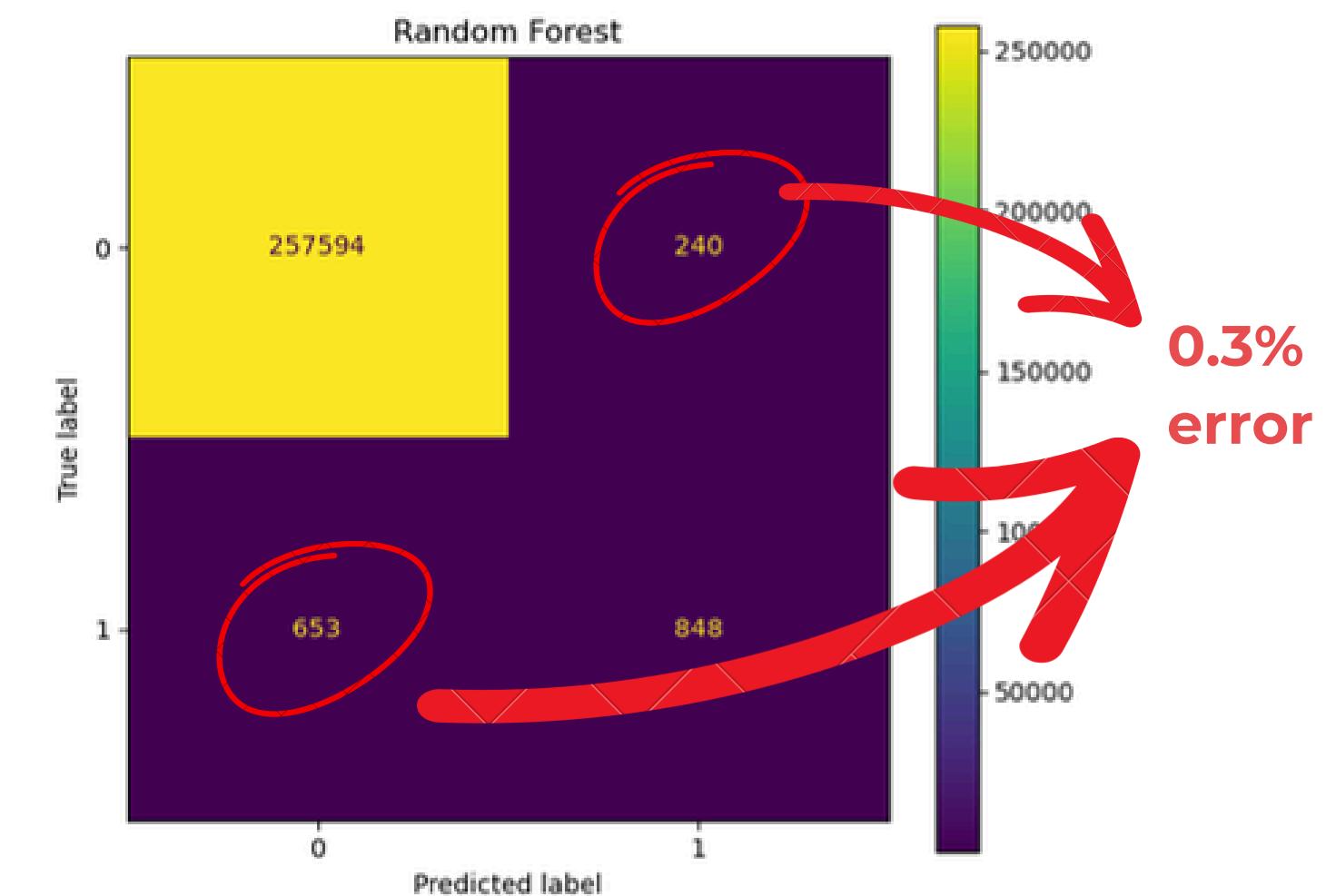
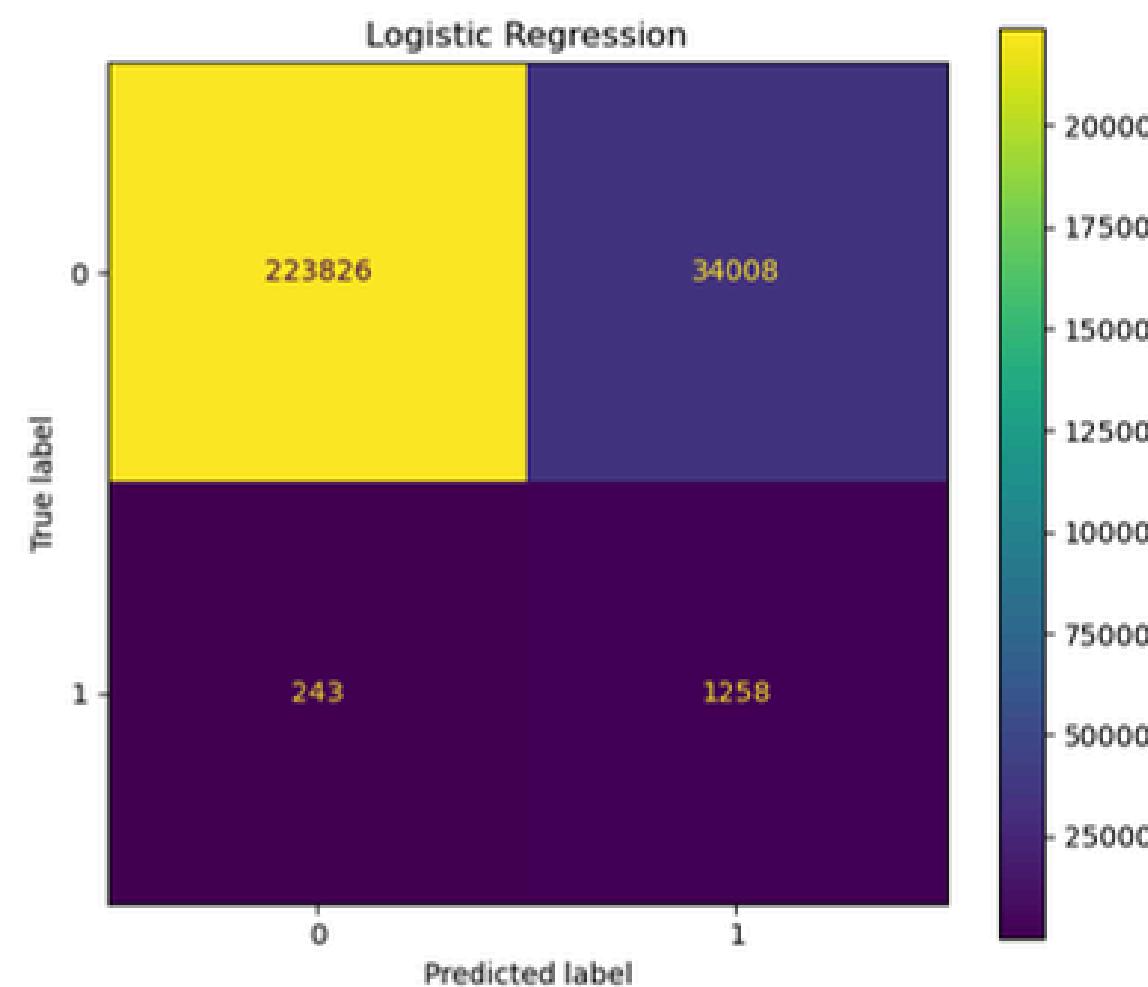
Akurasi tanpa Handling Imbalance : 0.9965565773998881				
	precision	recall	f1-score	support
0	1.00	1.00	1.00	257834
1	0.78	0.56	0.66	1501
accuracy			1.00	259335
macro avg	0.89	0.78	0.83	259335
weighted avg	1.00	1.00	1.00	259335

Akurasi dengan Handling Imbalance SMOTE : 0.9819191393371508				
	precision	recall	f1-score	support
0	1.00	0.98	0.99	257834
1	0.21	0.75	0.32	1501
accuracy			0.98	259335
macro avg	0.60	0.86	0.66	259335
weighted avg	0.99	0.98	0.99	259335

Akurasi dengan Handling Imbalance ADASYN : 0.9804307170262402				
	precision	recall	f1-score	support
0	1.00	0.98	0.99	257834
1	0.19	0.75	0.31	1501
accuracy			0.98	259335
macro avg	0.60	0.87	0.65	259335
weighted avg	0.99	0.98	0.99	259335

Results - Model Comparison

Berikut grafik Confusion Matrix yang menampilkan hasil prediksi model Logistic Regression dan Random Forest dalam mendekksi fraud:



Logistic Regression

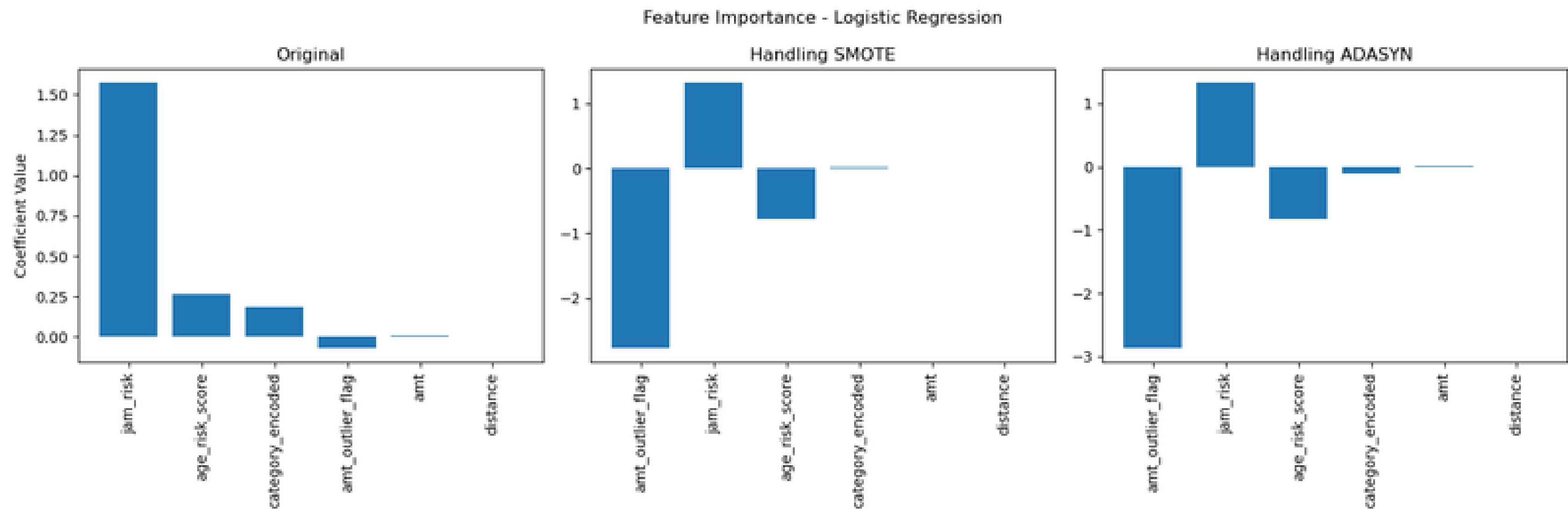
- Model ini cukup baik mengenali fraud ($TP = 1,258$), hanya sedikit $FN = 243$
- Tapi FP sangat besar (34.008) → model sering salah menuduh transaksi normal sebagai fraud
- ► Tingkat false alarm tinggi (tidak ideal untuk user experience)

Random Forest

- Random Forest sangat baik dalam mengenali non-fraud (FP hanya 240)
- Deteksi fraud ($TP = 848$) cukup baik, meskipun tidak setinggi Logistic Regression
- $FN = 653$ → masih ada fraud yang lolos, tapi trade-off ini wajar
- ► Model lebih stabil dan tidak banyak false alarm

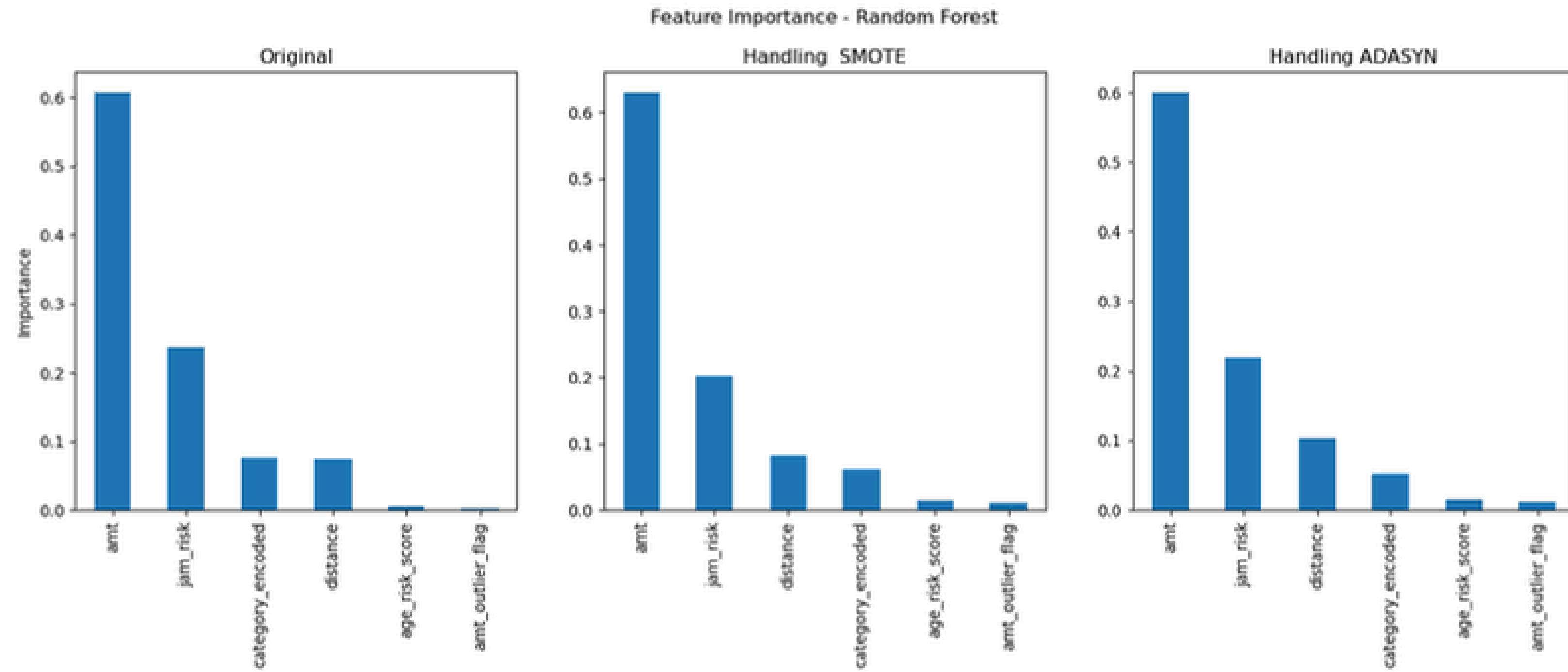


Results - Features Importance





Results - Features Importance



Conclusion

- Penipuan keuangan menimbulkan dampak **kerugian** yang komprehensif, tidak terbatas pada aspek **material**, melainkan juga merugikan dimensi **non-material** seperti reputasi dan kepercayaan.
- Dalam upaya mitigasi penipuan, **tim 360** telah merancang dan mengembangkan sistem deteksi berbasis **Pembelajaran Mesin (Machine Learning)** dengan metodologi berikut:
 - **Pemanfaatan Data Historis:** Model dikembangkan dan dilatih menggunakan data transaksi historis yang relevan.
 - **Analisis Rekayasa Fitur (Feature Engineering Analysis):** Dilakukan analisis mendalam untuk mengidentifikasi dan menciptakan fitur-fitur baru dari data mentah yang dapat meningkatkan kemampuan deteksi.
 - **Penanganan Isu Ketidakseimbangan Data (Data Imbalance Handling):** Implementasi strategi khusus untuk mengatasi ketidakseimbangan kelas dalam dataset, memastikan model tidak bias terhadap kelas mayoritas.
 - **Perbandingan Model:** Dilakukan evaluasi komparatif terhadap kinerja berbagai algoritma Pembelajaran Mesin untuk menemukan solusi yang paling optimal.
 - **Komparasi Hasil:** Analisis dan perbandingan mendalam terhadap hasil yang diperoleh dari setiap model.
- Dari hasil evaluasi, ditemukan bahwa Model **Random Forest**, bahkan tanpa penanganan spesifik terhadap ketidakseimbangan data, menunjukkan akurasi tertinggi. **Matriks Konfusi (Confusion Matrix)** mengindikasikan tingkat kesalahan yang sangat rendah, yaitu hanya **0,3%** dari keseluruhan prediksi. Sementara itu **Feature Importance** menunjukkan **nilai transaksi (amt)** merupakan faktor yang paling berpengaruh dalam penipuan keuangan.



Real-world Application

Model yang kami gunakan untuk project Fraud Detection ini bisa juga digunakan untuk :

1. Perbankan & Kartu Kredit

- a. Mampu mendeteksi transaksi mencurigakan secara real-time.
- b. Mengirimkan peringatan otomatis kepada nasabah saat terdeteksi aktivitas aneh.
- c. Melakukan penahanan sementara pada transaksi yang terindikasi sebagai penipuan.

2. E-Commerce & Marketplace

- a. Efektif dalam memblokir pembayaran palsu yang menggunakan kartu curian.
- b. Mengidentifikasi akun pengguna dengan perilaku tidak wajar atau mencurigakan.
- c. Menandai pola belanja yang anomali (misalnya, pembelian tengah malam dari lokasi geografis yang berbeda).

3. Fintech & Dompet Digital (Mobile Wallets)

- a. Menilai tingkat risiko transaksi dalam hitungan detik sebelum proses pembayaran selesai.
- b. Menyesuaikan limit atau izin transaksi berdasarkan analisis perilaku pengguna.

4. Audit Internal & Kepatuhan (Compliance)

- a. Memberikan wawasan berharga untuk pelaksanaan audit internal secara rutin.
- b. Mencegah kerugian finansial sebelum insiden terjadi.
- c. Memfasilitasi pelaporan yang akurat kepada pihak regulator.

5. Penguatan Sistem Keamanan

- a. Mengembangkan sistem keamanan berbasis Kecerdasan Buatan (AI) yang terus belajar dari pola-pola penipuan baru.
- b. Meningkatkan kepercayaan pelanggan dan reputasi positif perusahaan.





Future Improvement

Penyetelan Ambang Batas (Threshold Tuning)

Sesuaikan ambang batas prediksi untuk mengoptimalkan keseimbangan antara precision dan recall model.

Model yang Lebih Canggih (Advanced Models)

Pertimbangkan untuk mengimplementasikan model ensemble seperti XGBoost, LightGBM, atau CatBoost guna meningkatkan akurasi dan kinerja deteksi.

Penambahan Fitur (Feature Engineering)

Perkaya set fitur dengan memasukkan data relevan seperti frekuensi transaksi, selisih waktu antar transaksi, dan perbandingan lokasi pengguna dengan lokasi pedagang.

Pemodelan Runtun Waktu (Time-Series Modeling)

Evaluasi penggunaan pendekatan pemodelan runtun waktu, seperti Jaringan Saraf Berulang / Memori Jangka Pendek-Panjang (LSTM), untuk memanfaatkan pola urutan transaksi.





Let's Collaborate

The 360 Team

