Projet Data Science – SécuTransac

Détection intelligente de transactions bancaires frauduleuses

Introduction

La digitalisation croissante des services bancaires a considérablement facilité les paiements électroniques, mais elle a également entraîné une augmentation significative des fraudes financières. Les institutions bancaires doivent aujourd'hui traiter des volumes massifs de transactions, rendant les contrôles manuels inefficaces, coûteux et peu réactifs.

Dans ce contexte, le projet **SécuTransac** propose un système de détection intelligente de transactions bancaires frauduleuses, capable d'analyser automatiquement les transactions et de fournir une probabilité de fraude, afin d'aider les équipes métier dans leur prise de décision. Ce projet, réalisé dans le cadre d'un travail de groupe, couvre l'ensemble du cycle de vie d'un projet Data Science : analyse des besoins métier, cadre légal, exploration et préparation des données, modélisation, évaluation, déploiement applicatif et visualisation des résultats via un tableau de bord interactif.

1. Analyse des besoins métier

1.1 Contexte métier

La fraude bancaire représente un enjeu stratégique majeur pour les établissements financiers. Les pertes financières directes, combinées à l'impact sur la réputation et la confiance des clients, obligent les banques à mettre en place des systèmes de détection performants, fiables et rapides.

Les approches traditionnelles basées sur des règles fixes (seuils, listes noires) montrent rapidement leurs limites face à des comportements frauduleux évolutifs. L'exploitation des données transactionnelles et les techniques de Machine Learning constituent alors une solution pertinente pour détecter des schémas complexes et non linéaires.

1.2 Problématique

Comment détecter automatiquement les transactions bancaires potentiellement frauduleuses à partir des données transactionnelles afin de réduire les pertes financières et renforcer la sécurité des paiements ?

1.3 Objectifs du projet

- Développer un modèle de Machine Learning capable de prédire si une transaction est frauduleuse.
- Fournir une probabilité de fraude (%) pour chaque transaction analysée.
- Mettre à disposition une application interactive permettant de tester une transaction manuellement.

- Créer un dashboard dynamique qui se met à jour avec les transactions testées.
- Fournir des KPI et des visualisations pour le suivi en temps réel de la fraude.

1.4 Utilisateurs cibles

- Analystes fraude
- Équipes conformité / risk management
- Équipes data et sécurité
- Décideurs métiers souhaitant suivre l'évolution de la fraude

2. Cadre légal et réglementaire

2.1 Protection des données personnelles (RGPD)

Le projet respecte les principes du Règlement Général sur la Protection des Données (RGPD). Les données utilisées sont entièrement simulées à l'aide d'un script Python. Aucun client réel n'est impliqué. Les identifiants (clients, marchands, appareils, adresses IP) sont fictifs et ne permettent aucune identification réelle.

2.2 Décisions automatisées et explicabilité

Le modèle retenu, **XGBoost**, permet d'analyser l'influence des variables sur les prédictions. Cette capacité d'explicabilité est essentielle dans le secteur bancaire, où les décisions automatisées doivent pouvoir être comprises et justifiées.

3. Description et analyse exploratoire des données

3.1 Description du jeu de données

Le dataset contient 10 000 transactions simulées, avec environ 2 % de transactions frauduleuses, reflétant un contexte réaliste de fraude bancaire.

Chaque transaction contient les variables suivantes :

- transaction_id, timestamp, amount, merchant_id, customer_id
- transaction_type (paiement, retrait, virement, dépôt)
- country (FR, US, DE, ES, IT, RU, CN, NG, UA, BR)
- device_id, ip_address
- merchant_category (restaurant, technologie, divertissement, voyage, commerce de détail, santé, loisir, banque…)
- hour_of_day, day_of_week, is_fraud

Les pays à risque ont été définis sur la base d'analyses sectorielles et rapports internationaux : Russie, Chine, Nigeria, Ukraine et Brésil. Les types de transaction et catégories de marchands incluent toutes les catégories courantes utilisées en France et à l'international.

3.2 Préparation et nettoyage des données

- Suppression des doublons
- Conversion des timestamps en format datetime
- Création de variables temporelles (heure, jour_de_la_semaine)
- Définition d'une variable métier country_risk
- Encodage des variables catégorielles en français
- Normalisation des variables numériques via StandardScaler

3.3 Analyse exploratoire (EDA)

- Les transactions frauduleuses présentent généralement des montants plus élevés.
- Certaines catégories de marchands et certains pays concentrent davantage de fraudes.
- Les transactions effectuées la nuit ou tôt le matin sont plus à risque.
- Les transactions dans les pays à risque présentent un risque plus élevé.

4. Modélisation et expérimentation Machine Learning

4.1 Méthodologie

1. Séparation des données : 80 % entraînement / 20 % test
2. Gestion du déséquilibre des classes via **SMOTE**
3. Entraînement du modèle **XGBoost**
4. Évaluation via : accuracy, précision, rappel, F1-score, matrice de confusion, ROC-AUC

4.2 Choix du modèle

Le modèle XGBoost a été retenu pour :

- Sa robustesse face aux jeux de données déséquilibrés
- Sa capacité à capturer des relations complexes
- Ses performances élevées sur des problèmes de classification binaire

4.3 Évaluation des performances

Résultats obtenus après SMOTE et ajustement du modèle :

- Accuracy : ~93,6 %
- ROC-AUC : 0,9676

- Précision et rappel : équilibre satisfaisant entre détection des fraudes et faux positifs
- La matrice de confusion et la courbe ROC confirment la performance robuste du modèle sur les données simulées.

5. Architecture technique

5.1 Stack technique

- Langage : Python
- Librairies Data / ML : Pandas, NumPy, Scikit-learn, XGBoost
- Visualisation : Matplotlib, Seaborn, Plotly
- Application interactive : Streamlit
- Sauvegarde modèle : joblib

5.2 Pipeline de traitement

1. Génération et chargement des données simulées
2. Feature engineering et normalisation
3. Entraînement du modèle XGBoost
4. Évaluation et visualisation des performances
5. Sauvegarde du modèle et du scaler
6. Chargement du modèle dans l'application Streamlit
7. Prédiction en temps réel via formulaire utilisateur
8. Mise à jour dynamique du dashboard avec chaque transaction testée

6. Dashboard et indicateurs de performance

6.1 Objectifs

Fournir une vision synthétique et interactive de la fraude bancaire et des performances du modèle, en temps réel, basée sur les transactions testées par les utilisateurs.

6.2 Indicateurs clés (KPI)

- Nombre total de transactions testées
- Nombre et taux de transactions frauduleuses

- Montant total et moyen des fraudes
- Répartition des fraudes par : pays, type de transaction, catégorie de marchand, heure de la journée

6.3 Fonctionnalités

- Mise à jour dynamique après chaque test de transaction
- Filtres interactifs pour explorer les transactions
- Graphiques interactifs : barres, boxplots, histogrammes
- Visualisation de tendances et points chauds de fraude

7. Déploiement et perspectives

7.1 Déploiement

Le projet est déployé sous forme d'application Streamlit, avec possibilité de tester les transactions et d'afficher le dashboard en temps réel.

7.2 Perspectives d'amélioration

- Intégration de données réelles anonymisées
- Mise en place d'un flux temps réel
- Ajustement du seuil de décision métier
- Ajout d'explicabilité avancée (SHAP)
- Déploiement industriel via API

8. Conclusion

Le projet **SécuTransac** illustre l'intérêt concret de l'utilisation du Machine Learning pour la détection de transactions bancaires frauduleuses.
Grâce à une approche complète, couvrant l'ensemble du cycle Data Science — de l'analyse des besoins métier à la mise en œuvre d'un dashboard interactif —, le modèle XGBoost a démontré une performance robuste sur les données simulées, avec un taux de détection élevé (~94 %).

Les visualisations (matrice de confusion, courbe ROC) et les KPI interactifs permettent de suivre efficacement la fraude et d'aider les équipes métier à prendre des décisions éclairées.
Enfin, le projet ouvre la voie à des évolutions futures : exploitation de données réelles anonymisées, détection en temps réel, ajustement du modèle aux besoins métiers et déploiement industriel via Docker ou API.

Damien MARIE , Baayvin SOUBRAMANIEN , Hugo ALCARAZ