Communauté Économique et Monétaire de l'Afrique Centrale
(CEMAC)
Institut Sous-régional de Statistique et d'Economie Appliquée
(ISSEA)
Organisation Internationale

Filière *: Master Data Science Modélisation Statistique* ***Classe*** *: 2ieme année*

PROJET DATA MINING
DETECTION FRAUDE BANCAIRE

Rédigé par :

Hermann Gael **NGAMBOU KAMETCHA**

Sous la supervision de :

Jean Christophe BOBDA

ISE-Enseignant associé à l'ISSEA, PhD Candidate

Artificiel Intelligence

Année académique 2024-2025

SOMMAIRE

SIGLES ET ABREVIATION

ROC: Receiver Operating Characteristic curve

AUC: Area Under the Curve

LISTES DES FIGURES

LISTE DES TABLEAUX

RESUME

Ce projet vise à développer un système automatisé de détection de fraudes dans les transactions financières, dans un contexte où la numérisation croissante s'accompagne d'une complexification des stratégies frauduleuses. Face aux limites des approches classiques, des techniques d'apprentissage supervisé ont été mobilisées sur un jeu de données simulées issues de la plateforme Kaggle (PaySim). L'analyse exploratoire approfondie a permis la création de variables dérivées (écarts de soldes, ratios, indicateurs temporels, etc.) afin d'enrichir la représentation des comportements transactionnels. Pour pallier le fort déséquilibre entre fraudes et transactions légitimes, un suréchantillonnage synthétique (SMOTE) a été appliqué sur le jeu d'entraînement.

Trois algorithmes ont été évalués : la régression logistique, la forêt aléatoire et le SVM linéaire. La régression logistique, bien que présentant un excellent rappel (94 %), souffre d'une très faible précision (2,4 %) en raison d'une sur détection liée à une forte dépendance à une seule variable clé, induisant de nombreux faux positifs. La forêt aléatoire améliore la précision (avec un rappel autour de 70 %) mais présente un risque de surapprentissage, les prédictions étant concentrées autour de quelques variables dominantes. En revanche, le SVM linéaire se distingue par son équilibre global, avec une précision de 94 % et un rappel de 85 %, sans surapprentissage notable. L'analyse des poids montre une contribution plus homogène des variables explicatives, ce qui confère au modèle robustesse et capacité de généralisation.

En conclusion, le SVM linéaire a été retenu comme modèle final en raison de ses performances équilibrées, de sa stabilité entre l'entraînement et le test, et de sa meilleure exploitation des signaux issus de l'ingénierie des variables. L'analyse comparative de l'importance des variables a été déterminante pour comprendre les dynamiques sous-jacentes à la détection de la fraude. Des perspectives d'amélioration incluent l'intégration de modèles plus complexes (comme XGBoost ou les réseaux de neurones), le déploiement en temps réel, ainsi que la conception d'un tableau de bord interactif à destination des analystes.

ABSTRACT

This project aims to develop an automated fraud detection system for financial transactions, in a context where increasing digitalization is accompanied by more sophisticated fraudulent behaviors. To overcome the limitations of traditional approaches, supervised learning techniques were applied to a simulated dataset from the Kaggle platform (PaySim). An in-depth exploratory data analysis led to the creation of derived variables (balance differences, ratios, time-based indicators, etc.) to better capture transactional behavior. To address the significant class imbalance between legitimate and fraudulent transactions, synthetic oversampling (SMOTE) was applied to the training set.

Three algorithms were tested and compared: logistic regression, random forest, and linear Support Vector Machine (SVM). Logistic regression achieved excellent recall (94%) but suffered from very low precision (2.4%) due to over-detection caused by heavy reliance on a single key variable, resulting in many false positives. The random forest model offered better precision (with recall around 70%) but showed signs of overfitting, as predictions were driven by a few dominant variables. In contrast, the linear SVM model demonstrated a balanced performance, achieving 94% precision and 85% recall without noticeable overfitting. The analysis of feature weights showed a more even distribution of influence across variables, contributing to the model's robustness and generalization capacity.

In conclusion, the linear SVM was selected as the final model for its balanced performance, training-test stability, and effective use of engineered features. The comparative analysis of variable importance played a key role in understanding the underlying mechanisms of fraud detection. Future improvements may include exploring more advanced models (such as XGBoost or neural networks), deploying the system in real time, and developing an interactive dashboard for analysts.

INTRODUCTION

Problématique

La fraude bancaire représente un enjeu majeur pour les institutions financières et les utilisateurs de services numériques. Avec l'augmentation constante des transactions électroniques, les tentatives de fraude se multiplient, prenant des formes de plus en plus sophistiquées. Les méthodes traditionnelles de détection fondées sur des règles fixes ou des contrôles manuels se révèlent souvent inefficaces face à cette évolution. Il devient alors indispensable de recourir à des approches automatisées, capables de détecter en temps réel les comportements suspects à partir de grandes quantités de données transactionnelles.

Objectif générale

L'objectif principal de ce projet est de développer un modèle prédictif de détection de la fraude bancaire à partir de données simulées. Il s'agit notamment de comprendre en profondeur la structure des données disponibles, d'identifier les variables significatives, puis de concevoir un modèle de classification capable de distinguer les transactions frauduleuses des transactions légitimes.

Source de donnée et méthodologie

Les données utilisées dans cette étude proviennent de la plateforme **Kaggle**, plus précisément du jeu de données intitulé **PaySim**, qui simule des opérations bancaires sur une période de 30 jours. Le jeu de données contient plusieurs millions de transactions classées selon leur type, leur montant et d'autres caractéristiques liées aux comptes des utilisateurs.

La méthodologie adoptée pour ce projet suit les étapes suivantes :

- Exploration et compréhension des données (analyse descriptive)
- Prétraitement et ingénierie des variables
- Séparation des données en ensemble d'entraînement et de test
- Entraînement de plusieurs modèles de classification (régression logistique, Random Forest, Séparateur à Vaste Marge)
- Évaluation comparative des performances à l'aide de métriques adaptées (précision, rappel, F1-score, courbe ROC-AUC)

Description des variables de l'étude

Le jeu de données comprend les variables suivantes :

- **step :** unité de temps en heures (de 1 à 744)
- **type :** type de transaction (CASH-IN, CASH-OUT, DEBIT, PAYMENT, TRANSFER)
- **amount :** montant de la transaction
- **nameOrig :** identifiant du client émetteur
- **oldbalanceOrg :** solde initial du client émetteur
- **newbalanceOrig :** solde du client émetteur après transaction
- **nameDest :** identifiant du client bénéficiaire
- **oldbalanceDest :** solde initial du bénéficiaire
- **newbalanceDest :** solde du bénéficiaire après transaction
- **IsFraud :** variable cible indiquant si la transaction est frauduleuse (1) ou non (0)
- **isFlaggedFraud :** indicateur de transaction suspecte détectée automatiquement selon une règle (montant supérieur à 200 000)

Ces variables offrent une base riche pour modéliser le comportement transactionnel et repérer les anomalies susceptibles d'être liées à des fraudes.

I. ANALYSE EXPLORATOIRES DES DONNEES

1. Aperçu générale des données

Le jeu de données contient des informations sur des transactions bancaires simulées, incluant leur type, leur montant, les comptes impliqués, et une indication sur la présence ou non de fraude.

Le jeu de données contient environ **6,3 millions** de transactions réparties sur **11 variables**.

On retrouve des variables de type numérique (comme le montant, l'ancien solde, le nouveau solde) et catégorielles (comme le type de transaction). Le champ isFraud est binaire et constitue la variable cible.

À l'analyse des données, aucune valeur manquante n'est présente, ce qui facilite leur traitement. De même, aucun doublon n'a été détecté. Cela suggère une bonne qualité initiale des données du point de vue de leur complétude et unicité.

2. Analyse des variables principales

Dans cette sous-section de l'EDA, nous analysons les variables principales ainsi que les relations qui existent entre elles. Certaines variables ne seront pas explorées de manière approfondie à ce stade, pour des raisons analytiques pertinentes. Toutefois, dans le cadre de l'ingénierie des variables, elles pourront être transformées ou combinées afin de dégager des informations utiles. Il s'agit notamment de :

- **step :** variable peu informative en l'état, car codée en heures abstraites. Elle pourrait toutefois être transformée pour permettre une analyse temporelle plus fine (par jour, semaine, etc.).
- **nameOrig et nameDest :** identifiants alphanumériques non exploitables directement. Cependant, un encodage approprié ou une analyse de fréquence pourrait révéler des comportements suspects ou des utilisateurs récurrents impliqués dans des fraudes.
- **isFlaggedFraud :** cette variable présente une variance très faible (quasi systématiquement égale à 0), ce qui limite son utilité brute. Elle pourrait néanmoins être utilisée pour construire des indicateurs de détection croisée ou comme référence dans l'élaboration de nouvelles variables.

a) Statistiques descriptives des variables

Tableau 1:*statistiques descriptives des variables numériques*

	amount	oldbalanceOrg	newbalanceOrig	oldbalanceDest	newbalanceDes
count	6.362620e+06	6.362620e+06	6.362620e+06	6.362620e+06	6.362620e+06
mean	1.798619e+05	8.338831e+05	8.551137e+05	1.100702e+06	1.224996e+06
std	6.038582e+05	2.888243e+06	2.924049e+06	3.399180e+06	3.674129e+06
min	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00
25%	1.338957e+04	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00
50%	7.487194e+04	1.420800e+04	0.000000e+00	1.327057e+05	2.146614e+05
75%	2.087215e+05	1.073152e+05	1.442584e+05	9.430367e+05	1.111909e+06
max	9.244552e+07	5.958504e+07	4.958504e+07	3.560159e+08	3.561793e+08

Source : *nos travaux/donnée de la plateforme Kaggle intitulé PaySim / sortie python et Excel*

Le tableau 1 présente les statistiques descriptives des principales variables numériques relatives aux transactions financières, à savoir : **amount**, **oldbalanceOrg, newbalanceOrig**, **oldbalanceDest et newbalanceDest**. Ces variables révèlent une forte hétérogénéité,

caractérisée par des écarts importants entre les valeurs minimales, médianes et maximales, ce qui traduit une distribution très asymétrique à droite.

Le montant moyen des transactions (amount) s'élève à environ **180 000 FCFA**, avec une valeur maximale dépassant les **92 millions de FCFA**, ce qui met en évidence l'existence de transactions particulièrement élevées, susceptibles d'être exceptionnelles ou suspectes.

Le solde initial des expéditeurs (oldbalanceOrg) est également très dispersé : la médiane est faible (**environ 14 208 FCFA**) alors que la moyenne dépasse **800 000 FCFA**, ce qui suggère que la plupart des comptes ont peu de fonds, mais qu'une minorité dispose de soldes exceptionnellement élevés.

Les médianes nulles observées pour newbalanceOrig et oldbalanceDest indiquent que plus de 50 % des expéditeurs terminent leurs transactions avec un solde nul, et que de nombreux destinataires reçoivent des fonds à partir d'un solde initial nul. Le solde final des bénéficiaires (newbalanceDest) est en moyenne supérieur à leur solde initial, ce qui confirme logiquement la réception de fonds.

Les **figures 4 et 5** (en annexe) présentent des diagrammes en barres illustrant la distribution des variables qualitatives **type** (type de transaction) et **isFraud** (indicateur de fraude).

La majorité des transactions sont de type **CASH_OUT (environ 2,24 millions)**, **PAYMENT (≈2,15 millions)** et CASH_IN (≈1,4 million). Les types TRANSFER (≈530 000) et DEBIT (≈41 000) sont beaucoup moins fréquents.

Cette distribution reflète la prédominance des opérations de retrait et de paiement dans le jeu de données, tandis que les virements (type TRANSFER) sont relativement rares mais potentiellement plus sensibles à la fraude, comme le soulignent plusieurs études.

Concernant la variable isFraud, sur plus de 6,36 millions de transactions, seules 8 213 ont été identifiées comme frauduleuses, soit environ 0,13 % du total. Cela confirme un fort déséquilibre des classes, problématique classique en détection de fraude.

Les algorithmes de classification devront donc être conçus pour reconnaître ces cas rares, sans être dominés par la classe majoritaire (transactions non frauduleuses).

b) Analyse des relations entre les variables principales

La matrice de corrélation (figure 1) met en évidence les relations entre les principales variables numériques du jeu de données, y compris l'indicateur de fraude isFraud. Plusieurs enseignements peuvent être tirés de cette analyse :

Corrélations quasi-parfaites entre certaines variables de solde :

On observe une corrélation parfaite entre oldbalanceOrg et newbalanceOrig (1.00), ainsi

qu'une corrélation très forte entre oldbalanceDest et newbalanceDest (0.98). Ces relations traduisent une forte dépendance structurelle entre les soldes avant et après transaction, ce qui suggère une colinéarité. Dans le cadre de modèles linéaires tels que la régression logistique, cela peut nuire à l'interprétation des coefficients et à la stabilité des estimations. Une réduction de dimension ou la transformation de ces variables pourrait être envisagée.

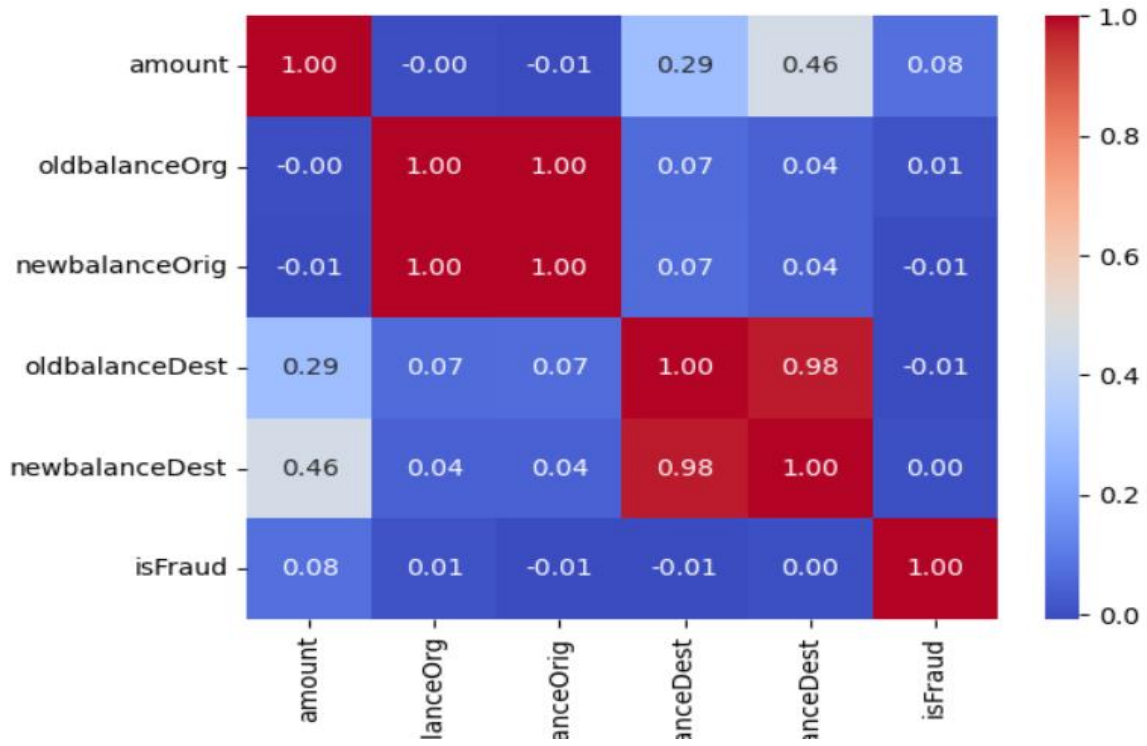
Faible corrélation entre isFraud et les autres variables :

L'indicateur de fraude présente des corrélations très faibles avec les autres variables numériques, toutes proches de zéro (comprises entre -0.01 et 0.08). Cela indique l'absence de lien linéaire direct entre les variables de solde ou le montant et la survenue d'une fraude. Ce phénomène est courant dans les problématiques de détection de fraude, où les signaux sont souvent faibles ou masqués, nécessitant des méthodes plus complexes ou la création de variables dérivées pour capturer les comportements suspects.

Comportement du montant (amount) :

Le montant de la transaction est modérément corrélé avec newbalanceDest (0.46) et plus faiblement avec oldbalanceDest (0.29), tandis qu'il est presque indépendant des soldes de l'émetteur (oldbalanceOrg et newbalanceOrig, corrélations proches de 0). Cela suggère que le montant impacte davantage le solde du destinataire que celui de l'émetteur.

Figure 1:matrix de corrélation des variables



Source *: nos travaux/donnée de la plateforme Kaggle intitulé PaySim / sortie python*

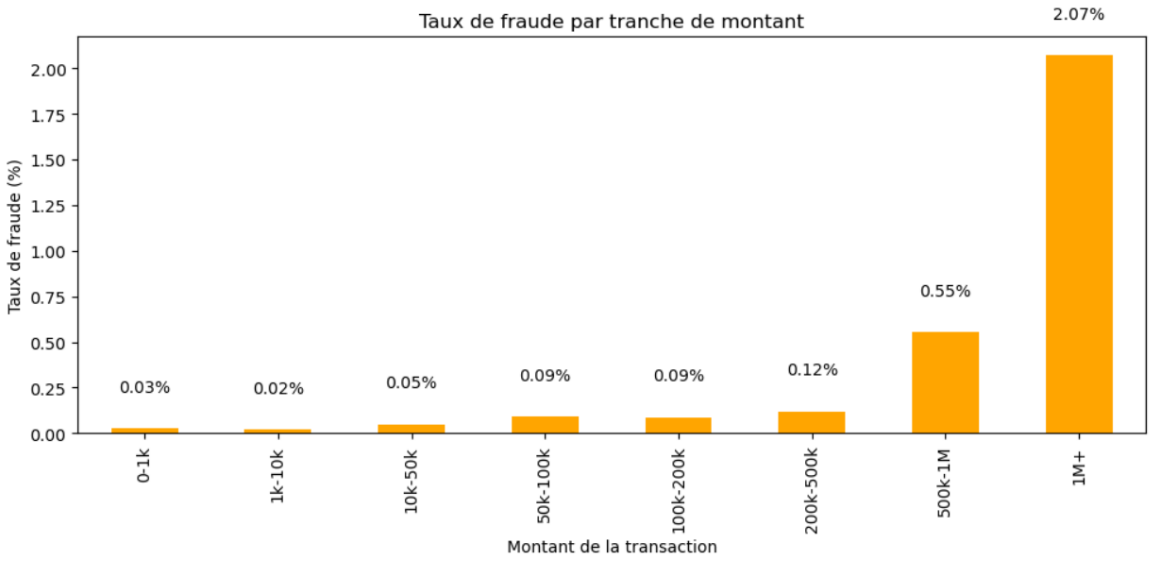
La figure 6 :(annexe) met en évidence, à travers deux boxplots, une nette différence de distribution des montants entre les transactions frauduleuses et non frauduleuses. Dans le cas des transactions non frauduleuses (isFraud = 0), les montants sont majoritairement concentrés entre 1 000 et 100 000 unités monétaires (entre 10^3 et 10^5), avec une médiane relativement basse et la présence de nombreuses valeurs extrêmes dépassant les 10^6, traduisant une grande dispersion des montants. À l'inverse, les transactions frauduleuses (isFraud = 1) présentent des montants globalement plus élevés, avec une médiane nettement supérieure et une concentration marquée dans la tranche 10^4–10^6. On observe également une moindre dispersion vers les petits montants, ce qui suggère que les fraudes se concentrent principalement sur des transactions à fort enjeu. Ces observations traduisent une stratégie potentielle des fraudeurs visant à maximiser les gains en ciblant des montants élevés tout en limitant le nombre d'attaques. Cette différence de comportement offre une piste intéressante pour le développement de modèles de détection, notamment en tenant compte du montant comme indicateur discriminant.

La figure 7 (annexe) : met clairement en évidence une forte concentration des cas de fraude dans deux types spécifiques de transactions : les opérations de type TRANSFER concentrent l'essentiel des fraudes, tandis que les CASH_OUT en regroupent également, mais dans une proportion bien moindre. À l'inverse, les types PAYMENT, DEBIT et CASH_IN n'enregistre pratiquement aucun cas de fraude, ce qui révèle une forte hétérogénéité dans la distribution du risque selon le type de transaction. Cette observation suggère que les fraudeurs ciblent préférentiellement certains canaux, probablement parce que ceux-ci permettent de transférer rapidement des montants importants ou présentent des vulnérabilités exploitables. De plus, l'analyse relative des taux montre que le taux de fraude associé aux TRANSFER est plus de quatre fois supérieur à celui des CASH_OUT, renforçant l'idée d'un usage stratégique de ce type d'opération dans les schémas frauduleux. En conséquence, le type de transaction apparaît comme une variable catégorielle fortement discriminante : sa simple prise en compte permet déjà de distinguer des zones à risque (TRANSFER et CASH_OUT) de zones à faible risque (PAYMENT, DEBIT, CASH_IN). Cette caractéristique en fait un excellent candidat pour l'inclusion dans un modèle de détection de fraude, qu'il s'agisse de modèles linéaires comme la régression logistique ou d'approches plus complexes comme les arbres de décision ou les forêts aléatoires

La figure 2 ci-dessous illustre l'évolution du taux de fraude en fonction des tranches de montants des transactions. On y observe que plus le montant d'une transaction est élevé, plus la probabilité qu'elle soit frauduleuse augmente. Ainsi, les transactions de faible valeur (inférieures à 1 000 unités monétaires) présentent un taux de fraude très bas, autour de 0,03

%, tandis que celles dépassant un million atteignent un taux de fraude supérieur à 2 %. Cette progression quasi monotone suggère une stratégie des fraudeurs visant à maximiser le gain en ciblant préférentiellement les transactions de grande valeur. Ce constat est d'autant plus important que, même si les petites transactions sont beaucoup plus nombreuses, leur poids dans la fraude globale reste marginal. L'analyse met donc en évidence une relation croissante entre le montant et le risque de fraude, soulignant l'intérêt d'accorder une attention particulière aux transactions élevées dans les systèmes de détection automatique.

***Figure 2**:Taux de fraude par tranche de montant*



Source *：nos travaux/donnée de la plateforme Kaggle intitulé PaySim / sortie python*

c) Comportements anormaux

Le tableau 2 ci-dessous met en évidence deux cas particuliers de transactions présentant des comportements atypiques, aux implications contrastées pour la détection de fraude.

Dans **le premier cas**, le solde de l'expéditeur ne diminue pas après la transaction, ce qui est incohérent d'un point de vue comptable. Toutefois, **seulement 41 cas** de fraude sont observés sur plus de **2 millions de transactions**, soit un taux extrêmement faible (~0.002 %). Ce comportement, bien que suspect en apparence, semble davantage refléter des anomalies techniques, des transactions annulées ou des enregistrements incomplets. Il présente donc peu d'intérêt pratique pour un modèle de détection automatisé, car il génère un grand nombre de faux positifs.

Dans **le deuxième cas**, le solde du destinataire reste inchangé après la transaction. Ce scénario concerne plus de **2,3 millions d'opérations**, dont plus de **4 000 cas de fraude**, soit un taux significatif de **0.176 %.** Ce type de comportement pourrait signaler des tentatives de dissimulation, comme des transferts vers des comptes fictifs ou des interruptions volontaires. Il

constitue ainsi un indicateur plus pertinent pour identifier des transactions potentiellement frauduleuses.

En conclusion, tandis que le cas de l'expéditeur inchangé semble peu exploitable en pratique, celui du destinataire inchangé apparaît comme un signal d'alerte utile, à intégrer dans une approche combinée de détection fondée sur plusieurs variables comportementales.

Tableau 2:*Analyse des soldes inchanche et lien avec la fraude*

Cas	Total des transactions	Nombre de fraudes	Taux de fraude
Cas 1 – Expéditeur inchangé	2 089 021	41	**0.002%**
Cas 2 – Destinataire inchangé	2 317 276	4 070	**0.176%**

Source *: nos travaux/donnée de la plateforme Kaggle intitulé PaySim / sortie python*

II. INGENIERIE DES VARIABLES

L'analyse exploratoire montre qu'aucune variable prise isolément ne permet de prédire efficacement la fraude. Cela souligne la nécessité de croiser, transformer et combiner plusieurs variables pour faire émerger des signaux pertinents. Ce besoin motive la mise en œuvre d'une ingénierie des variables (feature engineering) adaptée.

- Gestion de la colinéarité et création de variables dérivées

Certaines variables présentent une corrélation parfaite ou très forte, notamment :

oldbalanceOrg et newbalanceOrig (corrélation = 1.00)

oldbalanceDest et newbalanceDest (corrélation ≈ 0.98)

Pour atténuer les effets de colinéarité structurelle, nous proposons de créer des différences de solde :

diff_orig = oldbalanceOrg - newbalanceOrig

diff_dest = newbalanceDest - oldbalanceDest

Si nécessaire, l'une des deux variables corrélées pourra être supprimée.

- Corrélation du montant avec les soldes

Le montant de la transaction (amount) :

Est modérément corrélé avec newbalanceDest (0.46)

A une corrélation plus faible avec oldbalanceDest (0.29)

Est quasiment indépendant des soldes de l'émetteur (oldbalanceOrg, newbalanceOrig ≈ 0)

Cela justifie la création de ratios relatifs pour mieux capter les dynamiques sous-jacentes :

ratioAmountOrig = amount / (oldbalanceOrg + 1)

ratioAmountDest = amount / (oldbalanceDest + 1)

- Création d'indicateurs comportementaux

Afin de capter des comportements suspects, nous introduisons les variables suivantes :

no_balance_change_sender : indique si le solde de l'émetteur n'a pas changé malgré un montant > 0 (situation très suspecte).

isHighAmount : identifie les transactions à montant élevé, souvent ciblées par les fraudeurs.

hour (extrait de step) : capture l'heure de la journée, certaines plages horaires étant plus propices à la fraude.

isMerchantDest : identifie si le destinataire est un marchand (préfixe 'M'), souvent visé dans les fraudes.

isRiskyType : indique si la transaction est de type TRANSFER ou CASH_OUT, nettement surreprésentés dans les cas frauduleux.

En somme Les transformations proposées ne visent pas simplement à "embellir" les données : elles visent à mettre en lumière des comportements atypiques (incohérences de soldes, transferts suspects, montants anormalement élevés) et à fournir au modèle des signaux explicatifs puissants pour améliorer la détection des fraudes

III. MODELISATION

1. Objectif de la modélisation et méthodologie

L'objectif principal de cette phase est de **construire un modèle supervisé performant**, capable de **détecter efficacement les transactions frauduleuses** à partir des variables sélectionnées et transformées lors de l'étape d'ingénierie des variables.

Trois algorithmes de classification supervisée ont été testés et comparés :

- **Régression logistique**
- **Forêt aléatoire (Random Forest)**
- **Machine à vecteurs de support (Support Vector Machine - SVM)**

Stratégies d'équilibrage des classes

Compte tenu de la forte **déséquilibration des classes** (très faible proportion de fraudes), deux approches d'équilibrage ont été mises en œuvre :

- **Sur-échantillonnage synthétique (SMOTE) :**

Appliqué aux modèles de régression logistique et de forêt aléatoire, cette technique génère artificiellement de nouveaux exemples de la classe minoritaire (fraude) afin d'obtenir un ratio équilibré de 1 :1 dans le jeu d'entraînement.

- **Échantillonnage manuel** :

Pour le modèle SVM, un sous-échantillon du jeu initial a été constitué avec un ratio de 1:5 (une fraude pour cinq non-fraudes). Ce choix vise à conserver une représentation plus proche des proportions réelles tout en permettant un apprentissage efficace, en tenant compte de la sensibilité du SVM aux volumes de données.

Évaluation des performances

L'évaluation a été réalisée à partir de jeux de test distincts, non rééchantillonnés, pour garantir une mesure réaliste de la capacité de généralisation des modèles sur des données représentatives du contexte réel (avec fort déséquilibre).

2. Méthodologie

Pour chacun des modèles, les étapes suivantes ont été respectées :

- Prétraitement : centrage et réduction des variables via StandardScaler.
- Échantillonnage équilibré des classes.
- Séparation entraînement/test.
- Entraînement du modèle.
- Évaluation sur les jeux d'entraînement et de test avec :
    - Matrice de confusion
    - Rapport de classification (précision, rappel, F1-score)
    - Courbe ROC et AUC

3. Résultats compares des modèles

Tableau 3:*Resultats compares des modèles*

Modèle	Ensemble	Accuracy	Précision (Fraude)	Rappel (Fraude)	F1-score (Fraude)	ROC AUC
Régression logistique	Test	95.1%	**2.4%**	**94.3%**	4.7%	**0.9903**
	Train	96.0%	95.2%	96.8%	96.0%	0.9928
Random Forest	Test	**99.9%**	92%	70%	80%	0.9383
	Train	**100%**	100%	100%	100%	0.9999
SVM (linéaire)	Test	97.0%	**94%**	85%	**89%**	**0.9909**
	Train	97.0%	93%	85%	89%	0.9910

Source *：nos travaux/donnée de la plateforme Kaggle intitulé PaySim / sortie python*

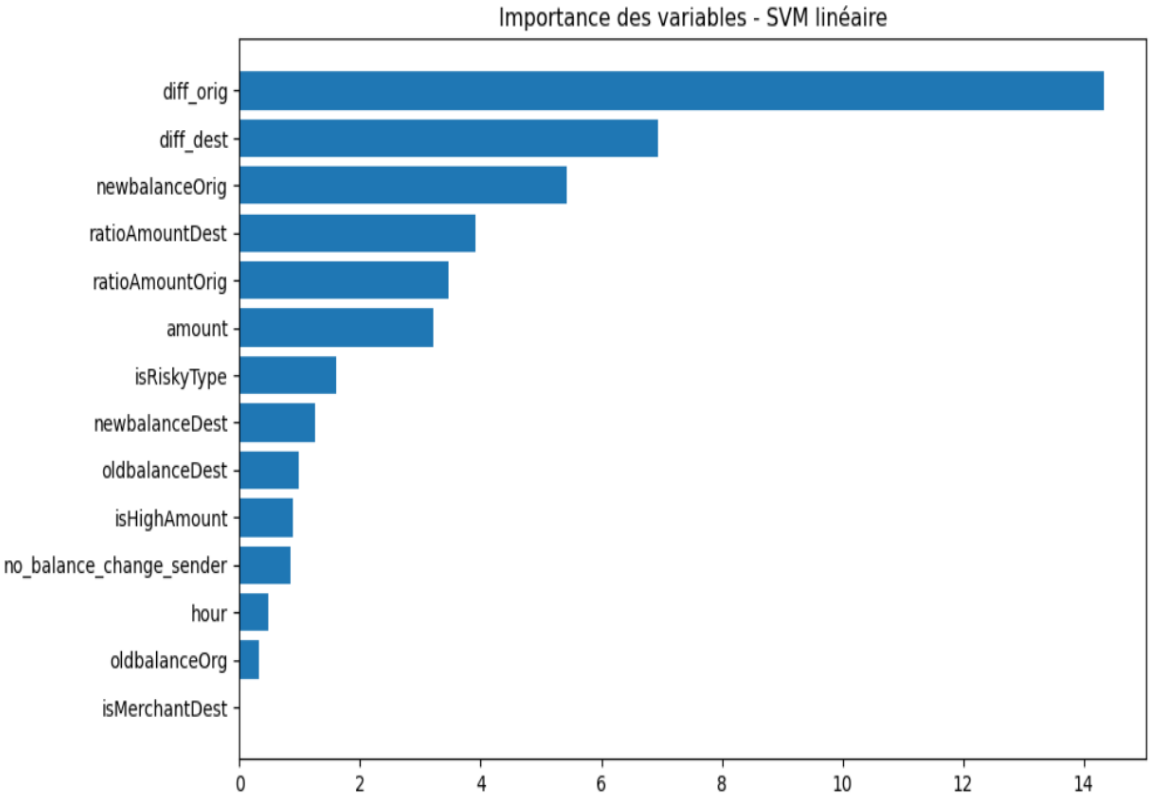
4. Interprétations des résultats

- **Régression logistique :** Ce modèle affiche un excellent rappel (94 %), ce qui signifie qu'il détecte efficacement la majorité des fraudes. Cependant, sa précision est très faible (2,4 %), générant un grand nombre de faux positifs. L'analyse des importances des variables (figure 8 en annexe) montre une forte dominance de la variable diff_orig (contribution maximale), suivie par isRiskyType avec une influence secondaire. Les autres variables ont une influence marginale. Cette forte pondération sur un petit nombre de variables indique un modèle simpliste qui tend à surévaluer certains facteurs, ce qui explique le déséquilibre entre précision et rappel.
- **Random Forest :** Ce modèle présente une bonne précision mais un rappel plus faible (70 %), ce qui signifie qu'il manque une part importante des fraudes. Le modèle montre des signes d'overfitting sur l'ensemble d'entraînement. L'importance des variables (figure 9, en annexe) met en avant diff_orig comme variable principale, avec ratioAmountDest et isRiskyType comme variables secondaires. L'arbre est donc plus complexe, optimisant fortement certaines interactions au détriment de la capacité à détecter toutes les fraudes.

- **SVM linéaire (modèle retenu) :** Le SVM offre un excellent compromis avec une précision élevée (94 %) et un rappel satisfaisant (85 %), sans signe de surapprentissage. L'importance des variables (figure 3) est plus équilibrée que dans les autres modèles, avec des contributions comprises entre 2 et 14. Les cinq variables les plus influentes sont diff_orig, diff_dest, newbalanceOrigin, ratioAmountDest, et ratioAmountOrigin. Les variables telles que hour, oldbalanceOrigin et isMerchantDest ont un impact plus faible (<4). Cette répartition plus homogène des importances contribue à une meilleure capacité de généralisation.

Compte tenu de la performance globale, de la stabilité entre entraînement et test, du bon équilibre entre précision et rappel, ainsi que de la compréhension fine des facteurs clés grâce à l'analyse des importances dans les différents modèles, le SVM linéaire a été retenu comme modèle final pour la détection des fraudes.

Figure 3:Importance des variables-SVM-lineaire



Source *: nos travaux/donnée de la plateforme Kaggle intitulé PaySim / sortie python*

CONCLUSION

Ce projet avait pour objectif principal de développer un système automatique de détection des fraudes dans les transactions financières, en s'appuyant sur des techniques de science des données. Nous avons adopté une démarche rigoureuse, allant de l'analyse exploratoire des données à l'évaluation comparative de plusieurs algorithmes de classification.

Dans un premier temps, une analyse exploratoire des données (EDA) a été réalisée afin de mieux comprendre la structure du jeu de données, détecter le déséquilibre marqué entre classes, et explorer les relations entre variables. Cette étape a permis d'identifier les caractéristiques pertinentes, de repérer des anomalies, et d'orienter la conception de variables dérivées.

L'EDA a notamment révélé :

- Un fort déséquilibre entre transactions légitimes et frauduleuses,
- L'importance du type de transaction dans la survenue des fraudes, notamment les types TRANSFER et CASH_OUT,
- Des incohérences dans les soldes avant et après certaines opérations.

Sur cette base, une ingénierie des variables a été menée, avec la création de nouvelles variables dérivées (ratios, différences de soldes, indicateurs horaires, types de clients, etc.) afin de mieux capturer les schémas typiques de fraude.

Côté apprentissage automatique, trois modèles supervisés ont été implémentés et comparés :

- La régression logistique : simple et interprétable, mais moins performante sur les fraudes rares ;
- La forêt aléatoire (Random Forest) : très performante, capable de capturer des interactions complexes tout en fournissant des indicateurs d'importance des variables ;
- Le SVM linéaire (Support Vector Machine) : efficace après équilibrage des classes, avec un bon compromis entre précision et rappel.

Les résultats montrent que la forêt aléatoire a fourni les meilleures performances globales sur l'ensemble de test, avec un rappel et une précision équilibrée, et un ROC AUC supérieur à 0,93. L'analyse des importances des variables a permis d'identifier les facteurs clés influençant la prédiction de la fraude, tels que le montant des transactions, la différence entre soldes avant et après, ainsi que le type de transaction.

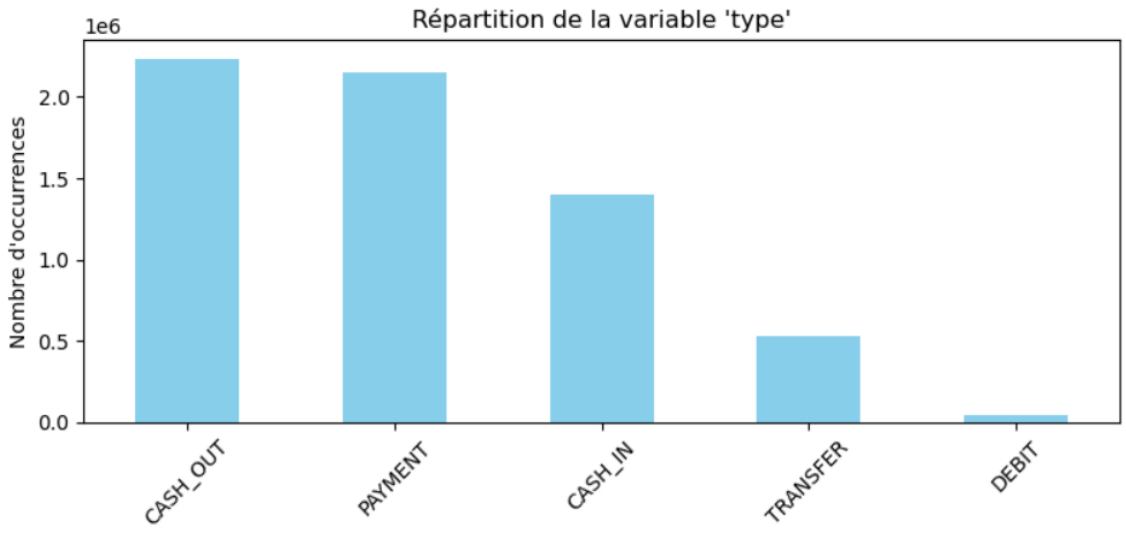
Enfin, ce projet met en lumière l'intérêt crucial des techniques d'équilibrage des classes (comme SMOTE ou le sous-échantillonnage contrôlé) pour améliorer la détection d'un phénomène naturellement déséquilibré.

Perspectives d'amélioration

- Ce travail pourrait être enrichi par :
- L'utilisation de modèles plus complexes, tels que les réseaux de neurones profonds ou les méthodes de gradient boosting (XGBoost, LightGBM) ;
- Le déploiement en temps réel avec un système d'alerte automatique ;
- La création d'un tableau de bord interactif pour les analystes ;
- L'intégration d'un système complet de détection en temps réel.

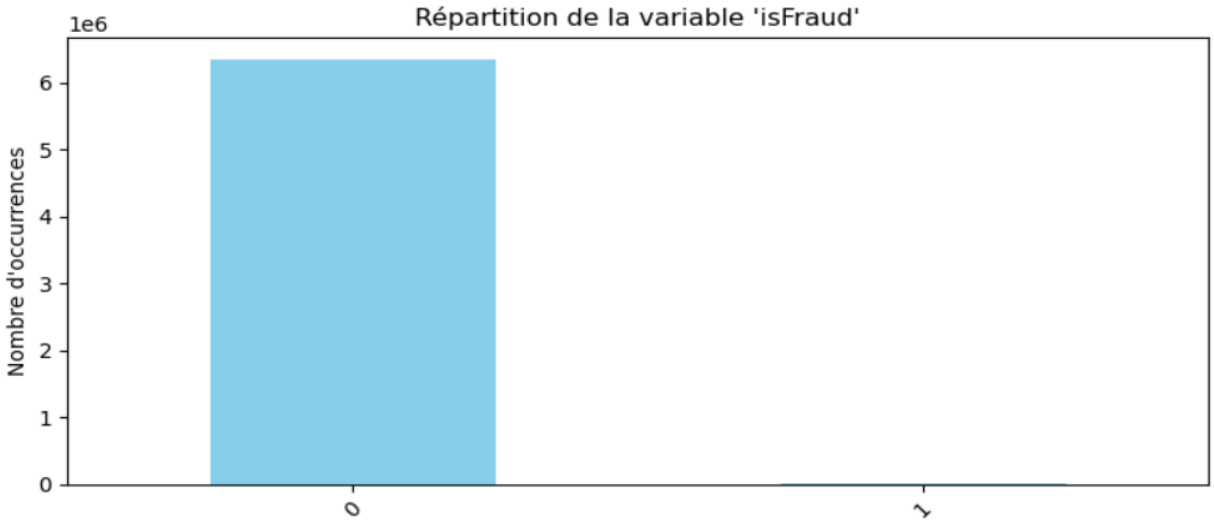
ANNEXE

***Figure 4**:Répartition de la variable "type"*



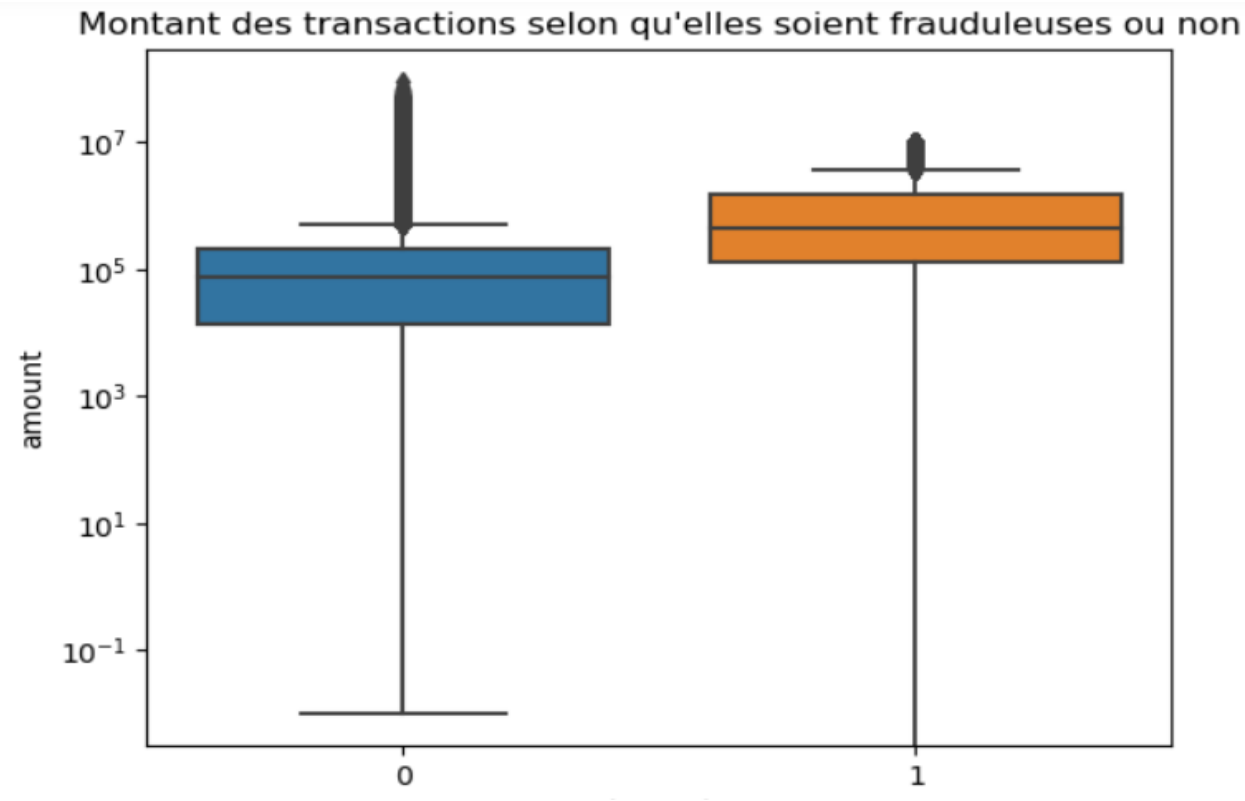
Source *：nos travaux/donnée de la plateforme Kaggle intitulé PaySim / sortie python*

***Figure** 5:Repartition de la variable "isFraud"*



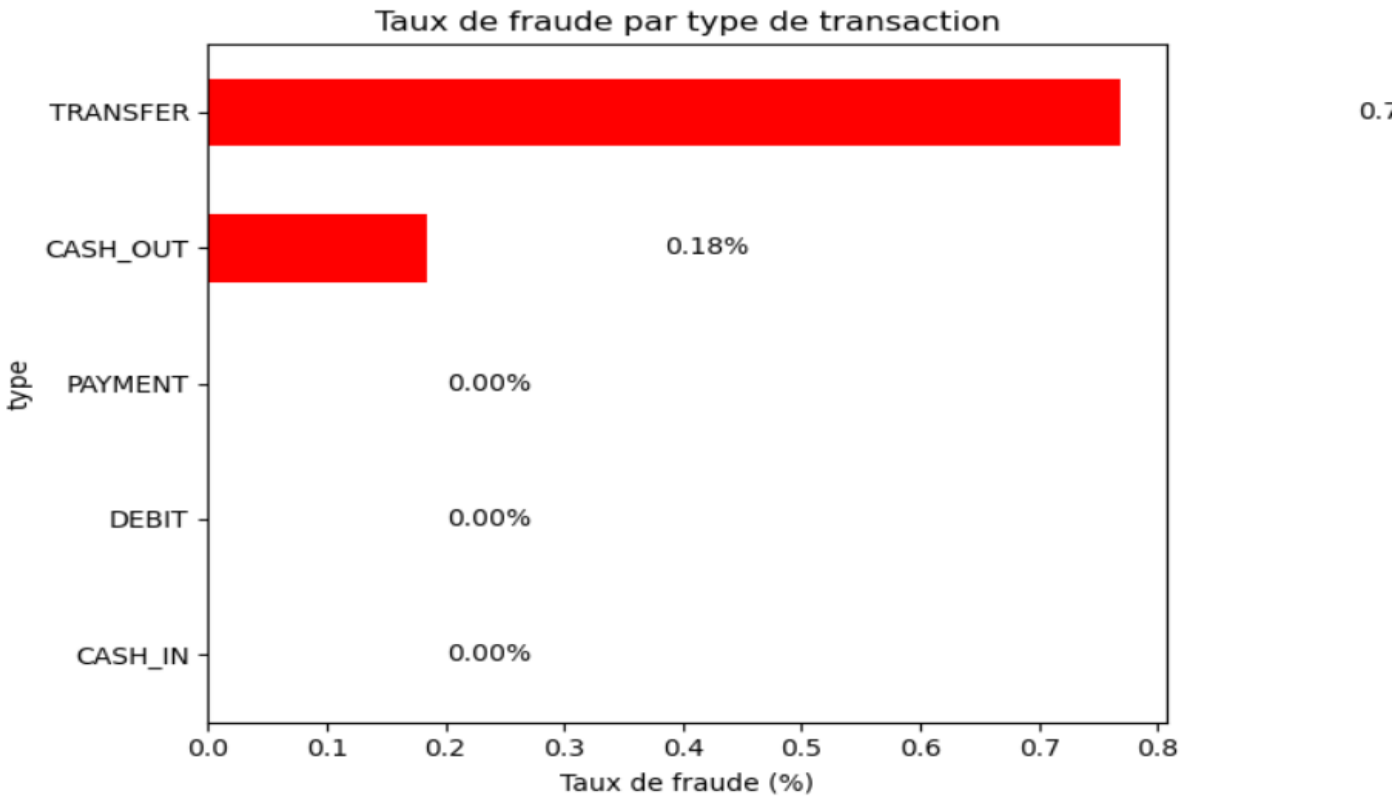
Source *：nos travaux/donnée de la plateforme Kaggle intitulé PaySim / sortie python*

***Figure 6:**Montant transaction selon la variable cible*



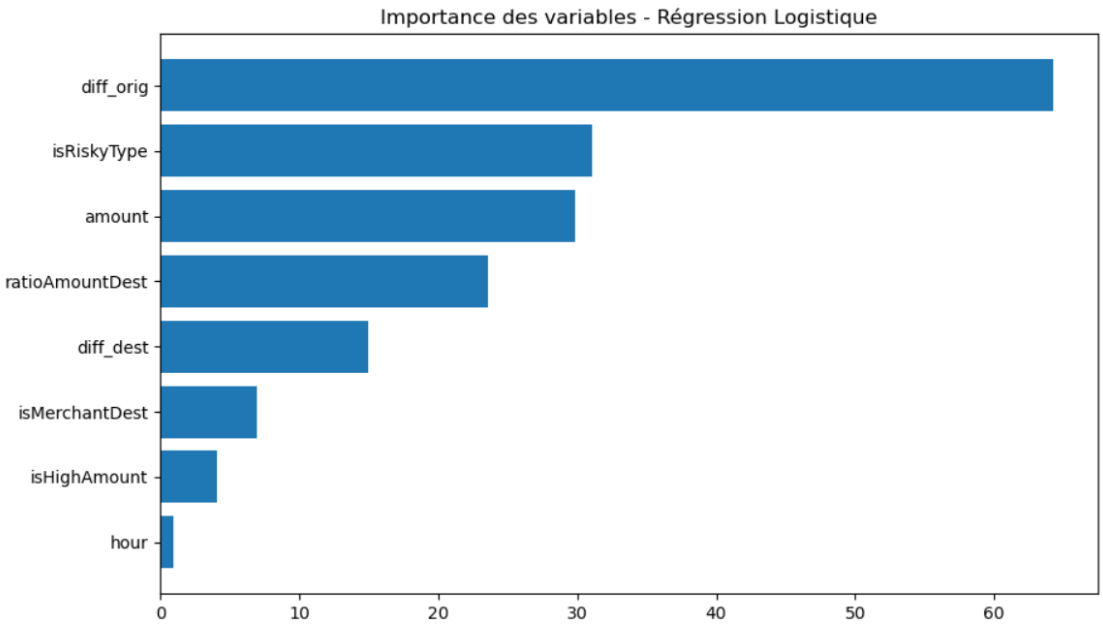
Source *：nos travaux/donnée de la plateforme Kaggle intitulé PaySim / sortie python*

Figure 7:taux de fraude par type de transaction



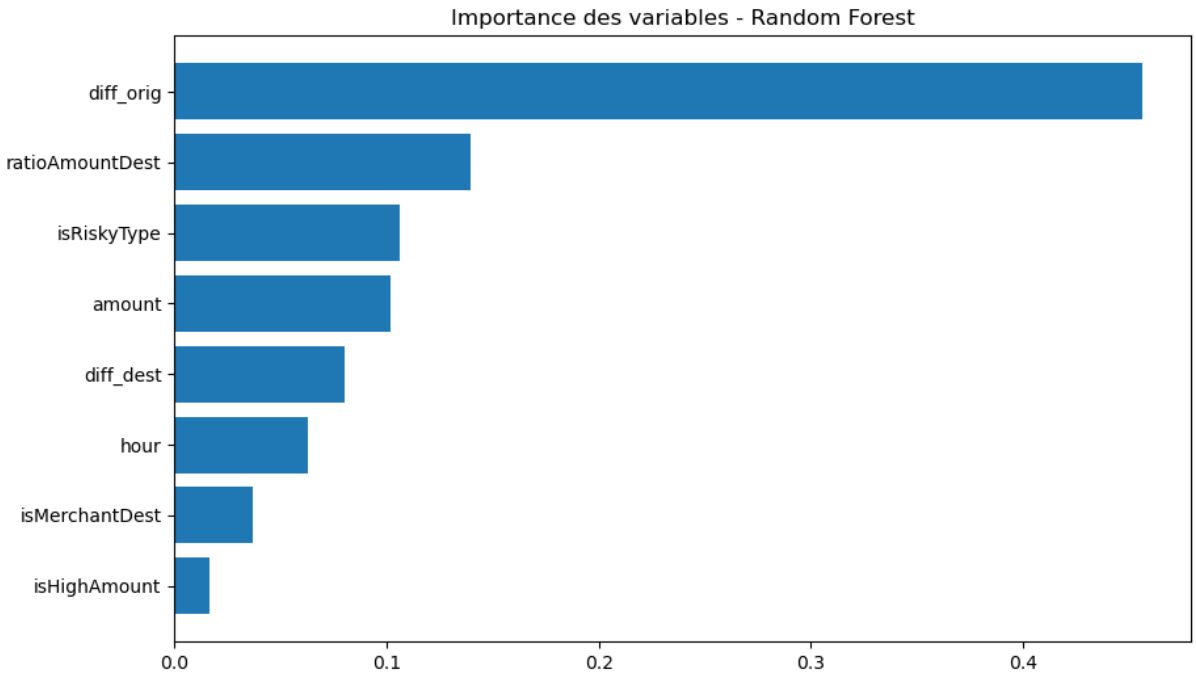
Source *: nos travaux/donnée de la plateforme Kaggle intitulé PaySim / sortie python*

Figure 8:Importance des variables-Regression linéaire



Source *: nos travaux/donnée de la plateforme Kaggle intitulé PaySim / sortie python*

***Figure 9**:Importance des variables -Random Forest*



Source : nos travaux/donnée de la plateforme Kaggle intitulé PaySim / sortie python