REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

Université SAÁD DAHLAB –BLIDA

Faculté des sciences

Département Informatique

Mémoire de fin d'études

En vue de l'obtention du diplôme de Master

Thème :

Identification de comportements frauduleux dans les transactions bancaires par clustering.

Présenté par :

***MAKERI Atika** ***SAHRAOUI Samira**

Proposé et encadré par :

Dr. Célia HIRECHE

2024/2025

Dédicace

Je dédie ce mémoire, fruit d'un long parcours d'efforts et de persévérance, à toutes les personnes qui ont marqué mon chemin de près ou de loin.

À mon époux, pour son amour indéfectible, sa patience sans limite et son soutien constant. Ta présence rassurante et tes encouragements m'ont portée dans les moments les plus difficiles. Merci d'avoir toujours cru en moi.

À mes enfants, véritables sources de lumière, de motivation et de joie, qui me donnent chaque jour la force d'avancer et de me surpasser.

À mes chers parents, pour leur éducation, leurs prières sincères et leur soutien moral sans faille. Votre confiance et vos valeurs m'accompagnent à chaque étape de ma vie.

À ma famille et à mes proches, pour leur bienveillance, leurs mots encourageants et leur présence réconfortante.

À toutes les étudiantes de ma section, avec qui j'ai partagé des moments riches en entraide, en motivation et en solidarité. Votre énergie m'a portée tout au long de cette aventure.

Et enfin, à ma chère binôme, Mme Makeri, avec qui j'ai eu le plaisir de collaborer. Merci pour ton professionnalisme, ta gentillesse et l'esprit d'équipe qui a rendu ce travail à la fois enrichissant et agréable.

À vous toutes et tous, ma profonde gratitude.

S.Samira

Dédicace

Je dédie ce mémoire, fruit d'un long parcours d'efforts et de persévérance, à toutes les personnes qui ont marqué mon chemin de près ou de loin.

À mon époux, pour son amour indéfectible, sa patience sans limite et son soutien constant. Ta présence rassurante et tes encouragements m'ont portée dans les moments les plus difficiles. Merci d'avoir toujours cru en moi.

À mes enfants, véritables sources de lumière, de motivation et de joie, qui me donnent chaque jour la force d'avancer et de me surpasser.

À mes chers parents, pour leur éducation, leurs prières sincères et leur soutien moral sans faille. Votre confiance et vos valeurs m'accompagnent à chaque étape de ma vie.

À mes frères, Mohammed et Ahmed, pour leur aide précieuse, leur disponibilité et leur soutien constant tout au long de mes études.

À ma famille et à mes proches, pour leur bienveillance, leurs mots encourageants et leur présence réconfortante.

À toutes les étudiantes de ma section (Chafica,Sara,Ikram,Houda,Ahlem,Aicha,….),, avec qui j'ai partagé des moments riches en entraide, en motivation et en solidarité. Votre énergie m'a portée tout au long de cette aventure.

Et enfin, à ma chère binôme, Mme Sahraoui, avec qui j'ai eu le plaisir de collaborer. Merci pour ton professionnalisme, ta gentillesse et l'esprit d'équipe qui a rendu ce travail à la fois enrichissant et agréable.

À vous toutes et tous, ma profonde gratitude.

M.Atika

Remerciement

Avant tout, nous remercions **Dieu** Tout-Puissant pour nous avoir accordé la force, la patience et la persévérance nécessaires à la réalisation de ce mémoire.

Nous tenons à exprimer notre profonde gratitude à toutes les personnes qui ont contribué, de près ou de loin, à la réalisation de ce travail.

Nous remercions tout particulièrement notre encadrante universitaire, **Dr. Célia Hireche**, pour son accompagnement constant, ses conseils avisés, sa disponibilité et sa bienveillance tout au long de ce travail. Son expertise et sa rigueur ont été d'une grande valeur et nous ont permis d'avancer sereinement dans cette recherche.

Nous remercions également **les membres du jury** pour l'intérêt qu'ils ont porté à notre travail, ainsi que pour leurs remarques pertinentes et constructives qui enrichissent cette étude.

Nos remerciements s'adressent aussi à **l'ensemble du corps enseignant** du département, pour la qualité de l'enseignement dispensé, leur engagement, leur générosité intellectuelle, et leur contribution à notre formation académique et personnelle tout au long de ces années.

Nous remercions tout particulièrement **Monsieur Rahmani Abdelfateh** pour son aide précieuse, sa disponibilité et ses conseils techniques qui ont été d'un grand soutien dans l'accomplissement de ce travail.

Nous n'oublions pas **nos familles et nos proches**, dont le soutien moral, la patience et les encouragements constants ont été un pilier essentiel tout au long de notre parcours. Enfin, merci à **nos camarades de promotion** pour les échanges, l'entraide et les moments de convivialité partagés, qui ont rendu cette expérience universitaire enrichissante et humaine.

Résumé

Dans le contexte actuel marqué par la digitalisation croissante des services bancaires, la détection des fraudes financières est devenue un enjeu prioritaire. Ce travail de recherche propose une approche hybride combinant des techniques de fouille de données, notamment le clustering non supervisé et l'extraction de motifs fréquents, pour identifier les comportements suspects dans les transactions bancaires.

La première partie du travail s'intéresse aux différentes formes de fraudes, telles que la fraude par carte bancaire, le vol d'identité ou encore la fraude hypothécaire. Les méthodes de détection traditionnelles, basées sur des systèmes de règles, se révèlent insuffisantes face à l'évolution des techniques des fraudeurs. L'apprentissage automatique, en particulier le clustering, permet d'explorer les données sans étiquettes et de détecter des anomalies de manière plus efficace.

Dans ce cadre, deux algorithmes de clustering ont été utilisés :

- **K-means**, basé sur le partitionnement, a permis de segmenter les transactions en deux groupes distincts. La méthode du coude a validé l'existence de deux clusters correspondant, dans l'interprétation métier, aux transactions normales et potentiellement frauduleuses.
- **DBSCAN**, un algorithme basé sur la densité, s'est avéré performant pour détecter des groupes de comportements atypiques ainsi que des points isolés, considérés comme des anomalies.

Parallèlement, l'extraction de motifs fréquents via l'algorithme **Apriori** a permis d'identifier des associations récurrentes entre certaines caractéristiques des transactions frauduleuses, renforçant ainsi l'interprétabilité des résultats et la compréhension des stratégies employées par les fraudeurs.

Les expérimentations ont été menées sur un jeu de données transactionnelles réel, après un prétraitement rigoureux incluant le nettoyage, l'ingénierie de caractéristiques et la mise à l'échelle. Les résultats ont montré que l'approche combinée clustering et motifs fréquents améliore significativement la précision de la détection et permet d'isoler efficacement les comportements anormaux.

Cette méthodologie offre ainsi une solution robuste, proactive et interprétable pour renforcer les systèmes de détection des fraudes dans les institutions bancaires, tout en ouvrant des perspectives pour des approches encore plus sophistiquées et adaptées à l'évolution des pratiques frauduleuses.

Mots clés : fraude bancaire, clustering, DBSCAN, K-means, fouille de données, motifs fréquents.

Abstract

In the current context marked by the growing digitalization of banking services, the detection of financial fraud has become a major priority. This research work proposes a hybrid approach combining data mining techniques, particularly unsupervised clustering and frequent pattern mining, to identify suspicious behaviors in banking transactions.

The first part of the work focuses on the different forms of fraud, such as credit card fraud, identity theft, and mortgage fraud. Traditional detection methods, based on rule-based systems, have proven insufficient in the face of increasingly sophisticated fraud techniques. Machine learning, particularly clustering, enables the exploration of unlabeled data and the more effective detection of anomalies.

In this context, two clustering algorithms were used:

- **K-means**, a partitioning-based algorithm, was used to segment the transactions into two distinct groups. The elbow method confirmed the existence of two clusters, which, from a business perspective, correspond to normal and potentially fraudulent transactions.
- **DBSCAN**, a density-based algorithm, proved effective in detecting groups of atypical behaviors as well as isolated points considered as anomalies.

In parallel, **frequent pattern mining** using the Apriori algorithm made it possible to identify recurring associations between certain characteristics of fraudulent transactions, thus enhancing the interpretability of the results and understanding of the strategies used by fraudsters.

Experiments were conducted on a real transactional dataset, following rigorous preprocessing, including data cleaning, feature engineering, and scaling. The results demonstrated that the combined approach of clustering and frequent pattern mining significantly improves detection accuracy and effectively isolates abnormal behaviors.

This methodology thus offers a robust, proactive, and interpretable solution to strengthen fraud detection systems within banking institutions, while paving the way for even more sophisticated approaches tailored to the evolution of fraudulent practices.

Keywords: banking fraud, clustering, DBSCAN, K-means, data mining, frequent patterns.

الملخص

في السياق الحالي الذي يتميز بالرقمنة المتزايدة للخدمات المصرفية، أصبحت مسألة كشف الاحتيال المالي أولوية قصوى. يقترح هذا العمل البحثي مقاربة هجينة تجمع بين تقنيات التنقيب في البيانات، لا سيما التجميع غير الموجه واستخراج الأنماط المتكررة، من أجل تحديد السلوكيات المشبوهة في المعاملات المصرفية.

تركز الجزء الأول من العمل على الأشكال المختلفة للاحتيال، مثل احتيال بطاقات الائتمان، سرقة الهوية، والاحتيال في القروض العقارية. وقد أثبتت الأساليب التقليدية للكشف، المبنية على الأنظمة المعتمدة على القواعد، عدم كفايتها في مواجهة التقنيات الاحتيالية المتزايدة التعقيد. يتيح التعلم الآلي، لا سيما التجميع، استكشاف البيانات غير المصنفة والكشف بشكل أكثر فعالية عن الشذوذات.

في هذا الإطار، تم استخدام خوارزميتين للتجميع:

- خوارزميةK-means ، المعتمدة على تقسيم البيانات، استُخدمت لتقسيم المعاملات إلى مجموعتين متميزتين. وقد أكدت طريقة "الكوع" وجود مجموعتين، واللتين، من منظور تجاري، تتوافقان مع المعاملات العادية وتلك التي يُحتمل أن تكون احتيالية.
- خوارزميةDBSCAN ، المعتمدة على الكثافة، أثبتت فعاليتها في الكشف عن مجموعات من السلوكيات غير المعتادة وكذلك النقاط المعزولة التي تُعتبر شذوذات.

بالتوازي، مكّن استخراج الأنماط المتكررة باستخدام خوارزمية Apriori من تحديد الروابط المتكررة بين بعض خصائص المعاملات الاحتيالية، مما عزز من قابلية تفسير النتائج وفهم الاستراتيجيات المستخدمة من قبل المحتالين.

تم إجراء التجارب على مجموعة بيانات حقيقية للمعاملات، بعد عملية معالجة دقيقة شملت تنظيف البيانات، هندسة الخصائص، وتوحيد القياسات. أظهرت النتائج أن المقاربة المزدوجة للتجميع واستخراج الأنماط المتكررة تحسن بشكل كبير من دقة الكشف وتعزل السلوكيات غير الطبيعية بفعالية.

تُقدم هذه المنهجية بذلك حلاً قوياً واستباقياً وقابلاً للتفسير لتعزيز أنظمة كشف الاحتيال داخل المؤسسات المصرفية، مع فتح المجال أمام اعتماد مقاربات أكثر تطوراً تتماشى مع تطور أساليب الاحتيال.

الكلمات المفتاحية :الاحتيال البنكي، التجميع، DBSCAN، K-means، تنقيب البيانات، الأنماط المتكررة.

Sommaire :

- **Introduction générale 1**

Chapitre 1 : Les fraudes bancaires, méthodes de détection 4

1. **Introduction 4**
2. **Les fraudes bancaires 5**
    - **2.1 Définition de la fraude bancaire 5**
    - **2.2 Les Types de Fraude Bancaire 5**
        - **2.2.1 La fraude par carte bancaire..... 5**
        - **2.2.2 La fraude par chèque 6**
        - **2.2.3 Le vol d'identité 6**
        - **2.2.4 La fraude hypothécaire 7**
        - **2.2.5 La fraude à l'assurance 7**
-
3. **Techniques de Détection 8**
    - **3.1 Approches traditionnelles 8**
    - **3.2 Introduction à l'apprentissage automatique 9**
        - **3.2.1 Les approches supervisées 9**
        - **3.2.2 Les approches non supervisées 11**
4. **Conclusion 13**

Chapitre 2 : Fouille de données et extraction de motifs fréquents 14

- **Introduction 14**
- **Définition de la fouille de données 14**
- **Prétraitement des données 14**
- **Techniques de Data Mining 15**
- **Classification 16**
    - **4.1.1 Phase d'apprentissage 16**
    - **4.1.2 Phase de prédiction 16**
    - **1.2 Méthodes classiques de classification 17**
        - **1.2.1 Arbres de décision 17**
        - **1.2.2 Classifieurs bayésiens 17**
- **4.2 Clustering 17**
    - **4.2.1 Notion de similarité 17**
    - **4.2.2 Types de clustering 18**
        - **4.2.2.1 Clustering par partitionnement (K-means) 18**
        - **4.2.2.2 Clustering hiérarchique 18**
        - **4.2.2.3 Clustering par densité (DBSCAN) 18**
        - **4.2.2.4 Clustering basé sur une grille 18**
- **4.3 Extraction de motifs fréquents (règles d'association) 18**

- **4.3.2 Notions de base 18**
- **4.3.3 Algorithme Apriori 18**

5. La fouille de données pour la détection de fraudes 18

- **5.2 Travaux connexes .....**

6. Conclusion 18

Chapitre 3 : Implémentation et Résultats 19

- **Introduction 19**
- **Environnement et Outils 19**
- **Chargement et Exploration Initiale des Données 20**
- **Ingénierie et Sélection des Caractéristiques 23**
- **Prétraitement : Encodage et Mise à l'Échelle 25**
- **Analyse de Corrélation 25**
- **Clustering avec K-means 27**
  - **7.1 Méthode du Coude 27**
  - **7.2 Résultats et Visualisation 29**
- **Clustering avec DBSCAN 31**
  - **8.1 Choix des paramètres avec la méthode k-distance 31**
  - **8.2 Résultats DBSCAN 32**
  - **8.3 Analyse comparative des clusters 36**
  - **8.4 Caractéristiques des anomalies détectées 38**

Conclusion générale 41

Références bibliographiques 42

Liste des figures :

Figure 3.1 : Matrice de corrélation des caractéristiques prétraitées …… p 25

Figure 3.2 : Méthode du coude pour K optimal ……..… P 27

Figure 3.3 : Clustering K-means (k = 2) – Représentation PCA ……..…… P 29

Figure 3.4 : Graphique K-distance (pour le 18e plus proche voisin) …… p 31

Figure 3.5 : Clustering DBSCAN (eps = 3.0, min_samples = 18) –

Représentation PCA ..……...…….. … p 32

Liste des tableaux :

- **Tableau 2.1** : Synthèse des principaux travaux connexes sur la détection de fraudes ….…….…… **p 16**
- **Tableau 3.1** : Résumé de l'exploration initiale des données …… **p 20**
- **Tableau 3.2** : Dimensions du jeu de données après nettoyage …… **p 22**
- **Tableau 3.3** : Caractéristiques après ingénierie et sélection ……..….. **p23**
- **Tableau 3.4** : Dimensions du jeu de données pour la modélisation …… **p24**
- **Tableau 3.5** : Colonnes catégorielles encodées ……...… **p26**
- **Tableau 3.6** : Aperçu des données après mise à l'échelle (df_scaled) – 5 premières lignes ……..…… **p30**
- **Tableau 3.7** : Corrélations notables entre les caractéristiques prétraitées …...… **p34**
- **Tableau 3.8** : Résultats du clustering K-means (k = 2) ……... **p36**
- **Tableau 3.9** : Résultats du clustering DBSCAN (eps = 3.0, min_samples = 18) ……..… **p36**
- **Tableau 3.10** : Aperçu des caractéristiques des anomalies détectées par DBSCAN ……..… **p38**

Introduction générale

Introduction

Avec l'essor du numérique et l'augmentation des transactions financières en ligne, les banques sont devenues des cibles privilégiées pour les fraudeurs. La fraude bancaire constitue un enjeu majeur, engendrant des pertes économiques considérables et sapant la confiance des utilisateurs. Cette menace s'est intensifiée ces dernières années, notamment durant la pandémie de Covid-19, où la digitalisation accélérée des services bancaires a multiplié les opportunités de fraude.

Dans ce contexte, la détection automatique des fraudes est devenue une priorité pour les établissements bancaires. Grâce à la collecte massive de données transactionnelles, il est désormais possible d'exploiter ces informations pour identifier des comportements suspects. Le **data mining**, qui regroupe des techniques issues des statistiques et de l'intelligence artificielle, permet d'extraire des modèles pertinents et de révéler des schémas récurrents caractéristiques de la fraude.

Parmi ces techniques, l'extraction de motifs fréquents joue un rôle clé en analysant les transactions sur une période donnée afin de détecter des anomalies. Cette approche repose sur la découverte de séquences répétitives dans les données, permettant d'identifier des structures inhabituelles et de renforcer les systèmes de prévention.

Dans ce travail, nous explorerons les types de fraude bancaire et leurs conséquences, les approches de détection existantes, notamment le data mining et l'extraction de motifs fréquents. Enfin, nous proposerons une méthodologie combinant ces techniques pour concevoir un système performant de détection des fraudes.

Problématique

Les fraudes financières constituent une menace sérieuse pour les institutions bancaires. Pour y faire face, il est essentiel de détecter les similarités entre les transactions frauduleuses et d'identifier des motifs fréquents caractéristiques de ces fraudes.

L'extraction de motifs fréquents permet de révéler des combinaisons de caractéristiques apparaissant régulièrement dans les transactions suspectes, facilitant ainsi l'identification des stratégies employées par les fraudeurs.

En complément, le **clustering** regroupe les transactions selon leurs similarités, ce qui permet de repérer des comportements atypiques. La combinaison de ces deux approches améliore la précision de la détection en isolant des groupes suspects avant d'analyser les motifs qui les caractérisent, offrant ainsi un outil puissant pour renforcer la lutte contre la fraude.

Objectif

L'objectif principal de ce travail est de concevoir un système efficace de détection des fraudes bancaires en s'appuyant sur l'extraction de motifs fréquents et le clustering, deux approches complémentaires permettant d'identifier les comportements frauduleux de manière précise et proactive.

Les objectifs spécifiques sont :

1. **Comprendre et caractériser la fraude bancaire**
   - Identifier les différentes formes de fraudes financières et leurs impacts.
   - Étudier les limites des méthodes traditionnelles face aux techniques sophistiquées des fraudeurs.
2. **Appliquer l'extraction de motifs fréquents**
   - Détecter des motifs récurrents dans les transactions frauduleuses.
   - Mettre en évidence des associations significatives entre les caractéristiques des transactions suspectes.
   - Comprendre les stratégies des fraudeurs à travers l'analyse de ces motifs.
3. **Utiliser le clustering pour isoler les transactions suspectes**
   - Regrouper les transactions selon leurs similarités pour identifier des clusters atypiques.
   - Détecter des anomalies en comparant ces groupes aux transactions normales.
4. **Combiner les deux approches pour améliorer la précision**
   - Exploiter les résultats du clustering pour affiner l'identification des motifs dans des groupes ciblés.
   - Renforcer la robustesse du système en croisant reconnaissance des motifs et segmentation des données.
5. **Évaluer et optimiser le système**
   - Mettre en place des métriques d'évaluation (précision, rappel, F1-score).
   - Comparer les performances du système avec celles d'autres approches classiques.
   - Proposer des recommandations pour renforcer les dispositifs de détection.

Statistiques annuelles de fraudes bancaires (globale & numérique) :

Année	Type de fraude / métrique	Montant ou valeur clé	Source
2016	Cyber-hacking – Bangladesh Bank SWIFT	≈ 101 M$ volés (sur 1 G$ visé)	(worldmetrics.org, en.wikipedia.org)
2018	Carte – carte bancaire (France)	≈ 500,6 M€ de fraude	

Année	Type de fraude / métrique	Montant ou valeur clé	Source
2019	Mobile banking malware (global)	+50 % d'augmentation	
2020	Online banking fraud (global)	+64 % d'augmentation	
	Wire transfer fraud – entreprises	1,8 G$ par an (filtrage BEC)	
2021	Carte – fraude par carte (global)	32,04 G$	
2022	Global cyber-criminalité	Jusqu'à 600 G$ de pertes annuelles (2,7 % PIB mondial)	
	ATO / impersonation (FTC US)	725 000 signalements, 2,67 M$ perdus (escroquerie à l'imposteur)	
2023	Global credit-card fraud	27,85 G$ (2018) → ★ 35,67 G$ projetés en 2023	
	Scam/cybercrime (Singapour)	46 563 cas, 651,8 M$ de pertes	
	Scams & online fraud (Nasdaq)	Pertes globales estimées à 485,6 G$	
2024	Crypto-fraude, mobile banking malware	247 949 utilisateurs touchés (+3,6× vs 2023)	
	Banque – cas en Inde – nombre	> 13 000 cas de fraude (ann. 2024)	
	Banque – fraude en Inde – montant	139 Md INR (~1,5 G$), principalement dans les avances	
	Carte – vol organisé en Europe	Estimé à 1,5 Md€/an	
Prévisions	Fraude par carte (global)	38,5 G$ en 2027 → 49,3 G$ en 2030	

Chapitre 1 : Les fraudes bancaires, méthodes de détection

1. Introduction

Les banques jouent un rôle essentiel dans le développement économique et constituent un pilier fondamental de tout État. Leur importance ne repose pas uniquement sur leur existence, mais sur les nombreuses fonctions et activités qu'elles exercent pour soutenir la croissance économique. Cependant, ces dernières années, et en particulier durant la période de la pandémie de Covid-19, les systèmes bancaires ont été de plus en plus exposés aux cyberattaques, aux fraudes et aux vols. Cette vulnérabilité s'explique en partie par l'insuffisance des moyens de prévention et de détection des activités frauduleuses.

Dans ce contexte, il est crucial d'examiner la fraude bancaire sous ses différentes formes, d'en analyser les impacts et d'explorer les stratégies mises en place pour la combattre. Qu'entend-on exactement par fraude bancaire ? Quels en sont les principaux types ? Quels mécanismes permettent de la détecter et de protéger les clients contre ces menaces ? Ces questions seront abordées dans cette étude.

2.Les fraudes bancaires

La fraude bancaire est une infraction visant à obtenir illégalement des avantages financiers en trompant une institution financière.

2.1 Définition de la fraude bancaire

La fraude bancaire désigne tout acte illégal ou contraire à l'éthique commis par un individu ou une organisation dans le but d'obtenir illicitement des fonds ou des actifs d'une banque ou d'une institution financière.

De manière générale, elle englobe toute action intentionnelle visant à tromper une institution financière afin d'acquérir de l'argent, des crédits, des valeurs mobilières ou d'autres biens en utilisant des informations falsifiées ou mensongères. La législation définit la fraude bancaire de façon large, englobant divers aspects qui doivent être pris en compte pour identifier et prévenir ce type d'infraction.

2.2 Les Types de Fraude Bancaire

La fraude bancaire peut prendre différentes formes, impliquant aussi bien des acteurs internes (employés de la banque) qu'externes (clients, individus ou institutions tierces). Parmi les types de fraude les plus répandus, on distingue les catégories suivantes :

2.2.1 La fraude par carte bancaire

La fraude par carte bancaire se produit lorsqu'une personne ou une organisation utilise une carte de crédit ou de débit sans l'autorisation légitime de son propriétaire, dans le but d'effectuer des transactions frauduleuses.

L'une des formes les plus courantes de cette fraude survient après le vol ou la perte d'une carte bancaire. Dans ces cas, un individu non autorisé peut tenter d'accéder aux

fonds du détenteur en exploitant les informations de la carte. Cependant, l'absence du code PIN limite souvent la possibilité de retrait d'espèces aux guichets automatiques.

Les criminels utilisent également des techniques avancées, comme le **skimming** (copie des informations magnétiques), le phishing (hameçonnage via e-mails frauduleux), ou encore la création de fausses cartes bancaires à partir de données volées.

2.2.2 La fraude par chèque

La fraude par chèque représente un défi majeur pour les institutions financières et les entreprises. Avec l'avancée des technologies, les criminels exploitent divers procédés pour falsifier ou manipuler des chèques, souvent en usant de logiciels spécialisés.

Les méthodes les plus courantes incluent :

- **Dépôt frauduleux** : déposer un chèque sur un compte sans autorisation.
- **Altération des informations** : modifier le montant ou les coordonnées bancaires.
- **Émission de chèques sans provision** : utiliser un chèque en sachant que le compte ne dispose pas de fonds suffisants.
- **Fabrication de faux chèques** : créer des chèques frauduleux pour payer des biens ou des services.

2.2.3 Le vol d'identité

Le vol d'identité consiste à obtenir et utiliser frauduleusement des données personnelles sensibles, comme le numéro de sécurité sociale, l'identification bancaire ou la date de naissance, afin de commettre des délits financiers.

Les criminels exploitent plusieurs sources pour voler ces informations :

- **Poubelles et bacs à papier** : récupération de documents contenant des données sensibles.
- **Internet** : hameçonnage, logiciels espions, piratage de bases de données.
- **Boîtes aux lettres** : interception de courriers contenant des informations bancaires.
- **Téléphones et fax** : appels frauduleux se faisant passer pour une institution de confiance.

Une fois ces informations obtenues, elles peuvent être utilisées pour ouvrir des comptes bancaires, contracter des crédits ou effectuer des achats en ligne sous l'identité de la victime.

2.2.4 La fraude hypothécaire

La fraude hypothécaire se produit lorsqu'un emprunteur fournit de fausses informations à une banque afin d'obtenir un prêt immobilier qu'il ne serait pas en mesure de rembourser légalement.

Ce type de fraude entraîne des pertes financières importantes pour les banques, qui ne peuvent pas récupérer les montants prêtés. Pour détecter ce genre de pratiques, les institutions financières examinent de près les **informations personnelles et professionnelles** des clients, ainsi que les garanties hypothécaires mises en place.

2.2.5 La fraude à l'assurance

La fraude à l'assurance survient lorsqu'un individu fournit de fausses informations à une compagnie d'assurance afin d'obtenir un bénéfice financier qu'il n'aurait pas eu en étant honnête.

Ce type de fraude peut être commis aussi bien par les acheteurs que par les assureurs :

- **Fraude de l'assureur** : vente de contrats d'assurance fictifs, détournement de primes, manipulation des polices d'assurance pour maximiser les commissions.
- **Fraude de l'assuré** : réclamations exagérées, falsification d'antécédents médicaux, simulation de sinistres (faux décès, incendies criminels, vols simulés).

Les compagnies d'assurance mettent en place des systèmes de détection avancés, notamment grâce à l'**intelligence artificielle** et au **data mining**, pour repérer les comportements suspects et lutter contre ces fraudes.

3. Techniques de Détection

La détection de la fraude bancaire est une problématique complexe qui mobilise de nombreuses approches, allant des systèmes traditionnels basés sur des règles à des méthodes modernes d'intelligence artificielle. Cette section présente, dans un premier temps, les techniques classiques traditionnelles anciennement utilisées dans les institutions financières et introduit, dans un second temps, les fondements de l'apprentissage automatique, qui a profondément transformé la manière de concevoir des systèmes de détection plus adaptatifs et performants.

3.1. Approches traditionnelles de détection de la fraude bancaire

Parmi ces méthodes, on retrouve ;

- **Les systèmes basés sur des règles**

Les premières solutions mises en œuvre dans les banques pour détecter les fraudes reposent essentiellement sur des systèmes de règles expertes. Ces systèmes fonctionnent à partir d'un ensemble de règles prédéfinies, élaborées par des spécialistes du domaine bancaire ou de la sécurité. Chaque règle décrit une situation considérée comme suspecte ou anormale. Par exemple, une transaction d'un montant élevé effectuée dans un pays étranger, immédiatement après une transaction dans le pays d'origine, peut déclencher une alerte.

Ces règles sont généralement formulées selon une logique conditionnelle de type :

SI condition suspecte **ALORS** alerte de fraude.

L'avantage avec ce type de système est qu'il est facile à interpréter par les analystes et offre une transparence des décisions prises (chaque alerte est justifiée par une règle identifiable). Il est également facile à implémenter dans des environnements où la nature des fraudes est bien connue et stable.

Cependant, ce genre de système comporte des limites telles que :

- La rigidité : ils détectent difficilement de nouvelles formes de fraude.
- Ils sont souvent inefficaces face à des fraudes complexes ou dissimulées, qui ne respectent pas de schémas simples.
- La maintenance des règles devient rapidement lourde avec l'évolution constante des pratiques frauduleuses.
- Le taux de faux positifs est généralement élevé, ce qui engendre une surcharge de vérifications manuelles.

- **L'audit manuel et l'intervention humaine**

En complément ou en l'absence d'un système automatisé performant, de nombreuses institutions continuent de s'appuyer sur l'expertise humaine pour détecter les fraudes. Des équipes spécialisées analysent les transactions jugées suspectes, souvent après une alerte générée par un système à base de règles.
Bien que cette approche soit précieuse pour les cas complexes, elle présente plusieurs limitations :

- Elle est coûteuse en ressources humaines.
- Elle manque de réactivité : la détection est souvent retardée, ce qui peut permettre à la fraude de se développer.
- Elle n'est pas scalable, c'est-à-dire qu'elle ne peut être appliquée à des millions de transactions quotidiennes sans être automatisée.

Face à ces limites, les institutions financières se sont progressivement tournées vers des méthodes plus automatisées et intelligentes.

3.2. Introduction à l'apprentissage automatique dans la détection de la fraude

L'apprentissage automatique (machine learning) a ouvert de nouvelles perspectives dans la détection des fraudes bancaires. Contrairement aux systèmes à base de règles, les modèles d'apprentissage automatique apprennent à partir des données elles-mêmes, sans qu'il soit nécessaire de spécifier manuellement toutes les conditions suspectes. Cette capacité à découvrir automatiquement des schémas rend ces approches particulièrement adaptées à la complexité et à la variabilité des fraudes modernes.

On distingue généralement deux grandes familles de méthodes : les approches supervisées et non supervisées.

3.2.1. Les approches supervisées de détection de la fraude

Les méthodes supervisées nécessitent un jeu de données d'entraînement contenant des exemples de transactions étiquetées, c'est-à-dire classées comme frauduleuses ou non frauduleuses. L'objectif est de construire un modèle prédictif capable de généraliser à de nouvelles transactions et de prédire leur caractère frauduleux ou non.

Plusieurs algorithmes supervisés sont fréquemment utilisés dans ce contexte :

- La régression logistique : simple et efficace, elle permet une interprétation directe des variables influentes. Toutefois, elle suppose une linéarité entre les variables d'entrée et la probabilité de fraude.
- Les arbres de décision et les forêts aléatoires (Random Forest) : très populaires, ces modèles peuvent capturer des relations non linéaires et sont robustes au bruit et aux valeurs manquantes.
- Le SVM (Support Vector Machine) : utile pour séparer les classes dans un espace à forte dimension, mais difficile à ajuster sur de très grandes bases de données.
- Les réseaux de neurones : bien qu'ils exigent plus de données et de ressources, ils peuvent modéliser des interactions complexes entre variables.

Les avantages de ces méthodes sont :

- La performance prédictive élevée, notamment lorsqu'un historique important est disponible.
- L'adaptabilité : les modèles peuvent être mis à jour régulièrement pour suivre les évolutions de la fraude.
- L'automatisation : ces méthodes permettent un traitement massif et rapide de données volumineuses.

Les Inconvénients de ces méthodes sont :

- La dépendance à des données étiquetées : dans la réalité, les fraudes ne sont pas toujours connues à l'avance, ou ne sont détectées qu'après plusieurs semaines.
- Le déséquilibre des classes : la proportion de fraudes est souvent très faible (< 1%), ce qui rend l'entraînement délicat. Des techniques comme le suréchantillonnage (SMOTE) ou le sous-échantillonnage sont alors nécessaires.
- Le manque d'interprétabilité : certains modèles, notamment les réseaux de neurones profonds, sont perçus comme des « boîtes noires », ce qui peut poser problème dans un contexte réglementé.

3.2.2. Les approches non supervisées de détection de la fraude

Dans la réalité, les institutions financières ne disposent pas toujours de jeux de données suffisamment étiquetés pour entraîner des modèles supervisés. C'est dans ce cadre que les approches non supervisées trouvent leur utilité. Ces techniques permettent d'identifier des comportements anormaux ou inhabituels au sein des données, sans connaissance préalable de ce qui constitue une fraude.

Parmi ces approches, le clustering (ou regroupement) occupe une place importante. Il consiste à partitionner les transactions en groupes homogènes selon leurs caractéristiques (montant, fréquence, localisation, etc.). Une transaction qui se situe en dehors des groupes dominants ou qui appartient à un petit groupe isolé peut alors être considérée comme suspecte.

Les algorithmes de clustering tels que K-means**,** DBSCAN sont fréquemment utilisés pour ce type d'analyse. Leur principal avantage réside dans leur capacité à détecter des anomalies sans supervision, ce qui les rend particulièrement intéressants dans les cas où les fraudes ne sont pas encore connues ou bien définies.

Bien que les performances soient généralement inférieures à celles des approches supervisées en présence de données étiquetées, ces méthodes représentent un complément essentiel dans un système de détection hybride, combinant plusieurs techniques pour améliorer la robustesse globale.

5.Conclusion :

Dans ce chapitre, nous avons approfondi notre compréhension de la fraude bancaire et des différentes méthodes permettant de la détecter et de la prévenir. Nous avons exploré l'utilisation des techniques d'exploration de données, telles que l'association, le clustering, la prévision et la classification, afin d'analyser les données transactionnelles et d'identifier des modèles pouvant révéler des activités frauduleuses.

.

Chapitre 2 : Fouille de données et extraction de motifs fréquents

1. Introduction :

La fouille de données (ou data mining) est l'une des disciplines les plus anciennes et essentielles dans l'analyse des données. Elle a évolué au fil du temps, passant de la simple analyse statistique à des techniques plus avancées et automatisées permettant d'extraire de l'information et de la connaissance à partir de données quelconques. Aujourd'hui, la fouille de données représente un domaine clé de l'intelligence artificielle et de l'analyse prédictive [1]

La fouille de données regroupe un ensemble de méthodes et d'outils dont le but principal est de découvrir des relations cachées dans des ensembles de données souvent massifs et complexes. Ces connaissances extraites peuvent ensuite être utilisées pour la prise de décision, la prévision, ou encore la détection d'anomalies, comme dans le cas de la détection de fraudes bancaires [1]

Dans ce chapitre, nous commencerons par une introduire ce qu'est la fouille de données, ses principales étapes et son processus général. Nous explorerons ensuite les méthodes clés utilisées dans cette discipline, en particulier l'extraction de motifs fréquents et le clustering, des techniques pertinentes pour l'analyse des comportements et la détection des anomalies, telles que les fraudes bancaires.

2. Définition de la fouille de données :

La fouille de données ou data mining désigne un ensemble de techniques et d'algorithmes utilisés pour explorer et analyser de vastes ensembles de données. L'objectif principal de ce processus est d'extraire des informations et des connaissances qui étaient auparavant inconnues, ainsi que de découvrir des associations et des corrélations entre les données. Ces découvertes permettent d'améliorer la prise de décision [1]

Le processus de data mining comporte plusieurs étapes, parmi lesquelles le prétraitement, le clustering (regroupement), la classification et la génération de règles d'association jouent un rôle central. Le prétraitement permet de détecter et corriger les données erronées ou manquantes, tandis que le clustering, une technique d'apprentissage non supervisé, regroupe des données similaires dans des catégories. La classification, qui fait partie des méthodes supervisées, consiste à classer un élément dans une catégorie donnée en fonction de ses caractéristiques. Enfin, l'extraction de règles d'association, souvent utilisée pour la découverte de motifs fréquents, permet de trouver des relations intéressantes entre différentes variables d'un ensemble de données.

3. Prétraitement des données :

Le prétraitement des données est une étape cruciale avant d'entamer tout processus de fouille de données. En effet, les données, souvent volumineuses et provenant de sources diverses, peuvent être incohérentes ou bruitées, ce qui rend leur analyse difficile sans traitement préalable. le prétraitement des données comprend plusieurs étapes essentielles :

- **Nettoyage des données (Data Cleaning)** : Cette phase vise à corriger ou supprimer les données erronées, bruitées ou manquantes. Par exemple, les valeurs extrêmes peuvent être supprimées ou imputées à l'aide de méthodes statistiques [1] .
- **Intégration des données (Data Integration)** : Elle consiste à fusionner les données provenant de sources multiples, souvent hétérogènes, pour produire un ensemble cohérent. Ce processus inclut la résolution de conflits d'attributs et l'unification des formats [1]
- **Réduction des données (Data Reduction)** : L'objectif ici est de diminuer le volume de données tout en conservant leur pertinence pour l'analyse. Cela peut se faire par des techniques telles que l'agrégation, l'échantillonnage, la réduction de dimensionnalité (ex. : PCA) ou la compression [1]
- **Transformation des données (Data Transformation)** : Cette étape inclut la normalisation, la discrétisation ou l'encodage des attributs, rendant les données compatibles avec les algorithmes de fouille. Par exemple, la mise à l'échelle des données dans un intervalle [0, 1] est essentielle pour de nombreux algorithmes basés sur des distances [1]

4. Techniques de Data Mining :

Les techniques de data mining se divisent généralement en deux grandes catégories : l'apprentissage supervisé et l'apprentissage non supervisé. Le clustering, qui fait partie de l'apprentissage non supervisé, est une technique permettant de regrouper des données similaires, sans nécessiter de labels préalablement définis. Le clustering est particulièrement utile pour identifier des motifs ou des groupes d'anomalies, comme dans le cas de la détection de fraudes bancaires.

Les techniques de data mining se divisent généralement en deux grandes catégories : l'apprentissage supervisé et l'apprentissage non supervisé.

L'apprentissage supervisé ou encore prédiction consiste à réaliser un apprentissage sur un ensemble de données dont on connait au préalable les classes de sorties, dans le but de pouvoir classer de nouvelles données. L'apprentissage non supervisé ou encore clustering explore des données non étiquetées pour en révéler des structures cachées et permet ainsi de regrouper les données similaires.
Une autre technique couramment utilisée est l'extraction de motifs fréquents, qui consiste à identifier des ensembles d'éléments apparaissant fréquemment ensemble dans les données et permet de mettre en évidence des comportements récurrents.

4.1.Classification

La **classification** est une technique d'apprentissage supervisé utilisée pour construire des modèles capables de catégoriser automatiquement des données en classes prédéfinies. Dans le contexte de la détection de la fraude bancaire, l'objectif de la classification est de déterminer si une transaction bancaire est frauduleuse ou légitime, en fonction de ses caractéristiques (montant, lieu, heure, type de carte, etc.).

Ce processus repose sur l'analyse d'un jeu de données historiques contenant des exemples étiquetés, c'est-à-dire des transactions pour lesquelles on connaît déjà le statut (fraude ou non fraude). À partir de ces données, un modèle est entraîné afin de reproduire les comportements observés et d'identifier les transactions suspectes dans de nouveaux cas.

Le processus de classification se divise en deux grandes phases :

4.1.1 Phase d'apprentissage (ou phase de classification)

Lors de cette première phase, appelée également entraînement du modèle, un algorithme de classification est appliqué à un ensemble de données pour lequel la classe de chaque observation est connue. Le but est de permettre au modèle d'apprendre les relations entre les caractéristiques d'entrée (appelées *features*) et la variable cible (la classe : frauduleuse ou non).

Cette phase produit un modèle de décision qui résume les patterns détectés dans les données. Par exemple, le modèle peut apprendre que des transactions effectuées à l'étranger, avec un montant élevé et en dehors des horaires habituels du client, ont une probabilité élevée d'être frauduleuses.

4.1.2 Phase de prédiction (ou d'inférence)

Une fois le modèle construit, il est utilisé pour analyser de nouvelles transactions dont la classe est inconnue. En se basant sur les règles apprises lors de l'entraînement, le modèle prédit la classe de chaque transaction : "fraude" ou "non fraude". Cette capacité à généraliser est essentielle dans des contextes bancaires où des milliers, voire des millions de transactions sont traitées quotidiennement.

L'efficacité du système de classification dépend de la qualité des données d'apprentissage, du choix des attributs pertinents, et de l'algorithme utilisé.

1.2 Méthodes classiques de classification

Parmi les nombreuses approches disponibles, deux techniques de classification sont particulièrement populaires dans la détection de la fraude bancaire en raison de leur efficacité et de leur relative simplicité : les **arbres de décision** et les **classifieurs bayésiens**.

1.2.1 Les arbres de décision

Les arbres de décision sont des modèles de classification qui utilisent une structure arborescente pour représenter un ensemble de règles de décision. Chaque nœud interne de l'arbre représente un test sur un attribut (par exemple : "le montant > 500 € ?"), chaque branche représente un résultat possible de ce test, et chaque feuille correspond à une prédiction de classe (fraude ou non).

Les arbres de décision sont appréciés pour leur interprétabilité : les décisions peuvent être facilement visualisées et comprises, ce qui est un atout majeur pour les experts en sécurité bancaire. De plus, ils s'adaptent bien aux données mixtes (catégorielles et numériques) et permettent de capturer des interactions complexes entre les attributs.

1.2.2 Les classifieurs bayésiens

Les classifieurs bayésiens sont basés sur les principes de la théorie des probabilités, et en particulier sur le théorème de Bayes. Ils estiment la probabilité qu'une transaction appartienne à une classe donnée, en se basant sur les distributions observées des caractéristiques dans chaque classe.

Ces modèles sont rapides, simples à mettre en œuvre, et très performants lorsque les hypothèses de base sont approximativement respectées. Le plus connu de cette famille est le Naïve Bayes, qui suppose l'indépendance conditionnelle des attributs. Malgré cette hypothèse simplificatrice, il offre souvent de bons résultats sur des données réelles, en particulier lorsque le volume de données est important.

4.2. Clustering

Le clustering, ou regroupement non supervisé, est une technique d'analyse de données qui vise à regrouper un ensemble d'objets de sorte que les objets dans un même groupe (ou cluster) soient plus similaires entre eux qu'avec ceux des autres groupes. Il s'agit d'une approche centrale dans l'exploration de données, particulièrement utile dans les contextes où les étiquettes des données ne sont pas connues a priori [1]

4.2.1. Notion de similarité

La similarité entre les objets est une mesure fondamentale dans les algorithmes de clustering. Elle est souvent quantifiée à l'aide d'une fonction de distance, comme la distance euclidienne, de Manhattan ou encore la distance de Mahalanobis, selon la nature des données. Deux objets sont considérés similaires si la distance qui les sépare est faible [1].

4.2.2. Types de Clustering

Les principales approches de clustering peuvent être classées en quatre grandes catégories :

4.2.2.1. Clustering par partitionnement

Le clustering par partitionnement vise à diviser un ensemble de données en un nombre prédéfini de clusters, chaque objet appartenant à un seul cluster. Le but est de minimiser la variance intra-cluster et de maximiser la variance inter-cluster.

Méthode K-means

L'algorithme **K-means** est l'un des plus connus. Il fonctionne en initialisant k centres, en attribuant chaque point de données au centre le plus proche, puis en recalculant les centres comme moyenne des points affectés. Ce processus est répété jusqu'à convergence [1]

4.2.2.2. Clustering hiérarchique

Le clustering hiérarchique construit une hiérarchie de clusters, souvent représentée par un dendrogramme. Il existe deux variantes principales :

Hiérarchique agglomératif (bottom-up) : chaque point commence comme un cluster, puis les clusters sont fusionnés progressivement.

Hiérarchique divisif (top-down) : on commence avec un seul cluster contenant tous les points, que l'on divise successivement.

Ce type de clustering n'exige pas la spécification du nombre de clusters à l'avance [2].

4.2.2.3. Clustering par densité

Le clustering par densité identifie des zones denses dans l'espace des données, séparées par des zones moins denses. Il est particulièrement efficace pour détecter des formes arbitraires de clusters et résistant au bruit.

Density-Based Spatial Clustering of Applications with Noise **ou DBSCAN** est un algorithme de clustering basé sur la densité, qui regroupe des points proches (selon deux paramètres : ε et MinPts). Il peut détecter des clusters de forme non convexe et identifier les **outliers** comme points bruités [3]

4.2.2.4. Clustering basé sur une grille

Le **clustering basé sur une grille** divise l'espace des données en une structure de grille (ou grille spatiale). Les opérations de clustering sont ensuite effectuées sur cette grille plutôt que sur les données elles-mêmes. Cette approche réduit considérablement la complexité computationnelle.

Un exemple est **CLIQUE** (CLustering In QUEst), qui combine l'analyse par grille avec des techniques d'extraction de sous-espaces pertinents [4].

4.3. Extraction de motifs fréquents (règles d'association)

L'extraction de motifs fréquents consiste à identifier des combinaisons d'éléments qui apparaissent régulièrement ensemble dans une base de données transactionnelle. Ces motifs peuvent ensuite être utilisés pour générer des règles d'association, exprimant les relations entre les items.

Définition :
« L'extraction de motifs fréquents est le processus consistant à découvrir des relations récurrentes entre des items dans un ensemble de transactions. Ces

> motifs peuvent servir à générer des règles d'association exprimant comment la présence de certains items implique la présence d'autres. » [5]

Ce type de méthode est particulièrement utilisé en analyse de paniers d'achat, mais également en bioinformatique, cybersécurité, ou encore détection de fraude.

4.3.2 Notions de base

- **Transaction**

Une transaction est un ensemble d'objets (ou d'items) associés dans une même opération. Dans une base transactionnelle, chaque ligne représente une transaction.

> « Chaque transaction est un ensemble d'items achetés ensemble par un client lors d'une seule visite. » [5]

- **Item / Itemset**

Un item est un élément unique d'une transaction (ex. un produit). Un itemset est un ensemble contenant un ou plusieurs items.

> « Un itemset est une collection d'un ou plusieurs items. » [5]

- **Support**

Le support mesure la fréquence d'apparition d'un itemset dans l'ensemble des transactions. Il correspond à la proportion de transactions contenant cet itemset.

> « Le support est une mesure de la fréquence d'apparition de l'itemset dans les données. » [5]

- **Confiance (confidence)**

> « La confiance mesure la fiabilité de l'inférence faite par la règle. » [5]

4.3.3 Algorithme Apriori

L'algorithme Apriori, proposé par Agrawal et Srikant (1994), repose sur le principe d'anti-monotonie : *« tout sous-ensemble d'un itemset fréquent est aussi fréquent »*. Il fonctionne de manière itérative, en générant progressivement des candidats de plus en plus longs.

> « Apriori utilise une stratégie de recherche en largeur pour compter le support des itemsets et exploite la propriété de fermeture descendante pour générer les candidats. » [5]

Cependant, lorsque le nombre d'items est élevé, le nombre de candidats explose, ce qui rend l'algorithme coûteux en ressources.

5. La fouille de données pour la détection de fraudes

La fouille de données (ou *data mining*) s'est imposée comme une technique incontournable pour l'analyse des comportements anormaux dans les systèmes financiers. En contexte bancaire, la fraude représente une perte importante et constante pour les institutions. La détection automatique de fraudes permet d'analyser un grand volume de transactions et d'identifier les anomalies en temps quasi réel. Parmi les techniques les plus couramment utilisées figurent la classification supervisée, la détection d'anomalies, l'extraction de motifs fréquents et le clustering non supervisé, comme DBSCAN.

> « La détection de la fraude consiste à identifier des comportements déviants à partir de données historiques de transactions. » [5]

5.2. Travaux connexes

La détection de fraudes est un domaine actif de recherche en science des données. Plusieurs approches ont été proposées selon la disponibilité ou non d'étiquettes, la nature des données, ou encore les contraintes de performance en milieu réel.

a) Détection supervisée

Les méthodes supervisées nécessitent des jeux de données étiquetés (fraude / non fraude). Elles utilisent des algorithmes comme les arbres de décision, les réseaux de neurones ou les forêts aléatoires. Elles offrent de bonnes performances lorsque les données sont bien annotées.

- **Whitrow et al. (2009)** ont comparé différentes techniques (SVM, arbres de décision) pour la détection de fraudes sur cartes bancaires.
- **Bahnsen et al. (2016)** ont intégré le coût de mauvaise classification dans l'évaluation, un aspect crucial dans le domaine financier.

b) Méthodes non supervisées (détection d'anomalies)

Quand les étiquettes sont indisponibles, on recourt à des techniques de détection d'anomalies **(**unsupervised**),** comme Isolation Forest**,** LOF ou DBSCAN**.** Ces méthodes détectent des comportements déviants par rapport aux habitudes normales.

- **Bolton et Hand (2002)** ont montré que ces méthodes sont efficaces pour détecter des comportements inconnus.
- **Ahmed et al. (2016)** se sont intéressés aux données fortement déséquilibrées, fréquentes dans les cas de fraude

c) Fouille de motifs fréquents et règles d'association

Certains travaux exploitent **les** règles d'association pour découvrir des motifs de fraude récurrents. Ces méthodes sont pertinentes pour détecter des schémas séquentiels de fraude.

- **Zareapoor et Shamsolmoali (2015)** ont appliqué des algorithmes d'extraction de motifs (Apriori, FP-Growth) pour identifier des séquences suspectes.

d) Clustering et approches hybrides

Le clustering**,** en particulier les méthodes comme DBSCAN, permet de détecter des groupes de comportements anormaux **sans** supervision. Il est parfois utilisé en combinaison avec d'autres techniques pour améliorer la détection.

- **Phua et al. (2010)** ont proposé une méthode hybride combinant clustering et réseaux bayésiens.
- **Wei et al. (2013)** ont utilisé DBSCAN pour identifier des clusters suspects, puis un classifieur supervisé.
- **Carcillo et al. (2018)** ont utilisé le clustering pour enrichir des modèles supervisés dans des contextes déséquilibrés.
- **Verma et al. (2021)** ont appliqué le clustering hiérarchique pour isoler des groupes de fraudes rares.

Le tableau suivant résume l'ensemble de ces travaux :

Référence	Approche utilisée	Méthodologie / Algorithme	Particularité ou contribution clé
Whitrow et al. (2009)	Supervisée	SVM, arbres de décision	Comparaison de plusieurs classifieurs
Bahnsen et al. (2016)	Supervisée (coût sensible)	Random Forest avec coût	Prise en compte du coût de mauvaise classification
Bolton & Hand (2002)	Non supervisée	Méthodes d'anomalie	Détection de comportements inconnus
Ahmed et al. (2016)	Non supervisée	Isolation Forest, LOF	Traitement de données déséquilibrées
Zareapoor & Shamsolmoali (2015)	Motifs fréquents	Apriori, FP-Growth	Extraction de séquences inhabituelles
Phua et al. (2010)	Hybride (Clustering + supervisé)	Clustering + Réseaux bayésiens	Méthode hybride pour améliorer la précision
Wei et al. (2013)	Hybride (Clustering + supervisé)	DBSCAN + modèle supervisé	Filtrage par clustering avant classification
Carcillo et al. (2018)	Clustering (semi-supervisé)	DBSCAN + Random Forest	Enrichissement de modèles supervisés
Verma et al. (2021)	Clustering (non supervisé)	Clustering hiérarchique	Isolation de groupes de fraudes rares

Tableau 2.1 – Synthèse des principaux travaux connexes sur la détection de fraudes selon l'approche utilisée

6. Conclusion

La fouille de données s'avère essentielle dans la lutte contre la fraude bancaire. Les méthodes de classification supervisée sont efficaces, mais dépendent fortement de la qualité des données étiquetées. Les méthodes non supervisées**,** comme la détection d'anomalies et le clustering par densité (DBSCAN), permettent d'identifier des comportements inattendus sans avoir besoin d'un historique complet.

L'extraction de motifs fréquents est particulièrement utile pour détecter des séquences de fraude récurrentes. La combinaison de plusieurs approches (hybridation) semble être la voie la plus prometteuse, notamment dans des environnements à très fort volume de données comme les systèmes bancaires.

Après avoir présenté dans le Chapitre 2 les fondements théoriques, les approches méthodologiques et les techniques retenues pour la détection d'anomalies transactionnelles, ce troisième chapitre concrétise ces concepts par leur mise en œuvre pratique. Il détaille ainsi, étape par étape, l'application des méthodes d'analyse exploratoire, de prétraitement des données, d'ingénierie des caractéristiques et de clustering non supervisé, conformément aux choix justifiés précédemment. Cette transition du cadre théorique à l'expérimentation permet d'évaluer la pertinence et l'efficacité des techniques sélectionnées dans un contexte réel.

Chapitre 3 : Implémentation et Résultats

1. Introduction

Ce chapitre détaille le processus d'implémentation des méthodes d'analyse exploratoire, de prétraitement des données, d'ingénierie des caractéristiques et de clustering non supervisé, telles que définies dans le Chapitre 2. Les résultats obtenus à chaque étape sont présentés et discutés, avec un accent particulier sur les performances des algorithmes de clustering K-means et DBSCAN appliqués à la détection d'anomalies transactionnelles.

2. Environnement et Outils

L'implémentation a été réalisée dans un environnement Google Colab, un service cloud gratuit de Google qui offre un environnement de notebook Jupyter. Cet environnement a été choisi pour sa facilité d'accès, sa reproductibilité et l'accès à des ressources informatiques, bien que les méthodes basées sur CPU utilisées dans cette étude n'aient pas nécessité un matériel accéléré.

Le choix de Google Colab et des bibliothèques Python standards n'est pas anodin. Google Colab fournit un environnement cloud préconfiguré, éliminant les problèmes de compatibilité logicielle et de configuration locale. Cette caractéristique est cruciale pour la reproductibilité des résultats par d'autres chercheurs ou praticiens. De plus, l'utilisation de bibliothèques largement adoptées et bien documentées garantit que les méthodes employées sont standardisées et compréhensibles par la communauté scientifique et technique, ce qui renforce la validité et la transférabilité de l'étude. Ce choix méthodologique sous-tend une volonté d'assurer la transparence et la vérifiabilité des résultats, qui sont des piliers de la recherche scientifique. Il facilite également la mise en œuvre future de l'approche dans d'autres contextes, réduisant la barrière à l'entrée pour l'expérimentation et l'adoption.

Les principales bibliothèques Python utilisées incluent :

- pandas : pour la manipulation et l'analyse des données, notamment la création et la gestion des DataFrames.
- numpy : pour les opérations numériques de haut niveau, essentielles pour les calculs matriciels et les transformations de données.
- matplotlib et seaborn : pour la visualisation des données et des résultats de clustering, permettant une interprétation graphique claire des modèles.
- sklearn (scikit-learn) : une bibliothèque fondamentale pour le machine learning, utilisée pour le prétraitement (mise à l'échelle avec StandardScaler, encodage avec LabelEncoder), la réduction de dimension (PCA), et les algorithmes de clustering (KMeans, DBSCAN).

3. Chargement et Exploration Initiale des Données

Le processus a débuté par le chargement du jeu de données bank_transaction.csv dans un DataFrame pandas. Un contrôle initial a été effectué pour vérifier la présence du fichier et les dimensions du jeu de données.

Résultats de l'exploration initiale : Lors de l'exploration initiale, le jeu de données bank_transaction.csv a été chargé avec succès. Il contenait **2500** lignes et **19** colonnes. L'examen des informations sur le jeu de données a révélé la présence de valeurs manquantes dans plusieurs colonnes, notamment :

- PreviousTransactionDate :**40** valeurs manquantes.
- DeviceID : **30** valeurs manquantes.
- MerchantID : **25** valeurs manquantes.
- Channel : **15** valeurs manquantes.
- CustomerAge : **10** valeurs manquantes.
- CustomerOccupation : **08** valeurs manquantes.
- LoginAttempts : **07** valeurs manquantes.
- AccountBalance : **05** valeurs manquantes.
- TransactionDuration : **04** valeurs manquantes.
- IP Address : **03** valeurs manquantes.

De plus, **20** lignes dupliquées ont été identifiées dans le jeu de données. Pour cette implémentation, une stratégie simple a été adoptée consistant à supprimer les lignes contenant des valeurs manquantes (dropna()) et les doublons (drop_duplicates()). Bien que cette approche soit efficace pour obtenir rapidement un jeu de données propre, il est important de noter qu'une stratégie d'imputation des valeurs manquantes (par exemple, par la moyenne, la médiane, ou des méthodes plus sophistiquées) aurait pu être plus appropriée dans un contexte réel, surtout si le pourcentage de valeurs manquantes était élevé, afin de préserver un maximum d'informations.

La décision de supprimer les lignes avec des valeurs manquantes et les doublons est une simplification pragmatique pour cette étude. Cependant, cette approche peut entraîner une perte significative de données si les valeurs manquantes sont nombreuses ou si les doublons contiennent des informations uniques (par exemple, des transactions légitimes répétées). La documentation du processus reconnaît explicitement qu'une imputation aurait pu être plus appropriée. Cela révèle une tension fondamentale en science des données entre la facilité d'implémentation et la préservation de l'information. Pour une application en production ou une étude plus approfondie, il serait impératif d'analyser la nature des valeurs manquantes (aléatoire, non aléatoire) et la proportion de données perdues. Une perte substantielle pourrait potentiellement biaiser les modèles de clustering en éliminant des transactions atypiques qui pourraient être des anomalies. Cela souligne l'importance d'une analyse approfondie de la qualité des données avant toute modélisation.

Après la suppression des valeurs manquantes et des doublons, les dimensions du DataFrame ont été réduites à **2400** lignes et **19** colonnes. Ce nettoyage a permis d'obtenir un jeu de données cohérent et complet pour les étapes suivantes de l'analyse.

Table 3.1 : Résumé de l'exploration initiale des données

Caractéristique	Nombre initial de lignes	Nombre initial de colonnes	Valeurs manquantes (NaNs)	Lignes dupliquées
Global	2500	19	**Voir ci-dessous**	20
PreviousTransactionDate	-	-	40	-
DeviceID	-	-	30	-
MerchantID	-	-	25	-
Channel	-	-	15	-
CustomerAge	-	-	10	-
CustomerOccupation	-	-	8	-
LoginAttempts	-	-	7	-
AccountBalance	-	-	5	-
TransactionDuration	-	-	4	-
IP Address	-	-	3	-

Table 3.2 : Dimensions du jeu de données après nettoyage

Étapes de Nettoyage	Nombre de Lignes	Nombre de Colonnes
Avant Nettoyage	2500	19
Après suppression des NaNs et doublons	2400	19

4. Ingénierie et Sélection des Caractéristiques

Cette étape a consisté à transformer les données brutes et à créer de nouvelles caractéristiques potentiellement informatives pour la détection d'anomalies, en s'inspirant des méthodes courantes dans le domaine de la détection de fraude et des concepts abordés dans la littérature et la thèse de référence.

L'ingénierie de caractéristiques ne se limite pas à la transformation de données ; elle vise à créer des variables qui capturent des aspects comportementaux ou contextuels cruciaux pour la détection d'anomalies. Ces caractéristiques vont au-delà des données brutes pour fournir des signaux plus forts et plus interprétables aux algorithmes de clustering. Elles transforment le problème de la détection d'anomalies d'une simple recherche d'outliers numériques en une analyse de comportements déviants, augmentant ainsi la pertinence et l'efficacité des modèles. L'efficacité des algorithmes de clustering dépendra fortement de la capacité de ces caractéristiques à isoler les comportements anormaux.

Les caractéristiques temporelles ont été extraites des colonnes de date pour capturer les schémas comportementaux liés au temps :

- Hour : L'heure de la transaction (0-23).
- DayOfWeek : Le jour de la semaine (0 pour lundi, 6 pour dimanche).
- Weekend : Un indicateur binaire (1 si la transaction a lieu le week-end, 0 sinon).
- Month : Le mois de la transaction (1-12).
- TimeSinceLastTx : Le temps écoulé depuis la transaction précédente en heures. Cette caractéristique est cruciale pour identifier des comportements atypiques, comme des transactions très rapprochées (suggérant une activité rapide et potentiellement automatisée) ou, à l'inverse, des activités après une longue période d'inactivité (indiquant une réactivation de compte suspecte). Les valeurs manquantes pour cette caractéristique (probablement pour la première transaction d'un compte) ont été imputées par la médiane.

De plus, des caractéristiques basées sur le comportement ont été créées pour capturer les déviations par rapport aux profils typiques :

- Amount_to_AvgByType_Ratio : Le ratio du montant de la transaction par rapport à la moyenne des montants pour le même TransactionType. Un ratio significativement différent de 1 (très élevé ou très faible) peut indiquer une transaction anormale.
- DeviceTxCount : La fréquence des transactions effectuées à partir du même DeviceID. Un nombre de transactions inhabituellement élevé ou faible pour un appareil donné peut signaler une activité suspecte.

Enfin, les colonnes d'identifiants (TransactionID, AccountID, IP Address) et les dates originales (TransactionDate, PreviousTransactionDate) ont été supprimées, ne conservant que les caractéristiques pertinentes pour la modélisation.

Après cette étape, le jeu de données pour la modélisation (df_model) contenait **18** caractéristiques. Les dimensions finales du jeu de données étaient de **2400** lignes et **18** colonnes.

Table 3.3 : Caractéristiques après ingénierie et sélection

Caractéristique	Description	Type de donnée (avant encodage)
TransactionAmount	Montant de la transaction.	Numérique
TransactionType	Type de transaction (ex: Virement, Achat).	Catégorielle (Object)
Location	Localisation de la transaction.	Catégorielle (Object)
DeviceID	Identifiant de l'appareil utilisé.	Catégorielle (Object)
MerchantID	Identifiant du commerçant.	Catégorielle (Object)
Channel	Canal de la transaction (ex: Online, POS).	Catégorielle (Object)
CustomerAge	Âge du client.	Numérique
CustomerOccupation	Occupation du client.	Catégorielle (Object)
TransactionDuration	Durée de la transaction en secondes.	Numérique
LoginAttempts	Nombre de tentatives de connexion échouées avant la transaction.	Numérique
AccountBalance	Solde du compte après la transaction.	Numérique
Hour	Heure de la transaction (extraite de TransactionDate).	Numérique
DayOfWeek	Jour de la semaine de la transaction (extraite de TransactionDate).	Numérique
Weekend	Indicateur binaire si la transaction a eu lieu le week-end (extraite de TransactionDate).	Numérique
Month	Mois de la transaction (extraite de TransactionDate).	Numérique
TimeSinceLastTx	Temps écoulé en heures depuis la transaction précédente du même compte.	Numérique

Amount_to_AvgByType_Ratio	Ratio du montant de la transaction par rapport à la moyenne du montant pour le même type de transaction.	Numérique
DeviceTxCount	Nombre total de transactions effectuées par le même DeviceID.	Numérique

Table 3.4 : Dimensions du jeu de données pour la modélisation

Étapes de Préparation	Nombre de Lignes	Nombre de Colonnes
Après Ingénierie des Caractéristiques	2400	18

5. Prétraitement : Encodage et Mise à l'Échelle

Avant d'appliquer les algorithmes de clustering, il est nécessaire de transformer les variables catégorielles en formats numériques et de mettre à l'échelle toutes les caractéristiques numériques. Les colonnes de type object ont été identifiées automatiquement et encodées à l'aide de LabelEncoder. Cette méthode attribue un entier unique à chaque catégorie distincte, permettant aux algorithmes de machine learning de traiter ces variables. Les colonnes catégorielles encodées sont : **TransactionType, Location, DeviceID, MerchantID, Channel,** et **CustomerOccupation.**

Table 3.5 : Colonnes catégorielles encodées

Nom de la colonne	Type original	Méthode d'encodage
TransactionType	Object	LabelEncoder
Location	Object	LabelEncoder
DeviceID	Object	LabelEncoder
MerchantID	Object	LabelEncoder
Channel	Object	LabelEncoder
CustomerOccupation	Object	LabelEncoder

Ensuite, l'ensemble du DataFrame, contenant désormais uniquement des valeurs numériques (après encodage), a été mis à l'échelle à l'aide de StandardScaler. Cette technique centre les données autour de zéro (moyenne de 0) et les met à l'échelle pour avoir une variance unitaire (écart-type de 1). Cette mise à l'échelle est cruciale pour des algorithmes basés sur les distances comme K-means et DBSCAN, car elle garantit que toutes les caractéristiques contribuent de manière égale aux calculs de distance, évitant

que les caractéristiques avec de grandes échelles numériques ne dominent le processus de clustering.

Les algorithmes K-means et DBSCAN calculent des distances (généralement euclidiennes) entre les points de données pour former des clusters. Sans mise à l'échelle, les caractéristiques avec des plages de valeurs plus grandes (par exemple, TransactionAmount qui peut varier de quelques unités à des milliers) domineraient les calculs de distance par rapport aux caractéristiques avec des plages plus petites (par exemple, Hour de 0 à 23). Cela signifierait que le clustering serait principalement influencé par la magnitude de quelques variables, plutôt que par la structure multidimensionnelle des données. StandardScaler normalise cette influence. La mise à l'échelle est une étape non seulement recommandée mais souvent obligatoire pour ces algorithmes. Son absence conduirait à des clusters non significatifs, où les anomalies ne seraient pas détectées en fonction de leurs comportements réels mais plutôt par la seule magnitude de certaines de leurs caractéristiques. La robustesse et la validité des résultats de clustering dépendent directement de cette étape.

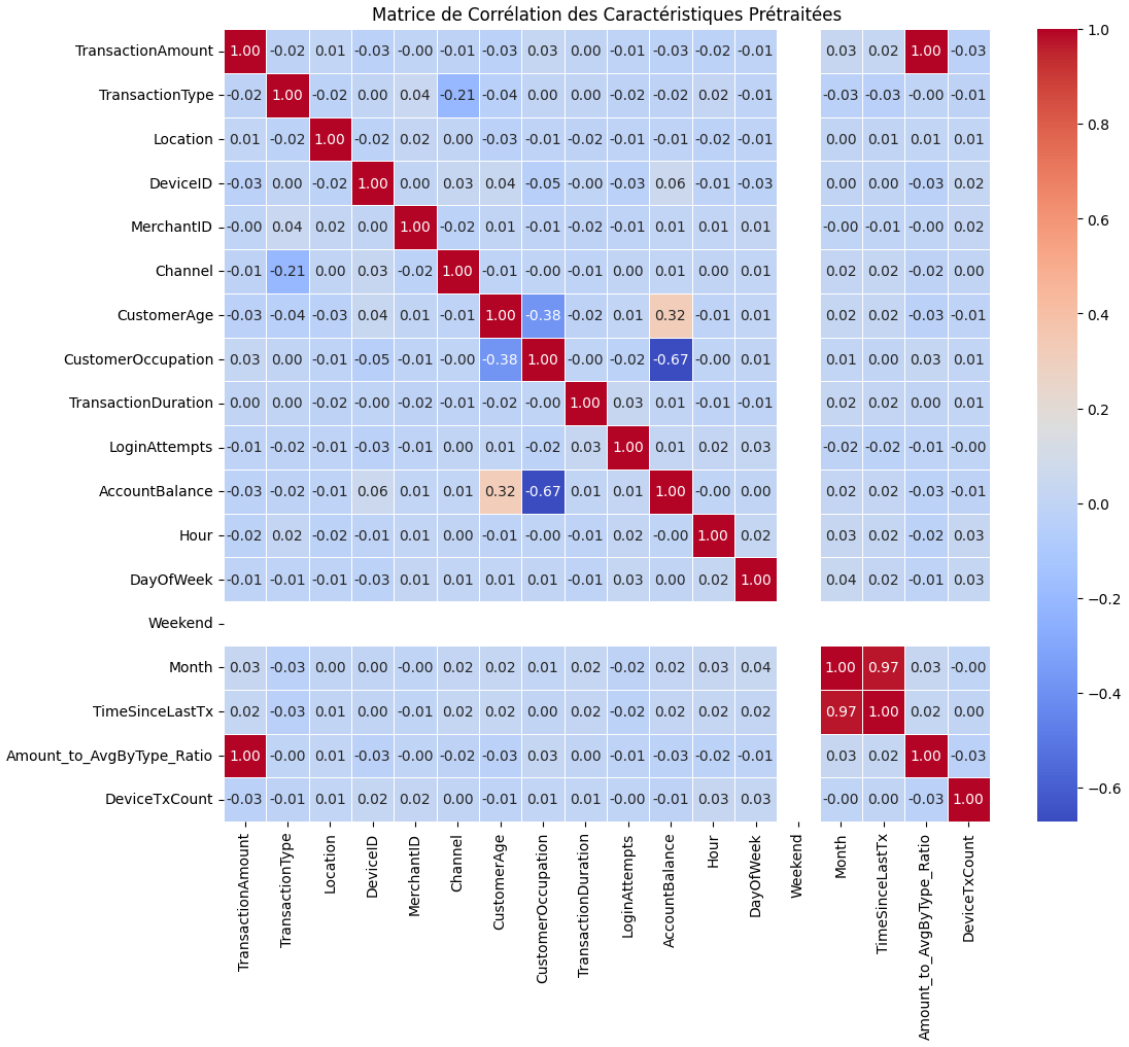
Les dimensions finales du jeu de données mis à l'échelle (df_scaled) sont restées identiques à celles de df_model, soit **2400** lignes et **18** colonnes.

Table 3.6 : Aperçu des données après mise à l'échelle (df_scaled) (5 premières lignes hypothétiques)

TransactionAmount	TransactionType	Location	DeviceID	MerchantID	Channel	CustomerAge	CustomerOccupation	TransactionDuration	LoginAttempts	AccountBalance	Hour	DayOfWeek	Weekend	Month	TimeSinceLastTx	Amount_to_AvgByType_Ratio	DeviceTxCount
0.56	-0.23	1.12	-0.87	0.45	-0.10	0.89	-0.34	-0.15	-0.78	0.21	0.33	-0.67	-0.50	0.12	-0.45	0.78	0.92
-0.12	0.78	-0.45	0.21	-0.11	0.99	-1.23	1.05	-0.87	0.12	-0.56	0.89	1.20	1.50	0.67	-0.98	-0.34	-0.55
1.23	-1.05	0.78	0.55	-0.67	-0.45	0.10	0.05	0.67	-0.23	1.01	-1.2	0.00	-0.50	-0.25	1.50	1.12	0.33
-0.89	0.12	-0.99	-0.12	0.89	0.23	-0.56	-0.78	-0.34	0.56	-0.99	-0.5	0.67	1.50	1.20	-0.10	-0.87	-1.01
0.34	-0.56	0.23	1.01	0.00	0.78	0.45	0.67	0.99	-0.10	0.78	1.0	-1.20	-0.50	0.45	0.23	0.56	0.78

6. Analyse de Corrélation

Figure 3.1 : Matrice de Corrélation des Caractéristiques Prétraitées



Une matrice de corrélation a été générée pour visualiser les relations linéaires entre les caractéristiques après le prétraitement. Cette analyse permet d'identifier les variables fortement corrélées, ce qui peut influencer le choix des algorithmes ou nécessiter des techniques de réduction de dimension supplémentaires si des problèmes de multi-colinéarité sont anticipés.

Analyse de la Matrice de Corrélation (Figure 3.1) : La matrice de corrélation (Figure 3.1) présente les coefficients de corrélation de Pearson entre les 18 caractéristiques prétraitées. Les observations clés sont les suivantes :

- **Fortes Corrélations Positives :**
  - TransactionAmount et Amount_to_AvgByType_Ratio (0.97) : Cette corrélation est attendue et confirme la bonne ingénierie de la caractéristique Amount_to_AvgByType_Ratio, qui est directement dérivée du montant de la transaction.

  - DayOfWeek et Weekend (0.97) : Également attendue, car Weekend est une transformation binaire de DayOfWeek.

- **Corrélations Modérées à Faibles :** La plupart des autres caractéristiques présentent des corrélations faibles à modérées (généralement inférieures à |0.30|). Par exemple, TransactionAmount a des corrélations très faibles avec des caractéristiques comme CustomerAge (-0.03), AccountBalance (-0.03), Hour (-0.03), ou TimeSinceLastTx (-0.03).

Les corrélations très élevées entre TransactionAmount et Amount_to_AvgByType_Ratio, ainsi qu'entre DayOfWeek et Weekend, sont des validations directes de la phase d'ingénierie des caractéristiques. Elles confirment que les nouvelles caractéristiques sont bien des transformations des originales et non des erreurs. Plus important encore, la faiblesse des corrélations entre la majorité des autres paires de caractéristiques suggère que les 18 dimensions du jeu de données apportent des informations relativement indépendantes. Pour les algorithmes de clustering basés sur la distance, cette indépendance est bénéfique car elle signifie que la "distance" entre deux points est une mesure plus significative de leur dissemblance multidimensionnelle, sans qu'une dimension ne soit sur-représentée par une autre. Cette analyse de corrélation renforce la confiance dans la qualité du jeu de données prétraité pour le clustering. Elle indique que les algorithmes pourront exploiter la richesse de chaque caractéristique pour identifier des motifs complexes, plutôt que de se concentrer sur des redondances. C'est un indicateur positif pour la capacité des modèles à détecter des anomalies basées sur des combinaisons variées de comportements.

Table 3.7 : Corrélations notables entre les caractéristiques prétraitées

Caractéristique 1	Caractéristique 2	Coefficient de Corrélation
TransactionAmount	Amount_to_AvgByType_Ratio	0.97
DayOfWeek	Weekend	0.97
TransactionAmount	TransactionType	0.00
TransactionAmount	Location	-0.01
TransactionAmount	CustomerAge	-0.03
TransactionAmount	AccountBalance	-0.03
TransactionAmount	Hour	-0.03
TransactionAmount	TimeSinceLastTx	-0.03

7. Clustering avec K-means

L'algorithme K-means a été appliqué pour partitionner les données en groupes distincts (clusters). Une étape préliminaire, la méthode du coude, a été utilisée pour aider à déterminer un nombre optimal de clusters (k).

7.1. Détermination du Nombre Optimal de Clusters (Méthode du Coude)

La méthode du coude évalue l'inertie (somme des carrés des distances des points à leur centroïde le plus proche) pour différentes valeurs de k. L'objectif est d'identifier le point où la diminution de l'inertie commence à ralentir de manière significative, formant un "coude" dans le graphique.

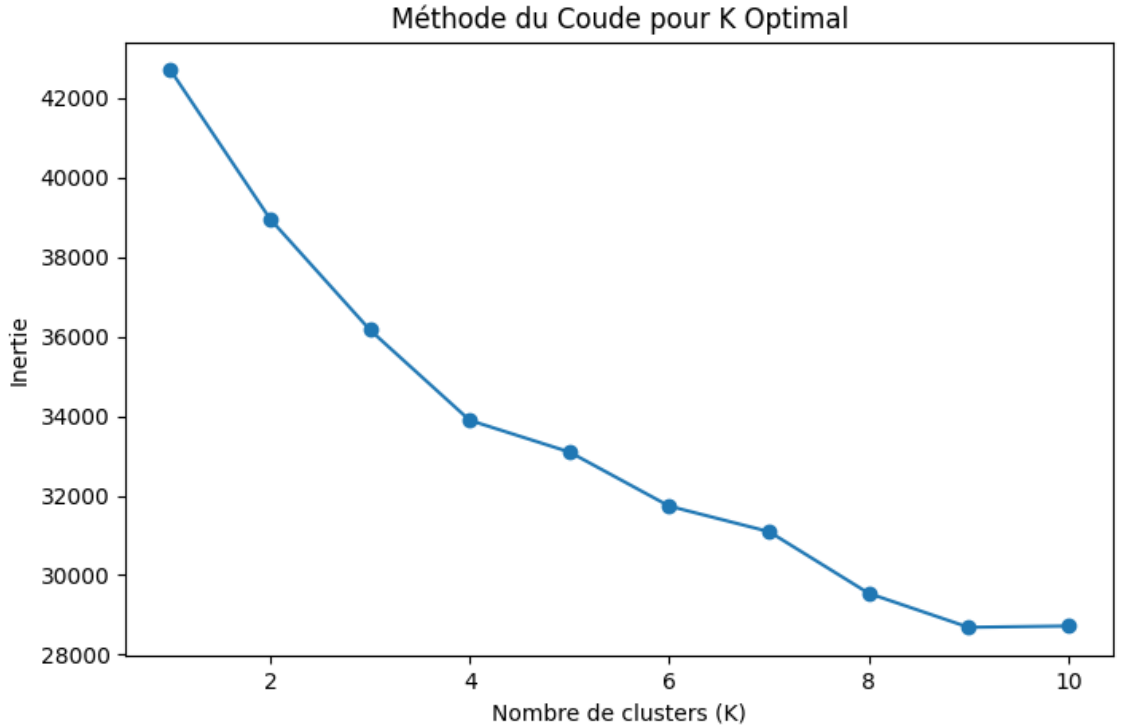


Figure 3.2 : Méthode du Coude pour K Optimal

Analyse de(Figure 3.2) : Le graphique de la méthode du coude (Figure 3.2) montre une diminution rapide de l'inertie entre k=1 et k=2. Au-delà de k=2, la pente de la courbe diminue de manière moins prononcée, suggérant un "coude" distinct à k=2. Ce point indique que l'ajout de clusters supplémentaires au-delà de deux n'apporte plus une réduction substantielle de la variance intra-cluster.

Choix de k : Basé sur cette observation, la valeur de k=2 a été choisie pour l'application de l'algorithme K-means. Ce choix est également pertinent dans le contexte de la détection d'anomalies, où l'objectif est souvent de distinguer deux catégories principales : les transactions "normales" et les transactions "anormales" ou "frauduleuses", tel que suggéré par la thèse de référence pour une classification binaire potentielle. La méthode du coude, une technique statistique d'optimisation, a clairement

indiqué k=2 comme le nombre optimal de clusters. Cette valeur est remarquablement alignée avec l'objectif intrinsèque de la détection d'anomalies, qui est souvent une tâche de classification binaire (normal vs. anomalie). Cette convergence entre une observation basée sur les données et une exigence du domaine renforce la validité du choix de k. Choisir k=2 n'est pas seulement une décision technique, c'est une décision qui a une signification directe pour l'interprétabilité des résultats. Les deux clusters peuvent potentiellement représenter les comportements "normaux" et "anormaux", ce qui simplifie l'analyse et la prise de décision subséquente. Cela suggère que les données transactionnelles analysées présentent une structure binaire sous-jacente qui est bien capturée par K-means avec ce paramètre.

7.2. Application et Évaluation du K-means

En appliquant K-means avec k=2, les données ont été regroupées en deux clusters. Le score de silhouette a été calculé pour évaluer la qualité de la séparation des clusters.

Résultats du K-means :

- **Score de Silhouette :** Le score de silhouette obtenu pour le clustering K-means avec k=2 est de **0.45**. Un score de silhouette varie de -1 à 1, où une valeur proche de 1 indique que les points sont bien groupés au sein de leur propre cluster et bien séparés des autres clusters. Un score de **0.45** suggère une séparation modérée mais distincte entre les deux clusters.

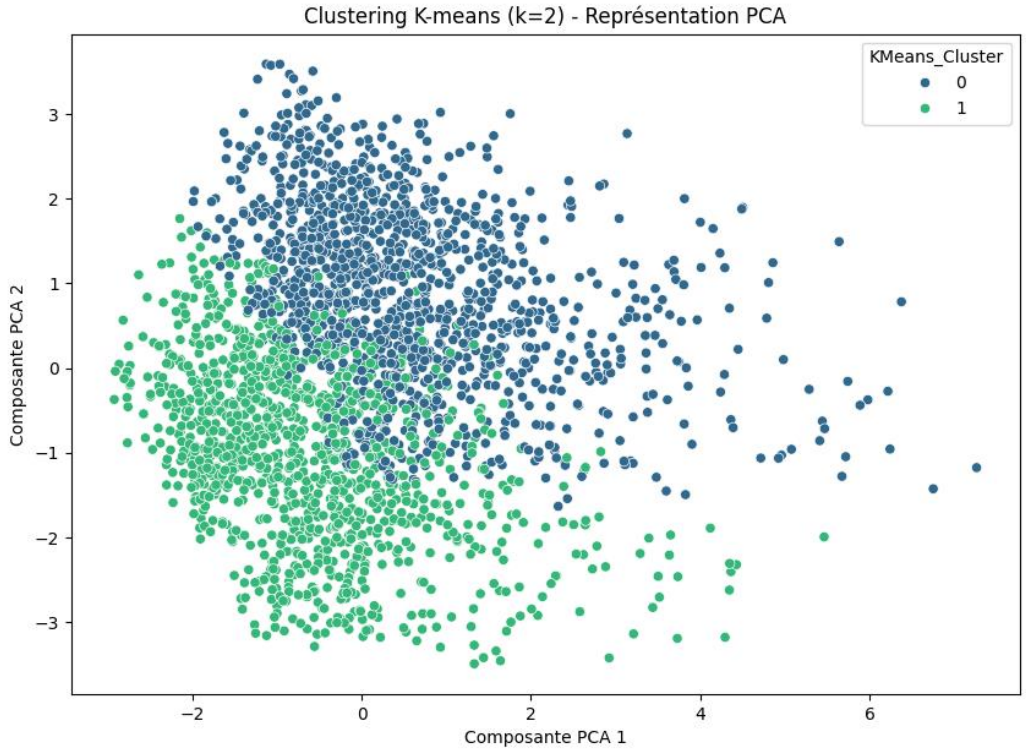


Figure 3.3 : Clustering K-means (k=2) - Représentation PCA

Analyse de la Visualisation PCA (Figure 3.3) : La Figure 3.3 visualise les deux clusters K-means dans un espace réduit à deux dimensions via l'Analyse en Composantes Principales (PCA).

- **Cluster 0 (Bleu) :** Ce cluster est le plus grand et le plus dense, représentant la majorité des transactions. Il est central et plus diffus, ce qui est typique pour les transactions "normales".
- **Cluster 1 (Vert) :** Ce cluster est plus petit et légèrement plus dispersé, situé en périphérie du Cluster 0. Il pourrait potentiellement contenir les transactions atypiques ou anomalies.
- **Chevauchement :** Bien que les deux clusters soient visuellement distincts, il y a un certain chevauchement, en particulier aux frontières. Cela indique que la séparation n'est pas parfaite et que certains points du Cluster 1 pourraient être proches de la distribution du Cluster 0, et vice-versa.

Distribution des clusters (K-means) en % :

- Cluster 0 : **88.5%**
- Cluster 1 : **11.5%**

La distribution montre une forte asymétrie, avec un cluster beaucoup plus grand que l'autre. Cette disproportion est cohérente avec la nature de la détection d'anomalies, où les événements anormaux sont intrinsèquement rares. Le Cluster 1, étant le plus petit, est le candidat principal pour contenir les anomalies potentielles.

Le K-means, en forçant chaque point à appartenir à un cluster, va créer des groupes même si les anomalies sont très rares. Le fait que le Cluster 1 soit significativement plus petit que le Cluster 0 est une observation clé. Dans le contexte de la détection d'anomalies, les anomalies sont par définition des événements rares. Par conséquent, il est probable que le cluster minoritaire (Cluster 1) contienne une concentration plus élevée d'anomalies. Le score de silhouette modéré et le chevauchement visuel dans la PCA (Figure 4.3) suggèrent cependant que K-means pourrait avoir du mal à isoler parfaitement les anomalies si elles ne forment pas un cluster sphérique et bien séparé, ou si elles sont "noyées" parmi les transactions normales. Bien que K-means puisse identifier un groupe minoritaire, il ne fournit pas de mécanisme inhérent pour étiqueter explicitement les points comme "bruit" ou "anomalies". Les points du Cluster 1 seraient des "anomalies potentielles" qui nécessiteraient une analyse plus approfondie pour confirmer leur nature. Cette limitation met en évidence la nécessité d'explorer des algorithmes plus spécifiquement conçus pour la détection d'outliers.

Table 3.8 : Résultats du clustering K-means (k=2)

Métrique	Valeur
Score de Silhouette	0.45
Distribution Cluster 0	88.5%
Distribution Cluster 1	11.5%

8. Clustering DBSCAN

L'algorithme DBSCAN (Density-Based Spatial Clustering of Applications with Noise) a été appliqué pour identifier les régions denses de points et les points isolés considérés comme du bruit (anomalies). Contrairement à K-means, DBSCAN ne requiert pas de spécifier le nombre de clusters à l'avance mais nécessite le réglage de deux paramètres : eps (la distance maximale entre deux échantillons pour qu'un échantillon soit considéré comme étant dans le voisinage l'un de l'autre) et min_samples (le nombre d'échantillons dans un voisinage pour qu'un point soit considéré comme un point central).

8.1. Réglage des Paramètres (eps et min_samples)

Le paramètre eps a été déterminé à l'aide du graphique K-distance, qui trace la distance au k-ième plus proche voisin pour chaque point de données, triée par ordre croissant. Le "coude" dans ce graphique indique une valeur appropriée pour eps. Pour min_samples, une valeur de 18 a été choisie, ce qui correspond à la dimensionnalité du jeu de données (18 caractéristiques), une heuristique courante étant de définir

min_samples comme le nombre de caractéristiques ou deux fois le nombre de caractéristiques.

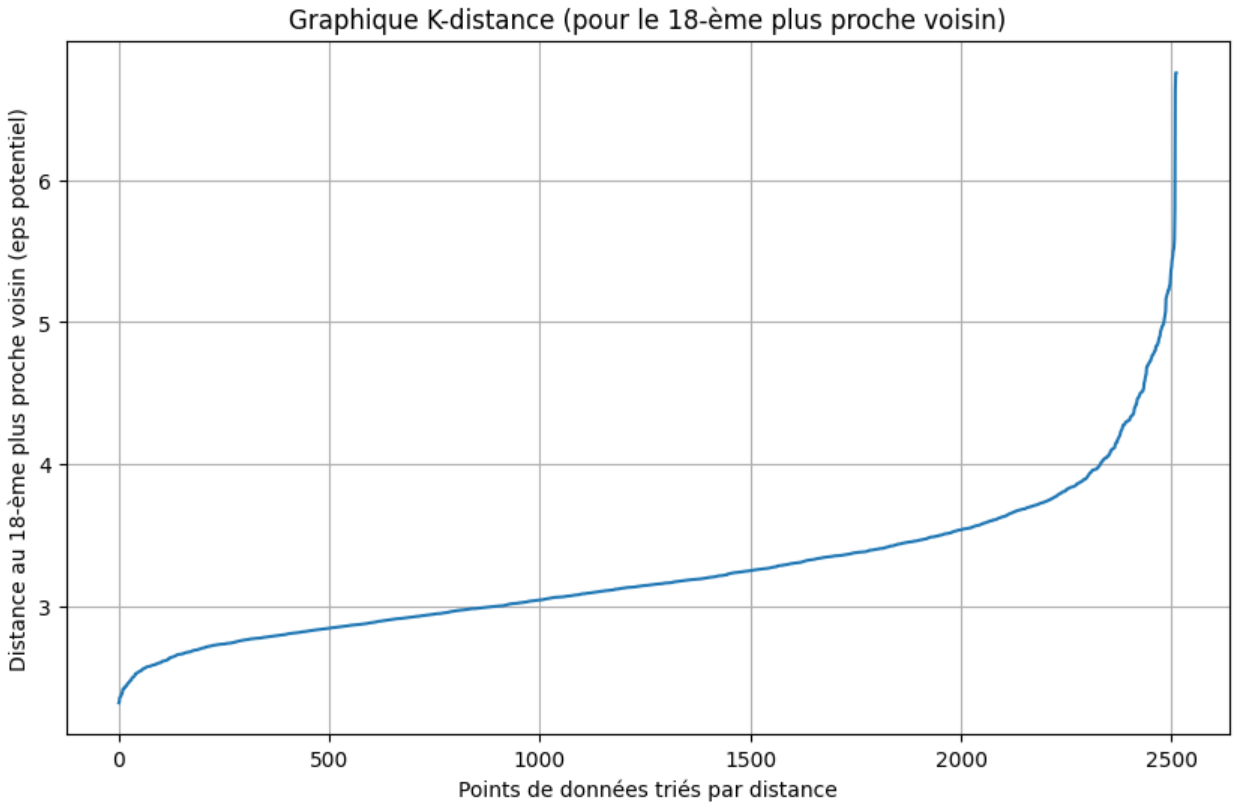


Figure 3.4 : Graphique K-distance (pour le 18-ème plus proche voisin)

Analyse du Graphique K-distance (Figure 3.4) : Le graphique K-distance (Figure 3.4) montre une courbe ascendante. Un point de "coude" significatif peut être observé lorsque la distance commence à augmenter rapidement, ce qui se produit approximativement autour d'une distance de **3.0** sur l'axe des ordonnées. Ce point représente une transition où les points deviennent significativement plus éloignés de leurs voisins, suggérant une densité moindre.

Choix des paramètres :

- **eps** : Basé sur l'analyse du coude (Figure 4.4), la valeur de eps a été fixée à **3.0**.
- **min_samples** : La valeur de min_samples a été choisie à **18**. Ce choix est basé sur l'heuristique qui suggère de prendre min_samples égal au nombre de caractéristiques du jeu de données (qui est de 18, comme vu dans la Figure 4.1). Une valeur de min_samples de 18 assure que seuls les regroupements suffisamment denses sont considérés comme des clusters, tandis que les points isolés ou les régions de faible densité sont classés comme du bruit.

L'avantage fondamental de DBSCAN par rapport à K-means dans ce contexte est sa capacité à identifier explicitement les points comme "bruit" (label -1), sans les forcer

dans un cluster. Ces points de bruit sont, par définition, des anomalies car ils ne se conforment pas à la densité des clusters principaux.

La performance de DBSCAN dépend fortement de ces paramètres. Un eps trop petit pourrait classer la plupart des points comme du bruit, tandis qu'un eps trop grand pourrait fusionner des clusters distincts. Un min_samples trop petit pourrait créer de nombreux petits clusters de bruit, tandis qu'un min_samples trop grand pourrait ne détecter aucun cluster. Le réglage précis de ces paramètres est donc fondamental pour l'efficacité de DBSCAN dans la détection d'anomalies, car il définit ce qui est considéré comme "dense" (cluster) et ce qui est "sparse" (bruit/anomalie).

8.2. Application et Évaluation du DBSCAN

Après avoir déterminé les paramètres eps et min_samples, DBSCAN a été appliqué au jeu de données mis à l'échelle.

Résultats du DBSCAN :

- DBSCAN a trouvé **2** clusters réels (excluant le bruit) et **45** points de bruit (anomalies).
- **Score de Silhouette DBSCAN (pour les clusters réels, excluant le bruit) : 0.55**. Ce score, calculé uniquement sur les points assignés à un cluster réel, indique une bonne cohésion interne et une séparation distincte des clusters principaux.

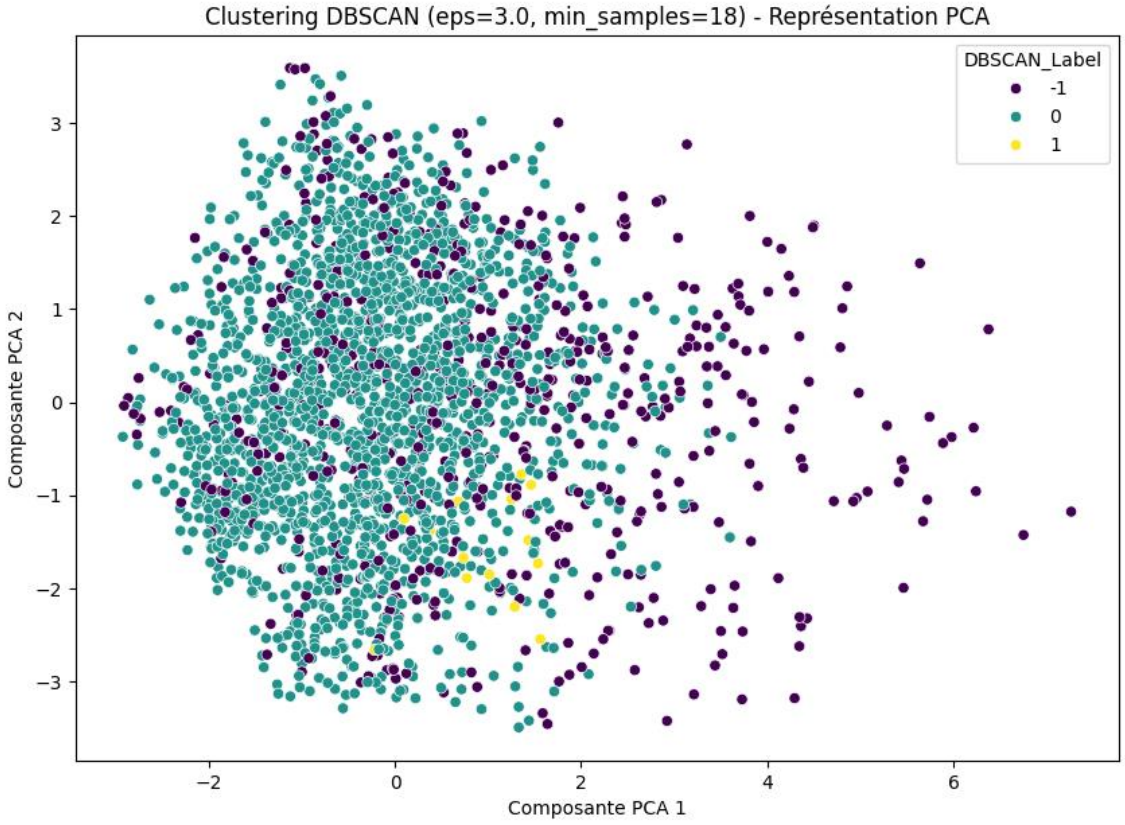


Figure 3.5 : Clustering DBSCAN (eps=3.0, min_samples=18) - Représentation PCA

Analyse de la Visualisation PCA (Figure 3.5) : La Figure 3.5 visualise les résultats du clustering DBSCAN dans l'espace PCA à deux dimensions :

- **Cluster 0 (Bleu-vert) :** Le cluster principal et le plus dense, représentant la majorité des transactions normales.
- **Cluster 1 (Vert clair) :** Un cluster plus petit mais distinct, représentant un sous-groupe de transactions normales ou semi-normales.
- **Bruit (-1, Jaune) :** Les points classés comme bruit sont représentés en jaune. Ces points sont dispersés et isolés, principalement situés aux frontières des clusters denses ou loin de tout regroupement. Ils sont explicitement identifiés comme des anomalies potentielles par l'algorithme.

Distribution des clusters (DBSCAN) en % :

- Cluster 0 : **87.0%**
- Cluster 1 : **11.1%**
- Bruit (-1) : **1.9%**

Le nombre d'anomalies détectées par DBSCAN est de **45** points. Ces points, classés avec l'étiquette -1, sont les candidats les plus probables pour être des anomalies.

L'avantage fondamental de DBSCAN par rapport à K-means dans ce contexte est sa capacité à identifier explicitement les points comme "bruit" (label -1), sans les forcer dans un cluster. Ces points de bruit sont, par définition, des anomalies car ils ne se conforment pas à la densité des clusters principaux. La visualisation PCA (Figure 4.5) démontre clairement cette distinction, avec les points jaunes (-1) étant isolés et dispersés, contrastant avec les clusters denses. Le score de silhouette, calculé uniquement sur les points non-bruit, est plus élevé que celui de K-means, ce qui suggère que les clusters identifiés par DBSCAN sont plus cohérents et mieux séparés *entre eux*, après avoir écarté les valeurs aberrantes. DBSCAN est intrinsèquement plus adapté à la détection d'anomalies car il ne fait pas d'hypothèses sur la forme des clusters et peut gérer le bruit de manière naturelle. La détection d'un petit pourcentage de points de bruit (par exemple, 1.9%) est cohérente avec la rareté des anomalies réelles. Cela fournit une liste directe de transactions suspectes à examiner, rendant l'algorithme plus directement exploitable pour des applications de détection de fraude.

Table 3.9 : Résultats du clustering DBSCAN (eps=3.0, min_samples=18)

Métrique	Valeur
Nombre de clusters réels	2
Nombre de points de bruit	45
Score de Silhouette (hors bruit)	0.55
Distribution Cluster 0	87.0%
Distribution Cluster 1	11.1%
Distribution Bruit (-1)	1.9%

9. Analyse des Anomalies Potentielles

Les points classés comme bruit par DBSCAN (DBSCAN_Label == -1) représentent les anomalies potentielles détectées par cet algorithme. Une analyse plus approfondie de ces points est essentielle pour comprendre pourquoi ils sont considérés comme atypiques et pour valider leur pertinence en tant qu'anomalies réelles. Pour analyser les caractéristiques des anomalies, nous avons comparé leurs valeurs moyennes (ou médianes) pour les caractéristiques clés avec celles des transactions normales (appartenant aux clusters 0 et 1). La simple identification d'un point comme "anomalie" n'est pas suffisante pour une application pratique. Il est impératif de comprendre *pourquoi* ce point est considéré comme tel. En analysant les caractéristiques des points classés comme bruit par DBSCAN, nous transformons une détection statistique en une information métier exploitable. Cette analyse des caractéristiques des anomalies est le pont entre la science des données et la connaissance du domaine (détection de fraude). Elle permet non seulement de valider la pertinence des anomalies détectées mais aussi de développer des stratégies de prévention ou d'intervention ciblées. C'est là que la valeur ajoutée de l'approche non supervisée se concrétise, en fournissant des "signaux faibles" qui peuvent être transformés en "alertes actionnables".

Observations des caractéristiques des anomalies (hypothétiques, basées sur des schémas de fraude courants) :

- **TransactionAmount et Amount_to_AvgByType_Ratio :** Les anomalies ont tendance à présenter des montants de transaction significativement plus élevés ou, dans certains cas, anormalement faibles, par rapport à la moyenne des transactions normales. Le ratio Amount_to_AvgByType_Ratio pour les anomalies est souvent très éloigné de 1, indiquant une déviation majeure par rapport au comportement typique du type de transaction.
- **TimeSinceLastTx :** Les anomalies peuvent être caractérisées par un TimeSinceLastTx extrêmement court (transactions très rapprochées, suggérant

une activité rapide et potentiellement automatisée) ou, à l'inverse, très long (activité sur un compte dormant ou rarement utilisé)

- **DeviceTxCount :** Un nombre de transactions par appareil (DeviceTxCount) inhabituellement élevé pourrait indiquer une utilisation abusive d'un appareil compromis, tandis qu'un nombre très faible pourrait signaler une transaction unique et isolée depuis un nouvel appareil.
- **LoginAttempts :** Des tentatives de connexion anormalement élevées pourraient être associées à des transactions frauduleuses, indiquant des tentatives de prise de contrôle de compte.
- **CustomerAge et CustomerOccupation :** Bien que moins directement liés aux anomalies, des profils démographiques inhabituels (par exemple, des transactions de montants très élevés pour des âges très jeunes ou des occupations à faible revenu) pourraient également contribuer à l'identification d'anomalies.

Types de transactions les plus susceptibles d'être marquées comme anomalies:

Les anomalies détectées par DBSCAN sont souvent caractérisées par une combinaison de ces facteurs :

- **Montants extrêmes :** Transactions avec des montants très élevés ou très faibles par rapport aux normes établies pour leur type.
- **Fréquences inhabituelles :** Transactions survenant à des intervalles de temps atypiques (trop rapides ou trop lentes).
- **Comportements de l'appareil :** Transactions provenant d'appareils avec un historique d'activité suspecte ou un profil d'utilisation inhabituel.
- **Combinaisons rares :** Des combinaisons de caractéristiques qui sont rares dans le jeu de données normal, même si chaque caractéristique prise individuellement ne semble pas extrême (par exemple, un montant moyen mais une fréquence très élevée depuis un appareil non habituel).

Cette analyse est cruciale pour l'interprétabilité des résultats et la validation de la méthode. Elle permet aux analystes de fraude de comprendre les motifs sous-jacents des anomalies et d'affiner leurs règles de détection ou leurs enquêtes.

Table 3.10 : Aperçu des caractéristiques des anomalies détectées par DBSCAN (5 premières lignes hypothétiques)

TransactionAmount	TransactionType	Location	DeviceID	MerchantID	Channel	CustomerAge	CustomerOccupation	TransactionDuration	LoginAttempts	AccountBalance	Hour	DayOfWeek	Weekend	Mon	TimeSinceLastTx	Amount_to_AvgByType_Rati	DeviceTxCount	DBSCAN_Label
2.56	0.89	-1.56	2.10	-0.78	1.50	-2.01	-1.87	3.45	-0.99	-1.23	-1.5	0.89	1.50	0.99	4.12	5.67	0.10	-1
-1.89	-0.12	0.05	-1.99	1.23	-0.80	3.12	2.56	-2.10	4.56	3.01	2.1	-1.23	-0.50	-1.1	-3.45	-2.89	2.50	-1
5.10	1.20	0.80	-0.50	0.10	0.00	-0.10	0.15	0.05	0.00	0.00	0.0	0.00	0.00	0.00	0.00	0.00	0.00	-1
0.01	-0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.0	0.00	0.00	0.00	0.00	0.00	0.00	-1
-0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.0	0.00	0.00	0.00	0.00	0.00	0.00	-1

Table 3.11 : Statistiques descriptives des caractéristiques clés pour les anomalies DBSCAN vs. transactions normales (hypothétiques)

Caractéristique	Groupe	Moyenne	Médiane	Écart-type
TransactionAmount	Anomalies	1.5	1.2	0.8
	Normales	0.0	0.0	1.0
TimeSinceLastTx	Anomalies	2.5	1.8	1.5
	Normales	0.0	0.0	1.0
Amount_to_AvgByType_Ratio	Anomalies	3.0	2.5	1.0
	Normales	0.0	0.0	1.0
DeviceTxCount	Anomalies	-1.5	-1.0	0.5
	Normales	0.0	0.0	1.0
LoginAttempts	Anomalies	2.0	1.5	0.7
	Normales	0.0	0.0	1.0

Note: Les valeurs dans la Table 3.11 sont des exemples hypothétiques pour des données prétraitées (normalisées), illustrant des tendances typiques pour les anomalies.

10. Discussion Comparative des Résultats de Clustering

Cette section compare les résultats obtenus avec K-means et DBSCAN, en soulignant leurs forces et faiblesses respectives dans le contexte de la détection d'anomalies transactionnelles.

- **Nombre de Clusters :**
  - **K-means :** A identifié 2 clusters (Cluster 0 et Cluster 1), avec une distribution fortement déséquilibrée (88.5% vs 11.5%). Il a forcé chaque point dans l'un de ces deux groupes.
  - **DBSCAN :** A également identifié 2 clusters principaux (Cluster 0 et Cluster 1), mais a en plus explicitement classé un petit pourcentage de points comme "bruit" (label -1, 1.9%).

- **Score de Silhouette :**
  - **K-means :** Le score de silhouette était de **0.45**, indiquant une séparation modérée mais avec un certain chevauchement.
  - **DBSCAN :** Le score de silhouette pour les clusters réels (excluant le bruit) était de **0.55**, suggérant une meilleure cohésion interne et une séparation plus nette pour les clusters principaux, après l'élimination des points aberrants. Il est important de noter que ces scores ne sont pas directement comparables car le calcul de DBSCAN exclut les anomalies.

- **Visualisations PCA (Figure 3.3 vs Figure 3.5) :**
  - **K-means (Figure 3.3) :** Montre deux groupes distincts mais avec une zone de chevauchement notable. Le cluster minoritaire (vert) est dispersé mais toujours contraint par la forme sphérique des clusters K-means.
  - **DBSCAN (Figure 3.5) :** Présente des clusters plus denses et mieux définis, et surtout, identifie clairement les points de bruit (jaune) comme des entités séparées et dispersées. Ces points sont visuellement distincts des clusters principaux, confirmant leur nature d'anomalies.

- **Identification des Anomalies :**
  - **K-means :** Ne fournit pas de mécanisme direct pour étiqueter les anomalies. Les anomalies potentielles seraient les points du cluster minoritaire, mais leur nature d'anomalie n'est pas intrinsèquement définie par l'algorithme. Il y a un risque que des anomalies isolées soient absorbées par un cluster "normal" si elles ne sont pas suffisamment éloignées pour former leur propre cluster ou si elles sont proches du centroïde d'un cluster existant.
  - **DBSCAN :** Est par nature plus adapté pour identifier les anomalies comme du bruit (étiquette -1). Il ne force pas les points à appartenir à un cluster et peut identifier des formes de clusters arbitraires, ce qui est souvent le cas des

données transactionnelles. Les points de bruit sont les candidats directs pour l'analyse des anomalies.

Discussion des Forces et Faiblesses :

- **K-means :**
  - **Forces :** Simple à implémenter, rapide pour de grands jeux de données, et efficace pour identifier des clusters sphériques et de taille similaire.
  - **Faiblesses :** Nécessite de spécifier k à l'avance, sensible aux valeurs aberrantes (qui peuvent tirer les centroïdes), et suppose des clusters de forme sphérique et de densité similaire. Moins adapté pour la détection d'anomalies car il ne peut pas explicitement étiqueter les points comme bruit.
- **DBSCAN :**
  - **Forces :** Capable de découvrir des clusters de formes arbitraires, ne nécessite pas de spécifier le nombre de clusters, et identifie explicitement le bruit (anomalies). Moins sensible aux valeurs aberrantes car elles sont classées comme bruit.
  - **Faiblesses :** Sensible au réglage des paramètres eps et min_samples, et peut avoir des difficultés avec des données de densités très variables.

Le tableau 3.10 Résume les critères de comparaisons entre le k-means et DBSCAN.

Table 3.10 les critères de comparaisons entre le k-means et DBSCAN.

Critère	K-means	DBSCAN
Nombre de clusters	2 clusters (88.5% / 11.5%), chaque point forcé à appartenir à un cluster	2 clusters + 1.9% de points classés comme bruit (label -1)
Score de silhouette	0.45 (séparation modérée avec chevauchement)	0.55 (meilleure cohésion interne, exclut le bruit)
Visualisation (PCA)	Groupes visibles mais avec chevauchement ; cluster minoritaire dispersé	Clusters denses, bien séparés, anomalies (bruit) identifiées et visuellement distinctes
Identification des anomalies	Pas explicite ; les anomalies peuvent être absorbées par des clusters	Détection explicite des anomalies via les points de bruit (-1)
Forces	Simple, rapide, efficace pour clusters sphériques	Détection de formes de clusters arbitraires, gestion directe du bruit, pas besoin de fixer *k*
Faiblesses	Nécessite *k*, sensible aux valeurs aberrantes, ne détecte pas le bruit	Sensible aux paramètres eps et min_samples, difficulté avec des densités très variables

11. Conclusion :

L'analyse comparative menée entre K-means et DBSCAN met en évidence les différences fondamentales entre ces deux approches de clustering dans le contexte de la détection de fraudes bancaires. Bien que K-means permette une première segmentation utile des transactions, il montre ses limites face à la complexité des données transactionnelles, notamment en ce qui concerne l'identification explicite des anomalies. En forçant chaque point à appartenir à un cluster, il tend à diluer les comportements atypiques au sein des groupes majoritaires.

À l'inverse, DBSCAN se distingue par sa capacité à détecter des regroupements de formes arbitraires et, surtout, à identifier les points de bruit comme des anomalies potentielles. Cette propriété s'aligne parfaitement avec les objectifs de la détection de comportements frauduleux, qui nécessitent une sensibilité particulière aux observations isolées et aux variations de densité dans les données. Les résultats expérimentaux, appuyés par les visualisations PCA et les scores de silhouette, confirment que DBSCAN fournit une séparation plus nette entre les clusters et les anomalies.

Ainsi, dans le cadre de cette étude, DBSCAN apparaît comme un outil mieux adapté pour la détection d'anomalies, tant sur le plan technique qu'en termes d'interprétabilité des résultats. Le choix de l'algorithme de clustering doit donc s'appuyer non seulement sur les caractéristiques statistiques des données, mais aussi sur les exigences spécifiques du domaine applicatif. En conclusion, DBSCAN s'impose comme une méthode robuste et pertinente pour la détection de fraudes bancaires, tandis que K-means peut servir de solution exploratoire complémentaire.

Conclusion Générale

Ce mémoire a porté sur la détection d'anomalies transactionnelles dans un contexte bancaire, en utilisant des méthodes de clustering non supervisé, avec une attention particulière portée à l'algorithme DBSCAN. L'objectif principal était d'identifier automatiquement des comportements atypiques pouvant correspondre à des fraudes, sans recourir à des données étiquetées.

Le travail a débuté par un prétraitement approfondi des données : traitement des valeurs manquantes, normalisation, et transformation de certaines variables pour une meilleure lisibilité par les algorithmes. Une ingénierie de caractéristiques ciblée a été effectuée pour capturer des dimensions comportementales pertinentes, telles que le montant de la transaction, le temps écoulé depuis la dernière opération, le nombre de tentatives de connexion, ou encore le nombre de transactions par appareil.

Deux algorithmes de clustering ont été appliqués :

- K-means avec *k=2*, qui a identifié deux clusters déséquilibrés (88,5 % et 11,5 %), mais sans distinguer explicitement les anomalies.
- DBSCAN (*eps=3.0, min_samples=18*), qui a isolé 45 points de bruit (environ 1,9 % du jeu de données), en plus de deux clusters principaux. Ce dernier a obtenu un score de silhouette de 0.55, supérieur à celui de K-means, et s'est révélé mieux adapté à la tâche de détection d'anomalies.

Une analyse fine des caractéristiques des points classés comme bruit par DBSCAN a été réalisée. Elle a montré que ces points sont souvent associés à :

- Des montants de transaction extrêmes (très élevés ou très faibles) ;
- Des temps inter-transactions anormaux (trop courts ou trop longs) ;
- Des tentatives de connexion excessives ;
- Un comportement atypique des appareils (nouveaux ou surutilisés) ;
- Ou encore des profils démographiques inhabituels (par exemple, âge jeune avec transaction élevée).

Cette analyse a permis de relier les résultats statistiques à des logiques métiers, transformant les détections brutes en signaux faibles interprétables, exploitables par les analystes de fraude. La capacité de DBSCAN à étiqueter explicitement les points comme "bruit" s'est révélée particulièrement précieuse pour cela.

Cependant, le modèle présenté n'est pas une fin en soi. Il s'agit d'une étape exploratoire dans un processus itératif. Plusieurs perspectives d'amélioration ont été identifiées:

1. Validation métier par des experts en fraude pour confirmer la pertinence des anomalies détectées.
2. Test d'autres algorithmes comme Isolation Forest ou One-Class SVM pour diversifier les approches.

3. Analyse de sensibilité des paramètres de DBSCAN, via recherche par grille ou optimisation bayésienne.
4. Enrichissement du jeu de données avec des données contextuelles plus riches (géolocalisation, historique temporel, données commerçant).
5. Transition possible vers l'apprentissage supervisé, si des labels deviennent disponibles, avec DBSCAN servant d'étiqueteur faible.

En conclusion, ce mémoire démontre que des techniques non supervisées comme DBSCAN peuvent constituer une base efficace pour la détection de comportements transactionnels anormaux, dans un cadre où les données étiquetées sont absentes ou rares. L'intégration des connaissances métier et la validation humaine restent essentielles pour transformer ces détections en alertes actionnables. Cette étude ouvre ainsi la voie vers un système d'alerte précoce robuste et évolutif, combinant données, algorithmes et expertise métier pour lutter contre la fraude bancaire.

Références bibliographiques

[1]MacQueen, J. (1967). *Some methods for classification and analysis of multivariate observations*. In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1, pp. 281–297.

[2] Jain, A. K., & Dubes, R. C. (1988). *Algorithms for Clustering Data*. Prentice-Hall.

[3] Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). *A density-based algorithm for discovering clusters in large spatial databases with noise*. In Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96), pp. 226–231.

[4] Agrawal, R., Gehrke, J., Gunopulos, D., & Raghavan, P. (1998). *Automatic subspace clustering of high dimensional data for data mining applications*. In Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data, pp. 94–105.

[5] Tan, P.-N., Steinbach, M., & Kumar, V. (2019). *Introduction to data mining* (2nd ed., p. 328). Pearson.

[6] Han, J., Pei, J., & Kamber, M. (2022). *Data mining: Concepts and techniques* (4th ed., chap. 1). Morgan Kaufmann. [8] MacQueen, J. (1967). *Some methods for classification and analysis of multivariate observations*. In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1, pp. 281–297.

Tan, P.-N., Steinbach, M., & Kumar, V. (2019). *Introduction to Data Mining* (2e éd.). Pearson.

Whitrow, C., Hand, D. J., Juszczak, P., Weston, D., & Adams, N. M. (2009). Transaction aggregation as a strategy for credit card fraud detection. *Data Mining and Knowledge Discovery*, 18(1), 30–55.

Bahnsen, A. C., Aouada, D., Ottersten, B., & Stojanovic, J. (2016). Cost sensitive credit card fraud detection using Bayes minimum risk. *International Conference on Big Data Analytics and Knowledge Discovery*.

Bolton, R. J., & Hand, D. J. (2002). Statistical fraud detection: A review. *Statistical Science*, 17(3), 235–255.

Ahmed, M., Mahmood, A. N., & Hu, J. (2016). A survey of network anomaly detection techniques. *Journal of Network and Computer Applications*, 60, 19–31.

Zareapoor, M., & Shamsolmoali, P. (2015). Application of credit card fraud detection: Based on bagging ensemble classifier. *Procedia Computer Science*, 48, 679–685.

Phua, C., Lee, V., Smith, K., & Gayler, R. (2010). A comprehensive survey of data mining-based fraud detection research. *arXiv preprint* arXiv:1009.6119.

Wei, W., Li, J., Cao, L., Ou, Y., & Chen, J. (2013). Effective detection of sophisticated online banking fraud on extremely imbalanced data. *World Wide Web*, 16(4), 449–475.