Apprentissage non supervisé appliqué au problème de détection de fraude

Jean-Charles Verdier [1] **Wanlin Li** [1] **Cédrick Jonathan Randriamilamina** [1]

1. Introduction

Les paiements par cartes de crédit constituent une part importante des transactions financières dans le monde. Elles offrent une facilité et une efficacité inégalée par d'autres types de paiement, particulièrement pour les transactions effectuées sur internet. Or avec une telle ubiquité vient des opportunités de fraude pour les criminels. Aux États-Unis seulement, on estime les coûts engendrés par la fraude fiscale à 32 milliards pour l'année 2013 [1]. À ce jour, un Américain sur 10 a été victime de fraude (avec un montant médian à 399$) [3]. Le plus récent rapport de la Banque Centrale Européenne comptabilise, pour l'année 2018, le montant total des transactions frauduleuses à plus de 1.8 milliard [14]. Depuis, de nombreuses solutions ont été implémentées afin prévenir cette fraude - comme l'ajout du code de sécurité CVV et l'arrivée des cartes magnétiques. Toutefois, les fraudeurs s'adaptent et développent des nouvelles techniques sophistiquées qui échappent aux systèmes de prévention actuels. Il s'avère donc important de développer des méthodes dynamiques permettant de prévenir ou du moins détecter la présence de transactions frauduleuses.

On distingue deux catégories de fraude : carte présente et carte absente. La première catégorie réfère aux scénarios où la carte est physiquement requise pour compléter un achat comme dans un magasin, par exemple. Elle implique généralement la duplication ou le vol d'une carte de crédit. Ce type de fraude est largement contré par l'arrivée des cartes à puce. En contraste, la catégorie carte absente réfère aux transactions par téléphone, internet ou par poste où la carte physique n'est pas requise. Cette taxonomie est nécessaire car les techniques pour compromettre les cartes de crédit varient en fonction de la présence ou l'absence physique de la carte. Dans notre travail, nous étudions plutôt la deuxième catégorie car elle représente aujourd'hui la majorité des fraudes [9] et parce que nos données proviennent de la compagnie Vesta qui se spécialise dans les transactions en ligne. Détecter manuellement des fraudes s'avère extrêmement onéreux en temps considérant le nombre élevé de transactions quotidiennes. En moyenne, une fraude est découverte après 72 heures d'analyse [1]. Pour cette raison, les techniques d'apprentissage automatique sont de plus en plus considérées car elles permettent une détection beaucoup plus rapide des fraudes. Les méthodes supervisées – qui utilisent les étiquettes lors de leur entraînement – sont délaissées en faveur des méthodes non supervisées car les premières ne permettent pas de détecter des nouveaux types de fraude et pour des considérations pratiques que nous verrons plus loin. Ainsi, dans ce travail, nous étudions quelques techniques de détection de fraude basées sur l'apprentissage non supervisé et comparons leur performance sur le jeu de données IEEE-CIS Fraud Detection (IEEE-FD) offert sur la plateforme Kaggle . Dans la prochaine section, nous définissons plus précisément le problème avant de présenter une revue de la littérature sur le sujet. Ensuite, nous analysons en rapidement les données de IEEE-FD avant de présenter les algorithmes utilisés.

2. Détection de fraude

Le problème de détection de fraude se généralise bien comme un problème de détection d'anomalie dans lequel les transactions frauduleuses représentent les anomalies par contraste aux transactions dites normales ou légitimes. La détection d'anomalie vise l'identification des observations qui se distinguent significativement d'un comportement attendu. Cette définition intuitive cache deux concepts sous-jacents: le concept de normalité et le concept de distance par rapport à celle-ci. Ces concepts prennent des significations différentes en fonction du cadre dans lequel on se trouve. Dans une approche probabiliste, par exemple, on définit l'ensemble des anomalies comme $A = \{x \in \mathcal{X} : \mathbb{P}^+(x) \leq \tau, \tau > 0\}$ où $\mathbb{P}^+$ représente la fonction de densité de la classe normale et $\mathcal{X} \in \mathbb{R}^d$ représente l'espace des données ([10]). Un cadre géométrique utiliserait plutôt une notion de distance numérique pour identifier les anomalies: $A = \{x \in \mathcal{X} : d(x) \leq \tau, \tau > 0\}$ où $d(x)$ représente une fonction de distance quelconque (euclidienne, Manhattan, Mahalanobis, etc.). Dans tous les cas, la tâche de détection de fraude se range dans la catégorie des problèmes de classification binaire où on tente de séparer les transactions en deux catégories : légitimes et frauduleuses. De plus, la détection de fraude présente des défis particuliers :

[*]Equal contribution [1]Département d'informatique, Université de Sherbrooke, Sherbrooke, Canada. Correspondence to: Jean-Charles Verdier <verj2009@usherbrooke.ca>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

- **Dérive conceptuelle**. Les transactions normales et frauduleuses changent avec le temps. D'un côté les comportements des consommateurs varient en fonction des saisons et journées de la semaine. De l'autre, les criminels innovent et développent de nouvelles techniques pour frauder leurs victimes. Les algorithmes d'apprentissage assument souvent que les données sont i.i.d (indépendantes et identiquement distribuées). Cette hypothèse permet de généraliser le critère de décision de l'ensemble d'entraînement à l'ensemble de test. Or, elle ne tient pas dans la détection de la fraude car la distribution des données change avec le temps.

- **Débalancement de classe**. Les anomalies sont beaucoup moins fréquentes que les transactions normales. Le pourcentage de fraude dans les données est de l'ordre de 1% seulement. On doit donc développer des méthodologies particulières pour entraîner un classifieur sans quoi on court le risque de sur-apprentissage sur la classe majoritaire.

- **Disponibilité des données**. Pour des raisons de confidentialité évidentes, les données de transactions financières ne peuvent pas être partagées publiquement. Lorsqu'elles le sont, les variables doivent être lourdement anonymisées de telle sorte qu'on ne sait plus ce qu'elles représentent.

- **Métriques de performance**. On doit choisir des métriques de performance qui sont robustes au débalancement de classe. Les mesures classiques comme Accuracy et AUROC doivent être abandonnées en faveur de mesures plus sensibles aux capacités prédictives sur la classe minoritaire (les fraudes). En effet, on peut mal classifier toutes les instances de la classe minoritaire et obtenir une excellente Accuracy (de l'ordre de .90 et plus). Quant à l'AUROC, il donne autant de poids aux prédictions sur la classe minoritaire que la classe majoritaire. Un excellent AUROC peut donc cacher de moins bonnes performances sur la classe minoritaire [5].

- **Apprentissage non supervisé**. Étant donné la dérive conceptuelle et des contraintes pratiques, un apprentissage non supervisé est favorisé par rapport à un apprentissage supervisé. En effet, puisque les anomalies ne sont pas fixes et changent avec le temps, un modèle supervisé aura de la difficulté à détecter de nouvelles fraudes. Étant donné le faible ratio de fraudes dans les transactions quotidiennes, il est très coûteux pour une entreprise de mettre en place une équipe dont la tâche est d'étiquetée chaque transaction.

3. Revue de la littérature

Dans cette section nous révisons quelques algorithmes de détection de fraude basées sur l'apprentissage automatique ainsi que quelques techniques d'ingénierie des données utilisées dans le problème. Les transactions peuvent être conçues comme une séquence discrète d'événements pouvant être modélisée par un automate de Markov à états cachés (MMC). Khan et al. [7] ont proposé ce modèle afin de produire une séquence de transactions. Un algorithme de clustering est appliqué à ces séquences afin de diviser les transactions en trois groupes définis par rapport à leur montant (petit, moyen et grand). Les nouvelles transactions sont comparées aux dix plus récentes et sont autorisées si l'intersection entre les deux ensembles est non nul. Les auteurs ne donnent toutefois aucune mesure de performance pour évaluer l'efficacité de leur approche [1]. Des séquences d'événements peuvent aussi être modélisés par des réseaux de neurones spécialisés comme les LSTM (Long Short Term Memory). Bontemps et al. [2] utilisent cette architecture pour produire des profils transactionnels pour chaque utilisateur. Chaque nouvelle séquence est comparée au profil attendu à l'aide d'une métrique de distance qui sert ultimement de mesure d'anomalie.

Des arbres de décision et des séparateurs à vaste marge (SVM) ont aussi été étudiés pour la détection de fraude. Sahin et Duman [12] ont comparé les performances des deux approches sur des données avec des taux de contamination différents. Les données sont divisées en trois groupes avec un ratio entre les anomalies et les transactions normales différent (1:1, 1:4 et 1:9). Les auteurs ont expérimenté avec sept SVM différents et quatre fonctions noyaux et ont constaté la supériorité des arbres de décision avec une exactitude variant entre 83.02 et 94.76. Toutefois, la mesure d'exactitude n'est pas appropriée dans les situations de débalancement de classe car les performances sur la classe minoritaire ont peu d'impact sur le résultat obtenu.

Des réseaux de neurones profonds, quoique dans une proportion beaucoup plus modeste, ont aussi été proposé. Van Vlasselaer et al., [15] ont appliqué une méthode de segmentation du marché (Récence, Fréquence, Montant) pour produire des nouveaux attributs discriminants. Trois classifieurs différents sont évalués sur 78 variables: régression logistique, forêt d'arbres décisionnels et un réseau de neurones [1]. Les auteurs utilisent encore la mesure biaisée d'exactitude afin d'évaluer la meilleure méthode.

La détection de fraude offre des possibilités intéressantes en termes d'ingénierie des données afin de générer de nouvelles variables discriminantes. L'entropie de Shannon peut être utilisée afin de quantifier l'information apportée par une nouvelle transaction pour un consommateur. Une grande entropie peut ainsi être indicatrice d'une fraude [4]. Certaines fraudes sont commises dans des courts laps de temps et

Table 1. Informations de base sur les données IEEE-CIS Fraud Detection

Nombre d'observations N	Nombre de variables D	Ratio d'anomalie ρ
590 540	434	0.035

peuvent être facilement détectés en associant chaque transaction à une variable delta qui représente la distance temporelle avec la plus récente transaction. Une petite valeur contribuerait ainsi à l'identification d'une fraude [6]. Une autre approche intéressante est l'utilisation d'un graphe biparti où les nœuds correspondent aux consommateurs et marchands et les arêtes représentent les transactions entre les parties [15]. Celles-ci sont pondérées par le volume de transactions et décroisent de manière exponentielle en fonction du temps. Un algorithme PageRank est utilisé pour mesurer l'exposition des nœuds à de la fraude et ultimement en retirer des attributs pertinents [8].

4. Descriptions des données

Les données proviennent de la base de données IEEE-CIC Fraud Detection et sont offertes publiquement sur le site Kaggle. Elles sont divisées en deux fichiers «identity» et «transaction» qui contiennent respectivement des informations sur les propriétaires de carte de crédit et les transactions. Les deux fichiers sont joints à l'aide de la clé «TransactionID». Il est important de noter que toute transaction n'est pas nécessairement liée avec un utilisateur, ce qui génère beaucoup de valeurs manquantes lors de la fusion des deux fichiers. D'ailleurs, on compte 590 540 observations avec 434 variables différentes dont la plupart ont été anonymisées pour des raisons de confidentialité. N'ayant pas accès à la signification de ces variables, notre analyse est fortement contrainte. Le tableau 1 résume les informations principales de notre jeu de données. Dans ce paragraphe, nous essayons de trouver des attributs permettant de bien discriminer les transactions frauduleuses.

4.1. Débalancement de classe

Avant, on note un fort débalancement de classe (voir 4.1). En effet, on compte 569 877 transactions légitimes pour seulement 20 663 transactions frauduleuses. Ces dernières ne représentent donc que 3.626% des transactions légitimes. Des stratégies de rééchantillonnage devront être considérées pour corriger la situation.

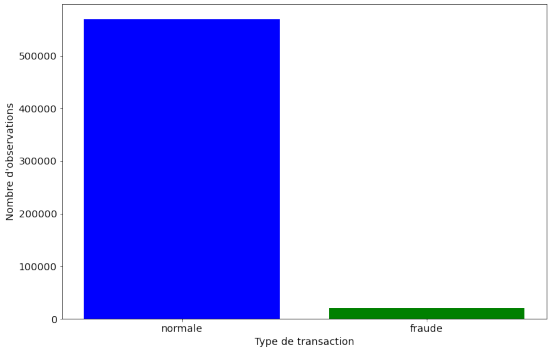

Figure 1. Débalancement de classe

4.2. Jours de la semaine

En premier lieu, vérifions la répartition des fraudes pendant la semaine. Les données originales ne contiennent pas directement cette information. Nous pouvons toutefois l'obtenir à partir du champs «TransactionDT» qui représente l'estampille temporelle de la transaction. Il suffit de diviser la valeur par le nombre de secondes dans une journées $(60 * 60 * 24)$ et calculer son modulo $\Big((60 * 60 * 24)\%7\Big)$. On constate que le jour 0 contient le plus grand nombre de transactions (4.2), mais que les transactions frauduleuses ne sont pas particulièrement présentes lors d'une journée en particulier (4.2).

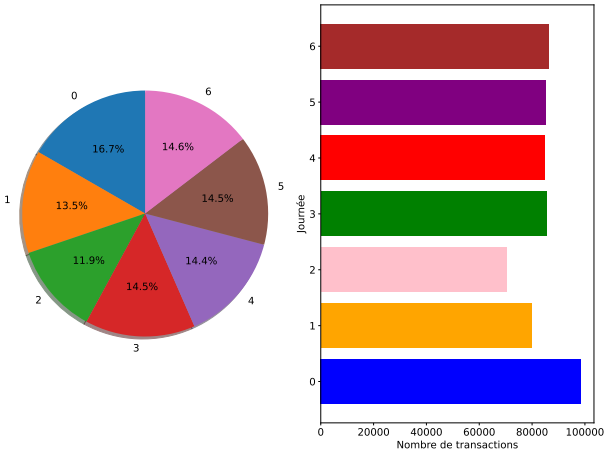

Figure 2. Nombre de transactions et pourcentage par jour

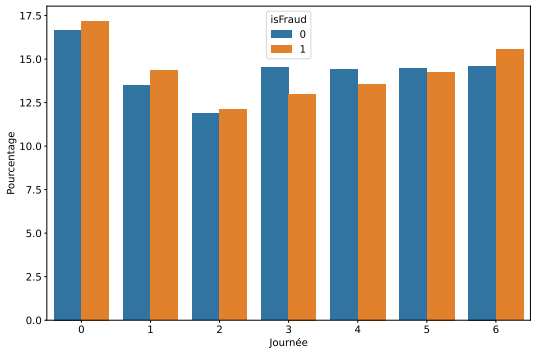

Figure 3. Proportion de fraudes par jour

4.3. Produits

Vérifions maintenant s'il existe une relation entre le type de produit et la fraude. La figure 4.3 indique que le produit «W» représente plus de 40% de toutes les fraudes et est suivi par «C» avec un peu moins de 40%. Toutefois, «W» représente 74.5% des données contre seulement 11.6% pour «C». Ce dernier est donc clairement surreprésenté dans les fraudes.

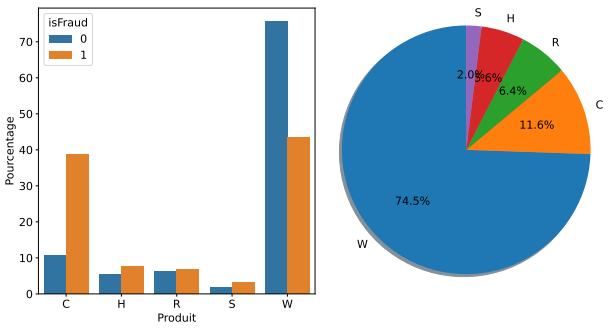

Figure 4. Représentation des produits parmi les fraudes

4.4. Attributs des cartes

Six attributs sont utilisés pour représentées les différentes cartes : «card1», «card2», «card3», «card4», «card5», «card6». Les variables «card4» et «card6» représentent respectivement le distributeur de la carte (Mastercard, Discovery ou Visa) et le type de la carte (crédit ou débit). Les autres champs sont difficiles à interpréter mais contiennent tous des valeurs numériques. Les histogrammes de chaque attribut (figure 4.4) révèlent que « card3 » contient majoritairement deux valeurs : 150 et 180. Les histogrammes des autres champs montrent une grande variété de valeurs différentes, ce qui implique qu'ils ne sont facilement discriminants. En analysant la distribution de « card3 » en fonction de la classe (figure 4.5), on constate que la probabilité de rencontrer une transaction frauduleuse après 150 augmente considérablement.

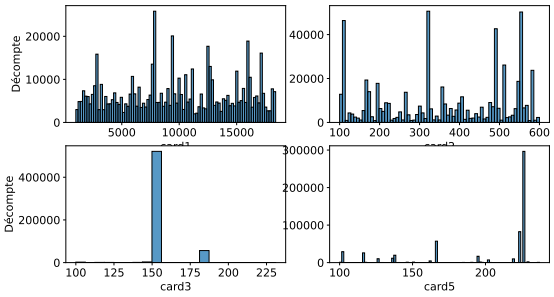

Figure 5. Histogrammes des attributs «card»

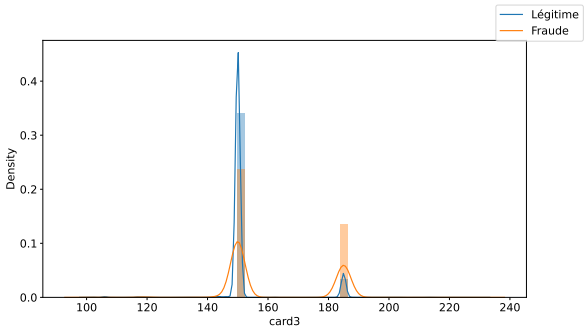

Figure 6. Distribution de «card3» en fonction de chaque classe

4.5. Fournisseurs des cartes

La plupart des transactions sont effectuées via les cartes Visa. Or la carte Discover est largement surreprésentée dans les transactions frauduleuses avec 7% de toutes les fraudes pour seulement quelques milliers de achats. Par contraste, seulement un peu plus de 3% des transactions par carte Visa sont frauduleuses alors que celles-ci compte entre $350\,000$ et $400\,000$ de toutes les transactions.

4.6. Domaine de email

Certaines des transactions sont effectuées sur internet et nous avons accès au nom de domaine associé à l'utilisateur de ces transactions. La figure 4.6 révèle que le domaine «protonmail.com» est relié à 40% des fraudes commises pour les achats en ligne.

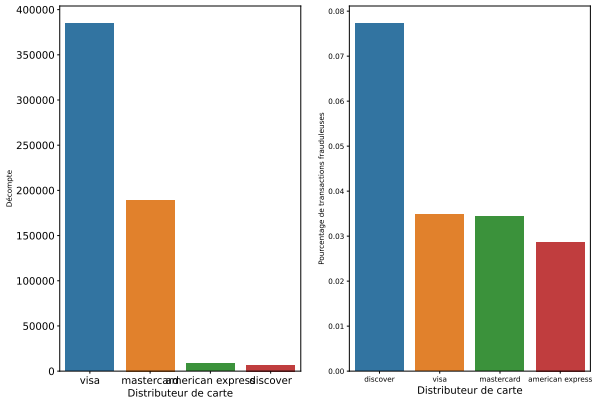

Figure 7. Représentation des fournisseurs de cartes parmi les fraudes

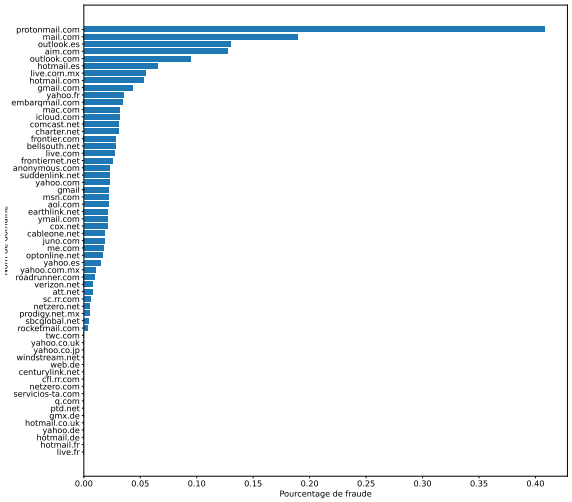

Figure 8. Nom de domaines et les transactions frauduleuses

Bref, on voit comment les fraudes varient en fonction des produits, des fournisseurs de carte et des noms de domaine des adresses emails. Celles-ci sont surreprésentées dans les produits «c» et pour la carte Discovery. De plus, l'attribut «card6» semble un bon indicateur de la présence ou l'absence de fraude avec un seuil clair à 160.

5. Expériences

Dans cette section, nous présentons brièvement les algorithmes de détection de fraude non supervisés utilisés, notre procédure d'entraînement sur les données de IEEE-CIS Fraud Detection.

5.1. Algorithmes

Nous sélectionnons les algorithmes en fonction de leur impact historique (en termes de citations) et de leur originalité. Nous tentons aussi de sélectionner une variété de modèles suivants chacun des approches différentes afin de permettre une interprétation des résultats plus intéressante.

DAGMM: Deep Autoencoding Gaussian Mixture Model [16]. Dans cette architecture, un auto-encodeur est entraîné pour apprendre un sous-espace dimensionnel permettant de représenter les données originales tout en minimisant l'erreur de reconstruction entre celles-ci et leur reconstruction à partir du sous-espace. L'erreur de reconstruction est concaténée avec la représentation latente et est fournie à autre réseau de neurones qui estime les paramètres d'une mixture gaussienne (GMM). Une fonction objectif conjointe permet d'entraîner les deux réseaux simultanément. Finalement, une fonction d'énergie (le logarithme de la vraisemblance) est utilisée comme mesure d'anomalie pour une observation quelconque. L'intuition derrière cette approche est que les anomalies sont incompressibles et vont générer des erreurs de reconstruction importantes comparativement aux instances normales.

DeepSVDD: Deep One-Class Classification [11]. Cette approche, beaucoup plus géométrique, nécessite moins d'hyperparamètres que DAGMM et n'admet aucune présupposition sur les données. Elle prend ses racines dans les méthodes SVDD qui minimisent le volume d'une sphère contenant les données, mais utilise en même temps un réseau de neurones pour apprendre une représentation des données. Autrement dit, un réseau de neurones est entraîné afin d'apprendre une représentation des données se trouvant dans une hypersphère de volume minimal. La distance euclidienne par rapport au centre de cette hypersphère est utilisée pour identifier les anomalies.

One-Class SVM [13]. Cette variante des SVM cherche une représentation des données et un hyperplan de marge maximal permettant de bien séparer les données projetées dans le nouvel espace dimensionnel. Une observation est jugée normale ou anormale en fonction de sa position par rapport à cet hyperplan. Ce problème peut être traduit par une formulation quadratique et ainsi être résolue à l'aide des multiplicateurs de Lagrange. La procédure d'optimisation est donc plus simple que celle d'un réseau de neurones, mais nécessite le maintien en mémoire de la matrice des noyaux qui croît en fonction du nombre de données fournies en entrée.

5.2. Procédure d'entraînement

Les réseaux de neurones sont implémentés à l'aide de PyTorch et sont optimisés par l'algorithme ADAM avec un taux d'apprentissage α obtenu par recherche d'hyperparamètre ($\alpha \in \{0.01, 0.001, 0.0005\}$). Nous évaluons aussi différente taille de lot: $16, 256$ et 1024. Nous utilisons l'implémentation de OC-SVM offerte par la librairie Scikit-Learn avec différents noyaux et différentes valeurs pour le paramètre ν. Les données sont séparées en deux sous-ensembles: l'ensemble d'entraînement et l'ensemble de tests. Les deux sous-ensembles se partagent 50% des données normales et l'entièreté des fraudes sont introduites dans les données de tests. Cette stratégie suit la tradition de détection d'anomalies où on assume que nos données sont largement normales [16, 11]. Nous évaluons aussi la sensibilité des approches à la corruption en injectant un taux de fraudes dans les données d'entraînement: 1%, 5% et 10%. Finalement, nous considérons les métriques precision, recall, f1-score et AUPR pour comparer la performance des algorithmes. Le seuil τ pour les trois premières mesures est déterminé en fonction du taux d'anomalies dans les données. Étant donné un ratio ρ de fraudes, on s'attend à ce que les $(1-\rho)^e$ plus grands scores soient associés à des anomalies [16].

5.3. Résultats et interprétation

Cette section sera remplie lorsque les expériences seront complétées.

6. Conclusion

En conclusion, IEEE-CIC Fraud Detection constitue une base de données assez complexe pour le problème de détection de fraudes. Son grand débalancement de classe, ses attributs hétérogènes et le grand nombre de valeurs manquantes rendent son traitement particulièrement difficile. De plus, l'absence de titres significatifs pour tous les attributs rend toute interprétation difficile. Nous avons tout de même trouvé des variables potentiellement discriminantes parmi les cartes, les types d'achats et les noms de domaine des adresses emails. Il sera intéressant de valider si ces variables seront toujours discriminantes lors de l'évaluation des modèles sur l'ensemble de test. En effet, la plupart des algorithmes d'apprentissage automatique supposent une distribution identique des variables entre le jeu d'entraînement et le jeu de test. Or, il n'est pas évident que ce soit le cas ici. Les distributions des fournisseurs de carte, des produits vendus et des noms de domaine sont appelées à changer avec le temps. Nous pourrons valider ou infirmer ces hypothèses en entraînant les données sur les modèles d'apprentissage non supervisés DAGMM, DeepSVDD et One-Class SVM.

Références

[1] Adewumi A.O. and Akinyelu A.A. "A survey of machine-learning and nature-inspired based credit card fraud detection techniques". In: ed. by Pat Langley. 2017.

[2] L Bontemps et al. "Collective anomaly detection based on long short-term memory recurrent neural networks". In: *Int J Adv Res Comput Commun Eng*. 2016.

[3] *Credit Card Fraud Statistics. Statistic Brain*. 2018. URL: https://www.statisticbrain.com/credit-card-fraud-statistics/.

[4] Kang Fu et al. "Credit Card Fraud Detection Using Convolutional Neural Networks". In: *Neural Information Processing*. Ed. by Akira Hirose et al. Cham: Springer International Publishing, 2016, pp. 483–490. ISBN: 978-3-319-46675-0.

[5] László Jeni, Jeffrey Cohn, and Fernando De la Torre. "Facing Imbalanced Data - Recommendations for the Use of Performance Metrics". In: vol. 2013. Sept. 2013. DOI: 10.1109/ACII.2013.47.

[6] Johannes Jurgovsky et al. "Sequence classification for credit-card fraud detection". In: *Expert Systems with Applications* 100 (2018), pp. 234–245. ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2018.01.037. URL: https://www.sciencedirect.com/science/article/pii/S0957417418300435.

[7] Ahmed AHE Khan MZ Pathan JD. "Credit card fraud detection using hidden markov model and K-clustering". In: *Int J Adv Res Comput Commun Eng 3*. 2014, pp. 5458–5461.

[8] Yvan Lucas and Johannes Jurgovsky. "Credit card fraud detection using machine learning: A survey". In: *CoRR* abs/2010.06479 (2020). arXiv: 2010.06479. URL: https://arxiv.org/abs/2010.06479.

[9] *Reproducible machine learning for credit card fraud detection - practical handbook*. URL: https://fraud-detection-handbook.github.io/fraud-detection-handbook/Chapter_3_GettingStarted/Introduction.html.

[10] Lukas Ruff et al. "A Unifying Review of Deep and Shallow Anomaly Detection". In: *CoRR* abs/2009.11732 (2020). arXiv: 2009.11732. URL: https://arxiv.org/abs/2009.11732.

[11] Lukas Ruff et al. "Deep One-Class Classification". In: *Proceedings of the 35th International Conference on Machine Learning*. Ed. by Jennifer Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, Oct. 2018, pp. 4393–4402.

[12] Duman E Sahin Y. "Detecting credit card fraud by decision trees and support vector machines". In: *Proceedings of the international multiConference of engineers and computer scientists*. 2011, pp. 1–6.

[13] Bernhard Schölkopf et al. "Support Vector Method for Novelty Detection". In: vol. 12. Jan. 1999, pp. 582–588.

[14] *Single euro payments area (SEPA)*. Mar. 2021. URL: https://ec.europa.eu/info/business-economy-euro/banking-and-finance/consumer-finance-and-payments/payment-services/single-euro-payments-area-sepa_en#:~:text=SEPA.

[15] V.V Vlasselaer et al. "Apate: A novel approach for automated credit card transactions fraud detection using network-based extensions". In: *Decision support systems*. 2015.

[16] Bo Zong et al. "Deep Autoencoding Gaussian Mixture Model for Unsupervised Anomaly Detection." In: *ICLR (Poster)*. OpenReview.net, 2018. URL: http://dblp.uni-trier.de/db/conf/iclr/iclr2018.html#ZongSMCLCC18.