

UFR de mathématique et d'informatique

Université de Strasbourg

MASTER MATHÉMATIQUES ET APPLICATIONS  
PARCOURS CALCUL SCIENTIFIQUE ET MATHÉMATIQUES DE L'INNOVATION

Mémoire de stage présenté par

José René PORTILLO

[rene.portillo@etu.unistra.fr](mailto:rene.portillo@etu.unistra.fr)

# MÉTHODES DE DÉTECTION DE FRAUDE POUR LES ASSURANCES

22 août 2025

Stage encadré par

Moisés RODRIGUEZ

[moises.rodriguez@shift-technology.com](mailto:moises.rodriguez@shift-technology.com)

Au sein de

SHIFT TECHNOLOGY

The logo for SHIFT TECHNOLOGY, featuring the word "SHIFT" in a bold, blue, sans-serif font. The letters are slightly shadowed, giving it a 3D appearance.



# Table des matières

<b>Table des matières</b>	<b>3</b>
<b>1 Introduction</b>	<b>5</b>
<b>2 Contexte</b>	<b>7</b>
2.1 L'Assurance . . . . .	7
2.2 La fraude à l'assurance . . . . .	8
2.3 Présentation de l'organisme d'accueil . . . . .	9
2.4 Force - La solution de détection de fraude . . . . .	10
2.4.1 Le logiciel . . . . .	10
2.4.2 Mapping . . . . .	11
2.4.3 Denoising et Reconstruction . . . . .	12
2.4.4 Scénarios et détection de fraude . . . . .	13
2.4.5 CI/CD et déploiement des scénarios et features . . . . .	14
2.4.6 Présentation des résultats au client . . . . .	15
2.5 Méthodes, outils et logiciels . . . . .	15
2.6 Les missions . . . . .	16
<b>3 Les missions</b>	<b>17</b>
3.1 Amélioration et création de scénarios . . . . .	17
3.2 Scénario de responsabilité civile . . . . .	18
3.3 Scénario de véhicules provenant de l'étranger . . . . .	18
3.4 Machine Learning . . . . .	19
3.4.1 Description des données . . . . .	20
3.4.2 Mise en production du modèle . . . . .	24
3.4.3 Développement des cas démos . . . . .	27
3.5 Recul d'expérience . . . . .	28
<b>4 Bibliographie</b>	<b>31</b>



# Chapitre 1

## Introduction

Ce rapport a pour objectif de décrire le travail effectué durant mon stage de fin de master effectué entre le 3 février et le 1er août 2025. Durant ce stage, j'ai été encadré par monsieur Moisés Rodriguez, Senior Data Scientist chez Shift Technology.

Shift Technology est une entreprise française fondée en 2014 par Jérémy Jawish, David Durlleman et Eric Sibony. L'objectif principal de l'entreprise est d'aider les fournisseurs d'assurances à détecter les sinistres frauduleux. En effet, Shift fournit une solution logicielle qui reçoit les données des assurances et des sinistres, et après un traitement algorithmique envoie une probabilité de fraude, avec une analyse quantitative et une explication de la suspicion. Ainsi, ils offrent d'autres services comme l'automatisation entière du traitement du sinistre, qui va de la réception jusqu'au paiement ; et la détection de documents frauduleux ou générées avec de l'intelligence artificielle.

Mon stage s'est déroulé au sein de l'équipe de Data Science, dans le pôle qui gère les clients issues de l'Espagne, l'Amérique Latine et tous les sinistres liées aux voyages. Cette expérience m'a permis de développer nombreuses compétences techniques et avoir un impact réel et direct avec le client.

Durant ces six mois de stage, j'ai eu comme objectif de travailler dans trois tâches différentes. Premièrement, la calibration de scénarios, c'est-à-dire les algorithmes de détection de fraude, déjà mis en production dans l'idée d'améliorer leur rendement et entretenir les relations avec le client pour leur présenter le travail effectué de façon hebdomadaire. Secondement, le développement de cas de démos pour l'équipe de Go To Market, avec l'objectif de montrer les capacités du logiciel. Troisièmement, la mise en production d'un modèle de machine learning pour la détection de fraude dans des sinistres avec des dégâts matériels et corporels.



# Chapitre 2

## Contexte

Par la suite, c'est pertinent de donner un peu de contexte du monde des assurances, du produit logiciel développé par Shift Technology et comment celui-ci fonctionne, et finalement donner les outils techniques ou logiciels qui ont été nécessaires au cours du stage.[7]

### 2.1 L'Assurance

L'assurance est un mécanisme de gestion des risques qui permet aux particuliers ou aux entreprises de se protéger financièrement contre des événements imprévus, appelés **sinistres**. Ces événements, qu'il s'agisse d'accidents, de dommages matériels, de maladies ou de décès, peuvent engendrer des pertes économiques significatives. L'assurance repose sur un contrat, la **police d'assurance**, qui établit les termes de la couverture, les obligations de l'assuré et de l'assureur, ainsi que les **primes** à payer, généralement mensuelles ou annuelles. Ces primes constituent le coût du service et sont calculées en fonction du risque évalué, de la probabilité d'occurrence d'un sinistre et de l'ampleur potentielle des dommages.

#### Principes fondamentaux de l'assurance

Le principe de l'assurance repose sur la mutualisation des risques : les primes versées par un grand nombre d'assurés financent un fonds commun permettant d'indemniser ceux qui subissent un sinistre. Ce mécanisme nécessite une évaluation actuarielle précise, réalisée par des experts en statistique et en probabilité, pour garantir l'équilibre financier de l'assureur. La police d'assurance détaille les **garanties** (les types de sinistres couverts), les **exclusions** (les cas où l'assurance ne s'applique pas), les **franchises** (montant restant à la charge de l'assuré) et les **limites de couverture** (plafond d'indemnisation). Par exemple, une assurance automobile peut inclure une garantie pour les collisions, mais exclure les dommages causés par une conduite en état d'ivresse.

#### Catégories principales d'assurance

Les produits d'assurance se divisent en deux grandes catégories : les assurances de biens et de responsabilité (**Property & Casualty, P&C**) et les assurances de personnes (**Health & Life, H&L**).

## Assurances Property & Casualty (P&C)

Les assurances P&C protègent les biens matériels et couvrent la responsabilité civile. Les biens assurés incluent les véhicules, les habitations, les locaux professionnels ou encore les équipements industriels. En cas de sinistre, comme un incendie, un vol ou un accident de voiture, l'assureur peut indemniser l'assuré par une compensation financière ou, dans certains cas, par le remplacement du bien endommagé.[2] Par exemple, pour une voiture accidentée, l'assurance peut couvrir les réparations ou rembourser la valeur du véhicule selon les termes du contrat.

La composante *Casualty* concerne la responsabilité civile, qui intervient lorsque l'assuré est tenu responsable de dommages causés à des tiers. Par exemple, dans un accident de voiture où l'assuré est fautif, l'assurance peut prendre en charge les frais médicaux des blessés ou les réparations des biens endommagés appartenant à autrui. Ce type de couverture est souvent obligatoire, comme la responsabilité civile automobile dans de nombreux pays.

## Assurances Health & Life (H&L)

Les assurances H&L protègent la santé et la vie des individus. Elles incluent les assurances santé, qui couvrent les frais médicaux (consultations, hospitalisations, traitements), et les assurances vie, qui prévoient des indemnisations en cas de décès ou d'incapacité permanente.[5] Par exemple, une assurance santé peut rembourser les coûts d'une opération chirurgicale, tandis qu'une assurance vie peut verser un capital aux bénéficiaires en cas de décès de l'assuré. Ces contrats précisent les maladies couvertes, les exclusions (par exemple, les maladies pré-existantes non déclarées) et les délais de carence (période pendant laquelle la couverture n'est pas encore active).

## Gestion des sinistres

Lorsqu'un sinistre survient, l'assuré doit effectuer une **déclaration de sinistre** (*insurance claim*) auprès de son assureur, généralement dans un délai stipulé par le contrat. Cette déclaration inclut des informations détaillées sur l'événement, telles que la date, les circonstances, et, si nécessaire, des preuves comme des photos ou des rapports de police.

La déclaration est ensuite transmise à un **expert en sinistres**, qui analyse le dossier pour vérifier si le sinistre est couvert par la police d'assurance. Cette évaluation implique une enquête (ex. : inspection des dommages sur un véhicule) et une estimation des coûts d'indemnisation. L'expert peut également détecter des tentatives de fraude, par exemple des déclarations exagérées ou des sinistres intentionnels. Une fois l'évaluation terminée, l'assureur décide du montant de l'indemnisation, déduit la franchise, et procède au paiement ou à la prise en charge des réparations.

## 2.2 La fraude à l'assurance

La fraude à l'assurance est l'acte de mentir ou décevoir le fournisseur d'assurance soit dans la déclaration du sinistre, soit dans la conception du contrat d'assurance. Ainsi, celle-ci arrive, quand l'assuré reçoit des compensations que normalement ne seraient pas octroyées.

Deux types de fraudes peuvent être décrits :

- **Fraude dure** C'est le fait d'inventer un sinistre ou des dégâts non-existants. De même c'est d'effectuer un sinistre de façon réfléchi et préméditée. Par exemple, en créant une



- collision de voiture, en mettant en scène un vole, ou en produisant un incendie.
- **Fraude simple** C'est le fait d'exagérer les dégâts ou les blessures d'un sinistre qui a vraiment eu lieu, mais ne se sont pas produits comme l'assuré le dépeint. Il se produit aussi quand l'assuré profite du sinistre pour faire passer comme dégâts des problèmes qui précédaient le sinistre ou parfois même le contrat d'assurance.

La fraude à l'assurance représente un défi économique majeur pour les assureurs et les consommateurs en France. Selon une étude de 2015 menée par l'Agence pour la Lutte contre la Fraude à l'Assurance (ALFA), le coût des fraudes s'élevait à environ 2,5 milliards d'euros, soit 5 % des primes d'assurance collectées [1]. Ce phénomène impacte non seulement les compagnies d'assurance, mais aussi les assurés, car les coûts liés à la fraude se répercutent sur les primes, augmentant ainsi les tarifs pour l'ensemble des clients. En plus, la gestion des sinistres est traitée plus lentement à cause des fraudeurs. Les assureurs investissent donc dans des technologies avancées, telles que l'intelligence artificielle, pour détecter et limiter ces pratiques frauduleuses.

## 2.3 Présentation de l'organisme d'accueil

Shift Technology est une licorne française, c'est-à-dire une startup estimée à plus d'un milliard d'euros. Elle compte avec plus de 500 salariés maintenant. En effet, basée dans le 12ème arrondissement de Paris, Shift Technology comporte des bureaux dans d'autres pays comme l'Espagne, le Royaume-Uni, les États-Unis, le Mexique, le Japon et le Singapour. [6] Les clients sont basés dans 25 pays différents, et comportent les grands groupes de l'assurance tels que AXA, Macif ou MetLife.



FIGURE 2.1 – Salariés de Shift Technology - Rencontre 2022

J'ai rejoint Shift Technology en tant que data scientist, un rôle central dans la conception des produits de l'entreprise. Chez Shift, les data scientists — appelés « DS » — sont organisés par régions géographiques. Du fait que les data scientists interagissent directement avec les clients ainsi qu'avec les experts en détection de fraude. À ce titre, la maîtrise native de la langue de la région concernée — l'espagnol, dans ce cas — est essentielle pour assurer une communication

fluide et efficace.

La direction globale des DS est assurée par M. Arnaud Grapinet. La région Europe-Amériques est quant à elle supervisée par M. Maxime Paul, tandis que les DS dédiés à l'Amérique latine et à l'Espagne travaillent sous la direction de Mme Sabrina Maldonado. Par la suite M.Néstor Gascue est le responsable de la partie Amérique Latine, et il surveille une équipe **PnC** et une **Health**. Durant ce stage j'étais sous l'encadrement de M. Moisés Rodriguez qui fait partie de cette première équipe d'Amérique latine PnC.

Néanmoins, il y d'autres pôles cruciaux au développement de l'entreprise et ses produits.

**-Recherche et Developpement** Il s'agit d'un groupe de data scientist, mais qui sont "indépendants", ils ne gèrent pas de clients. Et essayent de développer ou trouver des nouvelles fonctionnalités pour améliorer le produit ou la productivité des autres DS.

**- Les développeurs** Gèrent le front-end de la solution, la conception algorithmique et définissent les "standards" à être appliquées par les autres équipes.

**- Infra**L'équipe d'infrastructure veille à la stabilité, à la scalabilité et à la sécurité des systèmes informatiques qui soutiennent les produits de l'entreprise.

**-QA** Vérifient que la solution en production ne contient aucune erreur, gèrent la conception et l'implémentation des tests unitaires.

**-Delivery** L'équipe de livraison, composé des *Project Managers* accompagnent les clients dans l'implémentation des solutions, en assurant une intégration fluide et adaptée à leurs besoins.

**-Customer Success Management / Ventes** Ce pôle est chargé de maintenir une relation étroite avec les clients, en les accompagnant tout au long de leur expérience avec les produits Shift, et de développer de nouvelles opportunités commerciales.

L'interaction entre les équipes est très commune et le travail synergique est nécessaire, surtout quand un client vient d'être signé.

## 2.4 Force - La solution de détection de fraude

### 2.4.1 Le logiciel

Le produit principal chez Shift est **Force**, un logiciel qui aide les experts en sinistres à détecter la fraude. Le client s'identifie et des alertes sont envoyés pour les sinistres, clients ou intermédiaires suspects.

La solution Force s'adapte aux besoins précis du client, aux données et son format, les types de fraudes et les éléments que le client veut privilégier pour alerter. Malgré que plusieurs éléments de codes et la méthodologie soient les mêmes, la solution n'est pas en "prêt à porter", elle est taillée et fait en fonction du client.

Il existe deux versions de **Force** qu'on appelle **V1** et **V2**. Tous les nouveaux clients sont développés en V2, et il existe un effort constant de faire une migration de tous les clients de V1 vers V2. De plus, aucune nouvelle "feature" est développée en V1. Les différences entre les deux sont graphiques puisqu'elles n'ont même pas la même UI, d'architecture et d'implémentation, ainsi la logique de code entre les deux versions et les fonctions qui lui sont propres sont assez distinctes.

Dans les suivantes lignes nous allons détailler les étapes de la conception de Force pour chaque client en détail, et le process depuis l'envoi des données du client jusqu'à la détection des actes frauduleux. Ces étapes s'appliquent à tous les clients, et normalement sont travaillées par

2-3 data scientist par client.

## 2.4.2 Mapping

Pour comprendre le phénomène de fraude, particulièrement à l'international, il est essentiel d'étudier les spécificités locales : législations, pratiques d'assurance, modalités de remboursement et méthodes des équipes anti-fraude, qui varient selon les pays et les clients. La solution de Shift Technology vise à reconstituer les comportements frauduleux sous-jacents, nécessitant une maîtrise approfondie des aspects pratiques, matériels et légaux.

Une fois les spécificités du client comprises on peut commencer avec le **mapping**

Le *mapping* s'agit de transformer les données brutes fournies par le client en un modèle prédéfini par Shift. Comprendre en amont les données du client est essentiel : leur provenance, leur organisation, leur nature (par exemple, si elles sont fournies directement par les assurés), etc.

Les données sont généralement transférées via SFTP, d'un serveur du client vers un serveur de Shift. Le SFTP, ou *Secure File Transfer Protocol*, est un protocole sécurisé permettant le transfert de fichiers entre systèmes à travers une connexion cryptée. SFTP utilise une couche de sécurité SSH (Secure Shell) pour garantir la confidentialité et l'intégrité des données échangées, ce qui est essentiel pour protéger les informations sensibles des clients.

Elles sont extraites du serveur Shift par un programme nommé le **JobScheduler** et alimentent ensuite une base de données appelée **Raw Model**. L'objectif est alors de transformer ce *Raw Model* en un **Data Model** structuré et exploitable.

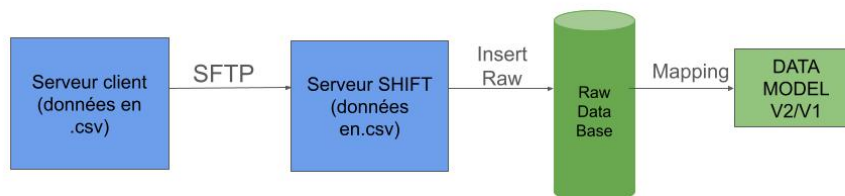


FIGURE 2.2 – Transfert des données et mapping.

Le **Data Model** repose sur une organisation spécifique, structurée en trois types de tables :

- **HUBS** : les hubs contiennent des informations clés sur des entités précises, telles que la déclaration d'un sinistre, une police d'assurance, un véhicule, ou un assuré. Dans le data model, un hub correspond à une table SQL qui regroupe le strict minimum d'informations nécessaires sur une entité. Par exemple, dans la table *Policy*, on trouvera uniquement des éléments comme *PolicyNumber* et *PolicyOwner*.
- **SPECIALIZATIONS** : ces tables contiennent les détails complémentaires d'un hub. Par exemple, dans une table de spécialisation liée à un assuré, on retrouvera le nom complet, la date de naissance, l'adresse, etc.

- **LINKS** : un link représente la relation entre deux hubs. La table de lien contient les clés primaires des deux hubs qu'elle relie, ce qui permet d'effectuer des JOINS entre eux.

### 2.4.3 Denoising et Reconstruction

Le denoising nous aide à trouver détecter et corriger des anomalies dans les données. En effet, un des enjeux est de savoir si deux individus avec des données légèrement différentes sont les mêmes ou pas. Les discrepancies peuvent venir de toutes les formes possibles, une faute d'orthographe, oubli d'une lettre muette, répétition d'un mot, etc. Ainsi, imposer que deux individus sont les mêmes que par l'égalité stricte de tous les champs est une règle beaucoup trop stricte et serait un obstacle à la reconstruction d'identités. La solution à ce problème consiste à nettoyer les données en effectuant des modifications, comme : enlever les espaces, passer à un texte phonétique, supprimer les caractères spéciaux.

Une fois les données nettoyées, les entités — telles qu'une personne, un véhicule ou une organisation — sont reconstruites. Ce processus permet de constituer un historique des entités associées aux sinistres et d'établir des liens entre elles.

Un aspect fondamental de la solution réside dans la capacité à identifier et à visualiser les relations entre les entités de la base de données. Cela permet de détecter des connexions sociales, financières ou autres. Par exemple, des assurés apparaissant à plusieurs reprises comme parties adverses dans différents sinistres peuvent indiquer des schémas frauduleux, facilitant ainsi leur détection et leur analyse par les équipes anti-fraude.

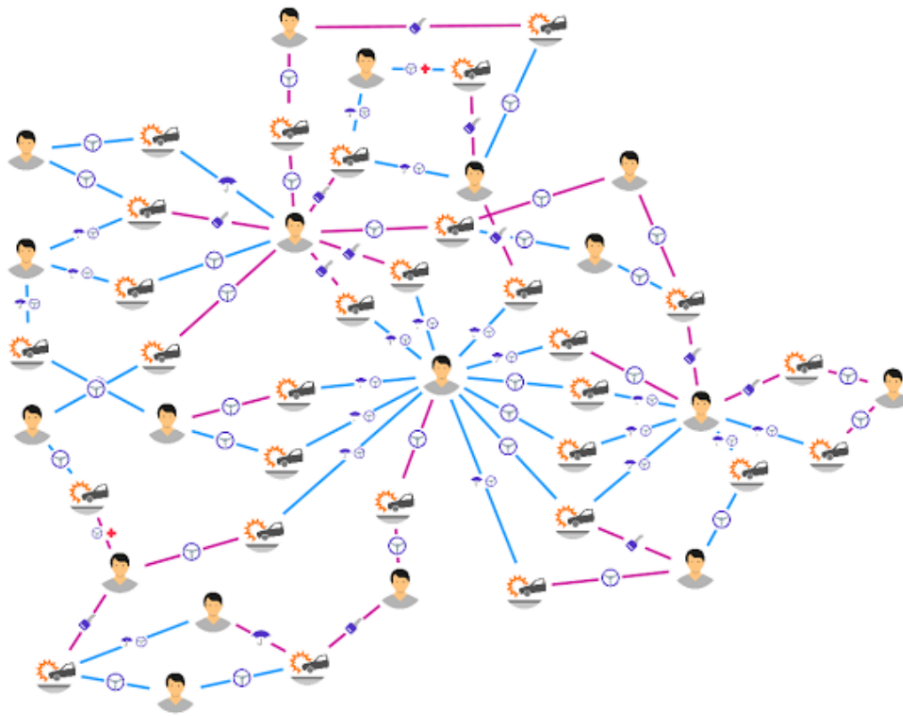


FIGURE 2.3 – Réseaux frauduleux présenté au client

#### 2.4.4 Scénarios et détection de fraude

Par la suite les données seront traitées par des algorithmes spécifiques qu'on appelle des **scénarios**. Un scénario de fraude correspond à un comportement frauduleux précis et défini avec le client en amont et avec son équipe de gestionnaires de fraude.

Les scénarios analysent les données ou une partie des données et lancent des alertes sous certaines contraintes. On retrouve globalement deux types de scénarios

- **Rule Based Scenario** : Il s'agit de scénarios qui déclenchent une alerte si, et seulement si, certaines conditions sont remplies. Les conditions à remplir sont nommées des **variables primaires** et sont souvent des conditions booléennes. Par la suite, on a des **variables secondaires ou explicatives** qui aident les gestionnaires de fraude à mieux comprendre la suspicion. On retrouve des scénarios standards valables pour tous les clients et toutes les régions géographiques, mais qui sont paramétrés spécifiquement pour chaque client.

L'exemple le plus parlant est celui du scénario de souscription récente : une alerte est déclenchée lorsqu'un assuré déclare un sinistre quelques jours seulement après la souscription du contrat d'assurance. Ce type de situation peut faire naître un soupçon quant à la possibilité que le sinistre soit survenu avant l'entrée en vigueur de la police. Cependant, les paramètres — comme le seuil de jours à partir duquel le scénario est déclenché — varient d'un client à l'autre. Tous ces détails sont éclaircis en amont de la mise en production.

Un scénario spécifique, lié à la géographie et au client est la détection de véhicules au Mexique provenant des États-Unis, acquis via des sites d'enchères, et sur lesquels on observe sur les photos, des dommages antérieurs. Ce genre de scénarios peut être développé en aval de la mise en production, en fonction des spécificités du client.

Les scénarios basés sur règles sont extrêmement importants, puisqu'en fonction des règles on a un soupçon très clair et précis qui peut aider fortement les investigateurs de sinistres, et donner un chemin pour l'investigation.

- **Machine Learning Scenarios** Il s'agit de scénarios entièrement basés sur des modèles de machine learning, en particulier sur des algorithmes d'apprentissage supervisé. Ces modèles sont entraînés à partir des données historiques du client, incluant notamment les sinistres identifiés comme frauduleux. Une fois le modèle entraîné, un score de probabilité de fraude est attribué à chaque nouveau sinistre. Un seuil de déclenchement est ensuite défini : si le score dépasse ce seuil, une alerte est générée.

Le développement de ce type de scénario présente plusieurs défis. Tout d'abord, le nombre de cas de fraude confirmés est généralement très faible par rapport à l'ensemble des sinistres, ce qui crée un fort déséquilibre de classes. Ce déséquilibre rend l'entraînement des modèles plus complexe et nécessite souvent le recours à des techniques spécifiques telles que le sur-échantillonnage, le sous-échantillonnage ou l'utilisation de métriques adaptées (AUC, F1-score, etc.) pour évaluer les performances.

Ensuite, contrairement aux scénarios basés sur des règles, ceux issus du machine learning ne reposent pas sur des signaux clairs et compréhensibles pour l'utilisateur final. Le client se retrouve souvent sans hypothèse initiale ou élément tangible pour orienter son enquête. Malgré tout, ces scénarios permettent de détecter des fraudes qui auraient passé inaperçus pour l'œil humain.

TABLE 2.1 – Scénarios de détection de fraude et variables associées

Scénario	Variables principales	Variables secondaires
Court délai entre la souscription au contrat d'assurance et la date du sinistre	Nombre de jours entre la souscription et le sinistre $\leq 30$	<ul style="list-style-type: none"> <li>— Nombre de jours entre la souscription et le sinistre</li> <li>— Montant du sinistre</li> </ul>
Même dommages déclarés deux fois	<ul style="list-style-type: none"> <li>— Moins de trois mois entre les dates des deux sinistres</li> <li>— Au moins deux parties du véhicule endommagées similaires</li> <li>— Premier sinistre payé directement sur le site</li> </ul>	<ul style="list-style-type: none"> <li>— Score de similarité des garanties souscrites</li> <li>— Présence de tiers ou témoins</li> <li>— Nombre de jours entre les deux sinistres</li> <li>— Nombre de parties endommagées similaires</li> </ul>
L'assuré déclare un vol plutôt que les dommages réels pour lesquels il n'est pas couvert	<ul style="list-style-type: none"> <li>— Le sinistre est un vol</li> <li>— L'assuré est couvert pour le vol mais pas pour les dommages matériels</li> </ul>	<ul style="list-style-type: none"> <li>— Présence de tiers ou témoins</li> <li>— Nombre de garanties de la police d'assurance</li> <li>— Valeur du véhicule</li> </ul>

### 2.4.5 CI/CD et déploiement des scénarios et features

Le processus de mise en production des différents scénarios et fonctionnalités suit un pipeline CI/CD structuré et automatisé. Une fois qu'une *Pull Request* (PR) est approuvée, des tests sont exécutés via GitHub. Ces tests vérifient notamment que les traductions ne sont pas cassées, qu'aucune exception non traitée n'est introduite et que les modifications respectent les standards du code.

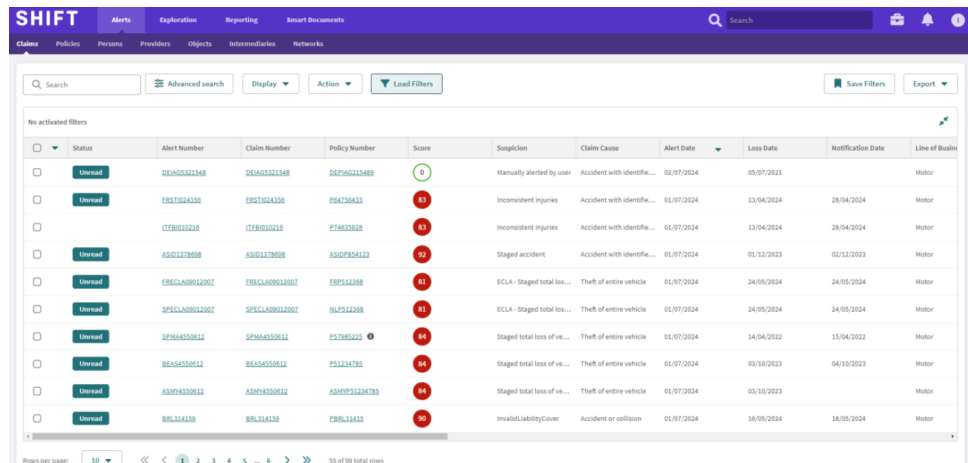
Lorsque les changements sont intégrés à la branche `develop`, utilisée par tous les développeurs, un **build** est lancé via TeamCity. Le build correspond à la compilation du code source et à la génération des artefacts nécessaires pour le déploiement, en s'assurant que le code est exécutable et que toutes les dépendances sont correctement résolues.

Le déploiement est ensuite orchestré par Octopus Deploy. Plusieurs produits de Shift sont concernés, tels que *Force* ou le *JobScheduler*. Chaque client (appelé *tenant*) dispose de plusieurs environnements dédiés : *staging*, *acceptation*, *préproduction* et *production*. Les trois premiers environnements disposent chacun d'un serveur de base de données associé, tandis que l'environnement de production possède son propre serveur.

Ainsi, une fois une fonctionnalité terminée, le déploiement est d'abord effectué en *préproduction*. Seuls si tous les tests et vérifications sont concluants, le déploiement est effectué de *préproduction* vers *production*, garantissant ainsi un processus sécurisé et contrôlé pour chaque tenant.

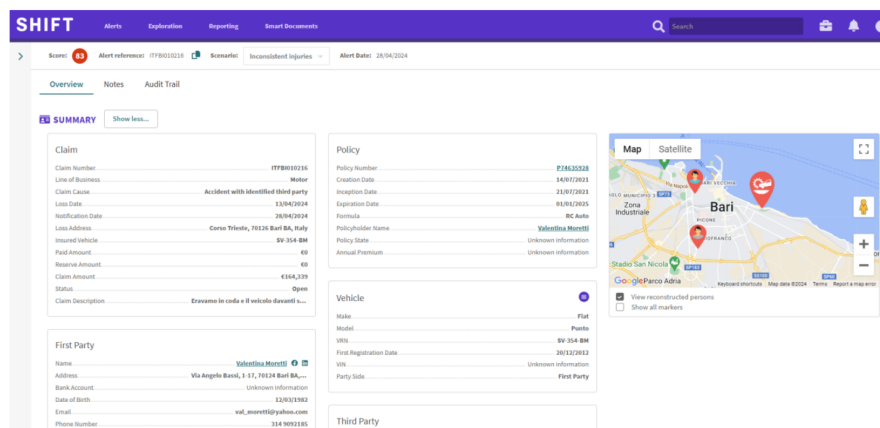
## 2.4.6 Présentation des résultats au client

Une fois tous ces éléments mis en place, on peut commencer à recevoir les sinistres et les données du client de façon journalière. Le traitement et analyse sont effectuées dans les heures qui suivent, et les enquêteurs de sinistres peuvent se connecter à une interface graphique où on retrouve toutes les alertes avec un score spécifique pour faire le tri de façon plus efficace.



Status	Alert Number	Claim Number	Policy Number	Score	Suspect	Claim Cause	Alert Date	Loss Date	Notification Date	Line of Business
Unread	DEAG0321348	DEAG0321348	DEPG0321348	0	Manually alerted by user	Accident with identified third party	02/07/2024	05/07/2021		Motor
Unread	FS17024106	FS17024106	FS17024106	81	Inconsistent injuries	Accident with identified third party	01/07/2024	13/04/2024	28/04/2024	Motor
Unread	ITF8010016	ITF8010016	ITF8010016	81	Inconsistent injuries	Accident with identified third party	01/07/2024	13/04/2024	28/04/2024	Motor
Unread	AS0137806	AS0137806	AS0137806	92	Staged accident	Accident with identified third party	01/07/2024	01/12/2023	02/12/2023	Motor
Unread	FRECLAS0012007	FRECLAS0012007	FRECLAS0012007	81	ECLA - Staged total loss	Theft of entire vehicle	01/07/2024	24/05/2024	24/05/2024	Motor
Unread	SPCLAS0012007	SPCLAS0012007	SPCLAS0012007	81	ECLA - Staged total loss	Theft of entire vehicle	01/07/2024	24/05/2024	24/05/2024	Motor
Unread	SPH4450012	SPH4450012	SPH4450012	84	Staged total loss of vehicle	Theft of entire vehicle	01/07/2024	14/04/2022	15/04/2022	Motor
Unread	BEA4450012	BEA4450012	BEA4450012	84	Staged total loss of vehicle	Theft of entire vehicle	01/07/2024	03/10/2023	04/10/2023	Motor
Unread	AS01450012	AS01450012	AS01450012	84	Staged total loss of vehicle	Theft of entire vehicle	01/07/2024	03/10/2023	03/10/2023	Motor
Unread	BRL314109	BRL314109	BRL314109	80	Invalid Liability Cover	Accident or collision	01/07/2024	18/05/2024	18/05/2024	Motor

FIGURE 2.4 – Interface Graphique de Force



Score: 81 Alert reference: ITF8010016 Scenario: Inconsistent injuries Alert Date: 28/04/2024

Overview Notes Audit Trail

SUMMARY Show less...

Claim

Claim Number: ITF8010016

Line of Business: Motor

Claim Cause: Accident with identified third party

Loss Date: 13/04/2024

Notification Date: 28/04/2024

Line Address: Corso Trieste, 70126 Bari BA, Italy

Insured Vehicle: 9V 354 BM

Paid Amount: 49

Reserve Amount: 49

Claim Amount: €164,339

Status: Open

Claim Description: Errore in coda e il veicolo davanti a...

First Party

Name: Valentina Moretti

Address: Via Angelo Bardi, 111, 70124 Bari BA, Italy

Bank Account: Unknown information

Date of Birth: 13/03/1982

Email: val\_moretti@yahoo.com

Phone Number: 344 9092185

Policy

Policy Number: P28120528

Creation Date: 14/01/2021

Inception Date: 15/01/2021

Expiration Date: 01/01/2025

Formula: RC Auto

Policyholder Name: Valentina Moretti

Policy State: Unknown information

Annual Premium: Unknown information

Vehicle

Make: Fiat

Model: Punto

VIN: 9V 354 BM

First Registration Date: 26/12/2013

VIN: Unknown information

Party Side: First Party

Third Party

Map Satellite

Map of Bari, Italy showing the location of the accident and the location of the vehicle.

Figure 2: Opening an alert

FIGURE 2.5 – Détail d’une alerte

## 2.5 Méthodes, outils et logiciels

### Outils et technologies utilisés

Au cours de ce stage, j’ai eu l’opportunité de travailler sur des projets variés, nécessitant la mobilisation de plusieurs technologies, langages de programmation et outils professionnels.

- **Langages de programmation** : Le développement back-end du logiciel *FORCE* est majoritairement en **C#**, donc ceci a constitué le langage principal de ce stage. Par ailleurs, l’ensemble des travaux relatifs à l’intelligence artificielle et à l’apprentissage automatique a été mené en **Python**, en s’appuyant sur des bibliothèques standards telles que *NumPy*, *pandas*, *matplotlib* et *LightGBM*.

- **Stockage et traitement des données** : Le stockage des données ainsi qu'un prétraitement partiel ont été effectués à l'aide de **SQL Server**.
- **Environnement de développement (IDE)** : L'IDE principal utilisé était **JetBrains Rider**, offrant une intégration fluide avec les différents composants de l'écosystème .NET, ainsi que des outils de débogage avancés.
- **Contrôle de version** : Le suivi des versions et la gestion collaborative du code ont été assurés via **Git**, avec l'assistance de l'interface graphique **Git Extensions**.
- **Intégration et déploiement continus** : Les outils **TeamCity** et **Octopus Deploy** ont été employés pour automatiser les processus d'intégration continue et de déploiement.

## 2.6 Les missions

Les missions au cours du stage furent

- **Suivi du client avec création, amélioration et calibration de scénarios classiques** : Chez Shift, la plupart des data scientists sont assignés à un ou plusieurs clients. Leur mission principale consiste à assurer le suivi de la solution logicielle, développer des scénarios adaptés et les présenter aux clients. Dans ce cadre, j'ai apporté un appui à mon encadrant de stage sur des projets impliquant trois clients d'Amérique latine. J'ai donc participé à l'amélioration et correction de nombreux scénarios, et au suivi du client.
- **Machine Learning** : J'ai entraîné plusieurs modèles pour deux clients en Espagne. Ce travail comprenait la génération de jeux de données, le prétraitement des variables, la détection de fuites (*leakage*) dans les données, l'entraînement des modèles ainsi que leur mise en production.
- **Développement de cas démos** : J'ai collaboré avec l'équipe *PreSales* dans le développement de quinze cas démos, destinés à illustrer les capacités techniques de la solution auprès de clients potentiels.



# Chapitre 3

## Les missions

### 3.1 Amélioration et création de scénarios

Les scénarios sont au cœur de la solution de Shift. Une fois les données correctement mappées, les scénarios analysent les sinistres et, si certaines conditions sont remplies, une alerte est envoyée au client. On appelle **alert rate** le pourcentage d’alertes envoyées par sinistres analysés. À partir du nombre d’alertes envoyées, on obtient le **taux de transformation**, qui correspond au pourcentage d’alertes s’avérant être des fraudes.

Ainsi, l’objectif est d’envoyer le moins d’alertes possible (réduire les faux positifs), tout en garantissant que les alertes envoyées soient effectivement frauduleuses (améliorer la précision). Pour réduire le nombre d’alertes, il est souvent nécessaire d’effectuer une calibration des scénarios, c’est-à-dire de modifier les conditions d’alertement.

Plusieurs modifications ont été apportées en collaboration avec le client au cours du stage. Voici une liste non exhaustive des ajustements effectués :

- Augmentation de l’alertement pour les véhicules haut de gamme dans plusieurs scénarios.
- Réduction, voire suppression, de l’alertement pour les véhicules appartenant à une même entreprise ou bénéficiant d’un contrat d’assurance spécifique.
- Diminution de l’alertement pour les véhicules ayant passé un contrôle technique récent.
- Élargissement des conditions du scénario lié à une souscription récente.

Une fois les modifications implémentées dans le code, une analyse d’impact est nécessaire afin d’évaluer l’effet de ces changements sur la performance des scénarios. Par exemple : si l’on diminue l’alertement, perd-on des cas de fraude ? Si l’on élargit les conditions pour générer plus d’alertes, observe-t-on une hausse des faux positifs ? Et surtout, dans quelle mesure ces ajustements sont-ils pertinents pour le client ?

L’analyse est effectuée dans les serveurs de préproduction à partir des sinistres enregistrés au cours des six derniers mois. Elle permet de présenter au client les éléments suivants :

- L’évolution du taux d’alertement.
- L’évolution du taux de transformation.
- Le nombre estimé de fraudes supplémentaires détectées ou, au contraire, perdues.

Dans le cas du scénario lié à une souscription récente, les conditions d’alertement ont été élargies. Sur 144 000 sinistres analysés en trois mois, le nombre d’alertes est passé de 112 à 146, tandis que les fraudes détectées sont passées de 3 à 6.

Au cours du stage, nous avons réalisé en moyenne une calibration par semaine.

Cependant, modifier les scénarios existants ne suffit pas. En début de trimestre, des actions sont définies pour effectuer des modifications majeures ou pour créer de nouveaux scénarios à partir de zéro. Voici deux exemples de scénarios que j'ai développés au cours de ce stage.

Dans les prochaines sections nous allons présenter deux scénarios que nous avons construits de zéro.

### 3.2 Scénario de responsabilité civile

L'assurance de responsabilité civile sert à indemniser un tiers en cas de dommages causés par le véhicule. Il existe également une extension de la responsabilité civile, c'est-à-dire que, si une autre personne que l'assuré conduit le véhicule, elle est également couverte.[3] Cette extension peut aussi protéger l'assuré lorsqu'il conduit un véhicule autre que le sien.

Il arrive parfois qu'un assuré bénéficiant de cette extension déclare avoir conduit un véhicule non assuré au moment du sinistre.

L'objectif est donc de développer un scénario qui, couplé à d'autres variables, permette de détecter des cas suspects d'utilisation de l'extension de responsabilité civile. Parfois, l'information liée à cette extension n'est pas explicitement présente dans les données du contrat d'assurance. Dans ce cas, il est nécessaire de rechercher cette information dans les notes rédigées par la personne ayant pris en charge le sinistre.

Pour cela, une requête `REGEX` est utilisée afin d'extraire toute mention éventuelle d'une extension de responsabilité civile.

Shift dispose d'un framework complet pour la détection de langage naturel. Ce framework est structuré par langue, et permet de définir des blocs ou concepts de base. Dans notre cas, il s'agit d'un concept très simple, qui regroupe toutes les variations possibles de l'expression « extension de responsabilité civile ».

Si on trouve la mention de "l'extension de responsabilité civile", avec d'autres variables définies avec le client comme le modèle du véhicule, la ville du sinistre, et autres, une alerte est envoyée au client.

### 3.3 Scénario de véhicules provenant de l'étranger

Dans le but de détecter des cas où les dommages subis par un véhicule pourraient être antérieurs au sinistre déclaré, j'ai développé un scénario spécifique basé sur l'origine du véhicule et la traçabilité de son historique.

Le scénario commence par vérifier si le véhicule impliqué est d'origine étrangère. Lorsqu'un véhicule est importé, il est possible qu'il ait déjà subi des dégâts avant d'arriver dans le pays d'assurance, sans que ces dommages ne soient explicitement mentionnés dans le dossier du sinistre. Pour cela, le scénario s'appuie sur le code VIN (*Vehicle Identification Number*), un identifiant unique attribué à chaque véhicule. Grâce au VIN, il est possible de déterminer si le véhicule provient des États-Unis.

Une fois cette information confirmée, une recherche est effectuée sur Internet, notamment sur des sites d'enchères automobiles spécialisés (tels que Copart ou IAAI), afin de vérifier si le véhicule a déjà été mis en vente avec des dommages visibles. Ces plateformes publient souvent des photos et des descriptions détaillées des véhicules accidentés ou endommagés.

Si le véhicule est retrouvé sur l'un de ces sites avec des dégâts antérieurs au sinistre déclaré,

une alerte est alors générée. Cette alerte peut être utilisée par le client pour ouvrir une enquête approfondie, notamment pour vérifier si le sinistre a été déclaré de bonne foi ou s'il s'agit d'une tentative de fraude consistant à faire passer des dégâts anciens pour des dommages récents.

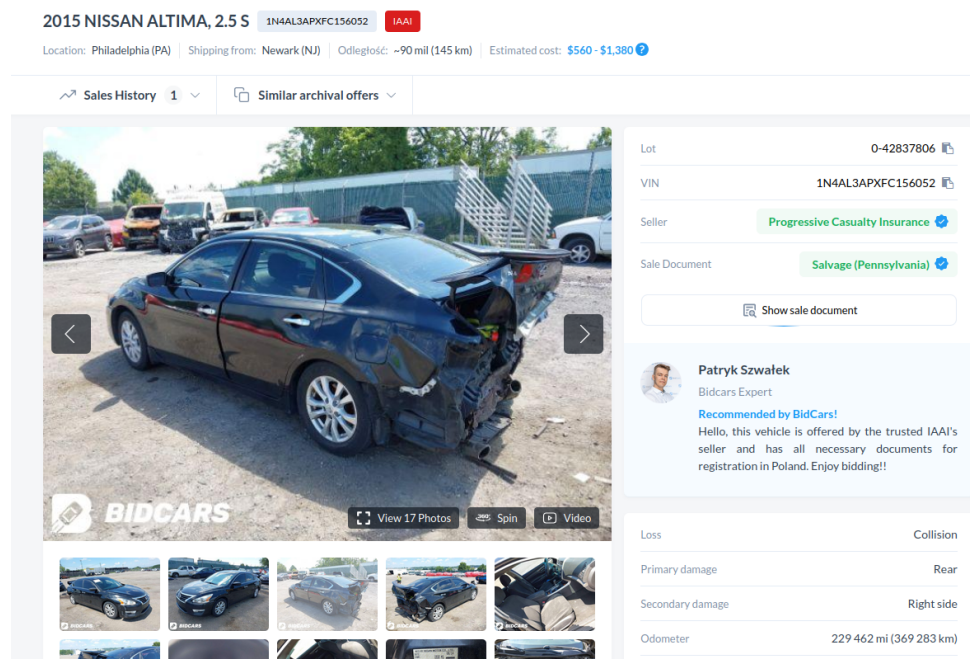


FIGURE 3.1 – Exemple de véhicule trouvé dans un site d'enchères

Dans ce scénario on voit bien l'intérêt de croiser des données internes (contrat, sinistre, VIN) avec des sources externes accessibles publiquement pour enrichir la détection de fraudes. Il s'inscrit dans une logique d'investigation proactive, visant à identifier des incohérences temporelles dans l'historique des véhicules assurés.

### 3.4 Machine Learning

L'objectif de ces scénarios c'est d'avoir des modèles de machine learning capables de détecter les cas frauduleux, mais qui n'ont pas un motif précis comme dans les scénarios à règle. Il s'agit donc d'un problème de **classification binaire**. L'une des contraintes les plus significatives est le faible nombre de cas frauduleux présents dans le dataset, auquel s'ajoute le fait que tous les éléments apparaissant comme "sans fraude" ne correspondent pas nécessairement à des sinistres réellement non frauduleux.

Dans le cadre de ce stage, j'ai tout d'abord travaillé sur le réentraînement d'un modèle de dédommagements corporels déjà en production pour une grande assurance espagnole. Cette première mission m'a permis de comprendre l'architecture de machine learning imposée dans ce type de scénarios ainsi que le processus de mise en production d'un modèle.

Par la suite, j'ai pu participer au développement complet d'un scénario de machine learning, appliqué aussi bien aux dédommagements matériels qu'aux corporels. Cela a impliqué toutes les étapes clés : génération du dataset, nettoyage et prétraitement des données, entraînement du modèle, validation, mise en production et enfin présentation des résultats au client.

## Présentation de LightGBM et contraintes du problème

LightGBM[4] est le framework choisi pour l'entraînement, il s'agit d'un framework open source développé par Microsoft, implémentant l'algorithme de *gradient boosting* basé sur des arbres de décision. Il est conçu pour être particulièrement rapide et efficace sur de grands volumes de données, tout en offrant une bonne capacité de généralisation. Parmi ses points forts, on note sa capacité à gérer directement des variables catégorielles, à traiter des données bruitées et à optimiser l'utilisation de la mémoire.

Dans notre cas, deux contraintes majeures rendent la détection de fraude complexe :

- **Forte déséquilibre des classes** : le nombre de sinistres frauduleux est très faible par rapport au nombre total de sinistres, ce qui entraîne un risque que le modèle privilégie la prédiction de la classe majoritaire.
- **Incertitude des labels "sans fraude"** : certaines observations étiquetées comme non frauduleuses peuvent en réalité être frauduleuses, mais n'ont pas été détectées comme telles au moment de la labellisation.

LightGBM propose plusieurs mécanismes permettant d'atténuer ces problèmes :

- **Pondération des classes** (*class weights*) : permet d'accorder plus d'importance aux exemples minoritaires, réduisant ainsi le biais en faveur de la classe majoritaire.
- **Gestion robuste des données bruitées** : grâce à la structure en arbres de décision et à l'optimisation par *leaf-wise growth*, LightGBM est moins sensible à la présence d'observations mal labellisées.
- **Efficacité sur grands datasets** : sa rapidité d'entraînement permet d'explorer plus de configurations d'hyperparamètres pour optimiser la détection de fraude malgré la rareté des exemples positifs.

## Données et préparation

### 3.4.1 Description des données

Les données utilisées dans le cadre de ce projet sont générées après le *mapping* et non directement à partir des données brutes du client. Le *dataset* contient plusieurs centaines de *features*, mais cela ne signifie pas que toutes seront exploitées. Une étape de **sélection de variables** est réalisée en fonction du modèle, du client et des données effectivement disponibles.

Les features du dataset incluent notamment :

- Identifiant du sinistre
- Prénom de la personne
- Âge de la personne
- Date du sinistre
- Marque du véhicule
- Temps écoulé entre le sinistre et sa déclaration
- Temps écoulé entre le sinistre et la souscription à l'assurance
- Nombre de sinistres précédents
- Nombre de blessés
- Nombre de véhicules impliqués
- Indicateur si le sinistre a eu lieu un jour férié
- Indicateur si le véhicule est de luxe
- Parties de la voiture endommagées
- Temps écoulé entre le sinistre et la déclaration des blessures corporelles
- Type de couverture de l'assuré

La variable cible (*label*) indique si le sinistre correspond ou non à un cas de **fraude**.

## Nettoyage et préparation des données

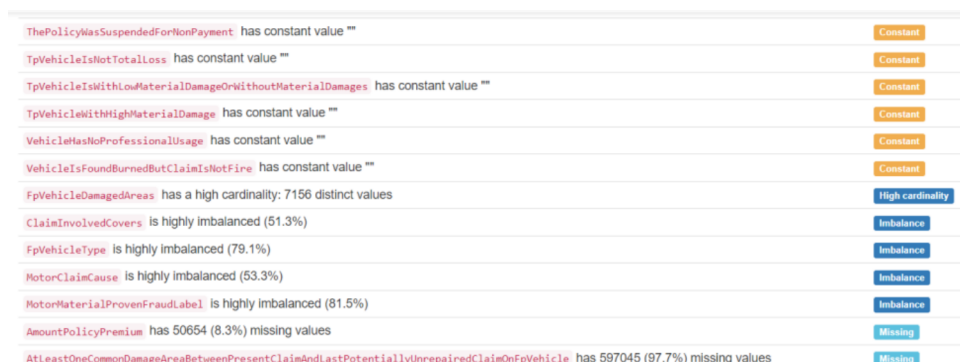
Nous disposons de deux *datasets*, *bodily\_injuries\_fraud* et *material\_damages\_fraud*, qui sont chacun divisés en deux sous-ensembles :

- Les données dites « historiques », correspondant à l'ensemble des sinistres enregistrés en 2024.
- Les données dites « récentes », correspondant aux sinistres des trois derniers mois, à savoir mars, avril et mai 2025.

Cette distinction est nécessaire car les données évoluent au fil du temps, et toutes les informations ne sont pas toujours disponibles dès le départ. Ainsi, le jeu de données « historique » est utilisé pour l'entraînement et le test du modèle, tandis que le jeu « récent » sert à la phase de validation. Bien que les deux ensembles contiennent des *features* globalement similaires, certaines variables peuvent présenter des différences, ce qui nécessite une harmonisation. Avant toute modélisation, une étape de **nettoyage et préparation des données** est effectuée afin d'assurer leur cohérence et leur qualité. Cette étape comprend :

- **Filtrage des sinistres dupliqués** : suppression des entrées présentant le même identifiant ou des informations identiques sur plusieurs variables clés.
- **Uniformisation des types de données** : vérification que les champs numériques, textuels et de dates sont correctement typés dans l'ensemble des fichiers.
- **Harmonisation des formats** : par exemple, s'assurer que les dates sont dans un format unique et que les noms de marques ou types de couverture ne présentent pas de variations d'orthographe.
- **Vérification des valeurs aberrantes** : détection et traitement des âges, durées ou montants manifestement incohérents.
- **Gestion des valeurs manquantes** : identification des champs incomplets et choix d'une stratégie adaptée (imputation, suppression, etc.).

On effectue aussi une comparaison entre les datasets historique et récents pour voir quelles variables sont présentes dans les données historique et pas dans les données récentes. Ainsi, cette comparaison, la vérification de valeurs aberrantes et gestion de valeurs manquantes peut être fait avec pandas et leur outil de profiling. Le package de profiling nous permet d'avoir une vue d'ensemble sur les données, et avoir des éléments quantitatifs précis pour chaque feature. On peut ainsi savoir combien d'éléments manquants nous avons, les différents écarts interquartiles si cela s'agit de variables quantitatives, combien de catégories avons nous s'il s'agit d'une variable catégorique et leurs effectifs correspondant.



ThePolicyWasSuspendedForNonPayment	has constant value ""	Constant
TpVehicleIsNotTotalLoss	has constant value ""	Constant
TpVehicleIsWithLowMaterialDamageOrWithoutMaterialDamages	has constant value ""	Constant
TpVehicleWithHighMaterialDamage	has constant value ""	Constant
VehicleHasNoProfessionalUsage	has constant value ""	Constant
VehicleIsFoundBurnedButClaimIsNotFire	has constant value ""	Constant
FpVehicleDamagedAreas	has a high cardinality: 7156 distinct values	High cardinality
ClaimInvolvedCovers	is highly imbalanced (51.3%)	Imbalance
FpVehicleType	is highly imbalanced (79.1%)	Imbalance
MotorClaimCause	is highly imbalanced (53.3%)	Imbalance
MotorMaterialProvenFraudLabel	is highly imbalanced (81.5%)	Imbalance
AmountPolicyPremium	has 50654 (8.3%) missing values	Missing
AtLeastOneCommonDamageAreaBetweenPresentClaimAndLastPotentiallyUnrepairedClaimOnFpVehicle	has 597045 (97.7%) missing values	Missing

FIGURE 3.2 – Liste de variables dans pandas profiling

Ce processus nous permet de comparer le data set à d'autres data set où on sait très bien que le modèle marche, ce qui pourrait nous permettre d'utiliser les mêmes paramètres lors de l'entraînement. On compare donc toujours deux entreprises qui ont à peu-près le même nombre

### FpVehicleType

Categorical

Distinct	9
Distinct (%)	< 0.1%
Missing	1615
Missing (%)	0.3%
Memory size	9.3 MiB

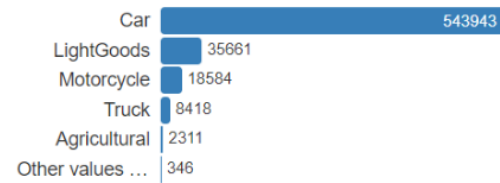


FIGURE 3.3 – Liste de variables dans pandas profiling

### NumberOfPreviousInvestigatedClaimsWithSimilarCircumstances

Real number (R)

Distinct	7
Distinct (%)	0.1%
Missing	598824
Missing (%)	98.0%
Infinite	0
Infinite (%)	0.0%
Mean	0.9266633483

Minimum	0
Maximum	6
Zeros	1723
Zeros (%)	0.3%
Negative	0
Negative (%)	0.0%
Memory size	9.3 MiB

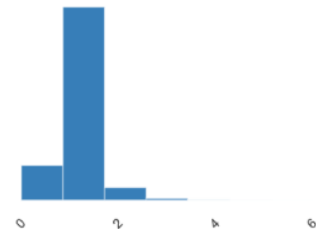


FIGURE 3.4 – Variable avec valeurs manquantes

de sinistres par an et le même taux de fraude.

## Entraînement

Voici les paramètres du modèle LightGBM concernant l'entraînement :

Parameter	Value
metric	F1_score
objective	Binary
verbosity	-1
eta	0.1
feature_fraction	0.9
min_data_in_leaf	10
max_delta_step	1
max_depth	10
lambda_l2	5

TABLE 3.1 – LightGBM paramètres d'entraînement

On peut utiliser aussi un module nommé LightGBMUDAHelper, qui permet de faire une sélection des features les plus importantes. Une optimisation est fait de sorte à utiliser qu'une partie des features. Ce module nous permet aussi de trouver les meilleurs hyperparamètres pour notre modèle.

## Pareto-front Plot

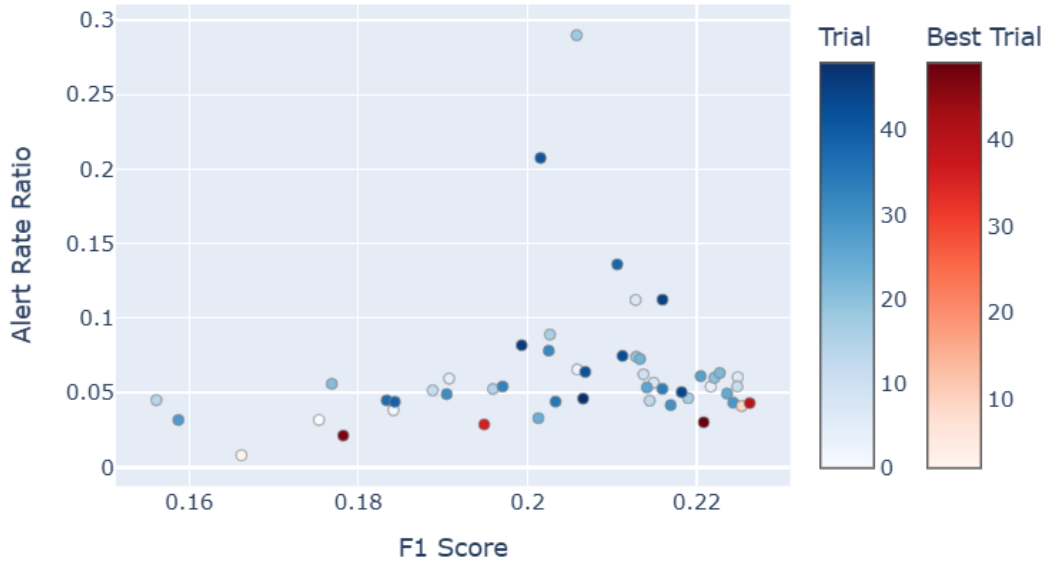


FIGURE 3.5 – Modèles en fonction des paramètres

## Métriques d'évaluation

Pour évaluer la performance du modèle, plusieurs métriques sont utilisées.

La **précision** (*Precision*) mesure la proportion de sinistres détectés comme frauduleux qui le sont réellement :

$$Precision = \frac{TP}{TP + FP}$$

où  $TP$  est le nombre de vrais positifs et  $FP$  le nombre de faux positifs.

Le **rappel** (*Recall*) mesure la proportion de sinistres frauduleux correctement détectés par le modèle :

$$Recall = \frac{TP}{TP + FN}$$

où  $FN$  est le nombre de faux négatifs.

Le **F1-score** combine la précision et le rappel (*Recall*) en un seul score harmonique :

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall},$$

ce qui permet de mieux évaluer la performance sur des classes déséquilibrées, comme c'est le cas pour la fraude.

Enfin, le **Alert Rate Ratio (ARR)** est calculé comme le ratio du taux d'alerte sur le dataset récent par rapport au dataset historique :

$$ARR = \frac{AlertRate_{recent}}{AlertRate_{historique}}.$$

Cette métrique est cruciale pour s’assurer que le modèle maintient une fréquence d’alerte cohérente dans le temps. Un ARR proche de 1 garantit que le modèle n’alerte ni trop ni trop peu sur les nouvelles données par rapport aux données historiques.

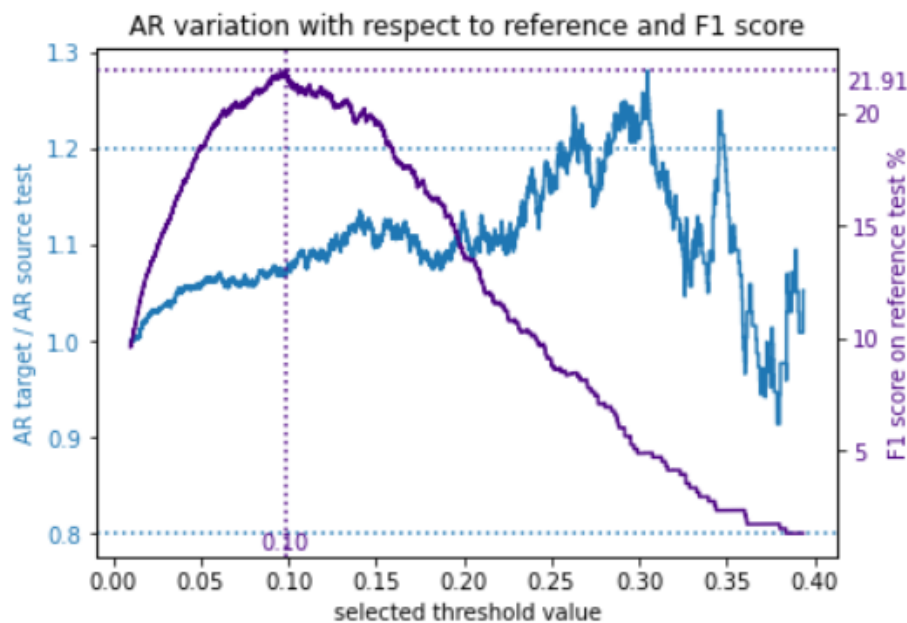


FIGURE 3.6 – Variation de l’alert rate

### Selection du modèle et déploiement

Le modèle est choisi en fonction de son alert rate et sa précision. Par la suite faut choisir un *threshold* en fonction du nombres d’alertes qu’on souhaite recevoir par mois. Dans ce cas on souhaite recevoir 2 alertes par mois pour les modèles de dommages corporels et 24 à 26 alertes par mois pour les modèles de dommages matériels.

#### 3.4.2 Mise en production du modèle

Une fois le modèle entraîné et validé, il est nécessaire de le rendre exploitable en production. Pour cela, plusieurs fichiers sont générés lors de la phase finale :

- des fichiers `JSON`, qui contiennent notamment la description du modèle `LightGBM` (poids, structure des arbres, paramètres retenus) ainsi que la liste des variables utilisées (features) avec leurs types;
- un fichier `YML`, qui sert de configuration et permet d’assurer la cohérence du déploiement (par exemple l’ordre des variables en entrée).

Ces fichiers sont ensuite intégrés dans un pipeline de déploiement. Dans notre cas, le processus repose sur un enchaînement entre **TeamCity** (build) et **Octopus** (déploiement). Le notebook ayant servi à l’entraînement n’est pas déployé : seule la version sérialisée du modèle est transférée vers le serveur interne.

Le serveur interne expose alors une **API REST**, qui rend le modèle accessible à des services applicatifs. Le logiciel en **C#** envoie une requête contenant les variables d’un sinistre au format attendu. Le serveur retourne en réponse une probabilité de fraude, c’est-à-dire une valeur comprise entre 0 et 1.



Performance on recent data

```
eval_benchmark["target"]["Table"][0]
```

	TN	FN	TP	FP	PP	Pr	Rd	F1	Alert Rate
Threshold									
0.01	12117	134	922	25567	26489	3.48	87.31	6.69	68.38
0.02	20931	274	782	16753	17535	4.46	74.05	8.41	45.26
0.03	25404	416	640	12280	12920	4.95	60.61	9.16	33.35
0.04	28324	505	551	9360	9911	5.56	52.18	10.05	25.58
0.05	30285	569	487	7399	7886	6.18	46.12	10.89	20.36
0.10	34577	780	276	3107	3383	8.16	26.14	12.44	8.73
0.15	35920	853	203	1764	1967	10.32	19.22	13.43	5.08
0.20	36481	900	156	1203	1359	11.48	14.77	12.92	3.51
0.25	36820	940	116	864	980	11.84	10.98	11.39	2.53
0.30	37078	960	96	606	702	13.68	9.09	10.92	1.81
0.35	37218	980	76	466	542	14.02	7.20	9.51	1.40
0.40	37345	998	58	339	397	14.61	5.49	7.98	1.02
0.45	37440	1012	44	244	288	15.28	4.17	6.55	0.74
0.50	37495	1017	39	189	228	17.11	3.69	6.07	0.59
0.55	37543	1026	30	141	171	17.54	2.84	4.89	0.44
0.60	37586	1032	24	98	122	19.67	2.27	4.07	0.31
0.65	37611	1038	18	73	91	19.78	1.70	3.14	0.23
0.70	37636	1042	14	48	62	22.58	1.33	2.50	0.16
0.75	37653	1046	10	31	41	24.39	0.95	1.82	0.11
0.80	37660	1047	9	24	33	27.27	0.85	1.65	0.09
0.85	37666	1050	6	18	24	25.00	0.57	1.11	0.06
0.90	37675	1051	5	9	14	35.71	0.47	0.93	0.04

FIGURE 3.7 – Tableau de performance du modèle en fonction du Threshold

Le **threshold** est ensuite appliqué. Si la probabilité reçue est supérieure au seuil défini, une alerte est déclenchée. Ce seuil n'est pas fixe : il dépend du type de modèle et du client. Par exemple, le modèle appliqué aux sinistres matériels peut avoir un seuil différent de celui utilisé pour les sinistres corporels.

Enfin, la **mise à jour des modèles** est un aspect essentiel du processus. Actuellement, elle est réalisée manuellement : lorsque de nouvelles données sont disponibles, il faut réentraîner un modèle, réexporter les fichiers JSON/YML et redéployer le tout. Cette étape est cruciale pour s'assurer que le modèle reste pertinent et ne repose pas sur des variables biaisées ou trop tardives. À titre d'exemple, un des modèles testés utilisait la variable indiquant si le client avait une représentation légale. Or, cette information est souvent disponible uniquement à un stade avancé du dossier, et elle est fortement corrélée à la fraude. Il a donc fallu réentraîner le modèle sans inclure cette variable afin d'améliorer sa robustesse et son applicabilité réelle.

## Résultats et présentation finale

Une fois le modèle choisi, il est d'abord nécessaire de tester le scénario en interne. Une analyse qualitative est réalisée en environnement de staging et de préproduction. Pour le scénario corporel, j'ai pu examiner la totalité des alertes générées afin d'identifier des cas pertinents et représentatifs. Pour le scénario matériel, une partie significative des alertes a également été analysée. Une attention particulière est portée sur la proportion d'alertes déjà couvertes par d'autres scénarios de détection afin d'évaluer la valeur ajoutée du modèle de machine learning.

Il est aussi essentiel de vérifier que le pipeline complet fonctionne correctement, de l'intégration du modèle jusqu'à son interaction avec le service C# qui interroge l'API. Cette étape garantit que la mise en production ne présentera pas de difficultés techniques et que la communication entre les différents composants est cohérente.

Lors de la présentation au client, l'objectif est de démontrer la pertinence du modèle. On met en avant des cas concrets qui ne sont pas détectés par d'autres scénarios et qui auraient normalement échappé au système de détection. De plus, le nombre total d'alertes attendues ainsi que le nombre minimal de fraudes détectées sont présentés. Les seuils (*thresholds*) sont également détaillés avec le volume d'alertes correspondant, en soulignant que leur ajustement est simple et peut être adapté à la demande du client.

Suspicion

Machine Learning - The Machine Learning model indicates that the current claim has a high probability of being fraud.

Export

Indication	Worth
Time between subscription and loss	4 weeks, 1 day
The insured vehicle is responsible	
Age of the insured vehicle	22 years
Driver's age	twenty-one
Number of bodily injuries in the accident	3
Number of bodily injuries in the insured vehicle	1
Number of bodily injuries in the other vehicle	2
Time between the accident and the last declaration of bodily injury of the affected persons of the VA	6 days
Time between the accident and the last declaration of bodily injury of those affected	2 months, 3 weeks
Number of third-party vehicles involved in the accident	1
The insured has not had another policy prior to subscription	
Number of policies of the insured	1
The insured does not have own damage coverage or has a high deductible	
The accident occurred at low speed	

FIGURE 3.8 – Exemple alerte Machine Learning

Enfin, cette présentation constitue la dernière étape avant la mise en production. Une fois le client convaincu de la valeur ajoutée et de la cohérence du modèle, celui-ci est déployé dans l’environnement opérationnel et intégré au processus global de détection des fraudes.

### 3.4.3 Développement des cas démos

Une partie importante de mon stage a consisté à développer une dizaine de *cas démos*, c’est-à-dire des scénarios de fraude simulés servant de vitrine commerciale. Ces cas jouent un rôle stratégique majeur : ils constituent la base des démonstrations réalisées auprès des futurs clients, mais aussi lors de discussions avec des experts métiers ou de présentations internes. Ils doivent donc être construits avec soin, en étant suffisamment **réalistes** pour refléter des pratiques crédibles dans le domaine de l’assurance, mais également légèrement **exagérés**, afin de mettre en avant toute la richesse fonctionnelle de la plateforme.

**Structure technique.** Chaque cas démo est implémenté en C#, selon une logique d’héritage et de **surcharge de paramètres**. La base repose sur un **cas général** de fraude — par exemple, un médecin qui facture des consultations inexistantes ou un patient qui tente de se faire rembourser une maladie préexistante. À partir de ce schéma commun, des déclinaisons spécifiques sont construites pour chaque géographie, afin de refléter les particularités locales tout en mutualisant le code.

Si le développement n’impliquait pas de défis algorithmiques complexes, il nécessitait une solide compréhension du **code existant** et de son organisation. En particulier, j’ai dû : - me familiariser avec le fonctionnement d’**ElasticSearch**, utilisé pour l’indexation et la recherche de sinistres dans l’interface, - comprendre les méthodes et fonctions internes permettant d’afficher correctement les éléments visuels associés à chaque cas, - et intégrer les scénarios dans la logique applicative globale, en respectant les conventions établies.

Ces tâches m’ont également permis d’améliorer ma rigueur dans la rédaction des *pull requests* (PR) et de mieux m’insérer dans un processus collaboratif impliquant plusieurs équipes.

**Processus de validation.** Les cas démos suivaient un cycle de validation structuré : 1. une première relecture par un *senior data scientist*, 2. une présentation par blocs à l’équipe *Pre Sales*, qui pouvait demander des ajustements en fonction des besoins de présentation, 3. une documentation finale sur *Confluence*, détaillant les choix effectués et leurs justifications.

**Dimension métier.** Les besoins à couvrir étaient définis conjointement par les **managers de l’équipe Data Science** et par les équipes **Sales** et **Pre Sales**, en fonction des contextes de démonstration. Cette étape m’a permis d’approfondir ma connaissance de la **terminologie propre au secteur de l’assurance** et de comprendre les différences culturelles et réglementaires dans la définition de la fraude selon les régions.

Au total, j’ai développé **sept à huit cas spécifiques pour l’Espagne** et **deux à trois pour le Mexique**. Tous ont été validés, mis en production et sont désormais utilisés activement dans les démonstrations clients. Ce travail a renforcé ma compréhension du code en C#, consolidé mes compétences en intégration logicielle, et illustré l’importance du lien entre développements techniques et valeur métier perçue par les clients.

### 3.5 Recul d'expérience

Au cours de ce stage, j'ai développé énormément de compétences, et j'ai pu mettre en pratique des éléments que j'avais appris durant le master CSML. Ainsi, les cours de *Traitement et fouilles de données* ou *Méthodes aléatoires* furent très utiles. À pas négliger aussi que des cours comme *Projet* m'ont donné un aperçu du CI/CD et de GitHub, et ce fut d'une grande aide dès le début du stage.

Ainsi, ce stage m'a permis de prendre confiance en moi, surtout au moment de coder que cela soit en C# ou n'importe quel langage orienté objet, je suis certain d'avoir pris beaucoup plus de rigueur, et une capacité à écrire du code lisible, claire et qui peut être facilement scalable.

De même, dans ce qui concerne le machine learning, cette expérience m'a permis d'aller au delà d'un notebook Jupyter et bien comprendre comment un modèle est mis en production, c'est quoi les contraintes et les difficultés, ainsi ce projet, représentant le défi le plus complexe en apprentissage automatique que j'ai relevé, m'a conduit à maîtriser toutes les étapes du processus, du nettoyage des données à la mise en production des modèles, générant ainsi des bénéfices économiques concrets pour le client.

En ce qui concerne la gestion de version et l'intégration continue, je dois dire que je comprends beaucoup plus les outils que cela soit Git pour la gestion ou TeamCity/Octopus pour le déploiement. En plus, d'être bien à l'aise avec un processus collaboratif.

Finalement, j'ai acquis un ensemble de compétences techniques et pratiques qui ont considérablement enrichi mon parcours en tant que data scientist. J'ai développé une compréhension approfondie du secteur de l'assurance et de la détection de fraude.

### Remerciements

Je finis ce rapport en remerciant toutes les personnes qui m'ont aidé tout au long de ce stage et pendant mon séjour à Paris.

Je souhaite tout d'abord remercier mon encadrant de stage, **Moisés Rodriguez**, pour sa patience, sa disponibilité et ses conseils constants au cours de ces six mois. Son accompagnement attentif a été déterminant pour la réussite de ce stage.

Je tiens à remercier la cheffe du département de Data Science pour les pays hispanophones **Sabrina Maldonado** qui m'accueilli et fait sentir dès le premier jour comme si j'avais toujours été membre de son équipe.

**Martin Calderón, Pablo García, Mario Parrón, et Kevin Sánchez** pour toute leur aide dans la partie machine learning, leur bienveillance et leurs explications.

**Marcos Akimoto et Alfredo Gudiño** pour leur appui avec les clients tout au long de ces six derniers mois.

**Néstor Gascue** aussi pour m'avoir fait confiance, pour son leadership, et feedback qui m'a permis de grandir en tant que jeune professionnel.

Un grand remerciement à tout le reste de l'équipe *Jawas* et les différents membres des autres équipes ; pour leur forte hospitalité.

Je tiens également à remercier le directeur du master, **Christophe Prud'homme**, pour son soutien académique et ses conseils.

Merci à **Anthony Gunes et Salma Massis** sans qui mon séjour à Paris aurait été complètement différent. Et **Giulio Carpi Lapi** pour m'avoir accompagné avec sa grande amitié dès le premier

jour de licence.



## Chapitre 4

# Bibliographie

- [1] ALFA. Comment mieux lutter contre la fraude? Technical report, ALFA, 2012. Document institutionnel.
- [2] Allstate. What is property and casualty insurance? <https://www.allstate.com/resources/what-is-property-and-casualty-insurance>, 2025.
- [3] Direction de l'information légale et administrative (Premier ministre). Assurance auto : qu'est-ce que la garantie responsabilité civile? <https://www.service-public.fr/particuliers/vosdroits/F31258>, 2025.
- [4] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm : A highly efficient gradient boosting decision tree. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 3149–3157, Long Beach, CA, USA, 2017. Curran Associates, Inc. Available at : <https://api.semanticscholar.org/CorpusID:3815895>.
- [5] ProAssurance. Different types of insurance – life vs. health vs. property & casualty. <https://proassurance.com/knowledge-center/different-types-of-insurance>, 2024.
- [6] Shift Technology. About shift technology : Inventing the future of insurance with ai. <https://www.shift-technology.com/about>, 2025.
- [7] Wikipédia. Assurance. <https://fr.wikipedia.org/wiki/Assurance>, 2025.