ZROUAMA Kaussaha Emmanuel Régis, YOUSSFI Hanane, REINETTE Mahlî, GOGUY Priscilla

***Compte Rendu : Projet VBA : Outil de détection de fraude***

***I) Objectifs du projet et contextualisation***

La fraude à l’assurance constitue un enjeu majeur pour les compagnies d’assurances. Chaque année, les tentatives frauduleuses se multiplient, portées par des fraudeurs de mieux en mieux organisés et des méthodes toujours plus sophistiquées. Parallèlement, les avancées en science des données, en intelligence artificielle et en automatisation permettent aux assureurs de renforcer leurs capacités de détection et d’analyse. Cette double dynamique :d’une part l’évolution des pratiques frauduleuses, d’autre part les progrès technologiques rend indispensable le développement de solutions innovantes pour lutter efficacement contre la fraude.

C’est dans ce contexte que s’inscrit notre projet, dont l’objectif principal est de concevoir un outil automatisé d’analyse et de détection des sinistres potentiellement frauduleux au sein d’un portefeuille d’assurances. Le système vise à repérer les anomalies et comportements suspects afin de mieux orienter les contrôles, optimiser les enquêtes et, in fine, réduire les pertes financières liées à la fraude.

***II) Méthodologie employée***

1. ***Les premières idées***

La phase initiale du projet a été marquée par plusieurs réflexions. Une première orientation devait être faite entre l’analyse de sinistres liés à l’automobile ou à l’assurance vie. Le choix s’est rapidement porté sur l’assurance automobile, jugée plus concrète, plus riche en données exploitables, et potentiellement plus simple à modéliser pour une première version d’un outil de détection.

Notre intention était de nous appuyer sur des règles simples mais robustes pour signaler des cas suspects : des indicateurs comme le moment de l’année (saison, météo), l’âge du conducteur, le délai de déclaration ou encore les termes utilisés dans la déclaration textuelle. Certaines idées plus complexes ont été brièvement envisagées, comme la reconstitution du sinistre à l’aide de graphes ou de modèles de machine learning supervisé. Toutefois, en raison de la complexité de ces approches et du temps limité, nous avons choisi de rester sur un projet plus pragmatique, mais opérationnel.

Un autre point essentiel dès le début a été de concevoir un outil accessible et intuitif. L’utilisateur final, souvent non expert en data science, devait pouvoir utiliser le fichier Excel, lancer les macros, et obtenir rapidement un score de fraude par sinistre, des graphiques. L’objectif est de fournir une aide à la décision, et non un jugement définitif.

1. ***Démarche et déroulé du projet***

Pour ce faire, nous avons adopté une démarche structurée en plusieurs étapes. Tout d’abord, une phase de réflexion collective a permis une mise en commun des idées afin de définir les enjeux, fixer les objectifs et établir une feuille de route claire. Nous avons ensuite procédé à l’identification des indicateurs pertinents, susceptibles de révéler des comportements anormaux ou suspects. Cette sélection a été déterminante pour les choix d’informations à intégrer dans la base de données. À partir de là, nous avons entamé la construction de la base de données. Le développement des outils de détection a ensuite reposé sur l’écriture de code en VBA. Nous avons aussi dû adapter la base de données, notamment en y intégrant des informations manquantes lorsque cela s’avérait nécessaire. Une fois les indicateurs opérationnels, nous avons mis en place un système de scoring. Chaque sinistre était évalué selon les indicateurs retenus : un score de 1 était attribué lorsqu’un indicateur suggérait une possible fraude, et 0 dans le cas contraire. Ce scoring, volontairement simple, peut ensuite être pondéré par l’utilisateur en fonction de ses priorités. Enfin, les résultats des analyses ont été restitués à travers une interface utilisateur : c’est donc la conception des *userforms* qui a constitué notre dernière étape.

Concernant les choix de données pour la base, ceux-ci ont été guidés par les indicateurs que nous souhaitions construire. La structure de la base s’articule autour des éléments fondamentaux d’un sinistre automobile, intégrant des informations à la fois administratives, temporelles, géographiques et descriptives. On y retrouve par exemple l’identité et l’âge de l’assuré, la date de souscription du contrat, la dernière consultation des données contractuelles, le coefficient bonus-malus (CRM), ainsi que des détails sur le sinistre (type, lieu, date, description, coût estimé) et des données annexes comme les conditions météorologiques au moment des faits. Bien que fictives, ces données s’inspirent étroitement des éléments que l’on peut trouver au sein d’une police d’assurance et lors de la déclaration d’un sinistre.

1. ***Les indicateurs que nous avons choisi de développer***

Nous avons élaboré un ensemble d’indicateurs simples, mais pertinents, qui permettent de capter différentes dimensions suspectes du sinistre :

* Âge incohérent : détection de conducteurs ayant un âge inférieur à l’âge légal pour conduire ou un âge très avancé.
* Incohérences temporelles :
  + Sinistre déclaré avant la date de souscription
  + Date du sinistre postérieure à la date de déclaration
  + Sinistre intervenant très peu de temps après la souscription (comportement suspect de contrat « opportuniste »)
* Incohérences météorologiques :
  + Météo décrite incompatible avec la nature du sinistre (par exemple, dégâts liés à la neige un jour déclaré « ensoleillé »)
  + Météo incohérente avec la saison (par exemple, neige en plein mois de juillet)
* Jour de survenance : Un sinistre déclaré un dimanche avec un usage du véhicule professionnel peut être plus suspect car le véhicule n’aurait pas dû être utilisé
* Coût du sinistre : Un montant anormalement élevé ou au contraire anormalement faible par rapport aux standards peut refléter une exagération ou une tentative de minoration pour éviter l’attention
* **CRM élevé :** Un CRM élevé peut-être le signe de beaucoup de déclarations d’accidents et donc le signe de fraude potentielle.
* **Dernière consultation des conditions générales du contrat :** Une consultation récente peut être le signe d’une déclaration d’un sinistre en fonction des garanties et donc d’une potentielle fraude.
* Nombre de sinistres par contrat : un nombre trop élevé de sinistres par contrat peut être suspecté de fraude.
* Analyse de la déclaration des faits :
  + En fonction du type de sinistre, le système dispose d’un lexique de mots ou expressions fréquemment associés à des cas de fraude (ex. : “oubli des clés” pour un vol, “soudain” pour un incendie). Chaque mot suspect trouvé ajoute des points à un score de suspicion.
  + Vérification de la cohérence météo : il compare les mots-clés de la description avec la météo déclarée. Par exemple, si un sinistre de type “inondation” est déclaré alors que la météo est “ensoleillée”, le système ajoute une pénalité importante au score de suspicion : des mots clefs selon la météo sont aussi attendus dans la description sinon c’est suspect.
  + Descriptions vagues : des phrases génériques comme “rien ne laissait présager” ou “tout à coup” sont aussi pénalisées, car elles manquent de précision et peuvent masquer un récit inventé.
  + Finalement, si le score cumulé dépasse un certain seuil, l’algorithme considère la déclaration comme potentiellement frauduleuse.

Ces indicateurs ont dont été implémentés via des fonctions en VBA et utiliser au sein de notre macro principale de détection des fraudeurs.

1. ***Scoring des indicateurs***

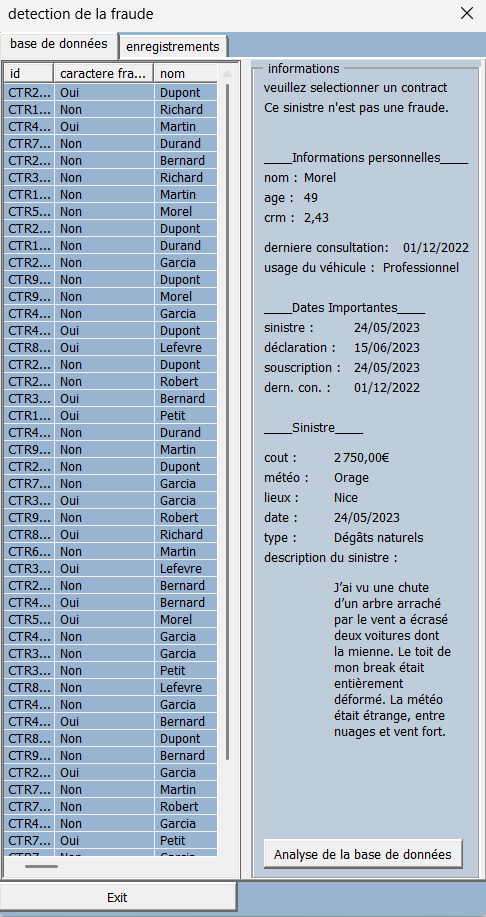
Afin d’agréger les différents signaux de fraude détectés, nous avons mis en place un système de pondération des indicateurs. Au départ, il s’agissait simplement d’un renvoi de nos fonctions. Un score +1 était affecté si notre indicateur indiquait un potentiel fraudeur et sinon 0. Par la suite, avec la création de notre *userform* nous avons permis à l’utilisateur de pondérer lui même les différents indicateurs par une valeur entre 0 et 1.

Chaque sinistre se voit donc attribuer un score global, calculé comme une somme pondérée des différents indicateurs activés.

Ce score peut être utilisé de plusieurs manières : pour filtrer les cas à examiner en priorité, pour générer des alertes automatiques au-delà d’un certain seuil ou encore pour segmenter le portefeuille en risque de fraudes (faible, modéré, élevé). A notre niveau, l’idée est que ce score puisse servir de filtre de priorité pour les gestionnaires : les sinistres ayant un score élevé peuvent être examinés en priorité ou transmis à un service spécialisé pour enquête et donc qu’il nous serve à produire nos résultats finaux affichés à l’utilisateur.

1. ***Création des userforms et rapports finaux de fraude***

Nous avons eu pour idée de concevoir une interface *userform*, permettant un dialogue direct entre notre base de données (par l’enregistrement de nouveaux sinistres, la suppression d’anciens sinistres et la visualisation de ces derniers) et elle. Ce système est implémenté par l’intermédiaire de deux autres feuilles excel, tres sombrement nommées : (*register* et *options*).

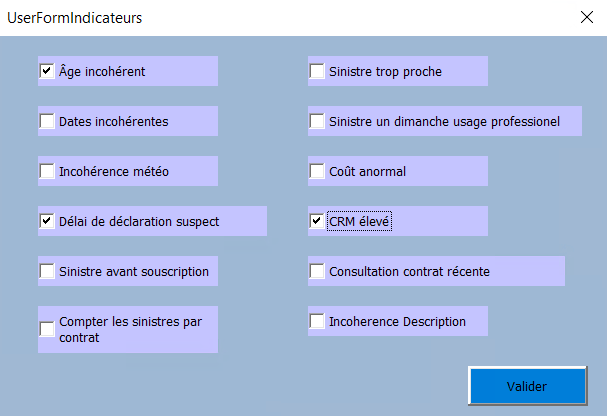
La premiere permet l’enregistrement et la moddification de nouveaux sinistres. Plus spécifiquement, elle permet de jouer avec nos données entrantes, sans comprometre notre base de données principale.

Une fois l’utilisateur satisfait, il peut envoyé les moddifications souhaitées dans la base de données principale («  Sinistres »).

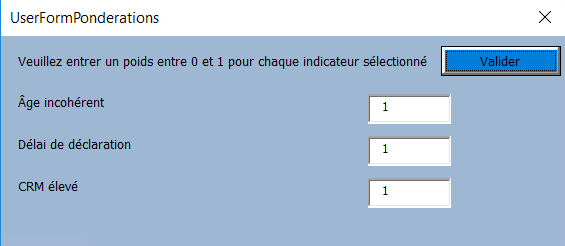
Elle peut etre appellée par l’intermédiaire de la *macro* : « Logiciel ».

La feuille : « options », n’a quant a elle que peu d’interet pour l’utilisateur lambda, elle ne sert que d’intermédiaire entre les differentes composantes de cette premiere interface (par exemple en permettant de stocker le nombres de sinistres enregistrés dans telle ou telle feuille, ou l’usage de telle ou telle macro). Elle fludifie les interactions entre les composantes (par exemple pour préciser qu’el sinistre l’utilisateur souhaite moddifier).

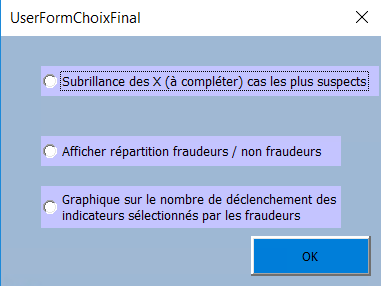
Notre premiere interface *userform*, permet egallement un dialogue avec une seconde interface par l’intermédiaire du bouton : « analyse de la base de données ».

******

Un second *userform* permet à l’utilisateur de sélectionner les indicateurs qu’ils souhaitent utiliser pour faire sa détection de fraude via des cases à cocher. Il s’agit donc uniquement de CheckBox dont on peut récupérer la valeur et savoir si la case été cochée ou non une fois que l’utilisateur aura validé ses choix.



Une fois qu’il a fait cela il va se servir d’un deuxième *userform* pour entrer les poids attribués à chaque indicateur. Comme indiqué précédemment, les valeurs permises sont comprises entre 0 et 1 et seront par défaut affectées à 1.



Enfin il dispose d’un troisième *userform* pour choisir les résultats qu’il veut afficher : veut-il afficher le tableur excel avec de la surbrillance sur les lignes de potentiels fraudeurs ou bien veut-il des graphiques sur sa base de données : quels indicateurs sont les plus souvent utilisés par les potentiels fraudeurs, combien de fraudeurs potentiels dans la base et leur répartition.

La majeure difficulté de cette étape a été de créer le deuxième *userform* de manière dynamique : le nombre de valeurs à rentrer dépendant du nombre d’indicateurs sélectionnés à l’étape d’avant. Il s’agissait donc au sein de notre deuxième *userform* de créer des objets ce que nous avons pu faire grâce à de la documentation.

De même, une autre difficulté a été de créer des graphiques en code VBA, là aussi les documents fournis en bibliographie nous ont été utiles, afin d’en apprendre plus et d’aboutir à nos graphiques.

***III) Notre programme et annotations***

1. ***Explication du code et des flux de macro***

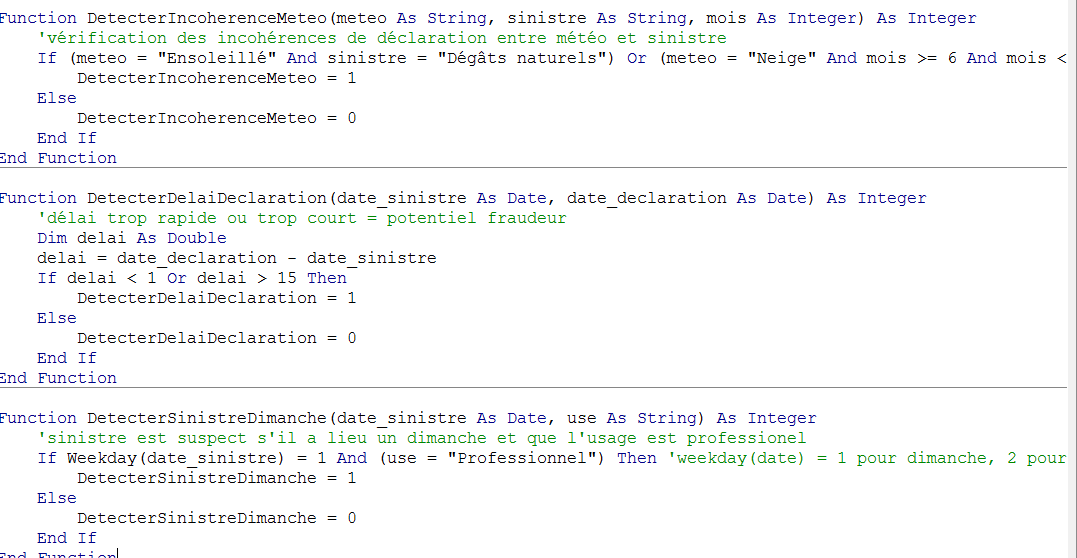
Premièrement, il est à noter que la première macro que doit lancer l’utilisateur (qui est d’ailleurs la seule donc doit s’occuper l’utilisateur) est celle nommée IdentifierFraudeursAvecChoix.

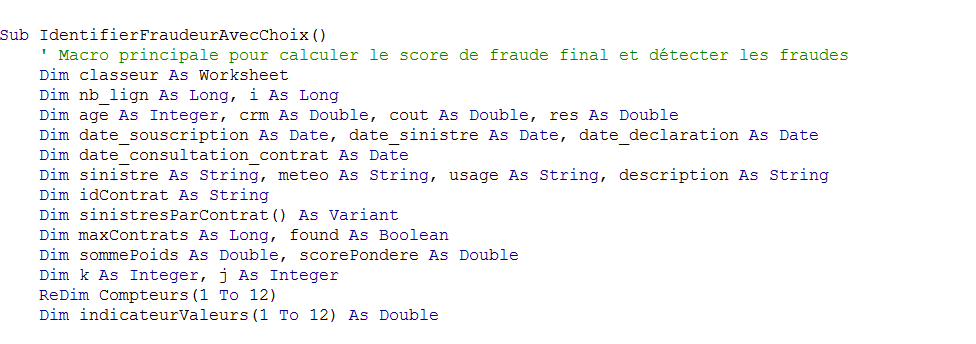
Cette macro aura plusieurs effets : premièrement elle chargera les deux premiers *userform* vu précédemment. Ce chargement des *userform* fait donc appel à la macro de lancement du deuxième *userform* nommée : UserForm\_Initialize(). De même ces userform lancent eux aussi des macro : lorsque les boutons de validation sont cliqués. Ce sont des macros qui permettent notamment de vérifier le type des données entrées.

Ensuite la macro principale permettra d’appeler les fonctions correspondantes aux indicateurs, et ce seulement pour les indicateurs sélectionnés afin d’éviter les calculs inutiles. Une fois qu’elle a fait cela elle calculera le score de fraude global pondéré et permettra de lancer le dernier *userform* et donc sa macro d’initialisation sur les choix d’affichage ainsi que des macro lorsque les boutons sont cliqués (notamment pour cacher la valeur qui n’est demandée que dans l’option 1) . Enfin c’est ce dernier *userform* qui lancera les macros suivantes :

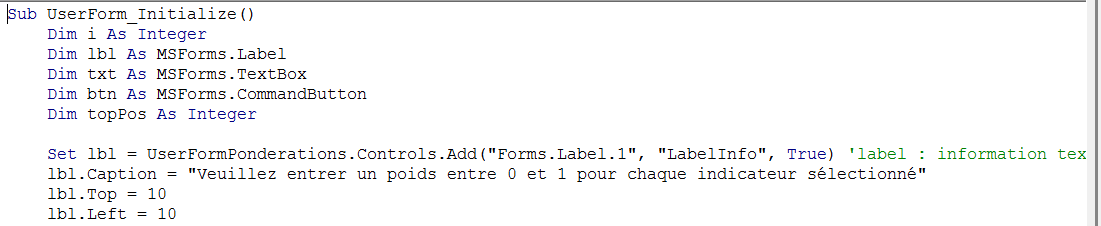
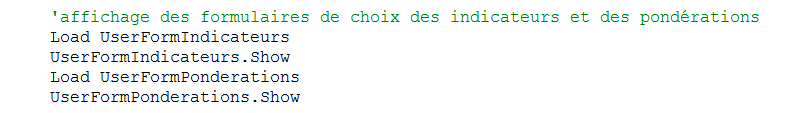
* si l’option de surbrillance a été choisie c’est la macro SurlignerTopXFraudeurs qui est lancée
* si l’option graphique de répartition des fraudeurs non fraudeurs a été sélectionnée c’est la macro AfficherGraphiqueFraudeurs qui est lancée
* si l’option graphique d’utilisation des indicateurs par les fraudeurs a été choisie c’est la macro AfficherGraphiqueIndicateurs qui est lancée

Tentons de résumer tout ça par une vue du code et des flèches indiquant les différents lancements de macros.

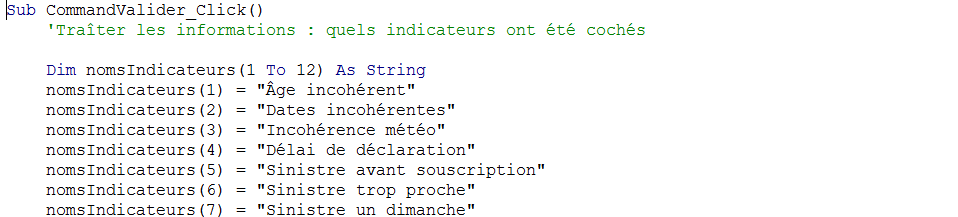
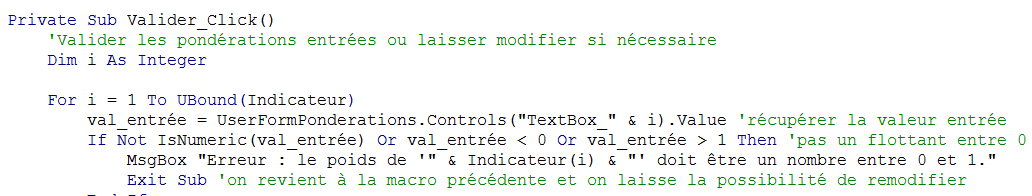
******

******

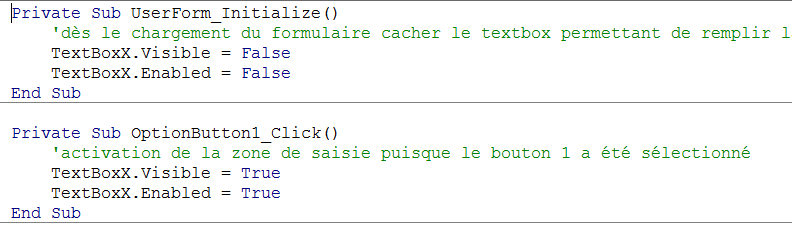
***Etape de lancement par la macro principale des fonctions de détection***

******

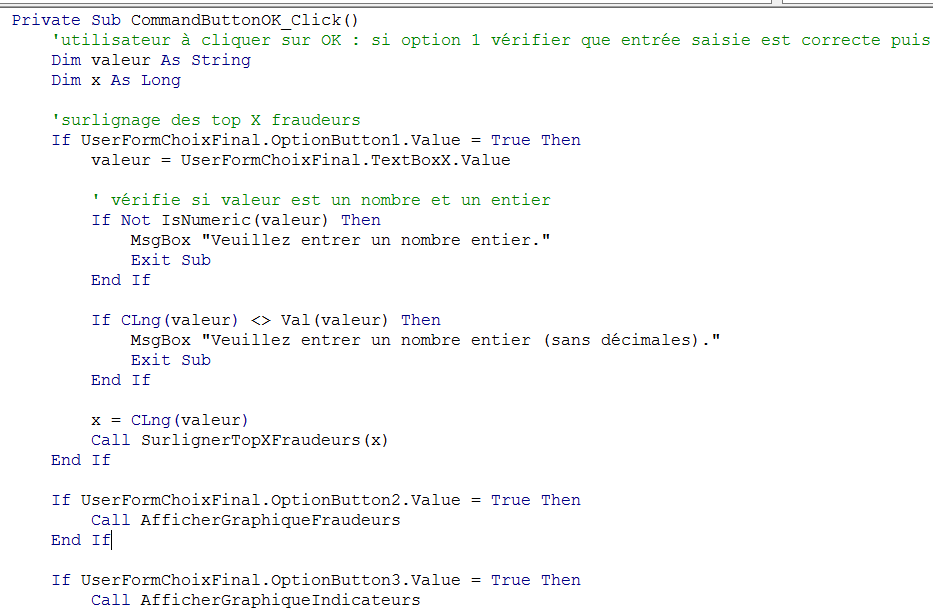
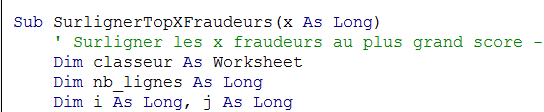
***Etape de lancement par la macro principale des userforms et macro initialisation du userform***

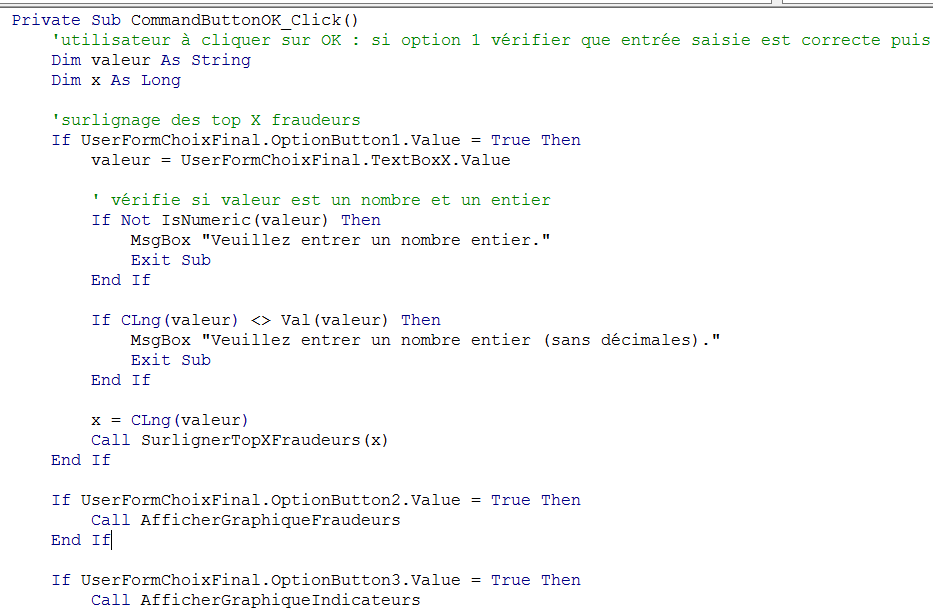
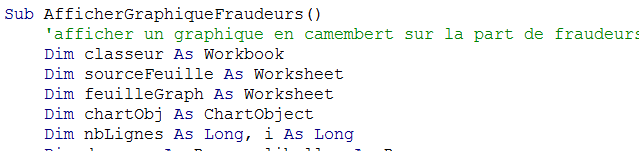
******

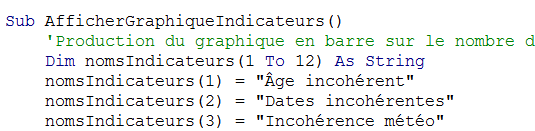
***Etape de lancement par les userform de macro : notamment validant l’entrée avec valider pour les 2 premiers userforms***

******

***Etape de lancement par le dernier userform lorsque les boutons sont cochés de macros privées pour afficher ou non une case à remplir***

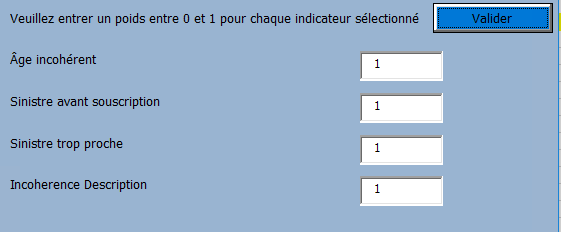
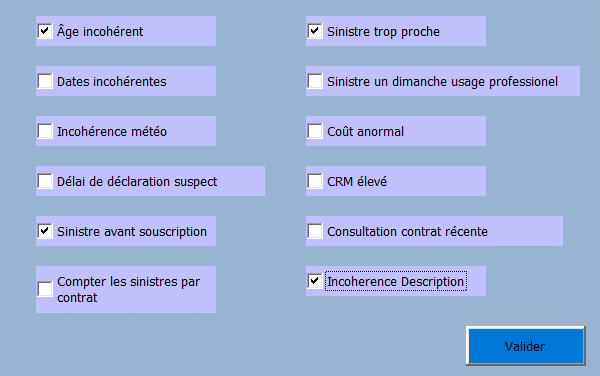
******

******

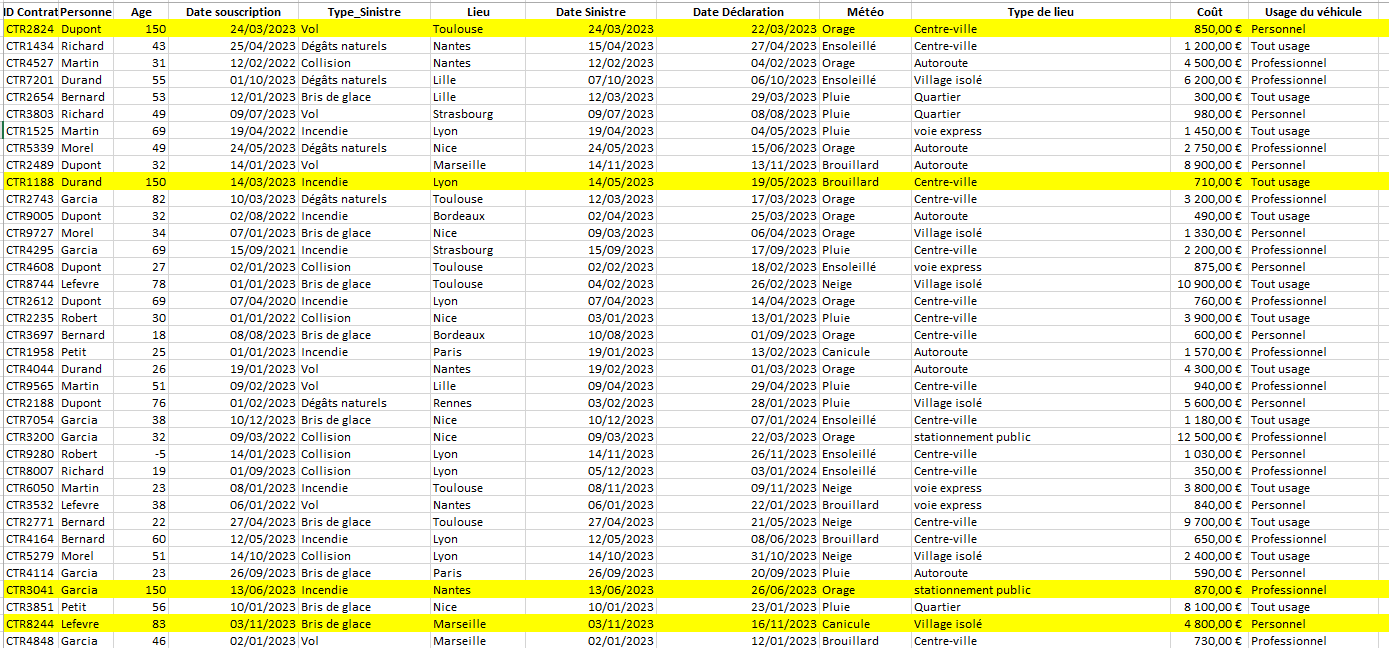
******

***Etape de lancement par le dernier userform des trois macro selon l’option sélectionnée***

1. ***Exemples de vues et résultats et analyse de ceux-ci***

******

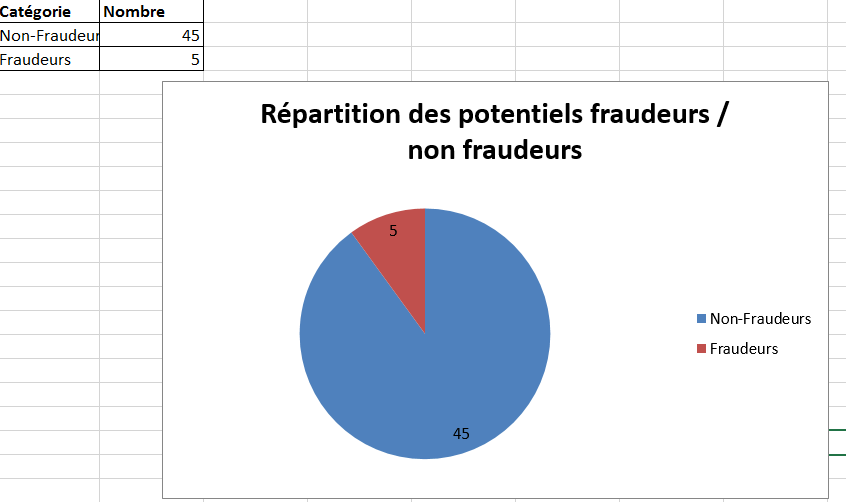
Nous nous plaçons donc dans le cas où l’utilisateur a fait les choix ci-dessus : il a décidé d’utiliser comme indicateur : l’incohérence sur l’âge, le sinistre avant souscription, le sinistre trop proche de la souscription et l’incohérence description. De plus, il a laissé les coefficients par défaut à 1.



Voici le résultat obtenu lorsque l’utilisateur décide de surligner les 10 premiers fraudeurs. Seulement 5 sont surlignés car il y a uniquement 5 fraudeurs potentiels dans la base détectés avec ces indicateurs. Ainsi pour l’analyse des résultats ces cas peuvent être analysés plus précisément notamment pour mettre en évidence des similitudes. Autrement, ces résultats peuvent aussi être utilisés par un service de gestion pour lancer une enquête sur ces cas suspects.

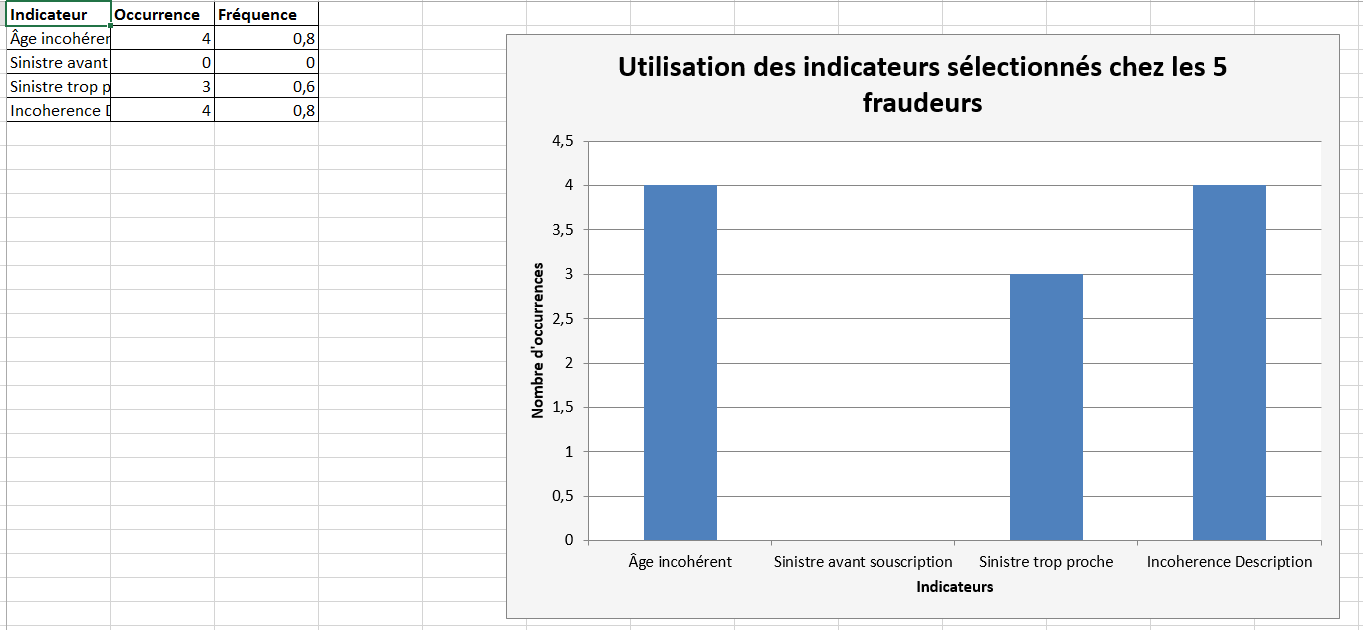
Du point de vue du rendu pur, l’utilisateur avait 10 cas surlignés il n’en a que 5 il peut déjà en tirer des conclusions sur le nombre de fraudeurs sur cette base avec ces indicateurs. De plus, il a accès à tous les détails de la déclaration, du contrat… Enfin, il peut aussi voir leur score de fraude et en déduire si certains sont potentiellement des fraudeurs vérifiant de nombreux points. Par exemple, ici un des fraudeurs à un score de 3 : il vérifie 3 indicateurs de fraude potentielle sur 4. En revanche, d’autres n’en vérifient que 2 et sont un peu moins visés par une fraude potentielle.

Ci-dessous, voici le résultat obtenu par l’utilisateur s’il choisir d’afficher le graphique sur la part de fraudeurs et non fraudeurs.

******

Cela offre une vue globale sur le pourcentage de fraude potentielle dans la base de données.   
 Dans ce cas présent par exemple : il y a 5 fraudeurs contre 45 non fraudeurs.

Enfin, voici la dernière visualisation s’il choisit d’afficher le graphique sur les indicateurs. Ce graphique permet de conclure quant aux indicateurs les plus utilisés par les potentiels fraudeurs et ceux qui ne sont pas pertinents.

******

Ainsi par exemple, on peut voir que l’indicateur sinistre avant souscription n’est pas pertinent tandis que âge incohérent ou incohérence description semblent bien plus pertinents car utilisés par 4 fraudeurs sur 5.

***IV) Limites du projet et sources d’amélioration***

Nous avons volontairement choisi des règles simples et compréhensibles. Si cela a l’avantage de la transparence, cela limite également la puissance de détection face à des fraudes plus sophistiquées. Des modèles statistiques ou des approches d’apprentissage automatique pourraient capter des combinaisons de variables plus complexes et très probablement moins arbitraires.

L’implémentation d’un neurone, que l’on aurait préalablement entrainer à déterminer une pondération optimale sur nos précédents contrats, ne nous apparait pas si extravagante.

De plus, il faut prêter attention à l’exploitation et l’interprétation des résultats qui dépendent fortement : des critères choisis et surtout des pondérations : des scores de 9 dans une expérience et de 0,1 dans une autre n’est pas comparables.

Le système de pondération est basé sur une appréciation subjective. Un calibrage plus rigoureux nécessiterait des données historiques de sinistres validés comme frauduleux ou non. Sans doute que cela nécessiterait une étude plus approfondie du fonctionnement des fraudeurs et des méthodes qu’ils utilisent le plus fréquemment. À terme, ces poids pourraient être calibrés à l’aide d’un historique de sinistres confirmés comme frauduleux.

L’analyse textuelle reste ici très simple et ne prend pas en compte le contexte des phrases ou les synonymes car les mots ont été définis d’avance.

Sur notre lancée, nous aurions pu extraire les rapports de sinistres (les « descriptions »), a l’intérieur de fichiers *word ou pdf,* vraisemblablement plus appropriés pour une description, qu’une cellule Excel.

Une extension possible consisterait à utiliser des techniques de traitement automatique du langage (NLP), comme l’analyse de sentiment, voire des modèles pré-entraînés pour détecter des schémas de rédaction typiques de fraudes. De même, notre système est sensible à la casse c’est encore un point qu’il faudrait améliorer par exemple via l’usage d’expressions régulières (motif à reconnaître).

Il nous faut prendre de la hauteur sur nos travaux et mettre en doute la qualité de notre code sur des bases de données autrement plus grandes. Notre *complexité* peut parfois être de l’ordre du *quadratique* et pourrait sans nul doute être améliorée à l’aide de doctrines plus inspirées.

Pour conclure nous tenons à souligner l’évidente faiblesse de notre code : la « sécurité ».

Nous sommes amenés à traiter des *données sensibles* et aucun moyen n’a été mis en œuvre pour sécuriser notre base de données contre des intrusions étrangères et potentiellement malveillantes.

Traiter de façon aussi légère des *données sensibles*, pourrait représenter un problème légal pour l’entreprise qui serait amené à utiliser notre code.

***V) Bibliographie***