RAPPORT PROJET – PSID

Présenté par :

RAMZI Réda 43013130

MAMMERI Amir 43010871

HENDALI Wassim 43016395

27 avril 2025

Nom du projet : TerrorTrack

Lien vers le github (Front et Back): https://github.com/redarmz/TerrorTrack

Lien vers la vidéo final : https://www.youtube.com/watch?v=eGwf2blo5xc

Lien vers le Drive du projet (vidéos, notebook, rapport) :

https://drive.google.com/drive/folders/1xxQvRAYxT0LkzLNC-xlNDX8BOCY0kIFZ?usp=drive_link

I. Introduction

Dans un monde où la sécurité publique est un enjeu majeur, l'accès à des outils capables de centraliser, analyser et visualiser les données liées aux attaques terroristes est devenu essentiel. **TerrorTrack** s'inscrit dans cette dynamique en proposant une application innovante permettant d'explorer, d'interpréter et de visualiser des informations complexes sur les attentats terroristes à travers le monde. Cette initiative se veut informative, pédagogique et stratégique, à la croisée des domaines de la data science, de la cybersécurité et de l'intelligence artificielle.

Grâce à l'exploitation d'un large jeu de données provenant de la **Global Terrorism Database (GTD)**, TerrorTrack ambitionne de rendre l'analyse du phénomène terroriste accessible, compréhensible et exploitable pour les chercheurs, journalistes, étudiants, décideurs publics ou simples citoyens intéressés par la géopolitique et les menaces sécuritaires.

Objectifs de l'application

L'objectif principal de **TerrorTrack** est de démocratiser l'accès à l'information sur les attaques terroristes mondiales, tout en fournissant des outils interactifs et puissants pour :

- Explorer dynamiquement les attentats à travers une interface cartographique conviviale ;
- Filtrer et visualiser les données selon divers critères (année, région, groupe terroriste, type d'attaque, etc.) ;
- Fournir des analyses statistiques synthétiques permettant d'identifier des tendances ou des foyers d'instabilité ;
- Détecter des schémas ou anomalies pouvant faire l'objet d'une investigation plus poussée ;
- Offrir une base solide pour des travaux de recherche ou de prévention.

Fonctionnalités clés

L'application **TerrorTrack** repose sur trois piliers fonctionnels majeurs :

1. **Visualisation interactive** :
    - Carte mondiale des attentats avec filtres multiples (date, pays, type d'arme, cibles, etc.).
    - Affichage détaillé de chaque attaque sélectionnée.
2. **Statistiques dynamiques** :
    - Graphiques temporels et géographiques sur l'évolution des attentats.
    - Histogrammes par pays, groupes terroristes, types d'attaque ou armes utilisées.
3. **Analyse exploratoire** :
    - Filtres combinatoires (par région, période, groupe armé, etc.) pour affiner les recherches.
    - Tableaux de données exportables pour usage externe.

Chaque fonctionnalité a été pensée pour offrir à l'utilisateur une expérience fluide, intuitive, et informative, que ce soit pour un survol rapide ou une analyse approfondie.

Public cible

TerrorTrack a été conçu pour s'adresser à une audience large, avec des niveaux de familiarité variés avec les données ou le sujet :

- **Chercheurs en sciences politiques, relations internationales, géopolitique** : pour analyser les dynamiques du terrorisme.
- **Journalistes et analystes** : pour illustrer et contextualiser leurs articles ou rapports.
- **Étudiants** : dans le cadre de mémoires, d'exposés ou d'apprentissages en data science ou sécurité.
- **Décideurs publics et ONG** : pour identifier des zones à risques, justifier des décisions stratégiques.
- **Citoyens curieux** : souhaitant mieux comprendre l'évolution du terrorisme à l'échelle mondiale.

Partie statistique

L'analyse statistique constitue une **fonctionnalité centrale** de **TerrorTrack**. L'objectif est de transformer un vaste corpus de données brutes en représentations lisibles, parlantes et exploitables, permettant aux utilisateurs de **dégager des tendances**, **d'identifier des foyers de violence**, ou encore de **mener des comparaisons entre périodes, pays ou groupes terroristes**.

L'application propose ainsi plusieurs visualisations dynamiques à partir des données de la GTD :

- **Évolution temporelle des attentats** : grâce à un graphique en courbes, l'utilisateur peut observer la croissance ou la décroissance des actes terroristes par année. Cette visualisation met en lumière des périodes clés (comme la hausse post-2001 ou la montée de Daesh entre 2013 et 2017).
- **Répartition géographique** : une carte interactive permet de visualiser les zones les plus touchées, avec une intensité de couleur proportionnelle au nombre d'attaques. Les foyers majeurs comme le Moyen-Orient, l'Afrique de l'Ouest ou l'Asie du Sud ressortent clairement.
- **Top des groupes terroristes** : un classement dynamique montre les groupes les plus actifs selon la période sélectionnée (Taliban, État Islamique, Boko Haram, etc.).
- **Typologie des attaques** : une répartition par type d'attaque (bombe, prise d'otage, fusillade, etc.) permet de voir l'évolution des méthodes utilisées.
- **Types de cibles** : une autre visualisation détaille les cibles privilégiées (civils, militaires, gouvernements, infrastructures, etc.), ce qui permet de mieux comprendre les motivations stratégiques derrière les attentats.

Ces visualisations sont conçues avec des outils modernes comme **Plotly**, **Chart.js** et **Leaflet**, et sont mises à jour dynamiquement selon les filtres de l'utilisateur. Elles visent non seulement à informer, mais aussi à susciter la réflexion et à appuyer des analyses plus poussées (géopolitiques, sociologiques, sécuritaires...).

Architecture métier

TerrorTrack repose sur une architecture modulaire combinant HTML/CSS/TypeScript pour le front-end, Django pour le back-end, et Google Colab pour l'exécution des modèles de Machine Learning. Cette organisation favorise la clarté, la modularité et l'intégration fluide entre les composants applicatifs et analytiques.

Frontend : (https://github.com/tonrepo/terrortrack-frontend)

— **Technologies :** HTML, CSS, TypeScript
— **Principales caractéristiques :** Le front-end a été développé en combinant HTML pour la structure, CSS pour le style, et TypeScript pour la logique et l'interactivité. Ce choix permet de concevoir des interfaces utilisateur claires, responsives et maintenables. TypeScript, avec son typage statique, apporte une sécurité accrue au développement, tout en facilitant la détection des erreurs à la compilation. L'usage conjoint de ces technologies standards du web garantit une compatibilité étendue et une personnalisation fine de l'expérience utilisateur.

Backend : (https://github.com/tonrepo/terrortrack-backend)

— **Technologie :** Django
— **Capacités :** Django a été choisi pour sa robustesse, sa sécurité intégrée, et sa capacité à accélérer le développement d'applications web grâce à son approche "batteries-included". Il fournit un système ORM puissant permettant de manipuler la base de données de manière déclarative, tout en favorisant une architecture claire en MVC. Le framework permet également une gestion native de l'authentification, des permissions, et de l'interface d'administration, ce qui en fait un choix pertinent pour construire rapidement un back-end complet et sécurisé. Son écosystème mature et sa documentation riche facilitent par ailleurs la collaboration et la maintenance du code.

Machine Learning :

— **Environnement :** Google Colab
— **Rôle et fonctionnement :** L'entraînement et le test des modèles de Machine Learning sont réalisés sur Google Colab, qui offre un environnement cloud collaboratif avec GPU/TPU en option. Cette solution a été retenue pour sa simplicité de prise en main, sa gratuité, et son intégration native avec les bibliothèques Python utilisées en data science (comme Pandas, Scikit-learn, TensorFlow, etc.). Elle permet de développer et d'expérimenter rapidement les algorithmes de détection d'anomalies ou de classification liés aux scénarios d'attentats traités dans le projet, tout en facilitant le partage et la reproductibilité des notebooks entre membres de l'équipe.

Pratiques de Collaboration et de DevOps

La réussite du projet **TerrorTrack** repose autant sur la qualité technique de l'application que sur la **dynamique de collaboration mise en place au sein de notre équipe**. Tout au long du

développement, nous avons adopté une approche structurée et agile, inspirée des **bonnes pratiques DevOps** et des principes de collaboration moderne.

Communication et coordination

Pour assurer un suivi régulier, fluide et efficace de l'avancement du projet, nous avons utilisé plusieurs outils complémentaires :

- **Discord** : utilisé comme canal principal de communication asynchrone et synchrone, il a permis d'échanger au quotidien, de partager des idées, de résoudre des problèmes rapidement, et de coordonner nos efforts à distance en dehors des séances encadrées.
- **Réunions régulières** : des points d'équipe hebdomadaires (parfois plus fréquents en période critique) ont été organisés pour faire le suivi de l'avancement, fixer les objectifs de sprint, répartir les tâches, et s'assurer que tout le monde reste aligné.
- **Séances de travail collaboratif encadrées** : lors des séances de cours, nous avons tiré parti de la présence des encadrants pour valider nos choix techniques, obtenir du feedback, résoudre des blocages, et structurer nos prochaines étapes.

Répartition des tâches

Le travail a été organisé de manière **modulaire** et **complémentaire** :

- Chaque membre s'est vu attribuer une ou plusieurs **responsabilités spécifiques** : traitement des données, visualisations, API backend, frontend, machine learning, documentation, etc.
- Nous avons régulièrement fait des **sessions de codage pair-à-pair**, particulièrement utiles pour résoudre les problèmes complexes et mutualiser les compétences.
- Nous avons aussi procédé à des **intégrations régulières** pour s'assurer que les composants développés séparément fonctionnent bien ensemble.

Suivi du code et DevOps

Pour le versionnage et la gestion du code, nous avons travaillé avec **GitHub**, où :

- Chaque fonctionnalité ou correctif faisait l'objet d'une **branche dédiée** ;
- Les **pull requests** étaient discutées et examinées collectivement pour garantir une bonne compréhension de chacun ;
- Un **workflow de merge propre** a été mis en place pour éviter les conflits et assurer une intégration fluide.

Afin d'assurer la **qualité du code**, nous avons utilisé :

- **CodeFactor** : un outil d'analyse statique de code directement intégré à notre GitHub, qui nous a permis d'identifier les problèmes de style, les duplications, les redondances ou les erreurs potentielles. Nous avons corrigé les points signalés au fil de l'eau afin de maintenir un code propre et lisible.

Travail collaboratif sur les notebooks

L'usage de **Google Colab** s'est révélé central pour la phase de traitement de données, d'analyse statistique et de machine learning. Il a permis :

- Une édition collaborative en temps réel des notebooks ;
- Le partage facile des résultats avec les membres de l'équipe ;
- L'intégration de visualisations intermédiaires (graphiques, cartes, courbes) ;
- La traçabilité des expérimentations de modèles ML grâce à l'historique des cellules.

En combinant ces outils et pratiques, nous avons pu garantir un travail **cohérent, itératif, transparent et de qualité**, dans une ambiance de **collaboration continue**. L'encadrement pédagogique et les retours des enseignants ont également joué un rôle fondamental pour structurer notre progression et orienter nos choix.

II. Personas utilisateurs

Afin de garantir une conception centrée utilisateur, nous avons identifié plusieurs profils représentatifs des futurs utilisateurs de **TerrorTrack**. Ces personas nous ont permis de mieux comprendre les attentes, les comportements et les objectifs des utilisateurs potentiels, et ainsi d'adapter l'ergonomie, les fonctionnalités et les visualisations de notre application en conséquence.

Persona 1 : Le Chercheur en Géopolitique – Dr. Karim El Mansouri

Profil

Karim, 43 ans, est professeur-chercheur en géopolitique et en relations internationales. Il travaille dans un institut de recherche basé à Bruxelles et se spécialise dans l'analyse des conflits asymétriques, du terrorisme et de leurs impacts géopolitiques. Il rédige des articles scientifiques, intervient dans des colloques, et conseille parfois des organisations internationales.

Besoins spécifiques

- **Accéder à des données fiables, structurées et historiques** sur les attentats à l'échelle mondiale, avec la possibilité d'analyser les évolutions sur plusieurs décennies.
- **Effectuer des analyses comparatives** entre régions (ex. : Sahel vs Moyen-Orient) ou entre groupes terroristes (ex. : Al-Qaïda vs Boko Haram).
- **Exporter des visualisations claires** pour illustrer ses publications, ou pour animer ses présentations en conférence.
- **Repérer les pics d'activité terroriste** pour les croiser avec des événements politiques ou militaires précis.

Utilisation de l'application

Karim utilise principalement les **filtres multi-critères** (par région, groupe, type d'attaque), les **timelines** et les **cartes thermiques**. Il exploite également l'export des données et la **consultation approfondie des fiches attentat**, disponibles dans l'interface.

Valeur ajoutée

TerrorTrack permet à Karim de **gagner un temps précieux** dans la collecte, la vérification et l'organisation des données. L'outil lui fournit des **bases empiriques solides** pour étayer ses thèses, tout en rendant ses interventions plus visuelles et percutantes.

Persona 2 : La Journaliste d'investigation – Jean Dupuis

Profil

Jean, 35 ans, est un journaliste indépendant spécialisé dans les affaires internationales et la sécurité. Il écrit pour plusieurs médias européens (Le Monde, Arte, Euronews) et réalise également des documentaires. Il enquête régulièrement sur les ramifications géopolitiques du terrorisme, notamment en Afrique, au Moyen-Orient et en Asie centrale.

Besoins spécifiques

- **Vérifier rapidement des informations** sur un événement : date, lieu, groupe impliqué, nombre de victimes, etc.
- **Contexualiser un fait d'actualité** grâce à une analyse régionale ou historique.
- **Accéder à des graphiques parlants** pour enrichir ses articles, dossiers ou reportages.

- **Croiser les données** entre plusieurs dimensions : zone géographique, cible, modus operandi.

Utilisation de l'application

Jean utilise principalement les **visualisations synthétiques**, la **recherche rapide** d'événements et les **statistiques globales**. Il apprécie les **infobulles descriptives**, les **filtres temporels simplifiés** et les **représentations par carte** qui permettent une lecture immédiate des tendances.

Valeur ajoutée

TerrorTrack devient pour Jean un **outil de veille stratégique et de vérification**. Il peut ainsi travailler de manière plus autonome, croiser ses sources, et proposer un journalisme appuyé sur des **données vérifiables, actualisées et visuellement impactantes**.

Persona 3 : L'Étudiant en cybersécurité – Mehdi Yilmaz

Profil

Mehdi, 24 ans, est étudiant en Master 2 Cybersécurité dans une université française. Passionné par la data science et la géopolitique, il prépare un mémoire sur l'analyse prédictive d'actes terroristes à partir de données ouvertes. Il possède des compétences avancées en Python, machine learning et visualisation de données.

Besoins spécifiques

- **Télécharger et manipuler le jeu de données GTD** dans ses propres notebooks.
- **Explorer des visualisations avancées** pour identifier des corrélations et poser des hypothèses de recherche.

- **Comprendre la logique du backend de l'application**, voire s’en inspirer pour ses propres projets.
- **Tester des modèles prédictifs** en s’appuyant sur les données fournies par TerrorTrack.

Utilisation de l’application

Mehdi navigue régulièrement entre le **frontend de visualisation**, le **notebook Colab fourni**, et les **dépôts GitHub** du projet. Il exploite les **dashboards statistiques** comme base d’analyse et utilise les **filtres combinés** pour isoler les cas pertinents à ses recherches.

Valeur ajoutée

TerrorTrack agit pour Mehdi comme un **cadre d’expérimentation concret**. Il peut observer comment les données sont structurées, comment elles sont visualisées, et s’en servir pour construire ses propres modules d’analyse. Il y trouve à la fois une **inspiration méthodologique** et une **source de données stable et accessible**.

III. Présentation des données

Sources des données

Les données principales utilisées pour cette analyse proviennent de jeux de données rigoureusement choisis sur la plateforme Kaggle, reconnue pour la qualité de ses ressources. La priorité a été donnée aux sources disposant d'un volume suffisant d'informations, permettant un entraînement optimal des modèles prédictifs, tout en assurant une représentation fidèle des tendances observées.

La phase de collecte des données a été guidée par des critères rigoureux, dont le principe était de disposer d'un volume de données suffisant pour garantir des analyses solides et des performances optimales en apprentissage automatique. En plus de la quantité, d'autres exigences ont été prises en compte, telles que la qualité des données, leur pertinence, ainsi que la fiabilité des sources utilisées.

Modification / Nettoyage des données

Dans le processus de nettoyage des données, une phase cruciale d'ajustement et de transformation des données a été mise en œuvre afin d'assurer leur cohérence, leur complétude et leur exploitabilité pour l'analyse. L'objectif de cette étape était de corriger les anomalies persistantes, de standardiser les valeurs manquantes et d'éliminer les incohérences résiduelles, tout en réduisant la dimensionnalité du jeu de données pour le rendre plus maniable.

Étape 1 : Suppression des données non pertinentes

- Grâce à un script automatisé, nous avons supprimé toutes les colonnes qui contenaient plus de 100 000 lignes vides afin d'optimiser la qualité des données.
- De plus, nous avons éliminé les lignes correspondant aux années antérieures à 1990. Le jeu de données final se concentre donc sur la période de 1990 à 2017.

Étape 2 : Traitement des colonnes avec incohérences

Nous avons ensuite procédé au nettoyage des colonnes présentant des incohérences ou des valeurs non logiques :

- Colonne imonth : suppression des lignes avec le mois "0" (20 lignes impactées), car un mois "0" est non valide.
- Colonne iday : suppression des lignes avec un jour "0" (892 lignes impactées), également non valide.
- Colonnes attacktype1 et attacktype1_txt : suppression des lignes vides (30 116 lignes impactées).

- Colonnes targetype1 et targetype1_txt : suppression des lignes vides (30 116 lignes impactées).

PS : les lignes vides dans les colonnes atacktype1, attacktype1_txt, targetype1, targetype1_txt sont toutes mêmes, par exemple si une ligne dans la colonne atacktype1 est vide, alors elle le sera dans les autres colonnes (d'où le 30116 lignes impacter pour ces 4 colonnes).

Étape 3 : Suppression des colonnes non informatives

Certaines colonnes n'apportaient pas d'informations pertinentes pour notre analyse. Nous les avons donc supprimées :

- Colonnes INT_LOG, INT_IDEO, INT_MISC, INT_ANY
- Colonnes crit1, crit2, crit3

Étape 4 : Gestion des valeurs négatives et inconnues

- Colonne nperps : remplacement des valeurs négatives et vides par -9 pour indiquer un nombre de terroristes inconnu.
- Colonne claimed : remplacement des valeurs vides par -9 pour indiquer l'absence d'information sur la revendication de l'attaque.
- Colonne weatype : remplacement des lignes vides par 13, identifiant pour "Unknown".
- Colonne weatype_txt : remplacement des lignes vides par "Unknown".

Résultat du nettoyage

- Fichier de départ : 135 colonnes et 181 690 lignes.
- Fichier nettoyé : 47 colonnes et 110 266 lignes.

Description des colonnes

Voici une description de certaines colonnes de notre dataset.

Colonnes temporelles et géographiques

- **iyear, imonth, iday** : Date de l'événement.
- **country_txt, region_txt, city, location** : Localisation textuelle à différents niveaux.
- **latitude, longitude** : Coordonnées GPS.
- **extended** : Attaque de plus de 24h (1 = oui, 0 = non).

Détails sur l'attaque

- **success** : L'attaque a-t-elle atteint son objectif ? (1 = oui, 0 = non)
- **suicide** : Attaque-suicide ou non.
- **attacktype1_txt**: Types d'attaque (bombardement, assaut armé, etc.).
- **weaptype1_txt**: Types d'armes utilisées (explosifs, armes légères, etc.).

- **weapsubtype1_txt** : Sous-types plus précis d'armes (grenade, AK-47, etc.).
- **motive** : Texte décrivant le motif de l'attaque.

Cibles

- **targtype1_txt** : Type de cible (militaire, civils, gouvernement, etc.).
- **targsubtype1_txt** : Sous-type, plus précis.
- **corp1, target1, natlty1_txt** : Organisation ciblée, description textuelle, nationalité.

Groupes responsables

- **gname, gsubname** : Nom du groupe terroriste principal ou associé.
- **individual** : Acte isolé ou non (1 = attaque individuelle).
- **claimed, claimmode_txt** : Revendication de l'attaque et mode utilisé (vidéo, appel, etc.).
- **compclaim** : Revendication concurrente de plusieurs groupes.

Dégâts humains et matériels

- **nkill, nwound** : Nombre de morts et blessés.
- **nkillter, nwoundte** : Morts/blessés côté terroriste.
- **property, propextent_txt, propvalue** : Dommages matériels et leur étendue.

IV. Intégration et Analyse des Visualisations

Intégration des visualisations dans l'application web

Étape 1 : Exportation des graphiques depuis Google Colab

Après avoir effectué le nettoyage et la préparation des données, les différentes visualisations ont été réalisées à l'aide de la bibliothèque Plotly dans un environnement Google Colab. Chaque graphique a été généré de manière interactive, puis exporté au format HTML à l'aide de la méthode fig.write_html("graph_X.html"), où X correspond au numéro ou au thème du graphique. Ce format d'exportation a été privilégié pour sa compatibilité avec le web, permettant une intégration simple et rapide sans nécessiter de traitement supplémentaire côté client.

Étape 2 : Organisation des fichiers dans l'architecture du projet

Les fichiers HTML obtenus ont été transférés dans l'arborescence du projet web, et plus précisément dans le répertoire frontend/public/plotly. Cette structure permet un accès direct aux fichiers via une URL relative, facilitant ainsi leur affichage dans les composants Vue.js via des balises .

Étape 3 : Intégration dans le tableau de bord Vue.js

Chaque fichier HTML correspondant à une visualisation a ensuite été intégré dans un composant Vue (Dashboard.vue) à l'aide de balises , encapsulées dans des sections dédiées à chaque graphique. Cela permet de structurer le tableau de bord de manière lisible et interactive, tout en maintenant une séparation claire entre les données, la logique métier (Django en backend) et la présentation (Vue.js en frontend).

Étape 4 : Communication avec le backend Django

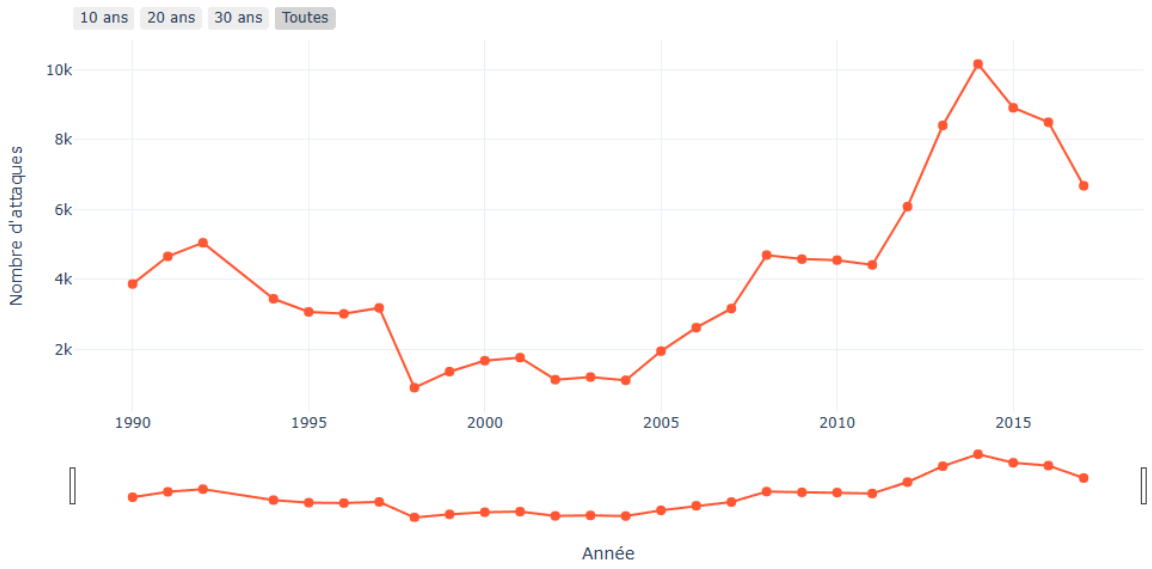
Le backend de l'application est géré avec le framework Django, utilisé principalement pour gérer la logique métier, la sécurité et l'architecture de l'application. Bien que les visualisations soient statiques (sous forme de fichiers HTML), Django permet de maintenir une structure cohérente du projet et de potentiellement étendre la logique en y intégrant des appels dynamiques ou des API REST si nécessaire. Ce processus a permis de relier efficacement la phase de préparation et d'exploration des données avec leur restitution visuelle au sein d'une interface web. L'exportation des graphiques sous format HTML et leur intégration dans l'interface Vue.js a facilité le déploiement rapide d'un tableau de bord interactif, sans recourir à des bibliothèques complexes côté client. Cette solution garantit une visualisation fluide, tout en respectant la séparation des responsabilités entre les différentes couches de l'application.

Analyse des visuels

Pour rédiger les analyses associées à chaque graphique, nous nous sommes appuyés sur une approche croisée mêlant l'exploitation des données brutes, la mise en perspective historique et l'analyse des grands événements géopolitiques survenus depuis 1990. À travers chaque description, nous avons cherché à dégager les grandes tendances, formuler des hypothèses explicatives, et surtout replacer les données dans leur contexte réel. L'objectif était de ne pas se limiter à une lecture purement statistique, mais d'enrichir les visualisations par des éléments concrets, des exemples marquants et parfois des récits humains représentatifs. Cette démarche permet de donner du sens aux chiffres, d'illustrer les dynamiques complexes du terrorisme mondial, et de proposer une grille de lecture à la fois analytique et sensible.

Voici quelques visuels que vous pouvez retrouver sur notre site web :

Évolution interactive du nombre d'attaques terroristes par année



Justification du choix du type de visuel :

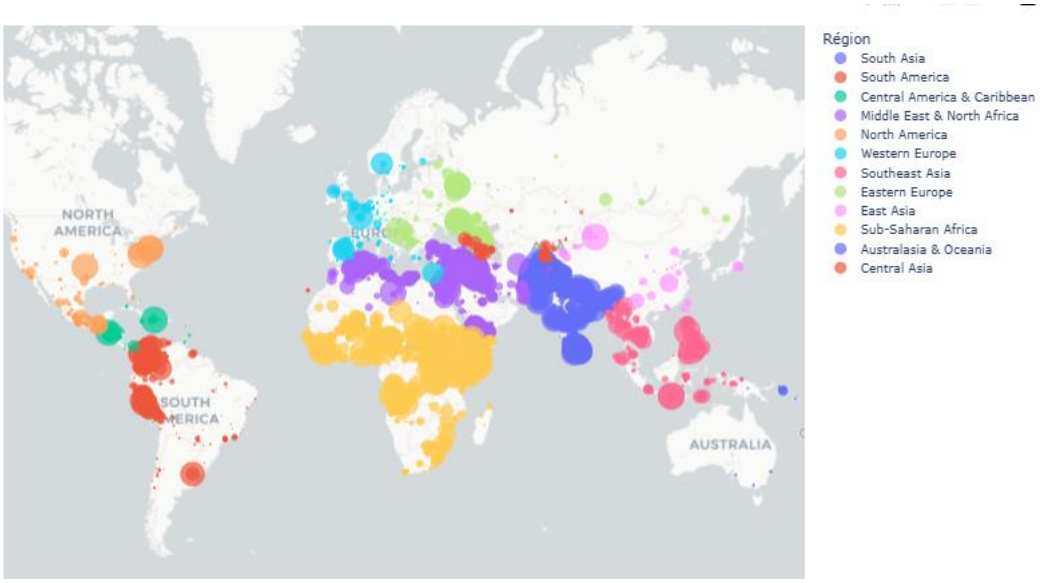
Ce graphique en courbe temporelle est particulièrement adapté pour illustrer l'évolution d'un phénomène sur le long terme. Il permet de visualiser de manière fluide les tendances, les pics et les creux dans la fréquence des attaques d'une année à l'autre. Grâce à la continuité visuelle qu'offre la ligne connectée, il est facile d'identifier les phases de hausse ou de baisse, et de les mettre en lien avec des événements historiques majeurs.

Interprétation :

Ce visuel met en évidence des fluctuations majeures dans l'intensité du terrorisme mondial au fil des décennies. La baisse notable autour de l'an 2000 contraste fortement avec l'explosion des attaques après 2010, culminant en 2014–2015, période dominée

par les actions de Daesh. La tendance post-2015 montre un léger recul, mais à un niveau toujours élevé. Ce graphique rappelle que les vagues de terrorisme ne sont pas aléatoires, mais fortement corrélées à des contextes géopolitiques précis.

Répartition des attaques terroristes (par région et gravité)



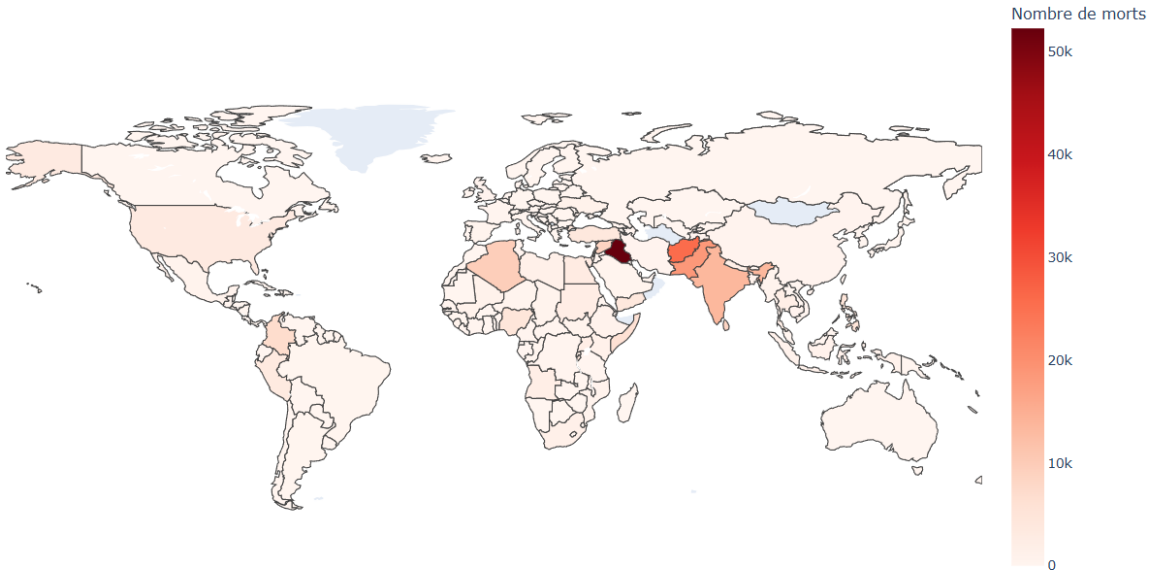
Justification du choix du type de visuel :

La carte de densité est le type de visuel le plus adapté lorsqu'il s'agit de représenter une variable spatiale à l'échelle mondiale. Ici, l'utilisation de couleurs par région et de tailles de points proportionnelles au nombre d'attentats permet une lecture instantanée des foyers de violence. Ce format est particulièrement utile pour mettre en évidence les zones chaudes du terrorisme mondial et comparer les différentes régions en un coup d'œil.

Interprétation :

La carte révèle une forte concentration d'attaques dans des régions politiquement instables ou en proie à des conflits armés prolongés, notamment en Asie du Sud, au Moyen-Orient et en Afrique subsaharienne. On remarque aussi des poches d'activité en Amérique latine, souvent liées à des guérillas ou des narco-conflits. En Occident, les attaques sont plus rares mais symboliquement marquantes. Cette répartition mondiale illustre bien que le terrorisme n'est pas un phénomène homogène, mais qu'il varie fortement selon les dynamiques locales, idéologiques et socio-économiques.

Carte des pays les plus touchés



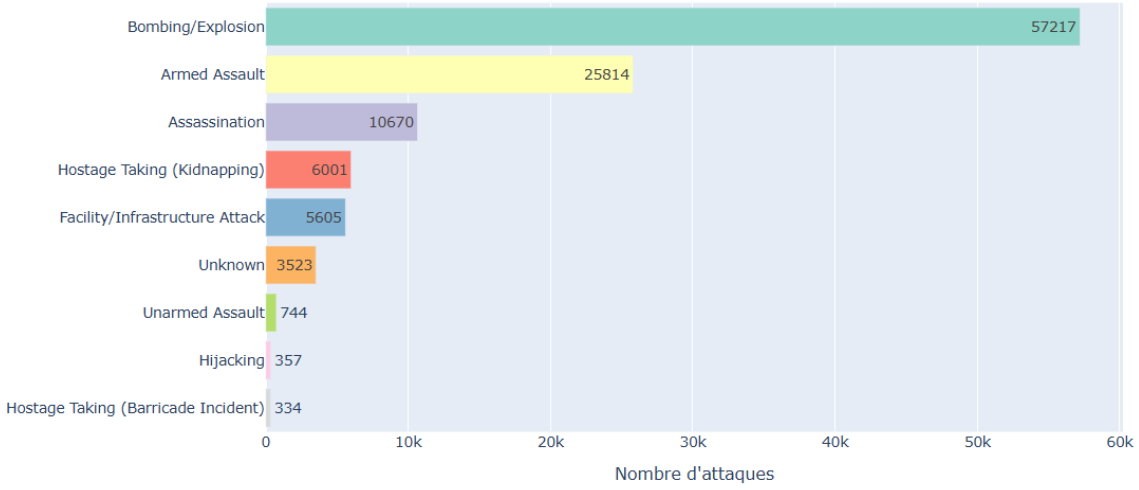
Justification du choix du type de visuel :

La carte choroplèthe, utilisant un dégradé de couleur pour représenter l'intensité d'un phénomène, est idéale pour visualiser la distribution spatiale d'un volume quantitatif, comme ici le nombre total de morts par pays. Le code couleur du rouge clair au rouge foncé permet une lecture immédiate de l'ampleur des pertes humaines, mettant en évidence les pays les plus affectés sans surcharge visuelle.

Interprétation :

Ce visuel met cruellement en évidence l'ampleur du coût humain du terrorisme dans certaines régions. L'Irak, profondément assombri sur la carte, concentre à lui seul une part dramatique des décès, en lien direct avec les conflits post-invasion de 2003 et la guerre contre Daesh. L'Afghanistan, le Pakistan, la Syrie, le Nigeria ou encore l'Inde figurent également parmi les nations les plus touchées. Ces zones sont caractérisées par des conflits prolongés, une faible sécurité intérieure et la présence de groupes armés actifs ciblant massivement les civils.
La carte ne montre pas seulement des chiffres, mais trace un portrait mondial de la vulnérabilité humaine face à l'instabilité et à la radicalisation.

Types d'attaque les plus fréquentes



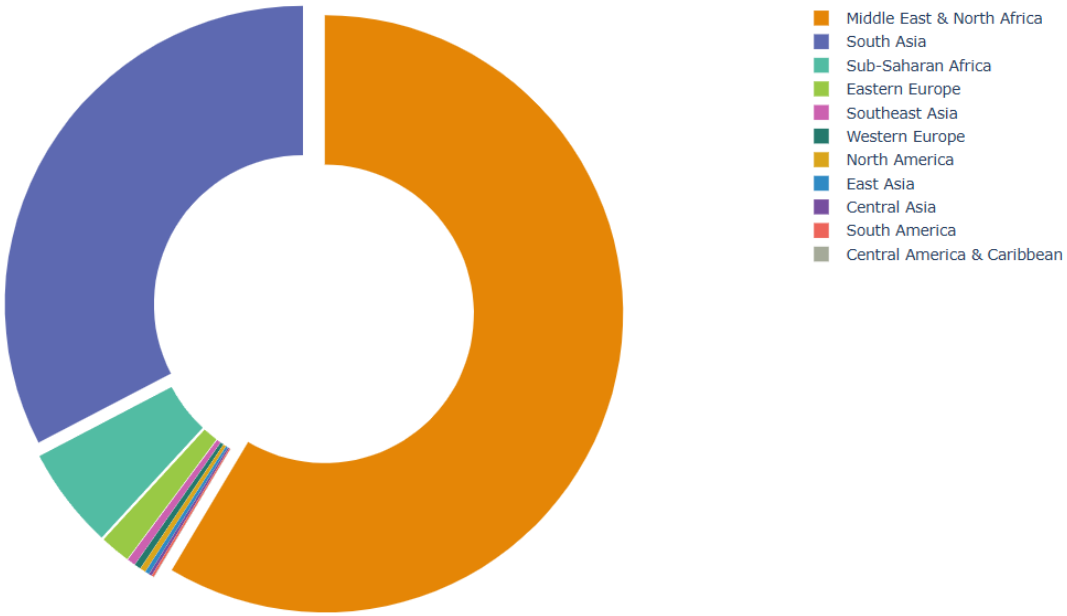
Justification du choix du type de visuel :

Le graphique à barres horizontales est parfaitement adapté pour comparer des catégories nominales sur un axe quantitatif. Il permet ici une lecture claire et hiérarchisée des différents types d'attaques, en mettant en évidence les plus fréquentes. L'affichage des valeurs sur chaque barre renforce la lisibilité, et l'ordre décroissant facilite l'identification des tactiques dominantes à l'échelle mondiale.

Interprétation :

Ce visuel révèle une nette prépondérance des attentats à la bombe ou par explosion, qui représentent à eux seuls plus de la moitié des attaques recensées. Suivent les assauts armés et les assassinats ciblés. Ces formes d'attaques partagent un point commun : elles sont rapides à exécuter, peu coûteuses, et fortement destructrices, tant en termes humains que symboliques. Les formes d'attaques plus complexes ou spécifiques, comme les détournements ou les incidents de barricade, sont minoritaires. Cela met en lumière la stratégie du terrorisme moderne : simplicité logistique, impact psychologique maximal.

Répartition des attaques suicides par région



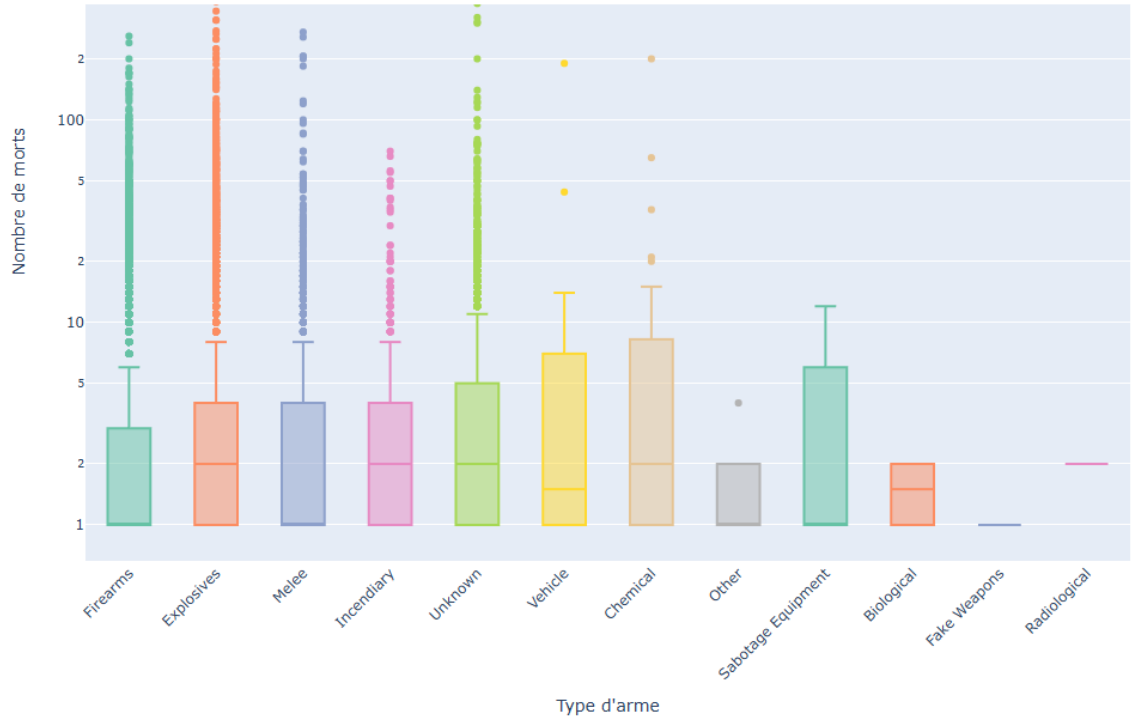
Justification du choix du type de visuel :

Le diagramme en anneau (ou camembert creux) est idéal pour représenter des partages proportionnels entre différentes catégories, ici les régions du monde. Son design circulaire met en valeur la part dominante de certaines zones géographiques tout en conservant une lisibilité visuelle. L'anneau permet également une meilleure aération que le camembert classique, ce qui améliore la lecture des petites parts régionales.

Interprétation :

Le graphique révèle une concentration écrasante des attaques suicides dans deux régions : le Moyen-Orient et l'Afrique du Nord d'un côté, et l'Asie du Sud de l'autre. Ces deux zones cumulent à elles seules l'écrasante majorité de ce type d'attentats. Cela témoigne de contextes géopolitiques propices à la radicalisation, où certaines idéologies extrémistes valorisent le suicide comme acte de guerre. Les autres régions, bien que touchées, restent marginales dans cette forme spécifique de violence.
Ce visuel met en évidence une stratégie régionale du terrorisme, où le recours au sacrifice humain s'inscrit dans des logiques de guerre psychologique, souvent à très forte charge symbolique.

Nombre de morts par type d'arme



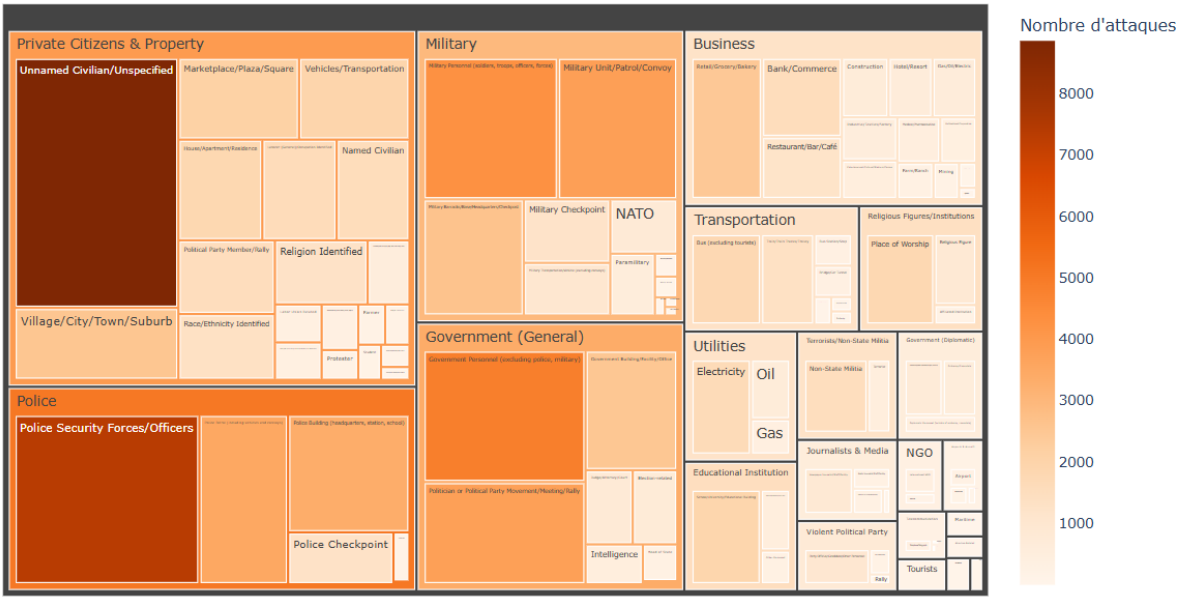
Justification du choix du type de visuel :

Le boxplot est particulièrement adapté pour comparer la distribution d'une variable quantitative (ici, le nombre de morts) entre plusieurs catégories (les types d'armes). Il permet de visualiser non seulement les valeurs médianes, mais aussi la dispersion, les valeurs extrêmes et la variabilité interne de chaque type d'arme. C'est un outil puissant pour révéler la létalité potentielle des différents moyens utilisés dans les attaques terroristes.

Interprétation :

Le graphique montre clairement que les armes à feu et les explosifs génèrent non seulement les bilans humains les plus lourds, mais aussi la plus grande variabilité, avec de nombreux cas extrêmes. Les attaques au véhicule, bien que moins fréquentes, peuvent produire des pics de mortalité significatifs, comme en témoignent les attentats de Nice ou Kaboul. Les catégories comme armes chimiques, biologiques ou radiologiques, bien que rares, présentent une capacité destructrice élevée, ce qui justifie une vigilance renforcée malgré leur faible occurrence. Ce visuel alerte sur la diversité des menaces : certaines armes sont courantes, d'autres marginales mais potentiellement catastrophiques.

Répartition de types de cibles visées



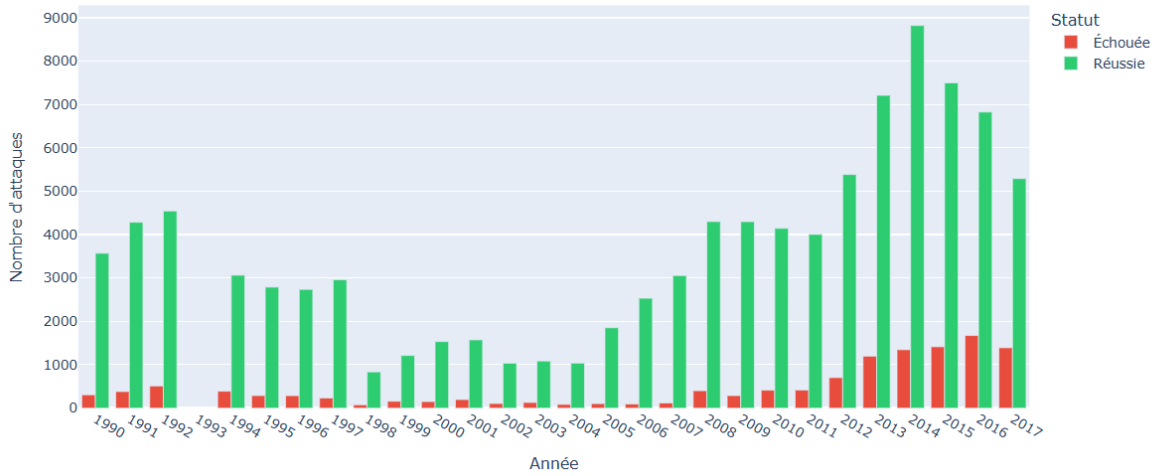
Justification du choix du type de visuel :

Le treemap hiérarchique est particulièrement efficace pour visualiser des catégories imbriquées avec des volumes associés, comme ici le nombre d'attaques en fonction du type de cible. Il permet de comparer rapidement les proportions, tout en offrant une structure claire et lisible des sous-catégories à l'intérieur de chaque groupe principal. L'utilisation d'un dégradé de couleurs renforce la perception de fréquence.

Interprétation :

Ce graphique met en évidence que les civils et leurs biens sont les cibles les plus fréquemment visées par les attaques terroristes. Les forces de l'ordre, les militaires, les gouvernements et les infrastructures civiles arrivent ensuite. Cette structure montre que le terrorisme cherche majoritairement à faire pression par la peur en ciblant les espaces de vie ordinaires, comme les marchés, les habitations ou les moyens de transport. Cela révèle une stratégie de désorganisation sociale par la violence symbolique et visible, visant à provoquer l'insécurité collective. Cette logique souligne l'importance de la protection des espaces publics dans les politiques de prévention.

Attaques réussies vs échouées par année



Justification du choix du type de visuel :

Le graphique en barres empilées est parfaitement adapté lorsqu'on souhaite comparer des sous-catégories dans une série chronologique. Ici, il permet d'évaluer non seulement l'évolution du nombre total d'attaques terroristes par année, mais aussi de distinguer visuellement leur statut (réussies ou échouées). L'usage de couleurs contrastées facilite une lecture intuitive et met en valeur les cas d'échec sans en détourner l'attention principale.

Interprétation :

Ce visuel met en évidence une dominance écrasante des attaques réussies, avec une augmentation notable à partir des années 2000, en particulier entre 2012 et 2015. Toutefois, la proportion d'attaques échouées tend à augmenter légèrement, signe d'une meilleure efficacité des mesures de contre-terrorisme. Ces résultats traduisent une tension croissante entre montée en puissance des groupes terroristes et renforcement des dispositifs de prévention. Le graphique rappelle également que chaque tentative déjouée est un succès discret, souvent lié à une intervention humaine, un renseignement ou une vigilance citoyenne.

V. Construction d'un modèle de prédiction du risque d’attaque terroriste

1. Introduction

Dans ce projet, nous avons cherché à prédire la probabilité qu'une attaque terroriste survienne dans un pays donné à partir d'informations contextuelles limitées. Cette problématique est particulièrement complexe car elle implique de travailler avec des données à forte sparsité, souvent incomplètes, tout en limitant le risque de surapprentissage. De plus, le respect de considérations éthiques dans la gestion des données a guidé nos choix techniques et méthodologiques.

Nous avons ainsi conçu un processus d'amélioration progressive des modèles, partant d'approches simples pour aboutir à une solution finale complexe, robuste, et adaptée à la réalité du domaine étudié.

2. Problématique

Le projet repose sur une hypothèse fondamentale : il serait possible de prédire le risque d'attentat à partir de quelques informations basiques sur un pays, telles que sa situation géopolitique, son niveau de développement économique ou la présence de groupes armés. Cependant, plusieurs défis majeurs sont rapidement apparus. Les données disponibles présentent un fort déséquilibre : les pays faiblement touchés par le terrorisme sont largement majoritaires dans l’échantillon, tandis que les événements les plus critiques sont relativement rares. Par ailleurs, certaines caractéristiques socio-économiques et politiques cruciales pour la compréhension du phénomène ne sont pas directement présentes dans les bases de données standardisées.

Face à ces limitations, une approche plus robuste et éthique a été adoptée. L’idée a été d’enrichir les données existantes à travers une analyse sociologique et géopolitique : en regroupant les pays selon des blocs régionaux ou des critères de stabilité économique, et en construisant de nouvelles variables explicites pour capturer des aspects structurels du risque. De plus, une attention particulière a été portée à la problématique de la sparsité : les techniques mises en œuvre visent à éviter que le modèle ne sur-apprenne sur les pays sur représentés, afin de permettre une généralisation correcte sur des zones moins documentées historiquement. Cette stratégie garantit ainsi une meilleure capacité de prédiction sur des contextes nouveaux ou sous-représentés, en préservant une approche équilibrée et responsable.

3. Méthodologie

La méthodologie suivie a été construite progressivement, en tenant compte des contraintes observées sur les données et de la nécessité de maintenir une approche éthique et robuste.

La première étape a consisté en une préparation minutieuse du jeu de données. L'objectif était de transformer les informations brutes disponibles en des variables exploitables pour la modélisation. Pour ce faire, plusieurs enrichissements ont été apportés. Une variable de risque (***risk***) a été créée à partir du nombre de victimes (***nkill***), distinguant ainsi les événements mortels des incidents sans victimes. De nouvelles caractéristiques, comme l'indicateur de revenu (***revenu_faible)*** et le statut démocratique (***est_democratie),*** ont été ajoutées en s'appuyant sur des regroupements géopolitiques reconnus. La situation de guerre (***en_guerre)*** et la présence de groupes armés (***groupes_active)*** ont également été intégrées afin d'enrichir le contexte géopolitique de chaque observation.

Afin de rendre le jeu de données plus homogène, un regroupement des pays a été effectué. Les pays les plus représentés dans les données ont été conservés tels quels, tandis que les pays moins présents ont été regroupés sous une catégorie générique ("Autres"). Ce choix permet de limiter la sparsité et d'éviter de biaiser l'apprentissage.

Sur le plan du pré-traitement, un encodage OneHot a été appliqué aux variables catégorielles (***region_txt, pays_simplifie, targtype1_txt),*** suivi d'une normalisation des variables numériques (***revenu_faible, est_democratie, indice_instabilite***). Cette double approche garantit que toutes les variables contribuent de manière comparable à l'apprentissage.

Enfin, pour remédier au déséquilibre entre classes (***risk = 1*** très minoritaire par rapport à ***risk = 0***), la méthode SMOTE (Synthetic Minority Over-sampling Technique) a été utilisée. Cette technique permet de générer artificiellement des exemples de la classe minoritaire, améliorant ainsi la capacité du modèle à apprendre des cas rares sans pour autant surcharger la base existante.

Cette méthodologie de préparation a été appliquée de manière homogène à tous les modèles développés, en adaptant parfois certaines étapes (comme l'ajout d'une composante non supervisée) pour enrichir la diversité des approches.

4. Approche par modèles successifs

La construction du système d'aide à la décision s'est faite par itérations successives, chaque modèle venant enrichir la complexité et la pertinence du précédent. Cette progression a permis d'intégrer progressivement plus d'informations contextuelles tout en améliorant la capacité de généralisation du modèle final.

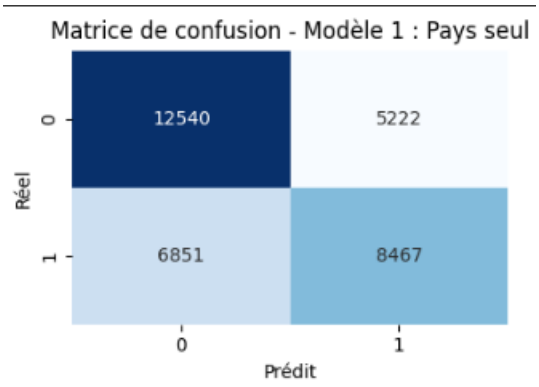
4.1 Modèle 1 : Apprentissage sur les données brutes **(Random Forest)**

Le premier modèle a été entraîné directement sur les données brutes du fichier d'attentats, en utilisant uniquement le pays (***country_txt***) et l'année (***iyear***) comme variables explicatives. Aucune transformation ni enrichissement n'était appliqué.

Pour la classification, nous avons utilisé un **Random Forest Classifier**, une méthode robuste contre le surapprentissage. Ce modèle a rapidement montré ses limites : il était incapable de généraliser sur les pays peu représentés, se contentant de mémoriser les cas fréquents (Irak, Afghanistan).

Techniques utilisées :

- Modèle : Random Forest
- Pas d'encodage, données brutes



4.2 Modèle 2 : Ajout de variables socio-politiques (**Random Forest**)

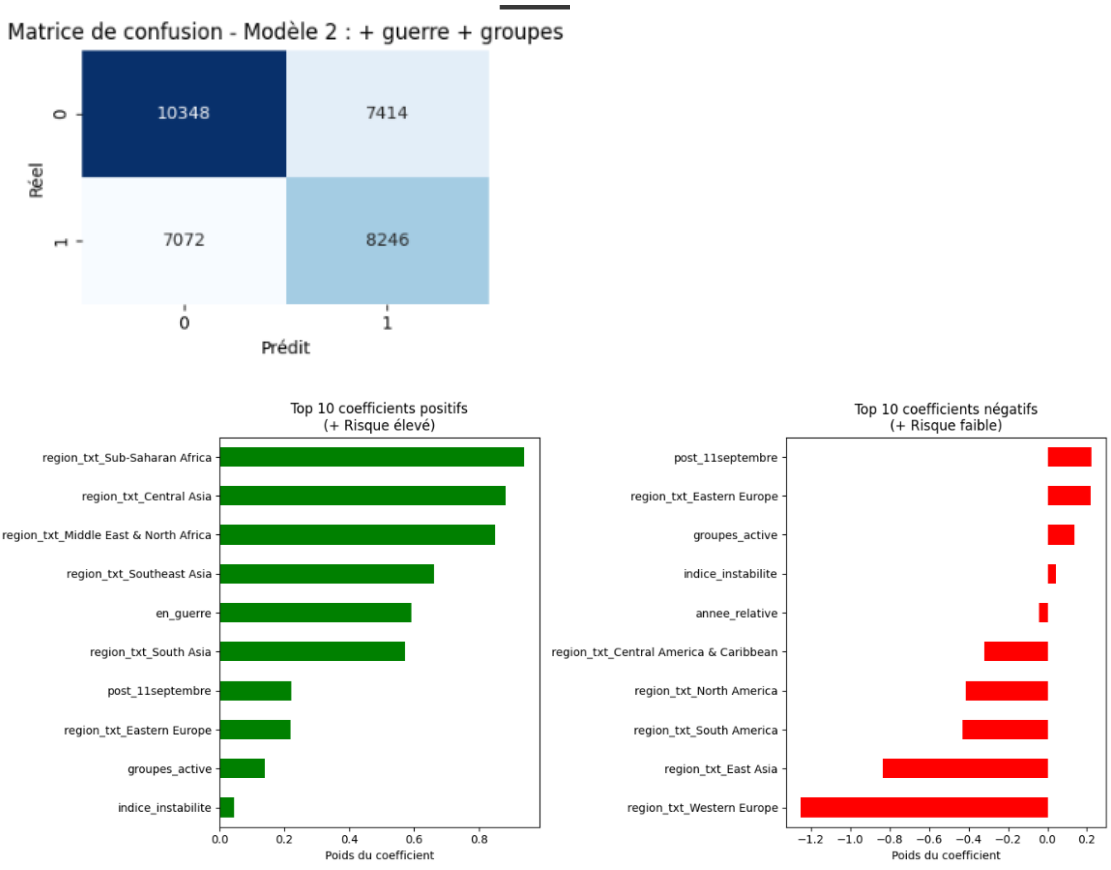
Afin de mieux prendre en compte la réalité géopolitique, un deuxième modèle a été développé en intégrant deux nouvelles variables : la situation de guerre (***en_guerre***) et la présence de groupes armés (***groupes_active***). Ces variables binaires ont permis d'ajouter une couche d'interprétation sur le niveau de risque latent dans chaque pays.

Le modèle utilisé est resté une **Random Forest Classifier**, pour conserver la continuité avec le modèle précédent, mais l'ajout de ces informations a permis d'améliorer légèrement la capacité du modèle à identifier les cas d'attentats mortels.

Technologie utilisée :

- Encodage : OneHotEncoder

- Modèle : Random Forest



4.3 Modèle 3 : Création d'un indice d'instabilité (Random Forest)

Afin de capturer la dynamique géopolitique des pays, nous avons introduit une nouvelle variable : **l'indice d'instabilité**.

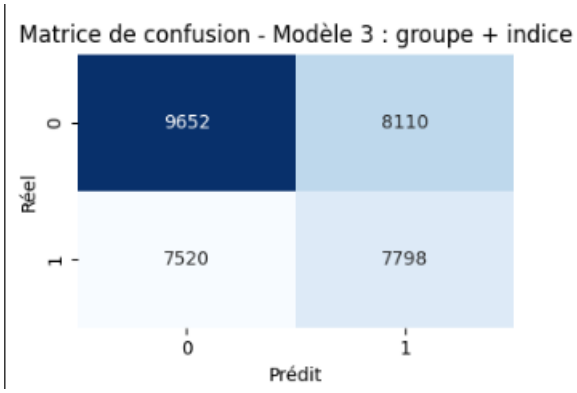
Basé sur la combinaison des indicateurs ***en_guerre*** et ***groupes_activ***e, cet indice synthétisait le niveau de dangerosité sociopolitique d'un pays. Un léger bruit aléatoire a été ajouté pour éviter l'apprentissage trop rigide.

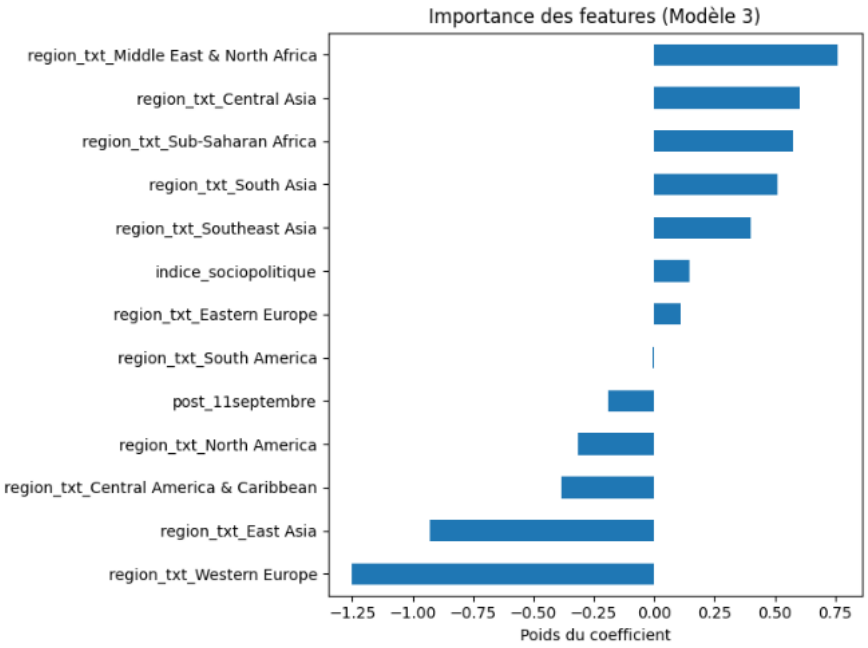
Le modèle reposait toujours sur une **Random Forest**, mais avec un dataset enrichi de cette nouvelle dimension.

Techniques utilisées :

- Modèle : Random Forest
- Ajout de la variable indice_instabilite

- Reprise du pipeline précédent (encodage, standardisa tion, SMOTE)





4.4 Modèle 4 : Premiers traitements non supervisés (***Random Forest***)

Le modèle 4 a constitué une véritable première fusion méthodologique : il a intégré l'ensemble des variables créées précédemment (revenu_faible, est_democratie, en_guerre, groupes_active, indice_instabilite), enrichies par des regroupements intelligents des pays (pays_simplifie) et des régions (region_txt).

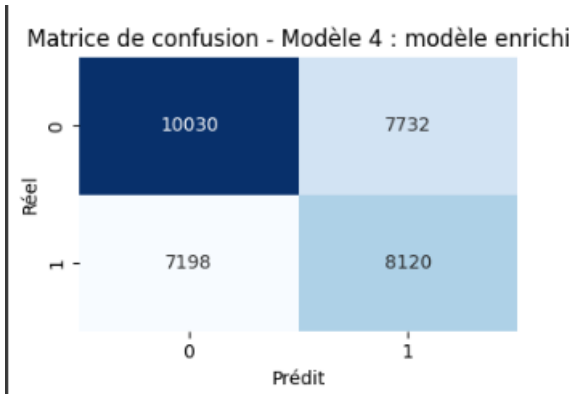
À ce stade, une stratégie de gestion de la sparsité a été systématiquement appliquée :

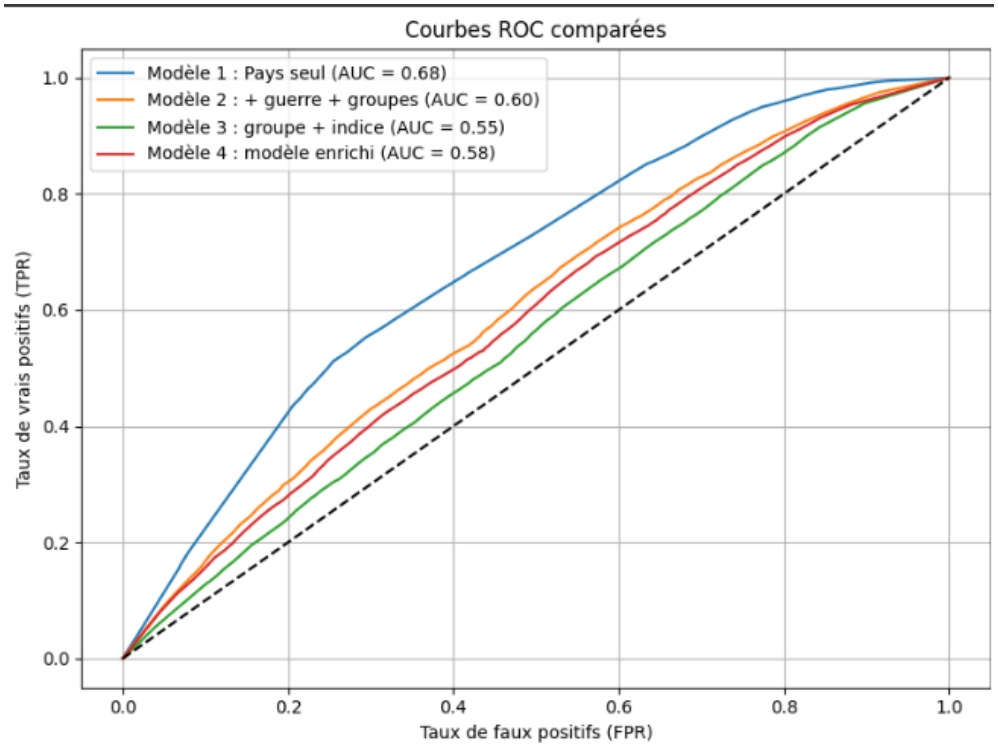
- Groupement des pays peu représentés sous la modalité "Autres"
- Introduction d'indicateurs synthétiques (plutôt que multiplication de catégories).

Le modèle était basé sur une **Random Forest optimisée** par validation croisée. Il a servi de socle pour les approches ultérieures.

Technologie utilisée :

- OneHotEncoder, StandardScaler
- Modèle : Random Forest + SMOTE (équilibrage)





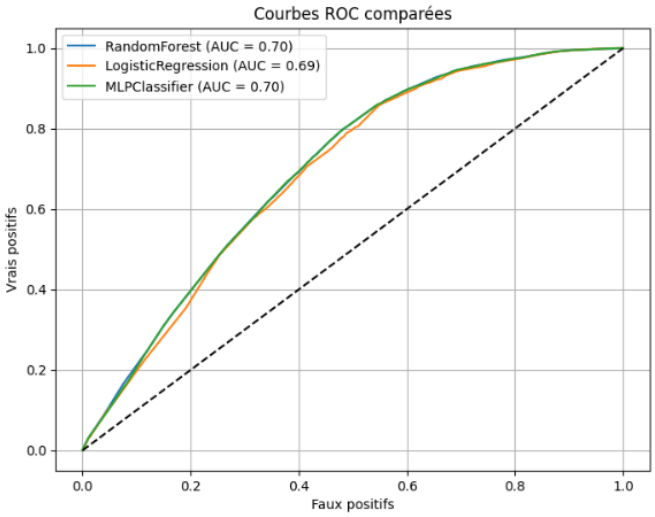
4.5 Modèle 5 : Croisement type de cible / région (**Random Forest, MLP, Logistic Regression)**

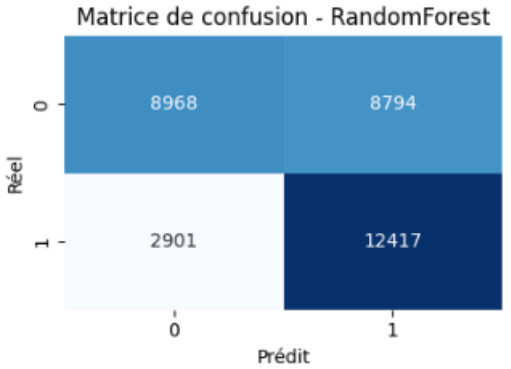
Dans un souci d'améliorer la spécificité du modèle, une nouvelle approche a été testée en utilisant directement deux variables plus précises : targtype1_txt (type de cible de l'attentat) et region_txt. Ces variables ont été encodées et croisées pour estimer la probabilité d'un attentat selon la combinaison type de cible / région.

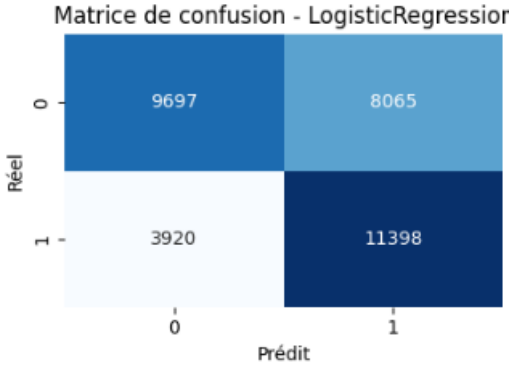
Le modèle utilisé pour cette tâche a été également un **Random Forest**, optimisé par validation croisée.

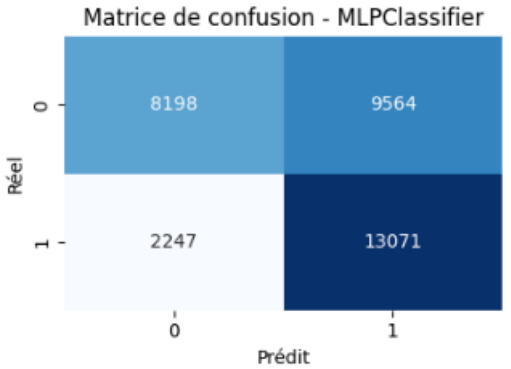
Technologie utilisée :

- OneHotEncoder
- Random Forest
- SMOTE (équilibrage)









4.6 Modèle 6 : Construction d'un modèle hybride avec réseau de neurones

Le sixième et dernier modèle représente l'aboutissement de la démarche méthodologique mise en place dans ce projet. Il vise à combiner plusieurs techniques supervisées et non supervisées afin d'améliorer la capacité de généralisation du modèle

tout en capturant les dynamiques complexes inhérentes aux phénomènes géopolitiques.

Le cœur de ce modèle repose sur l'utilisation **d'un réseau de neurones multicouches (MLPClassifier),** une architecture capable de modéliser des relations non linéaires entre les variables d'entrée et la variable cible. Avant l'entraînement, les données ont subi un enrichissement conséquent. Tout d'abord, les caractéristiques socio-économiques et politiques construites lors des modèles précédents ont été conservées : revenu du pays (***revenu_faible***), statut démocratique (***est_democratie***) et un indice d'instabilité calculé comme une combinaison pondérée de la situation de guerre et de l'activité des groupes terroristes dans le pays.

Pour capter davantage de structure cachée dans les données, une analyse non supervisée par KMeans clustering a été intégrée. Ce processus a permis de regrouper les attentats en 5 clusters distincts sur la base de leurs caractéristiques géographiques et socio-politiques. Chaque observation se voit donc enrichie d'une étiquette de cluster.

Par ailleurs, afin de réduire la dimensionnalité de l'espace et de conserver l'information la plus pertinente, une réduction par **Analyse en Composantes Principales (PCA)** a été appliquée. Deux nouvelles variables principales (***pca_1 et pca_2***) ont été extraites, résumant la majeure partie de la variance des données enrichies.

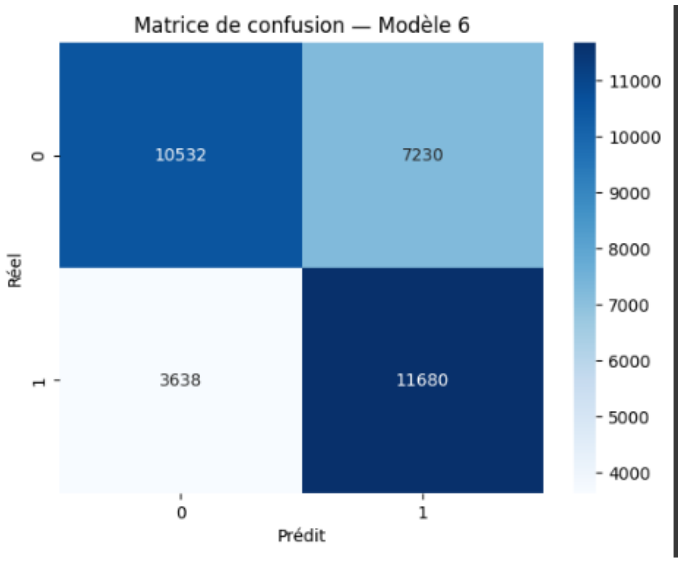
L'ensemble des variables finales utilisées pour l'entraînement comprend donc :

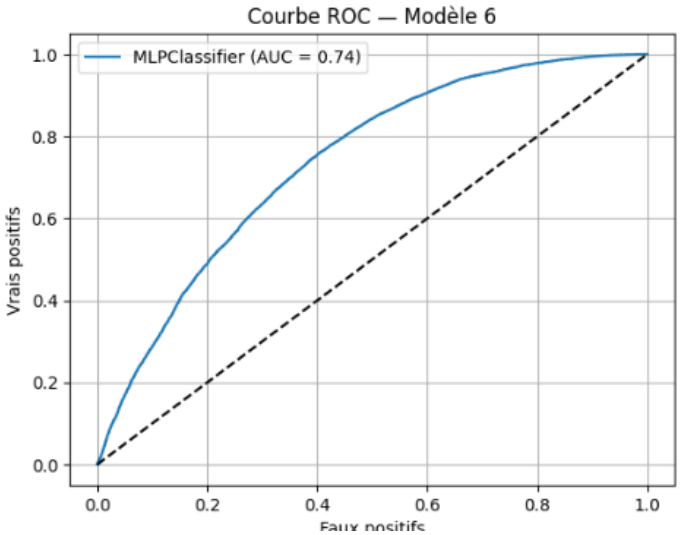
- Les variables catégorielles encodées par OneHotEncoder (région, pays simplifié, type de cible),
- Les variables socio-économiques standardisées ***(revenu_faible, est_democratie, indice_instabilite),***
- Le cluster issu de KMeans,
- Les deux composantes principales issues du PCA (***pca_1 et pca_2***).

Les données d'entraînement ont été équilibrées à l'aide de la méthode SMOTE, afin de corriger la forte déséquilibre entre les attentats ayant causé des victimes et ceux n'en ayant pas causé.

Enfin, le réseau de neurones a été optimisé via une recherche d'hyperparamètres utilisant GridSearchCV. Différentes configurations de nombre de neurones, d'activations (***relu, tanh***), de taux d'apprentissage (***learning_rate_init***) et de régularisation (***alpha***) ont été testées, et le meilleur modèle a été retenu en fonction du score F1 obtenu en validation croisée.

Ce modèle final, issu de la combinaison méthodique de traitements supervisés et non supervisés, offre un excellent compromis entre précision, capacité de généralisation, et robustesse face aux catégories rares, tout en respectant les contraintes d'éthique et de stabilité imposées au projet.


Matrice de confusion — Modèle 6
10532
7230
3638
11680
Réel
Prédit
0
1
11000
10000
9000
8000
7000
6000
5000
4000


Courbe ROC — Modèle 6
MLPClassifier (AUC = 0.74)
Vrais positifs
Faux positifs
0.0
0.2
0.4
0.6
0.8
1.0