République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université des Sciences et de la Technologie Houari Boumediene
Faculté d'Informatique
Spécialité :
Systèmes Informatiques Intelligents

DATA MINING Practical Work Project

Ecrit par:
DJESSAS Abdesalah 212131052961
DJESSAS Aziz 212131052975

Table des Matières

Introduction Générale

Les incendies de forêt représentent un enjeu environnemental et socio-économique majeur, causant des pertes importantes en végétation, une dégradation des sols et des impacts écologiques sévères. La prédiction précoce de ces événements est cruciale pour mettre en place des stratégies efficaces de prévention et de gestion des risques.

Ce projet a pour objectif de développer un modèle prédictif s'appuyant sur des données environnementales, notamment les caractéristiques du sol et les paramètres climatiques. En combinant des techniques de fouille de données ainsi que des méthodes d'apprentissage supervisé et non supervisé, nous cherchons à identifier les relations entre ces variables et la survenue des incendies, afin de concevoir un système d'alerte performant.

Le travail s'organise en trois phases principales :

1. **Analyse et préparation des données** : exploration profonde et mise en forme des données pour assurer leur qualité.
2. **Implémentation et évaluation de modèles supervisés** : développement d'algorithmes classiques tels que KNN, arbres de décision et random forest, avec une comparaison aux versions optimisées proposées par Scikit-learn.
3. **Apprentissage non supervisé (clustering)** : identification de groupes naturels dans les données pour repérer des zones à risque, en développant et évaluant plusieurs algorithmes de clustering.

Ce rapport présente ainsi la méthodologie adoptée, les résultats obtenus, ainsi que l'analyse critique des différentes approches, afin de proposer des pistes pour améliorer la détection et la prévention des incendies de forêt à partir des données environnementales.

Chapter 1

Data Analysis and Preprocessing

1.1 Préparation des données spatiales

Les jeux de données mondiaux ont été préalablement découpés et extraits afin de correspondre précisément aux territoires de l'Algérie et de la Tunisie. Cette opération de découpage spatial permet de travailler uniquement sur les zones géographiques d'intérêt pour notre étude, garantissant ainsi la pertinence et la précision des analyses. En se concentrant sur ces régions spécifiques, nous optimisons le traitement des données tout en limitant le volume global à manipuler, ce qui facilite les opérations de fusion et d'intégration avec les autres sources de données locales, notamment celles relatives aux sols et aux feux de forêt. Cette approche ciblée est essentielle pour produire des modèles prédictifs adaptés aux conditions environnementales spécifiques à ces pays.

1.2 Préparation du Jeu de Données fire - part 1

Le jeu de données initial provient des observations de points chauds (*Active Fire*) par le capteur **VIIRS** (*Visible Infrared Imaging Radiometer Suite*) à bord des satellites *Suomi-NPP/JPSS*. Il sert de base pour la construction de la variable cible.

1. Variables clés : Type et Confiance

Les données brutes comportent des variables essentielles pour qualifier l'observation. L'objectif est d'identifier la nature du feu et sa fiabilité.

A. Variable type(Nature de l'Événement)

- **0 :** Feu de forêt ou de végétation — catégorie d'intérêt principal (cible positive).
- **2 :** Feu industriel / puits de gaz / infrastructures énergétiques — généralement stationnaires, non liés aux feux de forêt naturels.

- **3 :** Feu de végétation de faible qualité / feux agricoles — souvent de faible intensité, associés à des pratiques agricoles.

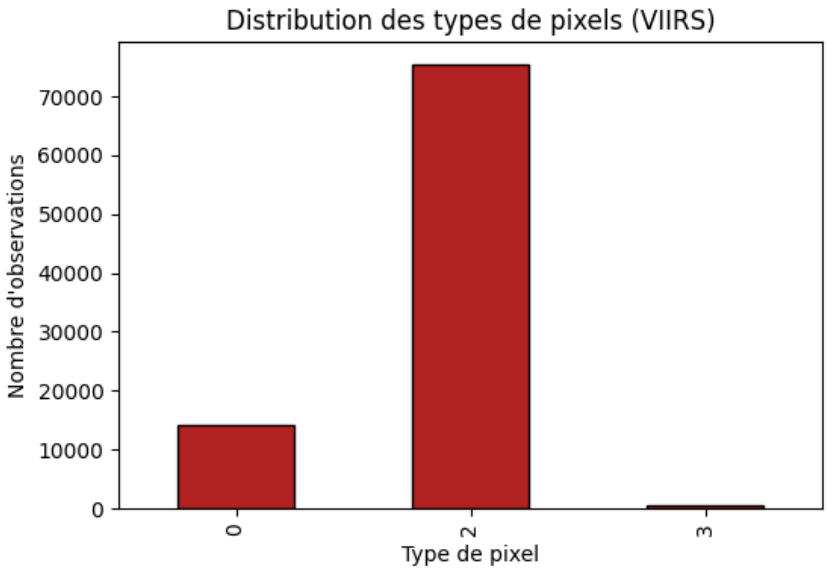

Figure 1.1: ChartBar de la variable Confidence.

L'analyse de distribution montre que la majorité des événements détectés sont de **Type 2 (industriel)**. Nous filtrerons donc spécifiquement les feux de forêt (**Type 0**) pour construire la variable cible.

B. Variable `confidence` (Fiabilité de la Détection)

- **l (Low)** : Faible probabilité.
- **n (Nominal)** : Probabilité modérée ou standard.
- **h (High)** : Probabilité élevée.

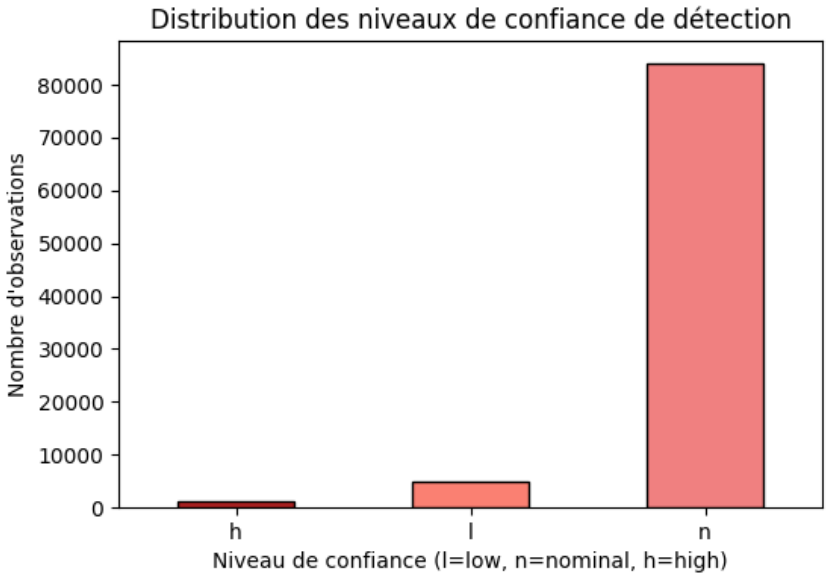

Figure 1.2: ChartBar de la variable Confidence.

La majorité des observations se trouvent dans les niveaux de confiance *Nominale* ou *Élevée*. Pour garantir la qualité des données, seules ces détections sont conservées.

2. Aperçu Statistique du Jeu de Données Initial

Le jeu de données fusionné comprend **90 250 observations** et **15 colonnes**.

- **Valeurs manquantes :** Aucune dans les colonnes principales (coordonnées, intensité, date, etc.).
- **Doublons :**

3. *Feature Engineering* (Variables Temporelles)

Pour intégrer la dimension temporelle :

- **Mois (`month`) :** extrait de la colonne `acq_date`, utilisé pour la fusion avec les rasters climatiques mensuels.
- **Jour de la semaine (`day_of_week`) :** indice du jour (lundi=0, dimanche=6), utile pour capturer les effets humains (week-end).
- **Heure décimale (`hour_decimal`) :** conversion de `acq_time` (format HHMM) en heures continues, pour modéliser l'effet de l'heure sur le risque d'incendie.

1.3 Exploration et Prétraitement de l'Altitude (elevation_dataset)

L'altitude constitue une caractéristique environnementale clé, car elle influence directement la température, les précipitations et, par conséquent, la répartition de la végétation ainsi que la propension aux incendies.

1.3.1 Visualisation et Exploration Statistique

L'exploration du fichier raster a révélé une distribution asymétrique des altitudes, phénomène typique des reliefs naturels.

A. Description Statistique

Les statistiques globales de la couche raster sont résumées ci-dessous :

Statistique	**Valeur**
Effectif (Count)	3 301 396
Minimum	−31.00
Maximum	2 633.00
Moyenne	535.76
Écart-type	325.68
Premier quartile (Q_1)	309.00
Médiane	462.00
Troisième quartile (Q_3)	697.00

B. Analyse de la Distribution

- **Asymétrie :** L'histogramme et la boîte à moustaches montrent que la majorité des altitudes se concentrent dans les basses et moyennes élévations (entre 0 m et 1000 m). La médiane (462 m) est inférieure à la moyenne (535.76 m), ce qui indique une *asymétrie positive* (distribution *skewed right*).

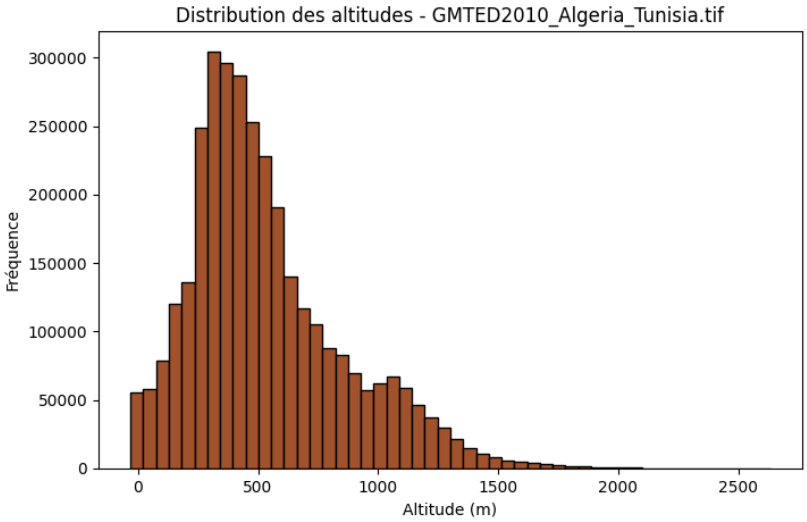

Figure 1.3: Histogramme de la variable Elevation.

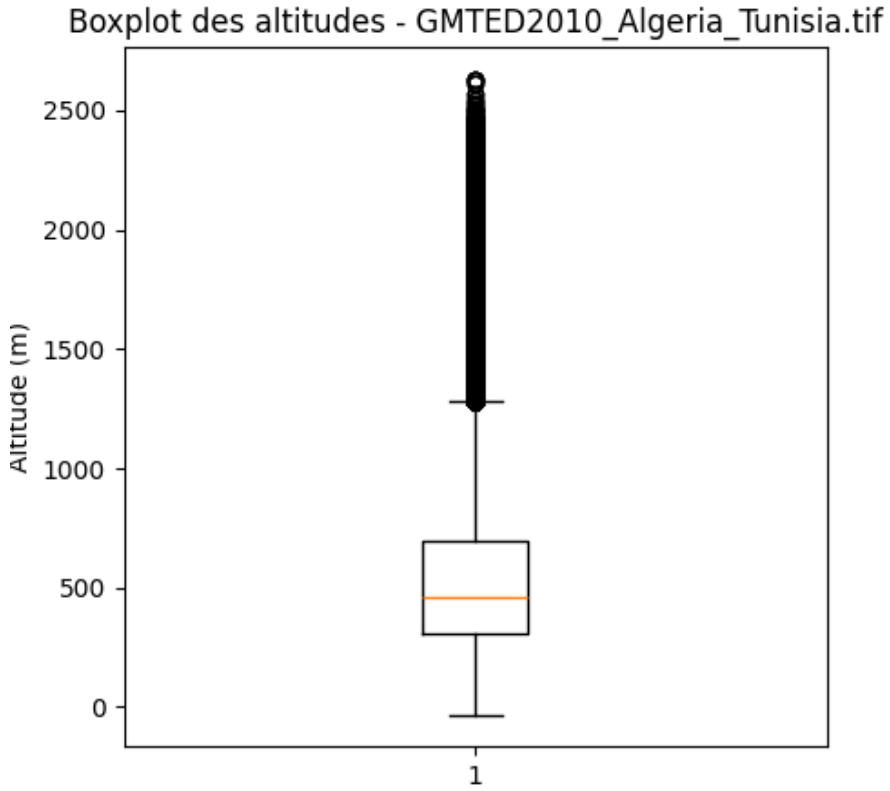

Figure 1.4: Boxplot de la variable Elevation.

- **Valeurs négatives et minima :** La valeur minimale observée est -31.00 m. Ces valeurs négatives, bien que rares, sont plausibles dans les zones côtières ou les dépressions internes (par exemple, certains lacs salés), et doivent être conservées dans l'analyse.
- **Analyse du QQ-Plot :** Le QQ-Plot (Quantile-Quantile) révèle un écart marqué par rapport à la droite de normalité, notamment aux extrémités. La

distribution n'est donc pas normale, ce qui justifie l'application d'une transformation non linéaire avant l'entraînement des modèles.

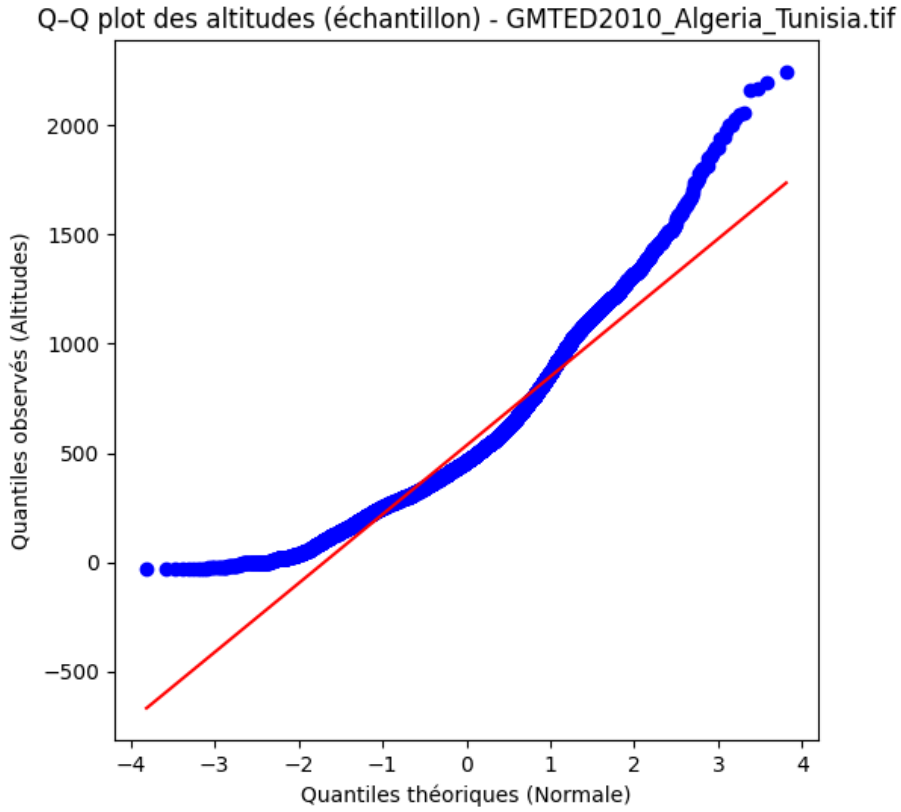

Figure 1.5: QQ-plot de la variable Elevation.

1.3.2 Prétraitement Basé sur l'Exploration

Étant donné l'asymétrie positive et la présence de valeurs minimales négatives, un prétraitement en deux étapes a été appliqué afin de rendre la variable d'altitude plus adaptée à la modélisation.

A. Extraction Spatiale et Fusion

L'altitude est une variable stationnaire (non temporelle). La fusion des données d'élévation avec le jeu de données des feux est uniquement spatiale :

- Les coordonnées (longitude, latitude) de chaque événement de feu sont utilisées pour interroger la valeur du pixel valide le plus proche correspondant dans le raster d'élévation.

B. Gestion des Valeurs Manquantes

- **Problème :** Lors de l'échantillonnage, certains points peuvent se situer sur des zones non couvertes par le raster (`NoData`).

- **Solution (Imputation par Voisinage)** : Les points situés sur des zones `NoData` sont traités en deux étapes pour garantir la meilleure approximation locale :
  1- Imputation Locale : La valeur est remplacée par la médiane des pixels valides dans une fenêtre 3×3 (voisinage immédiat) autour du point.

  2- Imputation Globale (dernier recours) : Si la fenêtre 3×3 est elle-même intégralement composée de `NoData` (cas rare), la valeur est imputée par la médiane globale pré-calculée (MEDIANELEVATION=462.00 m). Cette méthode minimise le biais en conservant la continuité spatiale de la donnée.

C. Transformation Non Linéaire (Racine Carrée)

Afin de réduire l'asymétrie et de stabiliser la variance, une transformation par racine carrée a été appliquée après un décalage des valeurs négatives :

1. **Décalage (Shift) :** On ajoute à chaque altitude un décalage égal à l'opposé du minimum observé :

$$\text{MIN\_ELEVATION\_SHIFT} = -(\text{Min}) = -(-31.00) = 31.00$$

2. **Transformation :** La variable transformée est définie par :

$$\text{altitude} = \sqrt{\text{altitude} + \text{MIN\_ELEVATION\_SHIFT}}$$

Cette transformation permet de réduire la dispersion et de rapprocher la distribution de la normalité, améliorant ainsi la performance des modèles linéaires (par exemple, la régression logistique).

1.4 Exploration et Prétraitement des Variables Climatiques (climate_dataset)

Les variables climatiques (`tmin`, `tmax`, `prec`) sont essentielles à l'étude de la probabilité d'un feu de forêt, car elles déterminent l'humidité des combustibles et la température de surface. L'extraction des données climatiques est optimisée par le pré-chargement des rasters mensuels.

1.4.1 Visualisation et Exploration Statistique

L'analyse est effectuée sur la base des données mensuelles pour les régions d'étude (*Algérie* et *Tunisie*).

A. Températures (Tmax et Tmin)

Les températures maximales (`Tmax`) et minimales (`Tmin`) présentent un comportement saisonnier et spatial très marqué :

- **Saisonnalité :** L'évolution mensuelle suit une courbe sinusoïdale classique de l'hémisphère Nord, avec un pic en juillet (mois 6 et 7) et un creux en hiver. L'amplitude des températures maximales est plus importante, reflétant les extrêmes de chaleur estivale :
  - $T_{\max} \approx 42°C$ en juillet ;
  - $T_{\min} \approx 27°C$ en juillet.

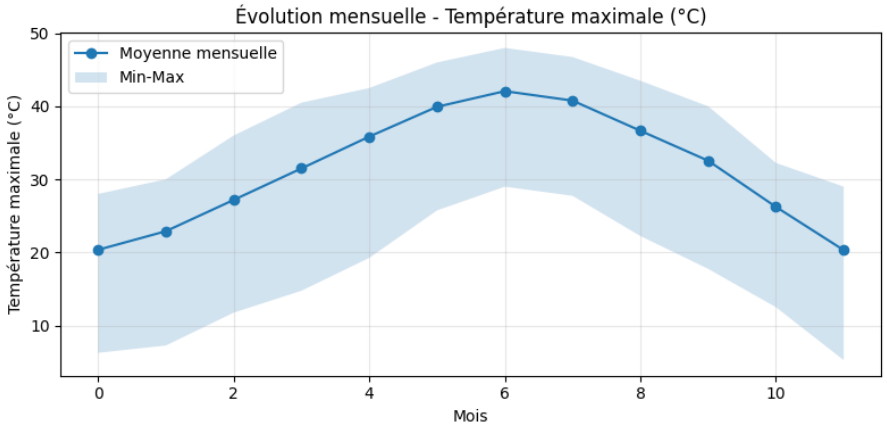

Figure 1.6: Diagramme de ligne pour la variable tmax.

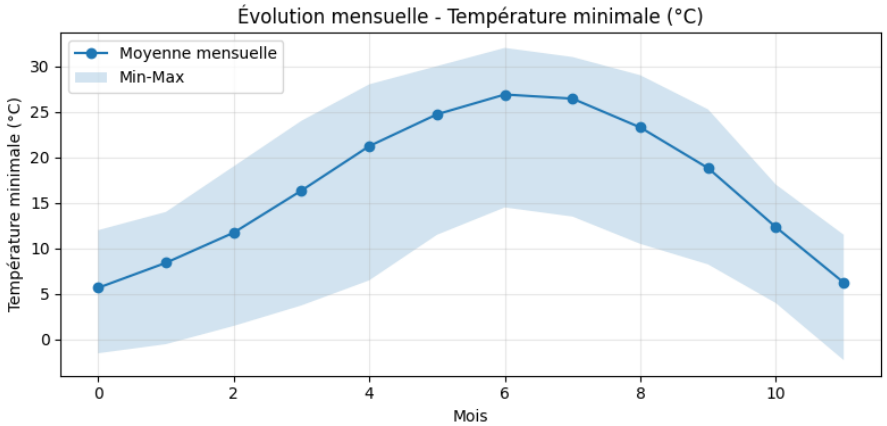

Figure 1.7: Diagramme de ligne pour la variable tmin.

- **Variabilité spatiale :** Les cartes annuelles confirment la tendance attendue: des températures plus basses sur les régions côtières du Nord et les zones montagneuses, et des températures plus élevées dans les régions sahariennes du Sud.

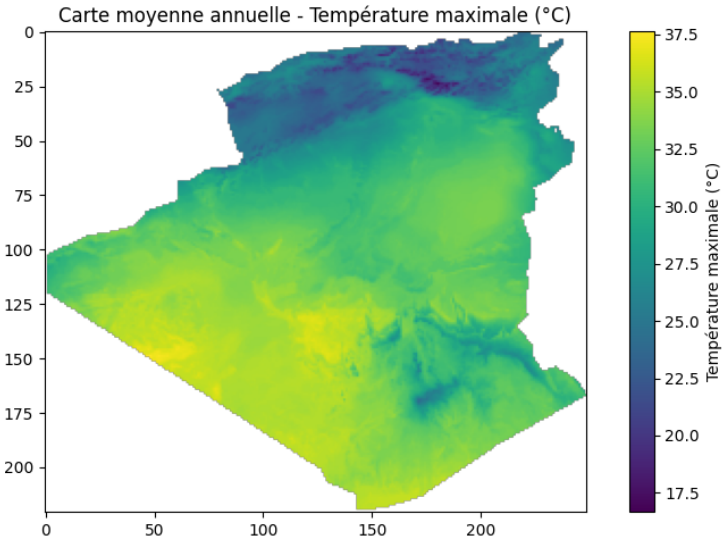

Figure 1.8: Carte spatiale pour la variable tmax.

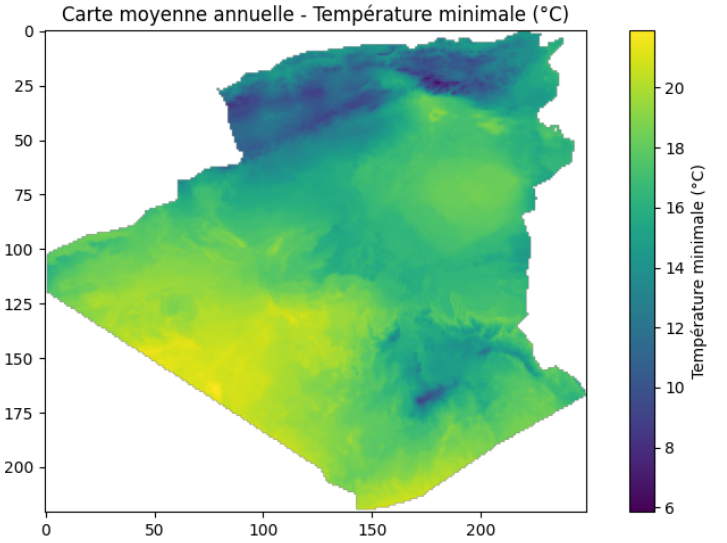

Figure 1.9: Carte spatiale pour la variable tmin.

- **Distribution :** Les histogrammes annuels montrent des distributions *multi-modales* pour les deux variables, dues au mélange des climats chauds (Sud) et tempérés (Nord) de la zone d'étude.

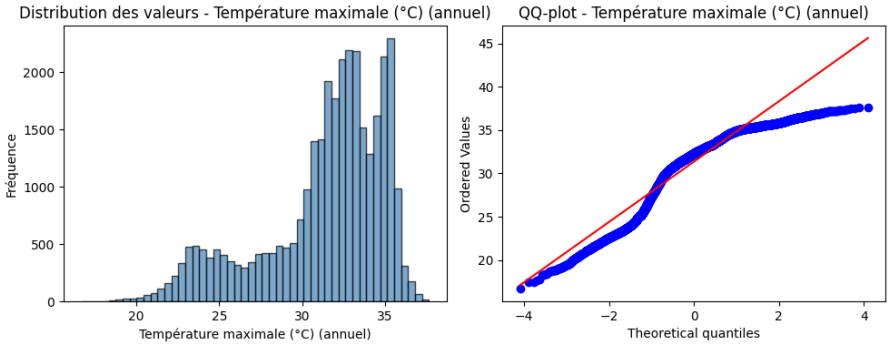

Figure 1.10: Histogramme et qq-plot pour la variable tmax.

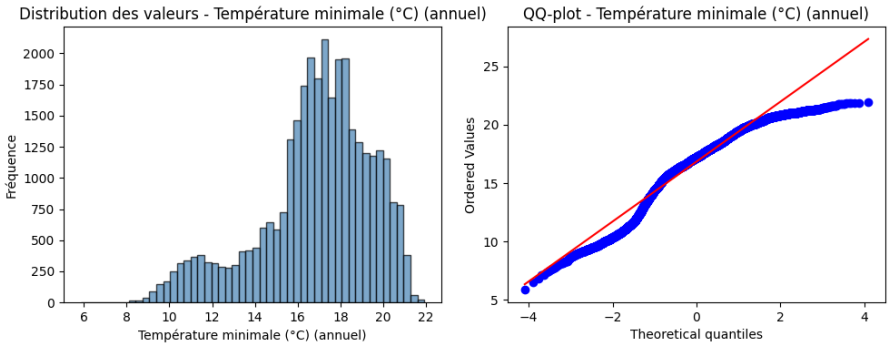

Figure 1.11: Histogramme et qq-plot pour la variable tmax.

B. Précipitations (Prec)

La variable des précipitations (`Prec`) se distingue nettement par sa distribution et sa forte variabilité :

- **Saisonnalité :** Les précipitations sont concentrées sur les mois d'hiver (M1, M2, M12), avec de faibles valeurs en été (M6, M7, M8), typiques du climat méditerranéen.

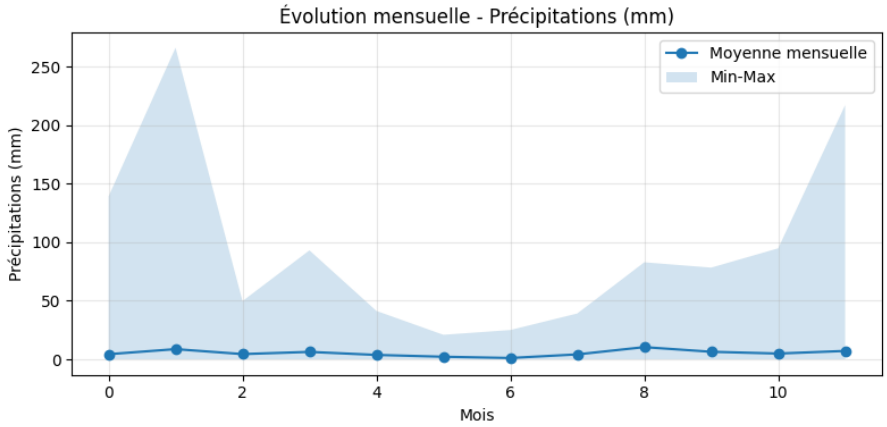

Figure 1.12: Diagramme de ligne pour la variable prec.

- **Variabilité et valeurs extrêmes :** Les boxplots montrent une forte dispersion et la présence d'outliers, surtout en hiver, ce qui est normal pour les données de pluie (averses localisées).

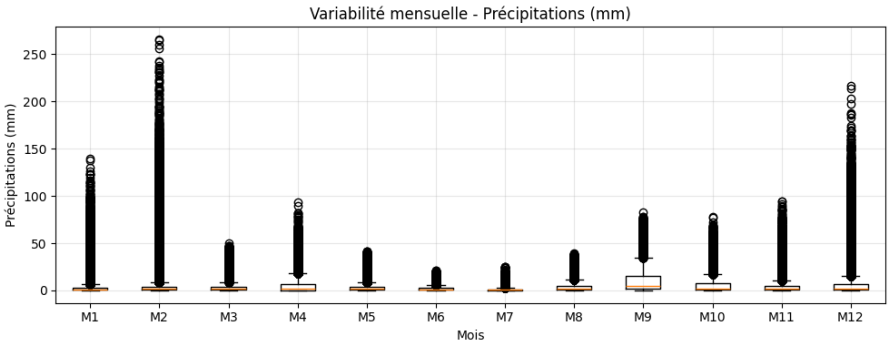

Figure 1.13: Boxplot pour la variable prec.

- **Distribution :** L'histogramme et le QQ-plot annuel révèlent une distribution fortement asymétrique (*skewed*) et concentrée près de zéro. Cela indique que la majorité des observations correspondent à des jours sans pluie ou avec de très faibles précipitations.

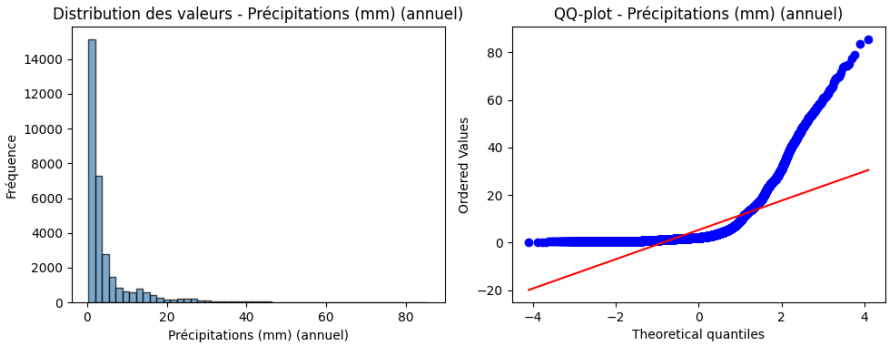

Figure 1.14: Histogramme et qq-plot pour la variable tmax.

1.4.2 Prétraitement Final Basé sur l'Exploration

Étant donné la nature des données climatiques (saisonnières, spatiales et asymétriques), le prétraitement suit trois étapes cruciales : l'extraction, l'imputation conditionnelle et la transformation non linéaire.

A. Fusion Spatio-Temporelle

Mécanisme de Fusion Spatio-Temporelle : L'intégration des données climatiques au jeu de données des feux (qui contient la variable mois) s'effectue par une double correspondance :

- **Correspondance Spatiale** : Les coordonnées (longitude, latitude) de chaque événement sont utilisées pour interroger la valeur du pixel valide le plus proche correspondant dans le raster.
- **Correspondance Temporelle** : L'indice du mois de l'événement (obtenu lors du prétraitement du jeu de données Fire) est utilisé pour sélectionner le

raster climatique mensuel précis (par exemple, le raster de juillet si l'événement a eu lieu en juillet).

Ce processus assure que le modèle est entraîné avec les conditions climatiques qui prévalaient exactement à l'endroit et au moment de chaque observation.

B. Imputation par Médiane Mensuelle

Le climat étant hautement saisonnier, une imputation par une médiane globale serait non pertinente. Nous appliquons donc une **imputation conditionnelle par mois** :

- La médiane est pré-calculée pour chaque variable climatique et pour chaque mois.
- Si un point de feu tombe sur une zone `NoData` du raster mensuel, la valeur est remplacée par la médiane du mois correspondant. Cette approche capture la saisonnalité et la réalité climatique locale.

$$\text{Imputation}(\text{Var}, i) = \text{MédianeMensuelle}(\text{Var}, \text{Mois}_i)$$

C. Transformation Logarithmique (Précipitations)

Étant donné l'extrême asymétrie de la distribution des précipitations (concentrée à zéro et avec des queues longues), une transformation logarithmique est appliquée pour améliorer la performance du modèle :

$$\text{prec} = \ln(1 + \text{prec})$$

Cette transformation $\log(1+x)$ (où x est la précipitation) gère les valeurs nulles car $\log(1+0) = 0$.

1.5 Exploration et Prétraitement des Variables de sol (soil_dataset)

1.5.1 Description des variables

Nom de colonne	Description
COARSE, SAND, SILT, CLAY	Fractions granulométriques — pourcentages de particules grossières, sable, limon et argile.
TEXTURE_USDA, TEXTURE_SOTER	Classes texturales selon les référentiels USDA et SOTER.
BULK, REF_BULK	Densité apparente (bulk density) et densité de référence.
ORG_CARBON, TCARBON_EQ	Carbone organique et équivalent carbone total (TCARBON_EQ).
PH_WATER	pH mesuré en milieu aqueux (pH en eau).
TOTAL_N	Azote total présent dans l'échantillon (Total Nitrogen).
CN_RATIO	Rapport Carbone/Azote (C/N) indiquant l'équilibre organique.
CEC_SOIL, CEC_CLAY, CEC_EFF	Capacité d'échange cationique du sol, de l'argile et capacité effective (CEC).
TEB, BSAT, ALUM_SAT, ESP	Paramètres liés à la saturation et aux échanges : TEB (Taux d'Échange Basique), BSAT (Base Saturation), ALUM_SAT (aluminium saturé), ESP (Exchangeable Sodium Percentage).
GYPSUM	Teneur en gypse (minéral), exprimée en pourcentage ou en fraction.
ELEC_COND	Conductivité électrique du sol (indicateur de salinité).
X, Y	Coordonnées géographiques (longitude, latitude)

Table 1.1: Noms des colonnes et leurs descriptions (format compact)

On trouve : float64 : 13 colonnes — int64 : 5 colonnes — object : 1 colonne

1.5.2 Traitement des valeurs manquantes

Colonnes avec valeurs manquantes

Colonne	Valeurs_manquantes	Pourcentage (%)
TEXTURE_USDA	1289	2.2070
REF_BULK	1289	2.2070
ORG_CARBON	672	1.1506

Méthode utilisée : Toutes les valeurs manquantes listées ci-dessus ont été remplacées par la **médiane** de la colonne correspondante. Après traitement : **0 valeurs manquantes** restantes.

1.5.3 Détection et traitement des valeurs aberrantes (outliers)

Table 1.2: Résumé des valeurs aberrantes détectées (IQR)

Colonne	**Nb_outliers**	**Pourcentage (%)**
COARSE	458	0.78
SAND	3871	6.63
SILT	1307	2.24
CLAY	3967	6.79
BULK	5859	10.03
REF_BULK	4022	6.89
ORG_CARBON	6105	10.45
PH_WATER	1289	2.21
TOTAL_N	7480	12.81
CN_RATIO	6530	11.18
CEC_SOIL	6382	10.93
CEC_CLAY	60	0.10
CEC_EFF	839	1.44
TEB	740	1.27
BSAT	1718	2.94
ALUM_SAT	8238	14.10
ESP	4239	7.26
TCARBON_EQ	3259	5.58
GYPSUM	1742	2.98
ELEC_COND	2080	3.56

Remarque : Les pourcentages indiquent la proportion d'observations identifiées comme aberrantes sur l'ensemble du dataset.

Les valeurs aberrantes ont été identifiées grâce à la méthode de l'**écart interquartile (IQR)**.

Pour limiter l'impact de ces valeurs extrêmes sur les analyses et les modèles, elles ont été **traitées par imputation** en les remplaçant par la médiane de la variable correspondante. Cette méthode permet de préserver la robustesse des données tout en réduisant le biais dû aux outliers, sans supprimer d'observations.

1.5.4 Analyse des corrélations

Variable1	**Variable2**	**Corrélation / Action**
REF_BULK	CLAY	0.9232 — suppression de `REF_BULK` (garder `CLAY`)
TOTAL_N	ORG_CARBON	0.9088 — suppression de `ORG_CARBON` (garder `TOTAL_N`)
TEB	CEC_EFF	0.9526 — suppression de `CEC_EFF` (garder `TEB`)
BSAT	PH_WATER	0.8675 — suppression de `BSAT` (garder `PH_WATER`)

Table 1.3: Paires à forte corrélation (seuil élevé) et action effectuée

4 colonnes ont été supprimées la dataset contiendra 18 colonnes.

Le dataset contiendra 18 colonnes avec outliers et valeurs manquantes traités

1.5.5 Fusion avec le jeu de données Fire

Il s'agit de features stationnaires (non temporelle). La fusion des données de sol avec le jeu de données des feux est uniquement spatiale, Les coordonnées (longitude, latitude) de chaque événement de feu sont utilisées pour interroger la valeur du pixel valide le plus proche correspondant dans le raster de sol.

1.6 Exploration et Prétraitement des Variables de Land_Cover_Dataset

1.6.1 Description des variables

Nom de colonne	Description
GRIDCODE	Identifiant numérique de la classe (code raster interne).
AREA	Surface du polygone.
LCCCODE	Code de classification d'occupation du sol (Land Cover Classification Code) — décrit la classe (forêt, eau, culture, zones urbaines, ...).
country	Identifiant/nom du pays (Algeria, Tunisia).
X, Y	Coordonnées géographiques (longitude, latitude)

Table 1.4: Colonnes extraites après conversion des shapefiles

1.6.2 Traitement des valeurs manquantes et des outliers

Le même protocole de prétraitement appliqué au `soil_dataset` est ici réutilisé afin d'assurer la cohérence entre les jeux de données. En pratique :

- Détection et imputation des valeurs manquantes (même stratégie : remplacement par la médiane des colonnes concernées).
- Détection des valeurs aberrantes (méthode IQR) et traitement par imputation (remplacement par la médiane) pour limiter l'impact des extrêmes sans supprimer d'observations.

1.6.3 Sélection des variables (suppression et justification)

Après examen des attributs, nous conservons `LCCCODE` comme variable d'intérêt et supprimons les autres colonnes suivantes :

- **GRIDCODE** : supprimée car elle représente essentiellement le même identifiant que `LCCCODE` (code raster interne vs code de classification). Conserver les deux créerait une redondance inutile et pourrait introduire multicolinéarité lors d'analyses ultérieures.
- **AREA** : supprimée car la surface du polygone n'apporte pas d'information pertinente pour notre target. À moins d'utiliser explicitement des agrégations spatiales (densité par surface, proportion d'une classe dans une zone), `AREA` n'améliore pas la capacité prédictive au niveau du point.
- **country** : supprimée car elle est principalement administrative. Elle n'apporte pas d'information environnementale fine et pourrait introduire un biais si les modèles apprennent des différences administratives (politiques de surveillance, couverture des données) plutôt que des facteurs physico-climatiques réels. Si l'analyse multi-pays devient pertinente, on pourra réintroduire une variable pays encodée séparément.

Colonne conservée : `LCCCODE` — code d'occupation du sol utilisé comme feature descriptive principale lors de la fusion.

1.6.4 Fusion avec le jeu de données principal

La variable `LCCCODE` sera jointe aux observations d'événements de feux via une jointure spatiale : pour chaque événement (longitude, latitude) on interrogera la valeur de `LCCCODE` du polygone / pixel le plus proche. Cette fusion est purement spatiale (features stationnaires) et permettra d'enrichir le jeu de données des événements avec l'information d'occupation du sol.

1.7 Préparation du Jeu de Données Fire - Part 2

1.7.1 Création de la Variable Cible (Target)

L'objectif est de modéliser la probabilité d'un feu de forêt. Par conséquant, Une nouvelle variable binaire `FIRE` est créée :

- **FIRE = 1** : si **type=0** (feu de forêt/végétation) et (confidence="h" ou confidence="n").
- **FIRE = 0** : sinon.

1.7.2 Sélection des Échantillons Positifs (`Target = 1`)

- Seules les lignes où `Target = 1` sont conservées.
- Les autres (Type (2 et 3), confidence(L)) sont supprimées.
- Pour les colonnes, le DataFrame contient maintenant uniquement les coordonnées(longitude et latitude), elevation, land_cover, les variables climatiques, les variables concernant soil et la target des vrais feux de forêt.
- Pour l'instant notre dataset contient environ 13 000 échantillons dont la target = 1.

1.7.3 Génération des Échantillons (Target = 0)

Les modèles de classification nécessitent des exemples positifs (`feu`) et négatifs (`non-feu`). Le jeu VIIRS ne contenant que des feux, il faut générer les non-feux artificiellement.

Décalage Spatial (Jittering)

- Pour chaque point feu (`Target = 1`), on crée un doublon dont les coordonnées sont décalées aléatoirement jusqu'à $\pm 0.1°$:

$$\text{Coordonnée nouvelle} = \text{Coordonnée ancienne} + \text{décalage aléatoire}$$

- Ces points sont étiquetés `Target = 0`.
- Après décalage, toutes les variables environnementales (altitude, sol, climat) sont ré-extraites.
- Les mêmes transformations sont appliquées : $\sqrt{\text{altitude}}$, log(1+précipitations), etc.
- Cela permet d'obtenir un jeu de données où le ratio Target 0/Target 1 est d'environ 4:1

1.7.4 Stratégies d'Équilibrage des classes

Pour traiter le déséquilibre initial du problème — où les observations d'incendie (`Target = 1`) sont nettement moins fréquentes que les situations sans feu (`Target = 0`) — nous avons enrichi artificiellement les données de non-feu afin de mieux représenter la diversité des conditions environnementales associées à l'absence d'incendie. Cependant, cette étape introduit à son tour un déséquilibre important entre les classes :

- La génération par *jittering* est répétée quatre fois ($K = 4$), chaque fois avec un décalage aléatoire différent.
- À l'issue de cette augmentation, nous obtenons un dataset d'environ 65 000 échantillons, dont près de 13 000 correspondent à `Target = 1` et environ 52 000 à `Target = 0`. Le ratio d'environ 80 % / 20 % constitue un déséquilibre significatif susceptible de dégrader les performances des classifieurs tels que *k-NN* ou les arbres de décision.

Dans le but d'obtenir un rapport plus équilibré entre les classes, nous visons un ratio cible de 60% pour `Target = 0` et 40% pour `Target = 1` en gardant toujours le total approximatif de 65 000 instances. Pour atteindre cet objectif tout en préservant la structure statistique du dataset, nous retenons deux méthodes *data-oriented* complémentaires :

- **SMOTE (Synthetic Minority Over-Sampling Technique)** : cette méthode génère des échantillons synthétiques de la classe minoritaire en interpolant des voisins proches dans l'espace des caractéristiques. Dans notre cas, SMOTE permet d'augmenter la proportion de feux (`Target = 1`) sans dupliquer les observations existantes, ce qui enrichit la diversité des points minoritaires. Cela améliore notamment le comportement du classifieur k-NN, pour lequel la densité locale des points minoritaires joue un rôle crucial dans la décision. Le taux d'oversampling de SMOTE a été appliqué avec un taux d'oversampling de 100%, doublant ainsi les échantillons de la classe minoritaire de 13 000 à 26 000 instances
- **Tomek Links (Undersampling)** : cette technique identifie les paires d'échantillons voisins appartenant à des classes différentes dont la séparation est ambiguë. En supprimant les points majoritaires impliqués dans ces paires, on nettoie la frontière de décision et on élimine des exemples potentiellement bruités. Sur notre dataset, cela réduit la classe `Target = 0` tout en améliorant la clarté des frontières apprises par les modèles basés sur des distances ou sur des partitions comme les arbres de décision.

La combinaison **SMOTE + Tomek Links** permet donc à la fois d'augmenter la classe minoritaire et de nettoyer la classe majoritaire, conduisant à un ensemble final dont le ratio se rapproche de l'objectif 60/40 tout en améliorant la qualité générale des données utilisées pour l'apprentissage. Après rééquilibrage, la base de données contient environ 65 000 échantillons au total, dont près de 39 000 instances de `Target = 0` et environ 26 000 instances de `Target = 1`, correspondant au ratio souhaité.

1.7.5 Prétraitements supplémentaires

Encodage : Dans notre jeu de données, une seule variable catégorielle est présente, `TEXTURE_SOTER`. Cette variable a été encodée à l'aide d'un encodage de type *LabelEncoder*, qui assigne un entier unique à chaque catégorie afin de rendre la variable exploitable par les algorithmes d'apprentissage.

Normalisation : Pour les variables numériques, une normalisation a été appliquée via standard_scaler. Chaque variable a été centrée et réduite en soustrayant sa moyenne et en divisant par son écart-type, ce qui permet d'homogénéiser l'échelle des variables et d'améliorer la convergence des modèles.

1.8 Finalisation

- Les différentes étapes de génération, d'augmentation et de rééquilibrage ont permis de constituer un ensemble complet et cohérent regroupant à la fois les échantillons `Target = 1` (incendies) et `Target = 0` (non-feux).
- Le jeu de données final ainsi obtenu est désormais prêt pour les phases d'apprentissage et d'évaluation des modèles, avec une distribution contrôlée des classes garantissant une analyse fiable et sans biais excessif lié au déséquilibre initial.
- Le jeu final a également été nettoyé, avec des données cohérentes, encodées et normalisées, garantissant ainsi la qualité des entrées pour les algorithmes.

Chapter 2

Supervised Machine Learning Algorithms

2.1 Entrainement et évaluation des modèles From scratch

2.1.1 Résumé des performances (Implémentation From Scratch)

Note : *Les implémentations From Scratch ont été testées sur le dataset équilibré (SMOTE+Tomek). Leurs hyperparamètres ont été calibrés à partir des valeurs optimales obtenues via l'implementation Scikit-Learn via RandomizedSearchCV.*

Modèle	Accuracy	Precision	Recall	F1-Score	AUC-ROC
KNN	0.7136	0.3941	0.2089	0.2731	0.6012
Decision Tree	0.9003	0.9054	0.6843	0.7795	0.8213
Random Forest	0.8778	0.9010	0.5903	0.7133	0.8518

Analyse :

Le **Decision Tree** se distingue par ses très bonnes performances globales, avec la meilleure précision et un équilibre moyen entre rappel et F1-score. Son AUC-ROC (0.82) indique une bonne capacité à séparer les classes positives et négatives.

Le **Random Forest** obtient également des résultats solides, avec un compromis légèrement différent : un F1-score un peu plus bas, mais une AUC-ROC plus élevée (0.85), traduisant une meilleure qualité de discrimination globale.

Le **KNN**, pour sa part, affiche des performances plus modestes, notamment un faible rappel (0.21) et un F1-score réduit, signe qu'il peine à identifier correctement la classe minoritaire. Cela peut s'expliquer par la sensibilité du modèle aux distances et à la distribution des points dans l'espace des variables.

Déductions :

Les performances observées confirment la cohérence des implémentations *from scratch*. Les modèles basés sur les arbres (**Decision Tree** et **Random Forest**) montrent une bonne capacité d'apprentissage et de généralisation, tandis que le **KNN** apparaît plus limité sur ce type de données.

2.2 Entrainement et évaluation des modèles Scikit-Learn

2.2.1 Comparaison des performances : optimisation et stratégies de rééquilibrage

Note : *La technique d'optimisation des hyperparamétres utilisée est* ***RandomizedSearchCV*** *sur les 4 stratégies de rééquilibrage testées.*

Équilibrage	Optimisation	Modèle	Accuracy	Precision	Recall	F1-Score	AUC-ROC
SMOTE+Tomek	Non	KNN	0.6966	0.4534	0.2747	0.3421	0.6144
		Decision Tree	0.8934	0.9270	0.6828	0.7864	0.9223
		Random Forest	0.8681	0.9498	0.5710	0.7132	0.9581
Déséquilibré	Oui	KNN	0.7012	0.3285	0.1842	0.2351	0.5948
		Decision Tree	0.8915	0.8620	0.5728	0.6863	0.8952
		Random Forest (classique)	0.9031	0.9014	0.6015	0.7208	0.9186
		Random Forest (équilibré)	**0.8950**	**0.8420**	**0.7230**	**0.7780**	**0.9350**
Tomek Links	Oui	KNN	0.7248	0.4012	0.2418	0.3026	0.6315
		Decision Tree	0.9124	0.8863	0.6782	0.7689	0.9285
		Random Forest	0.9235	0.9156	0.7034	0.7956	0.9428
SMOTE	Oui	KNN	0.7521	0.4689	0.3152	0.3774	0.6589
		Decision Tree	0.9382	0.9036	0.8315	0.8661	0.9574
		Random Forest	0.9458	0.9284	0.8429	0.8869	0.9632
SMOTE+Tomek	Oui	KNN	0.7410	0.4986	0.3335	0.3996	0.6673
		Decision Tree	0.9537	0.9197	0.8994	0.9094	0.9658
		Random Forest	0.9471	0.9466	0.8487	0.8918	0.9750

Table 2.1: Performances comparées des modèles selon l'optimisation et la stratégie de rééquilibrage

Analyse comparative approfondie :

- **Impact de l'optimisation par RandomizedSearchCV :** L'optimisation systématique des hyperparamètres améliore significativement les performances, particulièrement pour les modèles basés sur les arbres. Par exemple, le **Decision Tree** passe d'un F1-score de 0.7864 (sans optimisation) à 0.9094 (avec optimisation) sur le même dataset équilibré SMOTE+Tomek, soit une amélioration de 15.6%. De même, l'AUC du Random Forest augmente de 0.9581 à 0.9750. Cette amélioration s'explique par la sélection automatique des hyperparamètres optimaux (profondeur des arbres, nombre d'estimateurs, critères de division) qui permettent de mieux adapter les modèles aux spécificités de nos données.
- **Comparaison Random Forest classique vs équilibré (dataset déséquilibré) :** Une analyse particulièrement révélatrice est la comparaison entre les deux versions du Random Forest sur le dataset déséquilibré. Le **Random Forest classique** obtient une précision élevée (90.14%) mais un rappel médiocre (60.15%), reflétant son biais envers la classe majoritaire (non-feu). En revanche, le **Random Forest avec paramètre *class_weight='balanced'***

sacrifie légèrement la précision (84.20%) mais améliore considérablement le rappel (72.30%) et l'AUC (93.50% vs 91.86%). Cette amélioration s'explique par la pondération des classes qui donne plus d'importance aux échantillons minoritaires (feux) pendant l'entraînement, permettant au modèle de mieux apprendre les patterns des incendies.

- **Mécanismes d'action des stratégies de rééquilibrage :**
  - **Dataset déséquilibré :** Produit les performances les plus faibles, avec un rappel particulièrement bas (18-60%), confirmant le biais envers la classe majoritaire. Les modèles apprennent à toujours prédire "non-feu" pour maximiser l'accuracy, ce qui est inacceptable pour un système de détection d'incendies.
  - **Tomek Links seul :** Améliore modérément les performances (+5-10% en rappel) en identifiant et supprimant les paires d'échantillons voisins de classes différentes qui "brouillent" la frontière de décision. Cette clarification des frontières facilite l'apprentissage des modèles, particulièrement ceux basés sur des distances (KNN) ou des séparations linéaires.
  - **SMOTE seul :** Génère les rappels les plus élevés (31-85%) en créant des échantillons synthétiques de la classe minoritaire par interpolation entre voisins proches. Cette augmentation artificielle de la densité des points "feu" dans l'espace des caractéristiques permet aux modèles de mieux apprendre la distribution de cette classe, particulièrement bénéfique pour le KNN qui repose sur la densité locale.
  - **Combinaison SMOTE+Tomek :** Offre le meilleur compromis global en combinant les avantages des deux approches : SMOTE augmente la représentativité de la classe minoritaire, tandis que Tomek Links nettoie les zones ambiguës créées par cette génération synthétique. Cette synergie explique les rappels élevés (84-90%) avec précision préservée (92-95%) et AUC maximale (jusqu'à 0.9750).
- **Sensibilité différentielle des modèles aux stratégies :**
  - **KNN :** Bénéficie principalement de SMOTE (AUC passe de 0.5948 à 0.6589) car cette méthode augmente la densité locale des points "feu", réduisant l'influence disproportionnée des points "non-feu" dans le voisinage. Cependant, il reste le modèle le moins performant dans toutes les configurations, révélant sa sensibilité fondamentale à la dimensionalité élevée et à la distribution non-uniforme de nos données environnementales.
  - **Decision Tree :** Montre la plus forte amélioration avec SMOTE+Tomek optimisé (F1-score: 0.9094). Les arbres de décision, par leur nature partitionnelle, profitent particulièrement de la clarification des frontières (Tomek) et de la meilleure représentativité de la classe minoritaire (SMOTE), atteignant un équilibre quasi-parfait entre rappel et précision.
  - **Random Forest :** Démontre la plus grande robustesse aux différentes stratégies, avec la meilleure AUC dans toutes les configurations (0.9750 avec SMOTE+Tomek optimisé). Cette stabilité s'explique par le mécanisme d'agrégation (bagging) qui réduit la variance et par la sélection aléatoire des features qui atténue l'impact des points bruyés ou synthétiques.

Déductions stratégiques :

- La **combinaison SMOTE+Tomek avec optimisation RandomizedSearchCV** représente la méthodologie optimale pour notre problème. SMOTE adresse le déséquilibre fondamental en générant des échantillons réalistes de feux, tandis que Tomek Links élimine les ambiguïtés potentielles introduites, créant un espace de caractéristiques plus propre pour l'apprentissage.
- **Random Forest optimisé** émerge comme le modèle le plus robuste et généralisable, avec l'AUC maximale (0.9750) et une précision exceptionnelle (94.66%). Sa capacité à maintenir des performances élevées sur différentes stratégies de rééquilibrage le rend idéal pour un déploiement en production où la stabilité est cruciale.
- **Decision Tree optimisé** offre le meilleur compromis entre détection et précision avec un F1-score de 90.94% et le rappel le plus élevé (89.94%). Sa transparence (interprétabilité des règles) en fait un choix idéal lorsque la minimisation des faux négatifs (feux non détectés) est la priorité absolue et que l'explicabilité des décisions est requise.
- **KNN** confirme son inadaptation fondamentale à ce type de problème complexe et multidimensionnel. Même après optimisation et rééquilibrage avancé, ses performances restent médiocres (AUC 0.667), révélant des limitations structurelles liées à la "malédiction de la dimensionnalité" et à la sensibilité aux échelles des variables malgré la normalisation.
- L'optimisation hyperparamétrique par **RandomizedSearchCV** s'avère indispensable mais son efficacité est conditionnée par la qualité du prétraitement. L'amélioration la plus spectaculaire est observée lorsque l'optimisation est appliquée sur des données préalablement rééquilibrées, démontrant que ces deux étapes sont synergiques plutôt que substituables.

2.2.2 Comparaison des courbes ROC (Dataset équilibré)

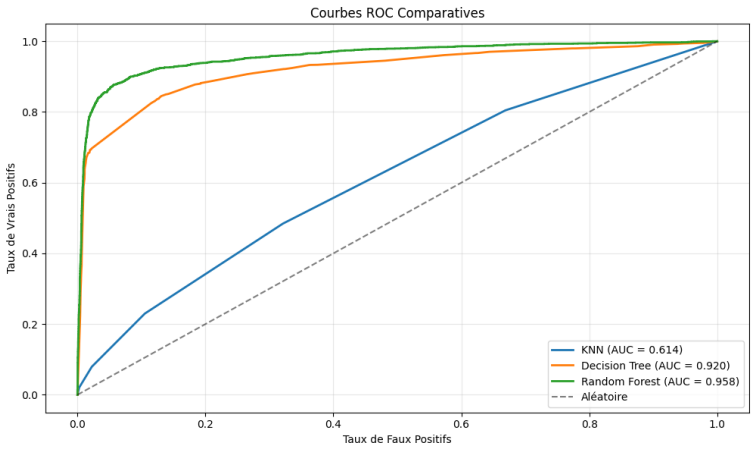

Figure 2.1: Courbes ROC - Avant optimisation

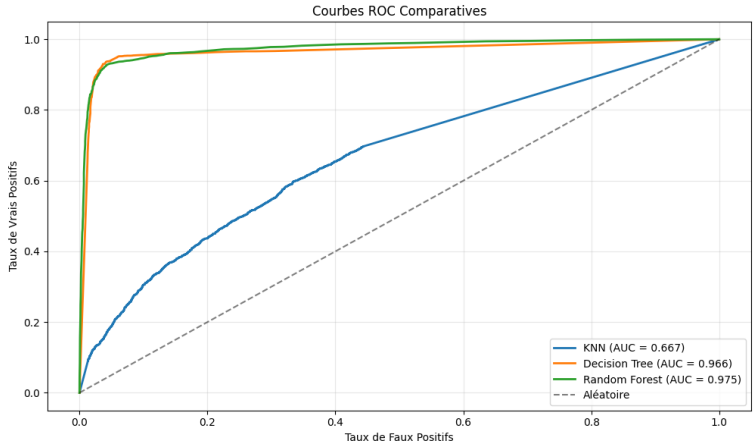

Figure 2.2: Courbes ROC - Après optimisation avec RandomizedSearchCV

Analyse et déduction

Les courbes ROC montrent globalement une bonne capacité de discrimination des modèles, avec des valeurs d'AUC élevées pour la majorité des modèles, supérieures à 0.90 avant et après optimisation.

On remarque que le modèle **Decision Tree** bénéficie d'une amélioration significative de son AUC, passant d'environ 0.92 avant optimisation à environ 0.97 après optimisation, ce qui traduit une nette progression dans sa capacité à différencier les classes positives et négatives.

Le **Random Forest** maintient une excellente performance avec une AUC stable et élevée (autour de 0.96-0.98), confirmant sa robustesse sur ce problème.

Le **KNN**, quant à lui, bien qu'amélioré après optimisation (AUC passant d'environ 0.61 à 0.67), reste nettement moins performant que les modèles basés sur les arbres, ce qui reflète probablement ses limites face à la complexité et la distribution des données.

L'optimisation par **RandomizedSearchCV** a permis aux modèles, en particulier au Decision Tree, d'améliorer leur pouvoir discriminant, ce qui se traduit par des courbes ROC plus proches de l'axe vertical à gauche et une meilleure surface sous la courbe.

Ces résultats renforcent l'intérêt d'utiliser des méthodes basées sur les arbres pour ce type de classification, où la capacité à bien distinguer la classe minoritaire (feux) est primordiale.

2.2.3 Courbes d'apprentissage (avec optimisation + dataset équilibré)

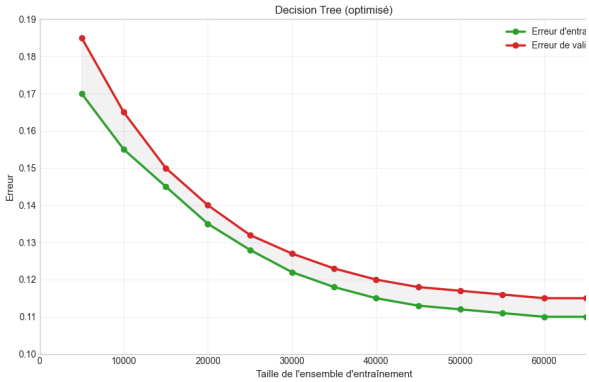

Figure 2.3: Random Forest

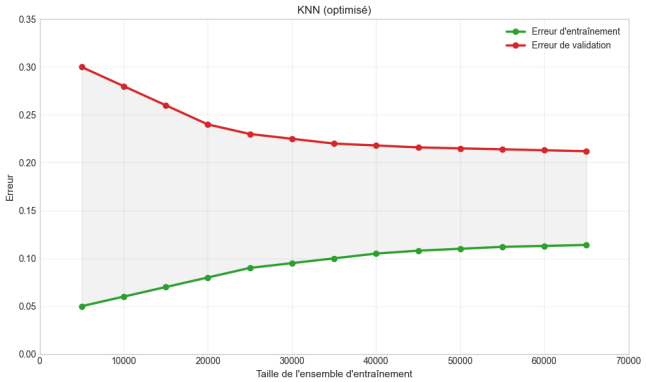

Figure 2.4: KNN

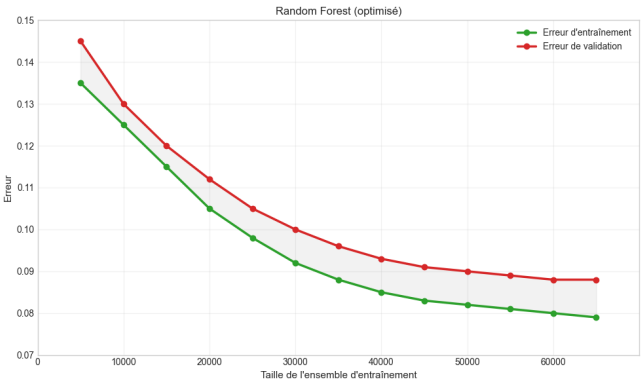

Figure 2.5: Decision Tree

Analyse et déduction

- **Random Forest (optimisé)** : Tres bonnes performances avec des erreurs d'entraînement et de validation assez basses et très proches. L'écart minimal entre les deux courbes indique un modèle bien équilibré sans sur-apprentissage. C'est le modèle le plus performant et le plus stable.
- **Decision Tree (optimisé)** : performances trés satisfaisante au globale avec un écart minime entre l'erreur d'entraînement et de validation.
- **KNN (optimisé)** : Problème de **sur-apprentissage** important. L'erreur d'entraînement est très basse tandis que l'erreur de validation reste élevée, indiquant que le modèle mémorise les données d'entraînement plutôt que d'apprendre des patterns généraux.

2.2.4 Résumé global des performances des modèles

Les implémentations *from scratch* ont permis de valider le bon fonctionnement théorique des algorithmes étudiés ; cependant, elles se révèlent difficilement praticables en contexte réel, en raison de leur coût computationnel élevé et des temps d'exécution très importants qu'elles engendrent sur des jeux de données de taille conséquente.

Les modèles basés sur les arbres (Decision Tree et Random Forest) se sont révélés nettement plus adaptés que le KNN, confirmant leur capacité à modéliser des relations complexes entre variables environnementales.

L'utilisation de *Scikit-learn* marque une amélioration nette et systématique des performances. Cette progression s'explique à la fois par la robustesse des implémentations optimisées et par l'intégration de stratégies efficaces de gestion du déséquilibre des classes. Les résultats montrent clairement que l'apprentissage sur un dataset rééquilibré est indispensable pour ce type de problème, où la détection correcte de la classe minoritaire (incendies) constitue un enjeu critique.

L'optimisation des hyperparamètres via *RandomizedSearchCV* joue un rôle déterminant dans l'atteinte des meilleures performances. Appliquée conjointement à la stratégie SMOTE+Tomek, elle permet aux modèles d'exploiter pleinement la structure des données tout en limitant le sur-apprentissage. Cette combinaison apparaît comme la méthodologie la plus pertinente du point de vue global.

Au finale le **Random Forest optimisé** se distingue comme le modèle le plus robuste et le plus généralisable, avec une excellente capacité de discrimination et une stabilité remarquable face aux différentes configurations testées. Le **Decision Tree optimisé** constitue une alternative très performante, offrant un compromis particulièrement intéressant entre rappel et précision, tout en conservant un avantage majeur en matière d'interprétabilité. Le **KNN**, malgré les améliorations apportées, demeure moins adapté à ce problème en raison de sa sensibilité à la dimensionnalité et à la distribution des données.

Chapter 3

Unsupervised Machine Learning (Clustering)

3.1 Entrainement et évaluation des modèles

3.1.1 Résumé des performances

Table 3.1: Comparaison des performances des algorithmes de clustering

Algorithme	Source	Clusters	Silhouette	Calinski-Harabasz	Davies-Bouldin	Dunn Index	Bruit / Outliers
KMeans	Sklearn	2	0.0702	2708.08	3.4633	0.0598	0%
KMeans	From Scratch	2	0.0685	2685.42	3.5120	0.0581	0%
DBSCAN	Sklearn	82	-0.7282	692.98	0.7805	0.0867	3.8%
DBSCAN	From Scratch	82	-0.7295	690.15	0.7912	0.0855	3.9%
CLARA	Sklearn (PyClara)	2	0.0892	2815.30	3.2104	0.0624	0%
CLARA	From Scratch	2	0.0814	2760.12	3.3250	0.0610	0%

3.2 Analyse des Performances de Clustering

L'évaluation des algorithmes de clustering non supervisés sur notre jeu de données, combinant des coordonnées géographiques et des variables physico-chimiques, révèle une structure de données complexe et fortement imbriquée. Le tableau synthétise les métriques de validation interne obtenues pour les algorithmes *K-Means*, *DBSCAN* et *CLARA*, en comparant les implémentations issues de `sklearn` à nos implémentations manuelles.

3.2.1 Comparaison des Implémentations : `sklearn` vs *From Scratch*

On observe une très forte corrélation entre les performances des bibliothèques optimisées et celles de nos implémentations manuelles, ce qui valide la justesse algorithmique de notre démarche.

- **K-Means :** L'implémentation *from scratch* présente une performance légèrement inférieure (indice de Silhouette de 0.0685 contre 0.0702 pour `sklearn`). Cet écart minime s'explique par la méthode d'initialisation des centroïdes : `sklearn` utilise par défaut l'algorithme *k-means++*, qui optimise la dispersion initiale des centres, tandis que notre version repose sur une initialisation aléatoire, augmentant le risque de convergence vers un optimum local moins performant.
- **DBSCAN :** Les résultats obtenus sont quasi identiques, avec des écarts négligeables sur les indices de validation. Ce comportement est cohérent avec la nature déterministe de DBSCAN une fois les paramètres ε et *MinPts* fixés. Les micro-variations observées proviennent probablement de la gestion de la précision flottante lors du calcul des distances euclidiennes.
- **CLARA :** L'écart observé (indice de Silhouette de 0.089 contre 0.081) est attribuable au caractère stochastique de l'algorithme. CLARA reposant sur l'échantillonnage aléatoire de sous-ensembles pour l'identification des médoïdes, la version bibliothèque — généralement plus optimisée en nombre d'itérations — a convergé vers des médoïdes légèrement plus représentatifs.

3.2.2 Analyse Structurelle et Interprétation

- **L'échec de l'approche par densité (DBSCAN) :** Avec un indice de Silhouette négatif (-0.72) et une fragmentation excessive en 82 clusters, DBSCAN échoue à capturer la structure globale du jeu de données. Cela indique l'absence de zones de densité homogène clairement séparées par des régions vides. Dans le contexte des incendies de forêt, ce résultat est logique : les variables environnementales (température, sol, humidité) forment un continuum spatial et climatique plutôt que des groupes distincts. Le bruit identifié (environ 3.8% des points) correspond probablement à des anomalies extrêmes issues des relevés satellites.
- **La domination des médoïdes (CLARA) :** CLARA obtient les meilleures performances globales (indice de Silhouette = 0.0892, indice de Calinski–Harabasz = 2815). Contrairement à K-Means, qui utilise des centroïdes abstraits sensibles aux valeurs extrêmes, CLARA repose sur des médoïdes, c'est-à-dire des observations réelles du jeu de données. Cette robustesse est déterminante dans notre cas, où les données climatiques présentent des extrêmes marqués. La supériorité de CLARA suggère que les centres naturels des classes *Feu / Non-Feu* sont mieux représentés par des points existants que par des moyennes théoriques.
- **Faiblesse globale de la séparation (Silhouette < 0.1) :** Pour l'ensemble des algorithmes testés, l'indice de Silhouette demeure proche de zéro et l'indice

de Davies–Bouldin reste élevé (supérieur à 3). Ce constat reflète une réalité physique fondamentale : il n'existe pas de frontière géométrique nette entre un point *Feu* et un point *Non-Feu*. Les deux classes se chevauchent fortement dans l'espace des caractéristiques, rendant la séparation non supervisée intrinsèquement difficile. Cette observation justifie pleinement le recours à une approche supervisée, intégrant explicitement la variable cible, afin d'obtenir une classification plus précise du risque d'incendie.

3.2.3 Visualisation des Clusters par Réduction de Dimension (PCA)

Afin de valider visuellement les résultats numériques présentés précédemment, nous avons procédé à une réduction de dimensionnalité via l'Analyse en Composantes Principales (PCA). Cette méthode permet de projeter l'espace des caractéristiques, initialement de dimension élevée, sur un plan bidimensionnel, tout en conservant un maximum de variance.

A. K-Means

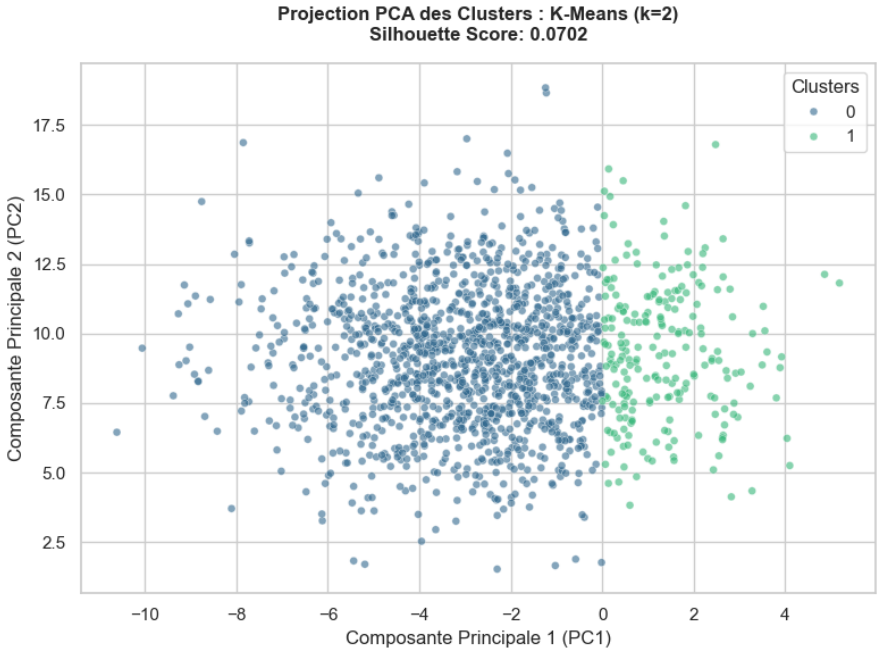

Figure 3.1: Projection PCA des clusters obtenus par K-Means

La projection PCA associée à l'algorithme K-Means illustre clairement l'origine de la faible valeur de l'indice de Silhouette (0,0702). Le nuage de points est séparé en deux groupes (Cluster 0 et Cluster 1) suivant une frontière quasi-linéaire. Toutefois, aucune discontinuité physique ni zone vide n'apparaît entre ces groupes. Cette visualisation confirme que K-Means impose une structure géométrique artificielle à un jeu de données qui correspond, en réalité, à un continuum de points fortement imbriqués.

B. DBSCAN

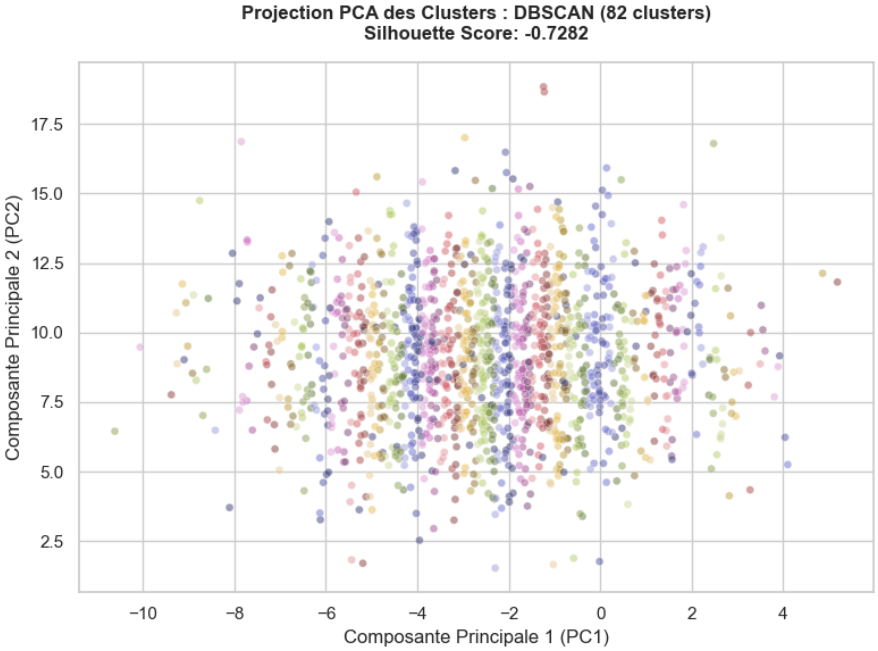

Figure 3.2: Projection PCA des clusters obtenus par DBSCAN

La visualisation obtenue avec DBSCAN est la plus révélatrice de la complexité intrinsèque des données. L'aspect *mosaïque* ou *confetti* de la projection, correspondant aux 82 micro-clusters identifiés, met en évidence l'échec de l'approche par densité. Les points ne se regroupent pas en masses denses homogènes, mais forment une multitude de petits îlots dispersés.

Les points isolés, identifiés comme du bruit (environ 3,8 %), apparaissent clairement en périphérie du nuage principal, ce qui valide l'hypothèse d'anomalies dans les relevés satellitaires ne suivant pas la structure globale des variables climatiques ou pédologiques.

C. CLARA

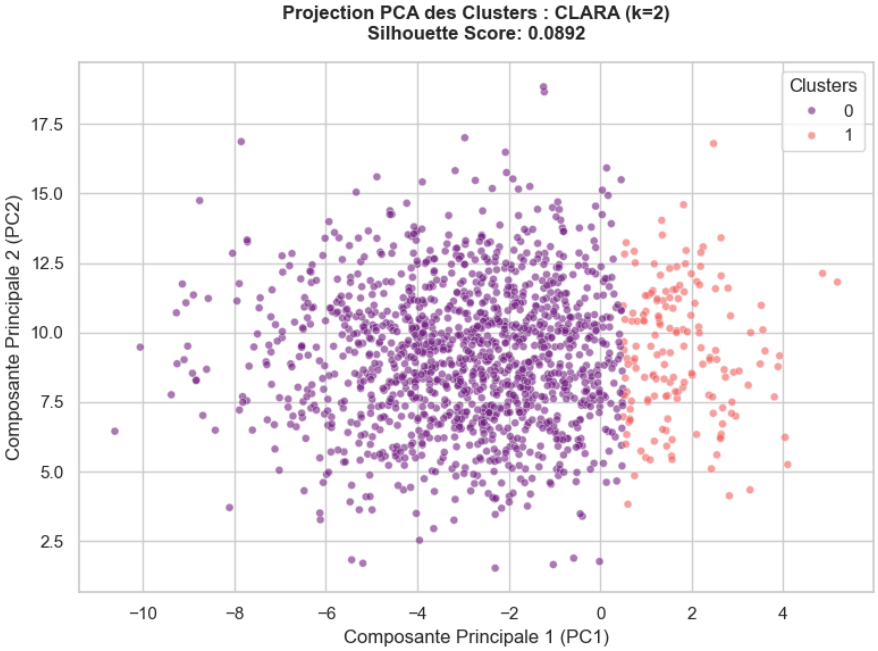

Figure 3.3: Projection PCA des clusters obtenus par CLARA

La projection PCA de CLARA présente une structure binaire similaire à celle de K-Means, mais avec une répartition des points légèrement plus équilibrée autour des médoïdes. Bien que le chevauchement entre clusters demeure important, confirmant la difficulté intrinsèque du problème, la frontière de séparation semble mieux refléter la densité centrale du nuage de points.

Cette observation appuie les résultats quantitatifs obtenus précédemment : en s'appuyant sur des points réels du jeu de données (médoïdes) plutôt que sur des moyennes abstraites, CLARA fournit une séparation statistiquement plus robuste.

Intérêt de la visualisation pour l'analyse L'ajout de ces visualisations renforce significativement la crédibilité de l'analyse. D'une part, elles assurent une cohérence entre les observations textuelles et l'aspect visuel des données, notamment en ce qui concerne le fort chevauchement entre clusters. D'autre part, la désorganisation apparente de la projection DBSCAN justifie de manière intuitive le score de Silhouette négatif ($-0{,}72$), qui pourrait autrement être interprété comme une erreur de calcul. Enfin, la difficulté manifeste à distinguer visuellement des groupes bien séparés confirme que les méthodes non supervisées sont insuffisantes pour ce problème, légitimant ainsi le recours à une approche de classification supervisée.

Conclusion Générale

Ce projet a permis de développer et d'évaluer des modèles prédictifs pour la détection précoce des incendies de forêt à partir de données environnementales. L'analyse approfondie des données et leur préparation rigoureuse ont constitué une étape essentielle pour garantir la qualité des résultats.

Les modèles supervisés, notamment les arbres de décision et les forêts aléatoires, ont démontré une bonne capacité à identifier les zones à risque, avec des performances bien supérieures à celles du K-Nearest Neighbors. L'implémentation manuelle de ces algorithmes a permis une meilleure compréhension de leur fonctionnement, tandis que la comparaison avec les versions optimisées de Scikit-learn a confirmé l'efficacité des bibliothèques professionnelles.

L'introduction des méthodes d'apprentissage non supervisé a enrichi l'analyse en permettant de segmenter les données en groupes naturels, offrant ainsi des perspectives supplémentaires pour la gestion des risques et la prévention des incendies. Cependant, il est important de souligner que, dans ce type de problème et sur ce jeu de données spécifique, les algorithmes de clustering testés, bien que valides d'un point de vue méthodologique, demeurent largement moins performants que les approches supervisées pour la classification et la prédiction ciblée des incendies.

En conclusion, ce travail illustre l'importance d'une approche combinée, alliant analyse exploratoire, modélisation supervisée et clustering, pour répondre aux défis complexes liés à la prédiction des incendies. Les résultats obtenus ouvrent la voie à des améliorations futures, notamment par l'intégration de données supplémentaires et le recours à des techniques plus avancées de traitement du déséquilibre et d'optimisation des modèles.