



# Wildfires in USA (1992-2015) Modélisation



**Prepared By The Fire Brigade :**  
Tiphaine Zimmowitch, Pierre Routel, Thibault Bezpalco  
Janvier 2024

## Table des matières

<b>I. Classification du problème.....</b>	<b>6</b>
<b>II. Exploration du dataset principal.....</b>	<b>8</b>
II.1. Préparation du dataset.....	8
II.1.a. Gestion des doublons.....	8
II.1.b. Changement de type.....	8
II.1.c. Création de colonnes d'intérêt.....	8
II.2. Visualisations et Statistiques.....	8
II.2.a. Statistiques.....	8
II.2.b. Visualisations.....	8
• Evolution géographique et temporelle du nombre de feux.....	8
• Corrélation entre le nombre de feux et la surface cumulée brûlée.....	12
• Distribution des classes de feux.....	14
• Les causes des feux.....	15
II.2.c. Constat.....	20
<b>III. Exploration du Dataset complémentaire.....</b>	<b>21</b>
III.1. Dataset "Végétation et météo USA".....	21
III.2. Préparation du dataset.....	21
III.2.a. Gestion des doublons.....	21
III.2.b. Gestion des valeurs manquantes.....	21
III.2.c. Changement de type.....	21
III.2.d. Nettoyage de la colonne de surface de l'écorégion niveau 3.....	21
III.3. Visualisations et Statistiques.....	22
• L'incidence du vent sur les classes de feux.....	22
• L'incidence de taux d'humidité du combustible végétal sur les classes de feux.....	23
• L'incidence de l'indice de biomasse sur les classes de feux.....	23
• L'incidence de l'écorégion (niveau 1) sur les classes de feux.....	24
• L'incidence du type de végétation sur les classes de feux.....	25
<b>IV. Fusion et préprocessing.....</b>	<b>26</b>
IV.1. Jointure.....	26
IV.2. Suppression des colonnes.....	26
IV.3. Réduction de la taille du dataset.....	26
IV.4. Encodage des variables cycliques.....	26
IV.5. Encodage des variables catégorielles.....	26
IV.6. Séparation des variables.....	27
IV.7. Scaling.....	27
<b>V. Modélisation et optimisation.....</b>	<b>28</b>
V.1. Sélection des features : panorama.....	29
V.1.a. KBest avec fonction "f_classif".....	29
V.1.b. KBest avec fonction "mutual_info_classif".....	30
V.1.c. KBest : premières constatations.....	31
V.2. Arbre de décision.....	32
V.2.a. Première approche avec plot_tree.....	32
V.2.b. Paramètre "class_weight".....	33

V.2.c. Causes humaines : regroupées ou non ?.....	33
V.2.d. Les features les plus importantes.....	34
V.2.e. GridSearch CV.....	36
<b>V.3. Régression logistique.....</b>	<b>38</b>
V.3.a. Paramètres de la Régression Logistique.....	38
V.3.b. Causes humaines : regroupées ou non ?.....	39
V.3.c. Les features les plus importantes.....	40
V.3.d. GridSearch CV.....	42
<b>V.4. Forêt aléatoire.....</b>	<b>45</b>
V.4.a. Paramètres “class_weight”.....	45
V.4.b. Causes humaines : regroupées ou non ?.....	46
V.4.c. Les features les plus importantes.....	48
V.4.d. GridSearch CV.....	51
<b>V.5. Boosting.....</b>	<b>52</b>
V.5.a. AdaBoost avec arbre de décision.....	53
V.5.b. HistGradientBoostingClassifier.....	53
<b>V.6. Under et oversampling.....</b>	<b>55</b>
V.6.a. SMOTEN seul.....	55
V.6.b. SMOTEN + Random Under Sampling.....	55
V.6.c. SMOTEN + Edited Nearest Neighbours.....	56
<b>VI. Interprétation des résultats.....</b>	<b>57</b>
VI.1. Features principales.....	57
VI.1.a. Durée.....	57
VI.1.b. Coordonnées géographiques.....	57
VI.1.c. Saisonnalité.....	59
VI.1.d. Vent.....	60
VI.1.e. Humidité de la végétation.....	61
VI.1.f. Indice de biomasse.....	62
VI.2. Pistes d'amélioration.....	63
<b>VII. Dataset corrigé et samplié.....</b>	<b>64</b>
VII.1. Problèmes identifiés.....	64
VII.1.a. Outliers : durée de feu.....	64
VII.1.b. Erreurs sur les dates.....	65
VII.1.b. Perte de données.....	66
VII.2. Entraînement.....	66
VII.2.a. Random Forest et jeu de données corrigé.....	66
VII.2.a. Random Forest et jeu de données corrigé et samplié.....	67
VII.2.a. Random Forest et jeu de données initial avec outliers.....	68
<b>Annexes.....</b>	<b>69</b>
Annexe 1 : Traitement du jeu de données principal.....	69
Annexe 2 : Description des Causes des Incendies.....	71
Annexe 3 : Carte des écorégions de niveau 1.....	74
Annexe 4 : Répartition géographique des types de végétation.....	75

## Contexte

Saviez-vous que les incendies de forêt brûlent en moyenne 6 millions d'acres par an aux États-Unis ?\*

Ou que les humains sont à l'origine d'environ 85% de ces incendies de forêt ?\*

Le gouvernement américain dépense environ 2 milliards de dollars par an pour éteindre les incendies de forêt. A cela s'ajoute la valeur des biens détruits qui dépasse les 6 milliards de dollars par an.

Outre l'aspect financier, il s'agit aussi d'un problème majeur pour l'écosystème des espaces américains, où le rythme des incendies domine le taux de rétablissement : après un incendie, les écosystèmes naturels sont détruits, touchant plus de 350 espèces d'animaux et de plantes.

Les enquêtes sur les incendies de forêt cherchent à en comprendre les causes afin que les agences puissent préparer et mettre en œuvre des stratégies de prévention.

C'est en comprenant les incendies de forêt que nous pourrons mieux planifier leurs effets potentiels souhaitables et indésirables.

Cependant, même si le gouvernement américain et les populations peuvent être préparés, ils ne peuvent pas prédire quand et où les incendies vont se produire. C'est ce que l'on va tenter de faire dans cette étude.

\*source : National Interagency Fire Center

## Objectif

Peut-on prédire l'ampleur d'un incendie, à savoir sa classe ?

## Cadre

Pour réaliser cette analyse, les données ont été extraites d'un ensemble de données Kaggle contenant 1,88 million d'incendies de forêt aux États-Unis : 24 ans d'enregistrements géoréférencés des incendies de forêt.

La publication de données contient une base de données spatiales des incendies de forêt survenus aux États-Unis de 1992 à 2015. Les enregistrements d'incendies de forêt ont été acquis à partir des systèmes de reporting des organisations fédérales, étatiques et locales de lutte contre les incendies. Les éléments de données de base suivants étaient requis pour que les enregistrements soient inclus dans cette publication de données : date de découverte, taille finale du feu et emplacement du point.

Les données ont été stockées sous forme de base de données SQLite. Elles ont ensuite été exportées vers un fichier .csv. Le fichier est importé dans un Jupyter Notebook, lu et manipulé à l'aide de la librairie Pandas du langage Python.

Provenant d'une publication scientifique, un dataset complémentaire, contenant les données de végétation et de météo sur un périmètre quasi identique au dataset Kaggle,, a été ajouté afin de tenter de parfaire le modèle.

Liens :

- dataset initial : <https://www.kaggle.com/ratman/188-million-us-wildfires>
- dataset complémentaire : article “Les incendies d'origine humaine augmentent le nombre de grands incendies de forêt dans les écorégions des États-Unis”  
<https://www.mdpi.com/2571-6255/1/1/4>

---

# I. Classification du problème

---

Le problème de machine learning de notre projet est une classification multi-classes. En effet, on cherche à modéliser la classe de feu en fonction des diverses données à notre disposition.

Les classes de feux sont définies selon la surface finale brûlée par le feu, comme suit :

- A : comprise entre 0 et 0.25 acres,
- B : comprise entre 0,26 et 9,9 acres,
- C : comprise entre 10,0 et 99,9 acres,
- D : comprise entre 100 et 299 acres,
- E : comprise entre 300 et 999 acres,
- F : comprise entre 1000 et 4999 acres
- G : supérieure à 5000 acres

On rappelle l'équivalence : 1 acre = 0,404686 hectare.

Les données à disposition proviennent de deux datasets :

- Kaggle :
  - Coordonnées géographiques
  - Date
  - Cause (foudre, criminelle, débris verts, feu de camp...)
  - Durée
  - ...
- Publication :
  - Vent
  - Humidité du “combustible” que constitue la matière végétale (feuilles, bois...)
  - Écorégion (niveaux 1 et 3)
  - Indice de biomasse de la zone, mesurant la “vie” sur la parcelle
  - ...

Le dataset étant très déséquilibré, on choisit d'utiliser le F1-score, qui est la moyenne harmonique de la précision (combien d'éléments prédits dans une classe sont pertinents ?) et du rappel (combien d'éléments au sein d'une classe sont détectés ?). Le F1-score permet de faire un compromis entre ces deux mesures.

Afin d'analyser plus finement les résultats, on utilise aussi la matrice de confusion et le rapport de classification, notamment pour avoir accès au F1-score de chaque classe.

Une autre métrique envisageable aurait été le F-bêta score, généralisation du F1-score.

Celui-ci permet d'ajuster la métrique d'évaluation en fonction de la préférence “métier” : souhaite-t-on privilégier le recall avec  $\beta > 1$ , et donc la diminution des faux négatifs, ou bien la précision avec  $\beta < 1$ , et donc la diminution des faux positifs ?

---

## II. Exploration du dataset principal

---

### II.1. Préparation du dataset

#### II.1.a. Gestion des doublons

Le dataset comportait quelques doublons d'ID. Les lignes associées ont été par sécurité supprimées.

#### II.1.b. Changement de type

Afin d'alléger l'espace mémoire occupé par le dataset, toutes les variables catégorielles se sont vues attribuées un type "category".

#### II.1.c. Création de colonnes d'intérêt

Les datetimes de début et de fin de feu ont été créées à partir des dates et des horaires de début et de fin de feu. Cela a permis de créer une variable avec un poids important dans la modélisation : la durée du feu en minutes.

Une nouvelle colonne "CAUSE\_DESCR\_HUMAN" regroupe toutes les causes d'origine humaine et se distingue de la foudre (lightning) ou de l'absence de cause (missing/undefined).

Annexe 1 : traitement du jeu de données principal

Annexe 2 : description des causes des incendies

### II.2. Visualisations et Statistiques

#### II.2.a. Statistiques

Dans l'étude de la dispersion de nos données, nous avons tracé des boxplots. En présence de valeurs trop extrêmes, ceux-ci n'étaient pas pertinents à la lecture et nous avons décidé de ne pas les mettre en avant.

#### II.2.b. Visualisations

- Evolution géographique et temporelle du nombre de feux

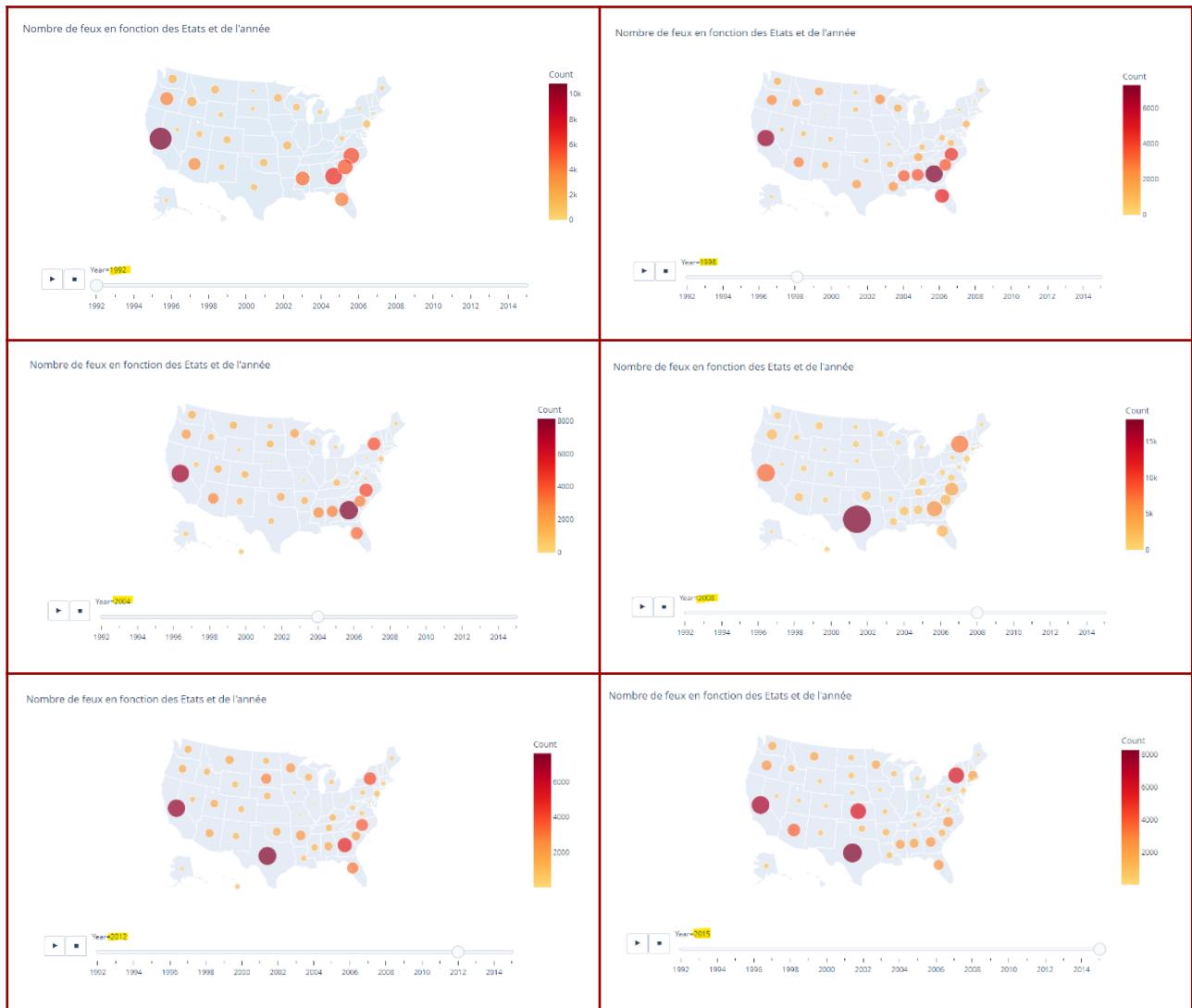
Introduction sur l'évolution géographique et temporelle du nombre de feux aux USA grâce à un graphique de dispersion géographique animé.

Sur la bubble map, chaque Etat voit son nombre annuel de feux figuré par une bulle de taille et de couleur variable en fonction de l'année.

Cette bubble map ci-dessus nous permet d'avancer que la présence des feux évolue sur tout le territoire américain avec le temps.

En effet, des états qui n'étaient pas ou peu touchés au début de la période comptent par la suite des feux, par exemple le Kansas ou encore les états au nord de la côte est.

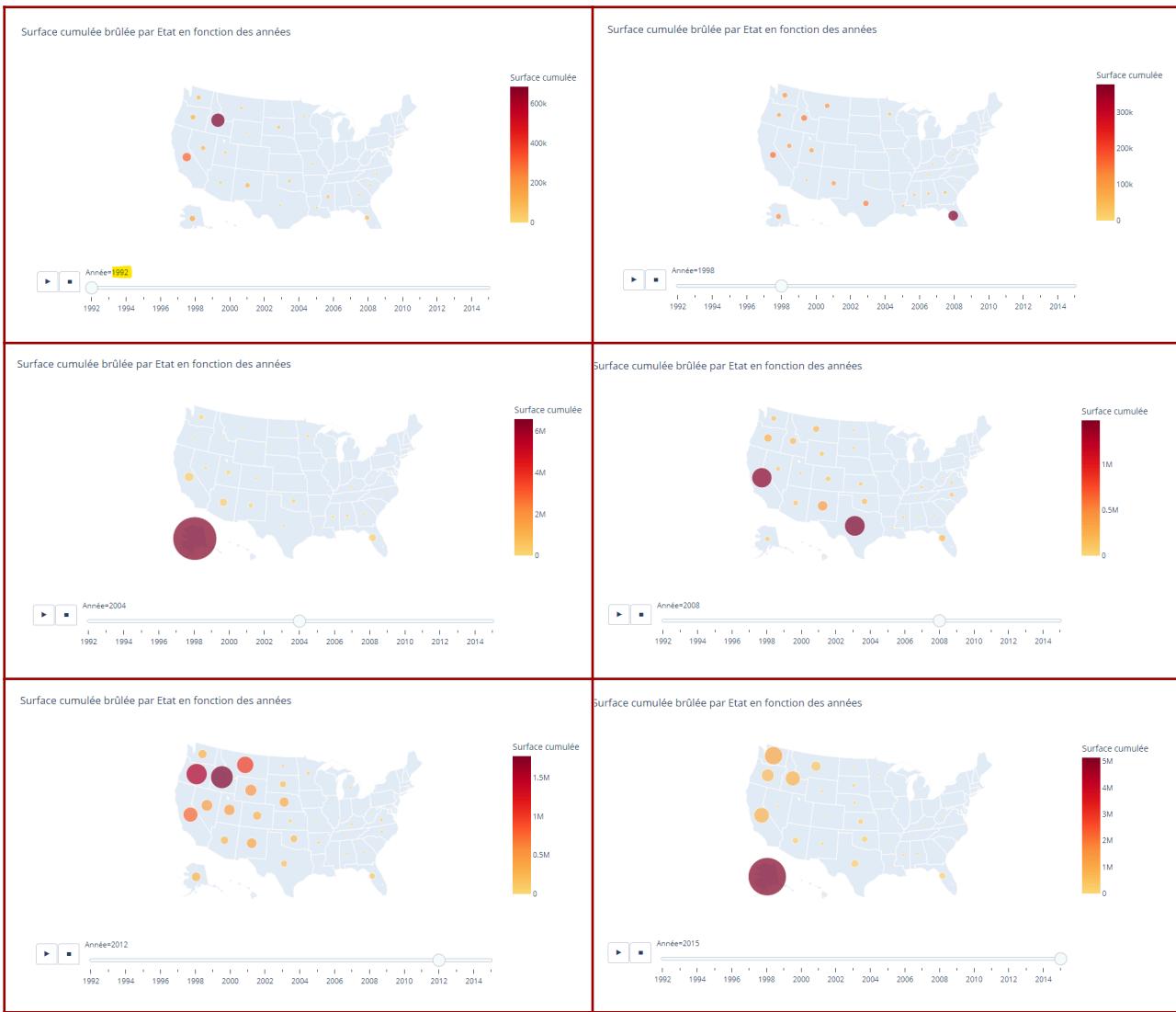
L'État du Texas, qui était peu touché par les feux entre les années 1990 à 2005, a vu son nombre d'incendies s'accroître fortement depuis 2006 et sans discontinuer depuis.



Nous remarquons aussi que les Etats de la côte sud-est sont fréquemment touchés, avec des variations temporelles non négligeables.

La Californie quant à elle, est constamment touchée.

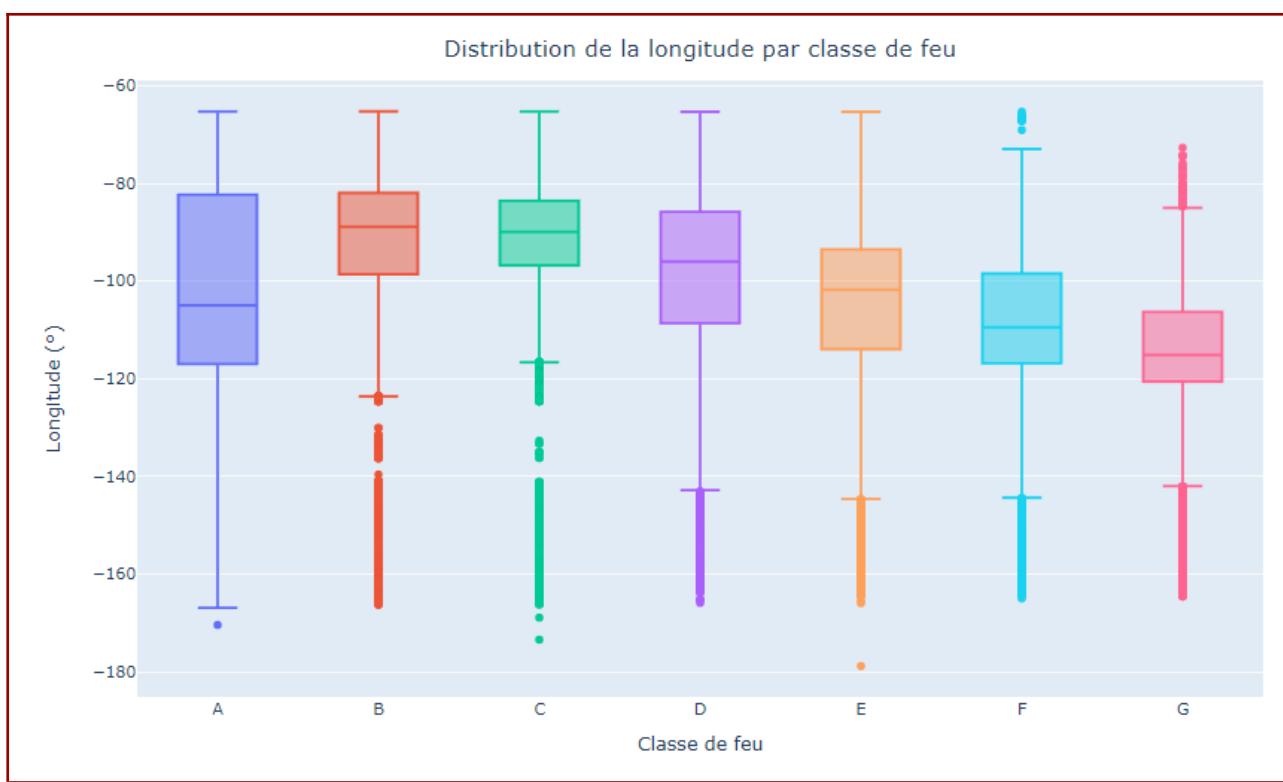
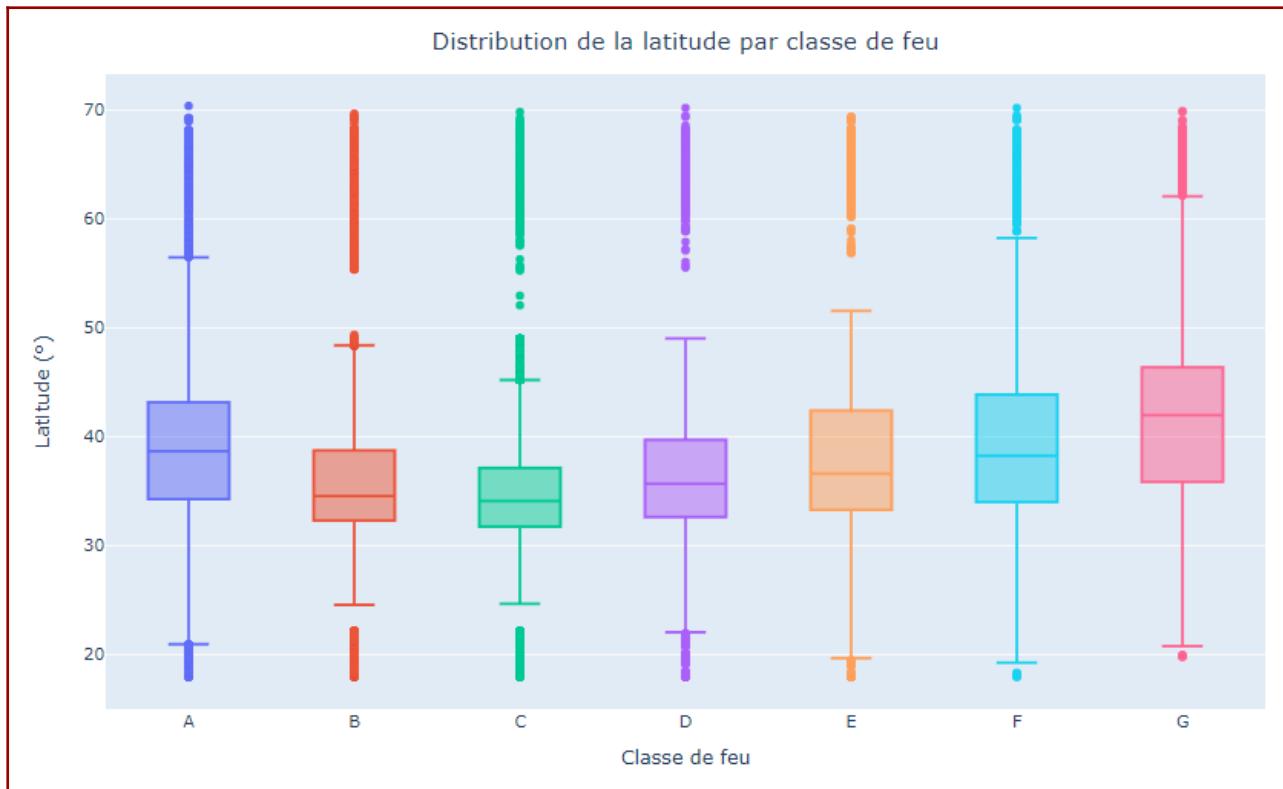
Sur la bubble map suivante, c'est le cumul annuel de surface brûlée par état qui est représenté par une bulle de taille et de couleur variable.



L'étude comparative de ces deux bubble maps nous montre que la corrélation entre le nombre de feu et la surface brûlées cumulées n'est pas assurée.

Le cas de l'Alaska est particulièrement révélateur de ce fait : alors que le nombre de feux est bien moins important que dans des états comme la Californie, la surface brûlée en Alaska est souvent n°1 dans le classement des surfaces brûlées.

On peut préciser l'analyse géographique en croisant la classe avec les latitude et longitude des feux.



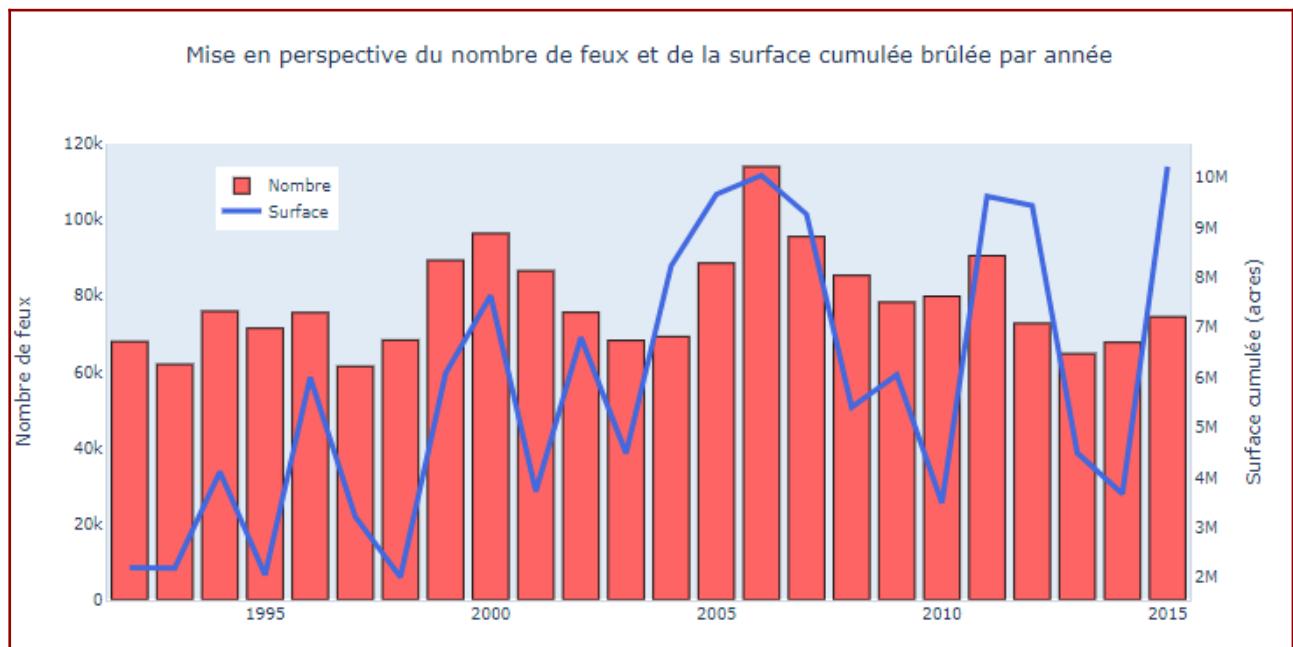
Au niveau de la longitude, on pressent un poids conséquent de la côte Ouest dans les feux de grande classe. En effet, comme on le verra, une part importante des feux de grande classe touche l'Alaska, la Californie ou encore l'Idaho ou l'Etat de Washington, qui se trouvent tous à l'Ouest des Etats-Unis.

- Corrélation entre le nombre de feux et la surface cumulée brûlée

À l'échelle du pays, ce constat quant à l'absence de corrélation stricte entre nombre de feux et surface cumulée brûlée est de nouveau vérifié dans la figure suivante :

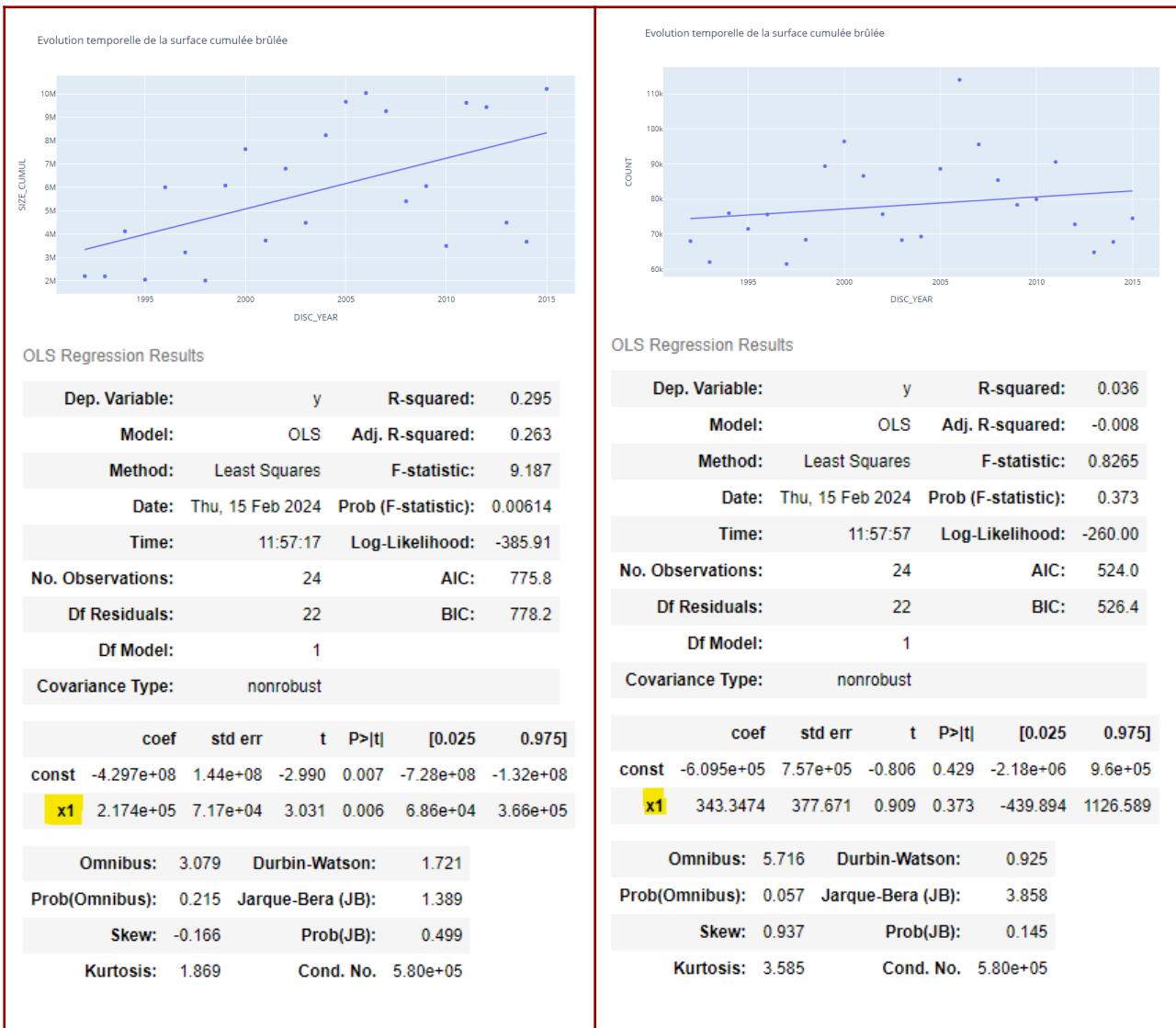
- 2001-2002 : alors que le nombre de feux diminue, la surface cumulée brûlée augmente

- 2013-2014 : alors qu'il y a une augmentation du nombre de feux, la surface cumulée brûlée diminue



Dans le but de se figurer les grandes tendances temporelles, on peut en première approximation utiliser une régression linéaire entre le nombre annuel de feux et l'année d'une part, et entre la surface annuelle brûlée et l'année d'autre part.

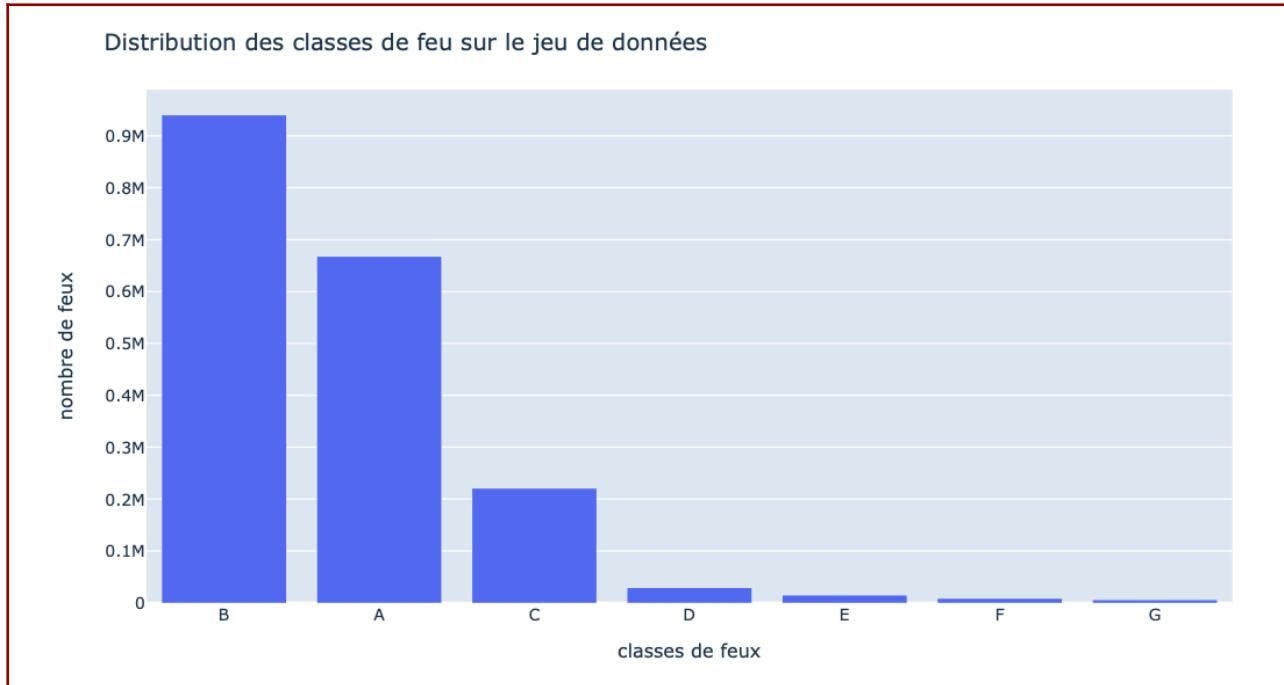
Bien entendu, au vu des variations annuelles importantes, cette régression linéaire n'est pas la meilleure modélisation du phénomène et ne peut être utilisée pour des conclusions précises.



On aboutit à deux constatations différentes. Le nombre de feux ne semble pas évoluer de manière significative. Au contraire, la surface cumulée brûlée semble croître de manière non négligeable : si on s'en tient au coefficient directeur de la régression, la surface aurait été multipliée par 2,5 environ sur la période d'analyse.

Pour rester sur la surface, il est important d'introduire la notion des classes de feu issue de la nomenclature américaine (source : National Interagency Fire Center et Climatecheck) de classification par taille des incendies, cette variable sera d'ailleurs notre variable cible pour l'étude de nos prédictions.

- Distribution des classes de feux



Les pourcentages surlignés représentent la proportion de chaque classe sur le total des feux enregistrés.

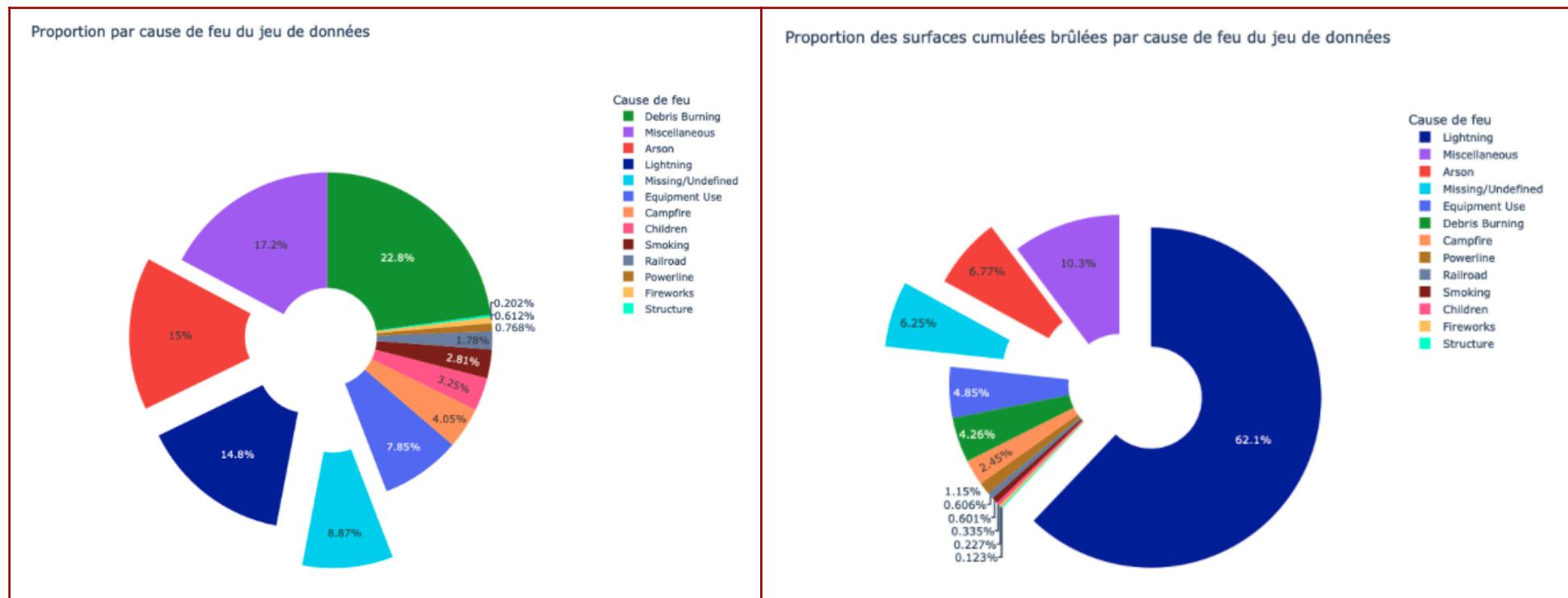
- A = supérieur à 0 mais inférieur ou égal à 0,25 acre >> 35,5 %
- B = de 0,26 à 9,9 acres >> 50,0 %
- C = de 10,0 à 99,9 acres >> 11,7 %
- D = de 100 à 299 acres >> 1,5 %
- E = de 300 à 999 acres >> 0,8 %
- F = de 1 000 à 4 999 acres >> 0,4 %
- G = supérieur à 5 000 acres >> 0,2 %

Nous remarquons que 97% des feux sont de classes A, B et C. Les surfaces détruites sont en très grande majorité “modérées”, à savoir pas plus de 99.9 acres.

Nous avons créé le même histogramme qui évolue en fonction des années qui peut être consulté dans le notebook. Juste avant le début des années 2000, la classe B devient majoritaire en termes de nombre de feux alors que c'était la classe A qui dominait jusqu'ici.

- Les causes des feux

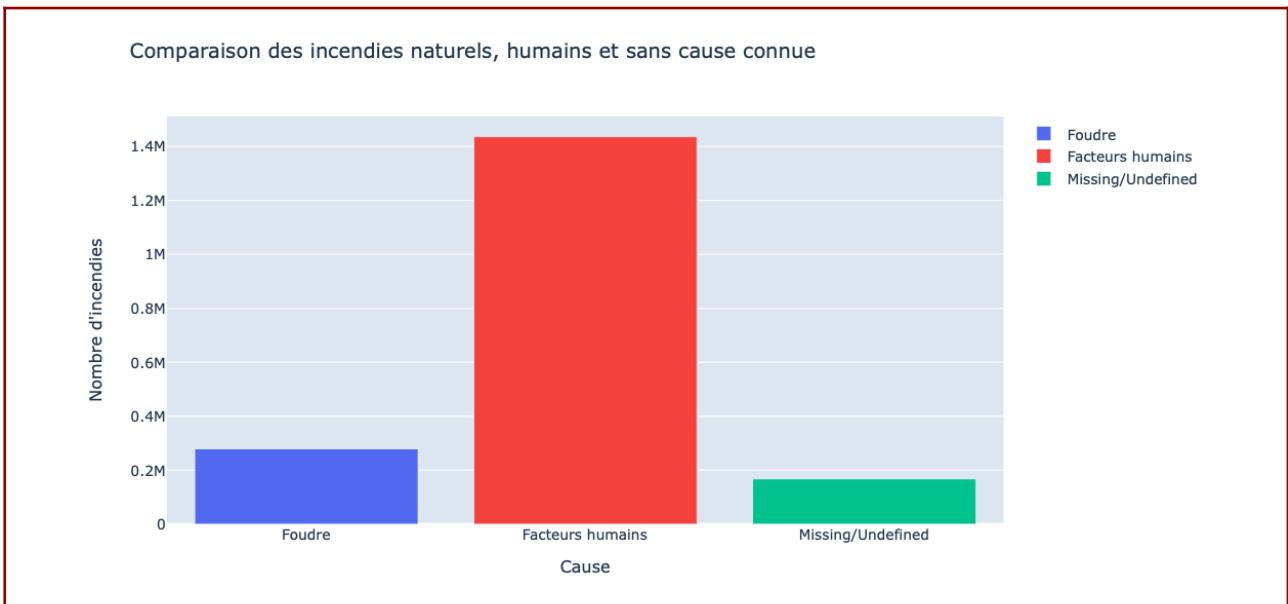
Une autre variable catégorielle indispensable dans notre étude est la cause des feux, représentée dans les figures suivantes.



Comme souligné en introduction, les causes d'origine humaine représentent 76% du camembert (dont 15% sont volontaires puisque criminels) tandis que les causes dites naturelles (foudre) représentent 15 %.

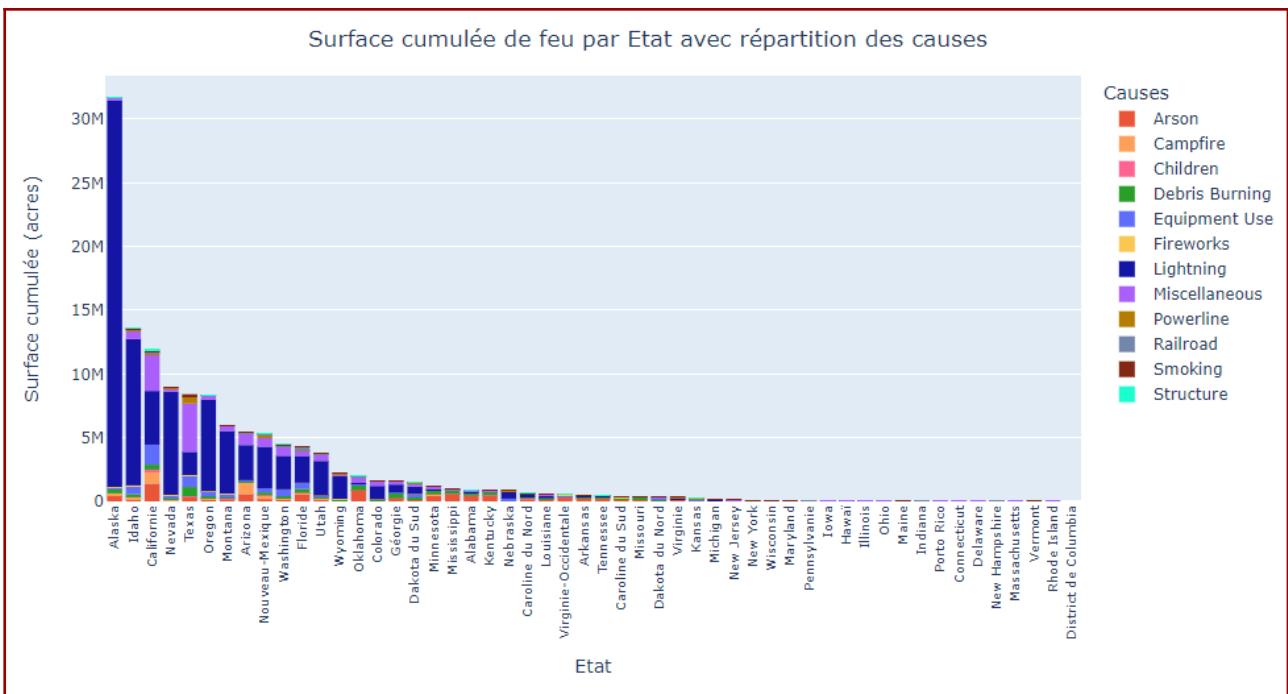
On constate encore une fois une différence de point de vue en fonction de l'axe d'analyse choisi. Prenons l'exemple des feux de débris végétaux : alors qu'ils représentent la première raison du nombre de feux, ils ne sont finalement que la sixième cause de la surface brûlée. À l'inverse, la foudre n'est que la quatrième cause du nombre de feux mais représente plus de la moitié de la surface brûlée.

Il faut donc vraiment prêter attention à la variable analysée.



Cette figure nous montre qu'il est pertinent d'isoler les causes inconnues (missing/undefined) qui représentent 9 % du total des feux enregistrés et 6 % des surfaces cumulées. On observe bien que les feux d'origine humaine restent prédominants.

Nous axons maintenant notre étude sur les 52 Etats américains avec les deux figures suivantes.



Ce graphique illustre que certains états subissent des dégâts bien plus importants que d'autres.

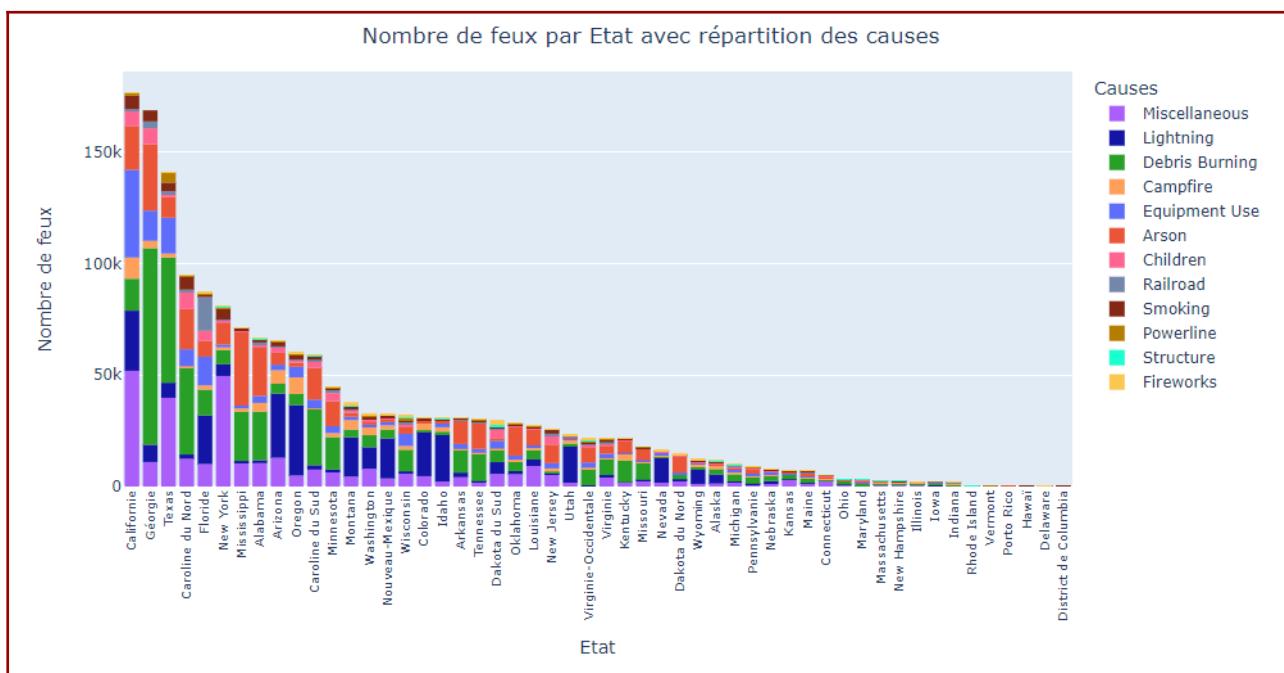
Notamment l'État d'Alaska (AK) dont le territoire brûlé cumulé dépasse de loin les autres États. L'Alaska est principalement touchée par des feux de foudre, mais nous reviendrons sur les différentes causes plus tard.

Avec le graphique suivant, nous pouvons déjà observer que la corrélation entre surface brûlée et nombre de feux comptés est faible. Si nous reprenons l'Alaska, nous pouvons voir que l'État est en 35ème position sur le nombre de feux déclarés, là où il était premier, et de loin, pour la surface.

Nous constatons l'inverse pour l'état de Géorgie(GA).

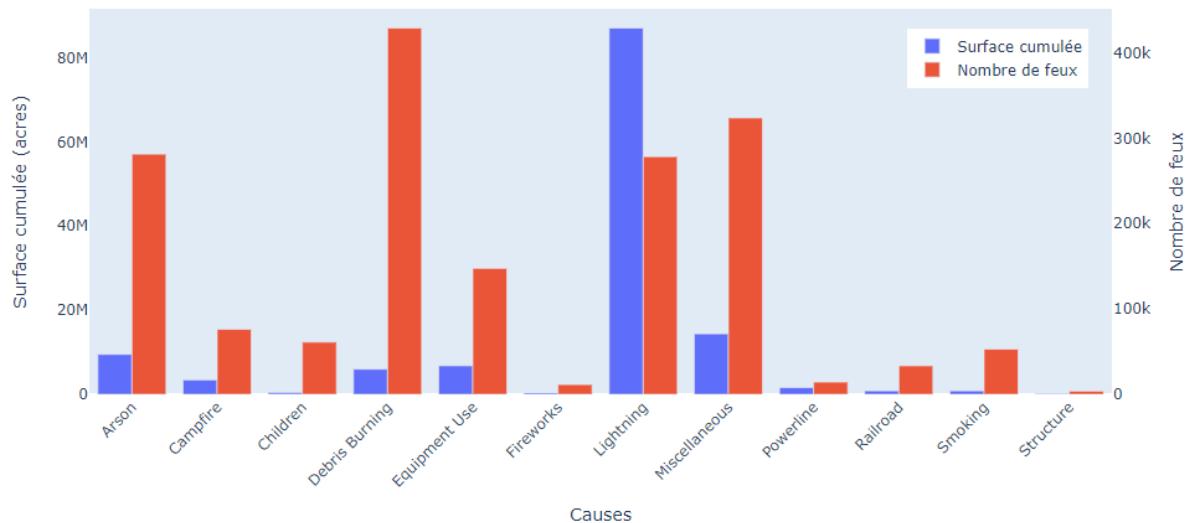
Ceci s'explique par la cause du feu : là où il y a beaucoup de feux de débris végétaux en Géorgie, l'Alaska est majoritairement touchée par des feux dûs à la foudre, feux qui entraînent une surface brûlée plus importante que les autres causes.

Nous observons aussi que l'État de Californie (CA) est quant à lui dans le top 5 de chacune des figures, de même pour le Texas (TX).



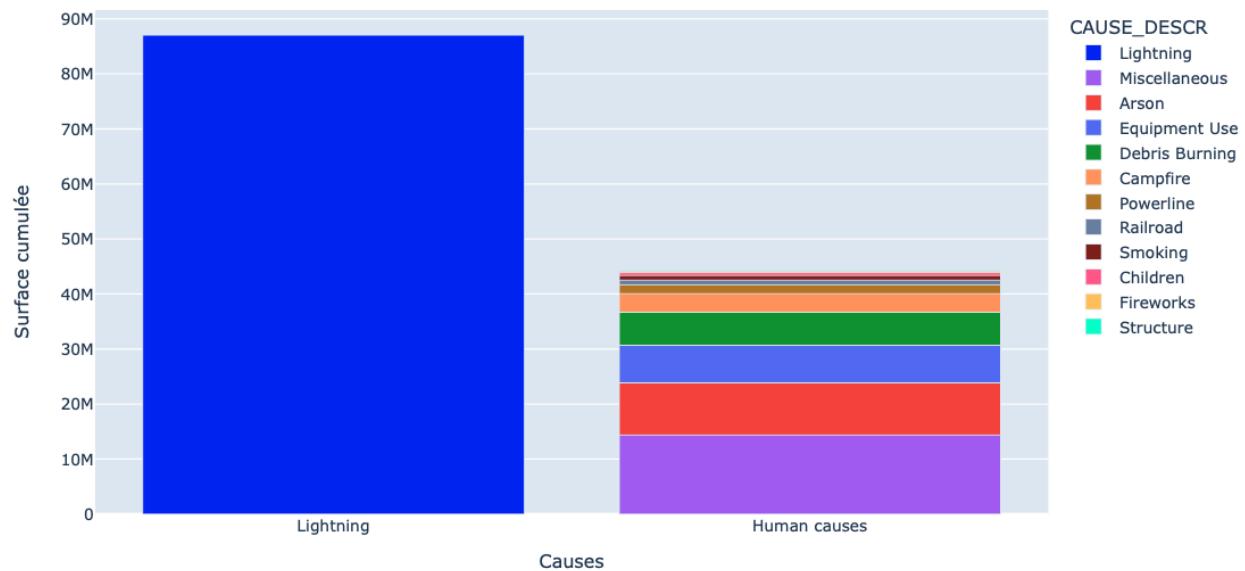
L'histogramme suivant donne une vue d'ensemble des deux figures précédentes.

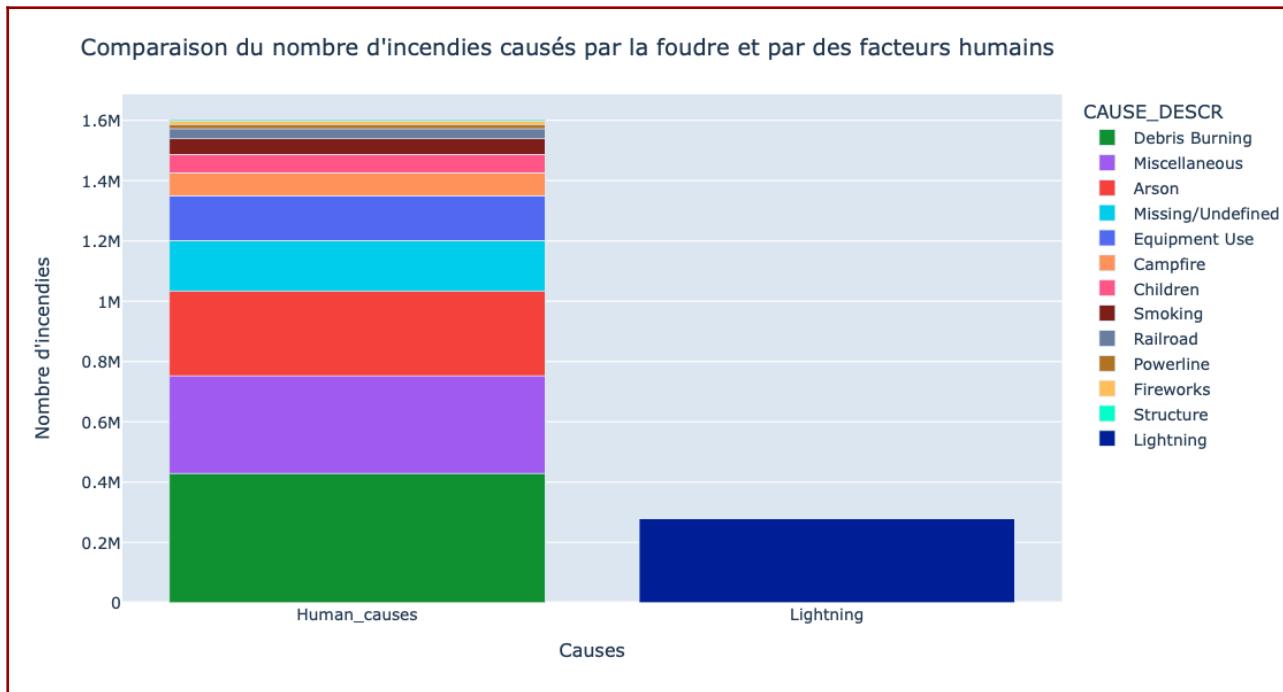
Mise en perspective du nombre de feux et de la surface cumulée brûlée par cause



Les deux figures suivantes permettent de détailler la différence entre la surface et le nombre de feux d'origine humaine et naturelle par cause.

Comparaison des surfaces cumulées brûlées par la foudre et par des facteurs humains





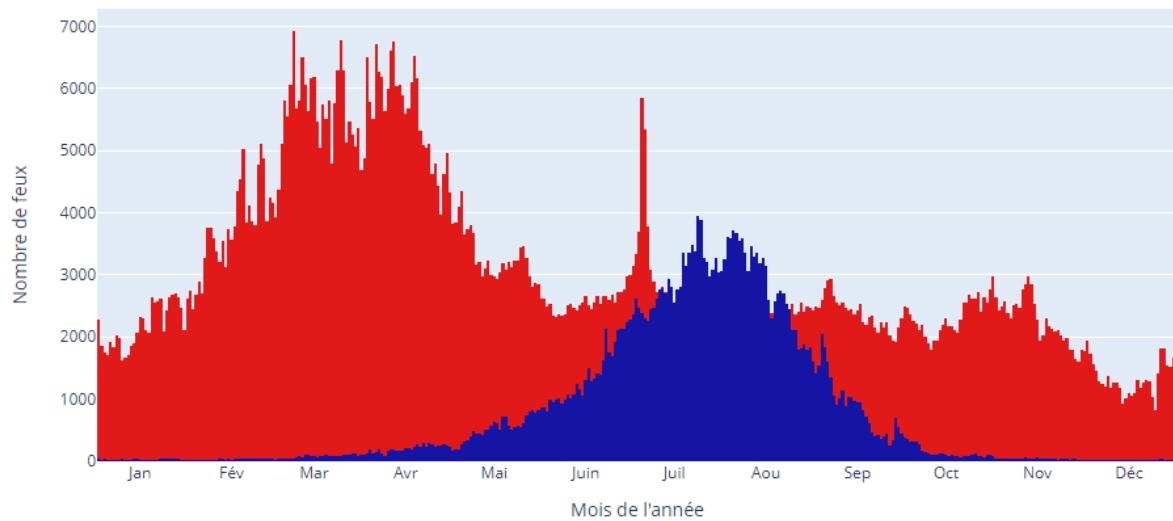
La comparaison des deux figures illustre qu'il n'y a pas de corrélation entre le nombre de feu par cause et la surface cumulée provoquée. Exemple pour illustrer notre propos : la cause Lightning.

Ceci se confirme aussi avec la cause Debris Burning qui correspond à 23% des feux d'une part, mais seulement à 4% de la surface cumulée d'autre part.

Pour donner un ordre de grandeur, la surface brûlée par la foudre représente 85 millions d'acres, soit équivalente à la surface cumulée de tous les parcs nationaux américains.

L'histogramme suivant montre la distribution des incendies selon le jour de l'année. Le contraste entre la foudre d'un côté et les causes humaines et de l'autre est saisissant. La saisonnalité des feux diffère.

Distribution selon le jour de l'année des feux : foudre VS causes humaines



En effet, les feux dûs à la foudre se concentrent sur la période “chaude” de l’année, à savoir l’été.

Les feux d’origines humaines sont caractérisés par une saisonnalité différente. Bien qu’il y ait au minimum plus d’un millier de feux journaliers, nous constatons :

- une forte augmentation de feux de d’origines humaines au printemps (mars et avril), probablement liée au brûlage des végétaux morts.
- un pic en juillet (correspondant très certainement aux feux provoqués par les feux d’artifice lors de la fête nationale du 4 juillet).

### II.2.c. Constat

Ces précédentes visualisations et le preprocessing nous confirment que nous sommes face à un dataset très vaste et dispersé.

Afin de mieux répondre à notre question : “Peut-on prédire l’ampleur d’un incendie ?”, il nous a semblé pertinent d’apporter à notre étude un nouveau Dataset utilisé dans l’analyse de l’article « Les incendies d’origine humaine augmentent le nombre de grands incendies de forêt dans les écorégions des États-Unis » par R. Chelsea Nagy, Emily Fusco, Bethany Bradley, John T. Abatzoglou et Jennifer Balch. Cet article a été accepté pour publication dans la revue Fire le 22 janvier 2018.

## **III. Exploration du Dataset complémentaire**

### **III.1. Dataset “Végétation et météo USA”**

Pour comprendre les conditions environnementales dans lesquelles de grands incendies de forêt se sont produits dans différentes écorégions, nous avons exploré l'influence de l'humidité du combustible végétal, la vitesse du vent, le type de végétation/environnement biophysique et les conditions de biomasse de grands incendies de forêt déclenchés par l'homme et la foudre.

Annexe 3 : carte des écorégions de niveau 1

### **III.2. Préparation du dataset**

#### **III.2.a. Gestion des doublons**

Le dataset comportait quelques doublons d'ID. Les lignes associées ont été par sécurité supprimées.

Les colonnes en doublet du dataset initial ont été supprimées : localisation, dates, causes, surface.

#### **III.2.b. Gestion des valeurs manquantes**

Quelques milliers de lignes, comportant des valeurs manquantes relatives au vent et à l'humidité du combustible végétal, ont été supprimées.

#### **III.2.c. Changement de type**

Afin d'alléger l'espace mémoire occupé par le dataset, toutes les variables catégorielles se sont vues attribuées un type “category”.

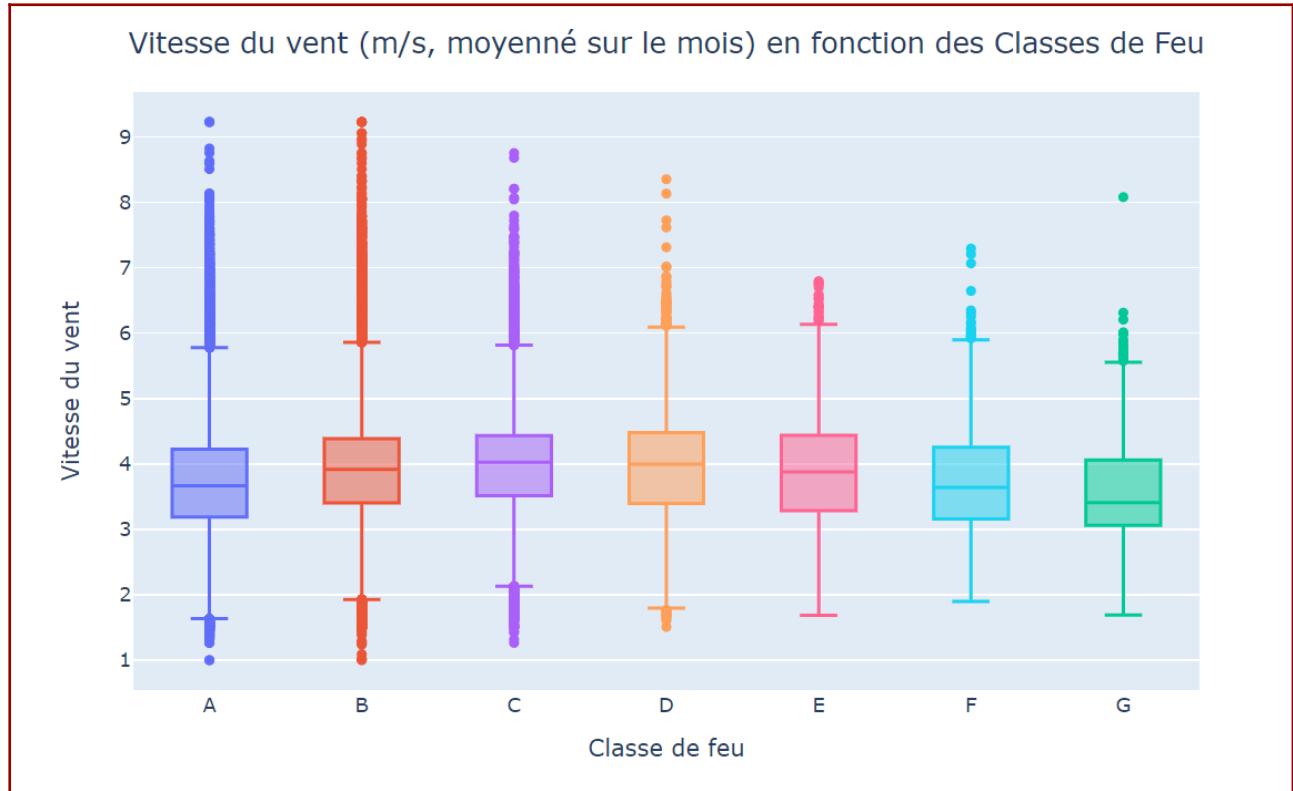
#### **III.2.d. Nettoyage de la colonne de surface de l'écorégion niveau 3**

Certaines écorégions de niveau 3 comportaient des valeurs aberrantes, de l'ordre du kilomètre carré là où en réalité même les zones les plus petites font plusieurs milliers de kilomètres carrés.

Ces valeurs ont été harmonisées en prenant la valeur maximale (en général majoritaire) propre à chaque écorégion de niveau 3.

### III.3. Visualisations et Statistiques

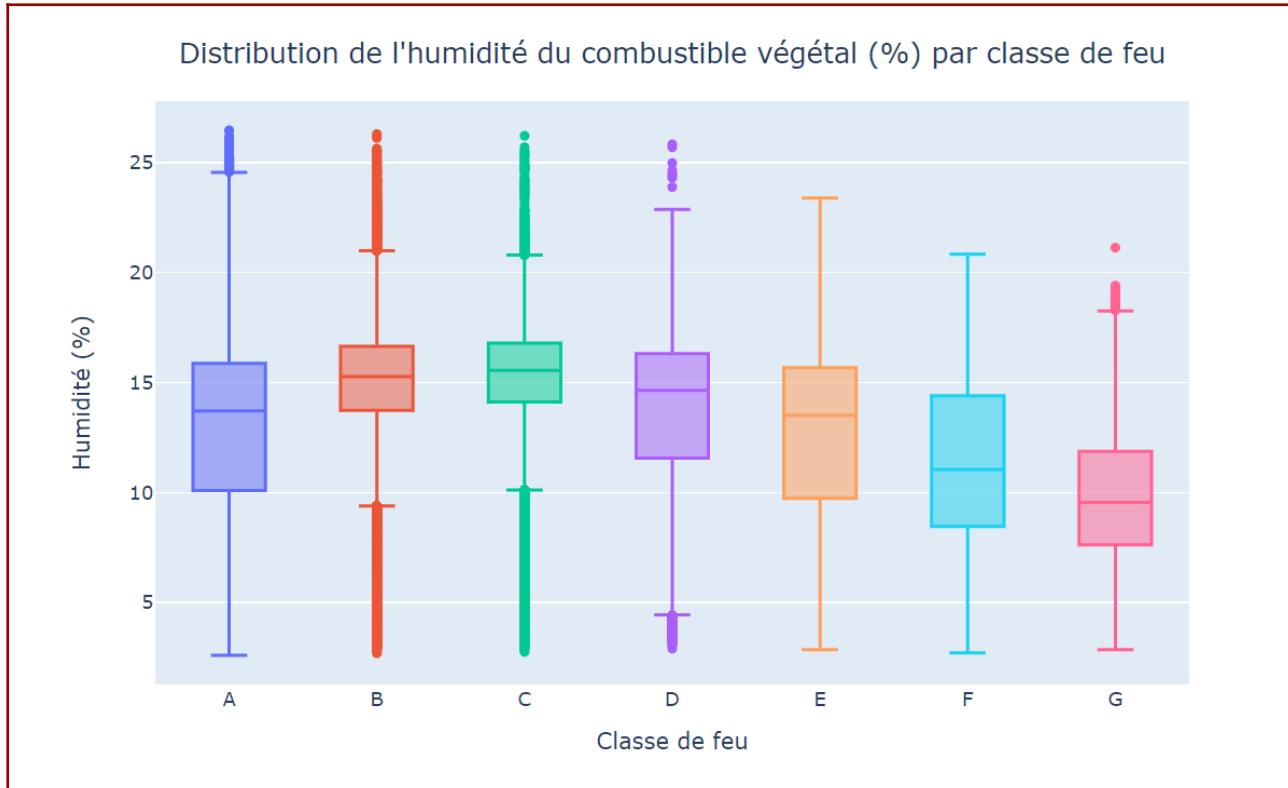
- L'incidence du vent sur les classes de feux



Bien que l'on sache que les vitesses de vent plus élevées sont liées à des incendies plus importants , nous n'avons pas trouvé de lien entre la taille moyenne des grands incendies et la vitesse moyenne du vent à l'échelle de l'écorégion.

L'agrégation des conditions environnementales en une moyenne d'écorégion et en moyennes climatologiques peut masquer les tendances observables au niveau des incendies individuels provoqués par le vent.

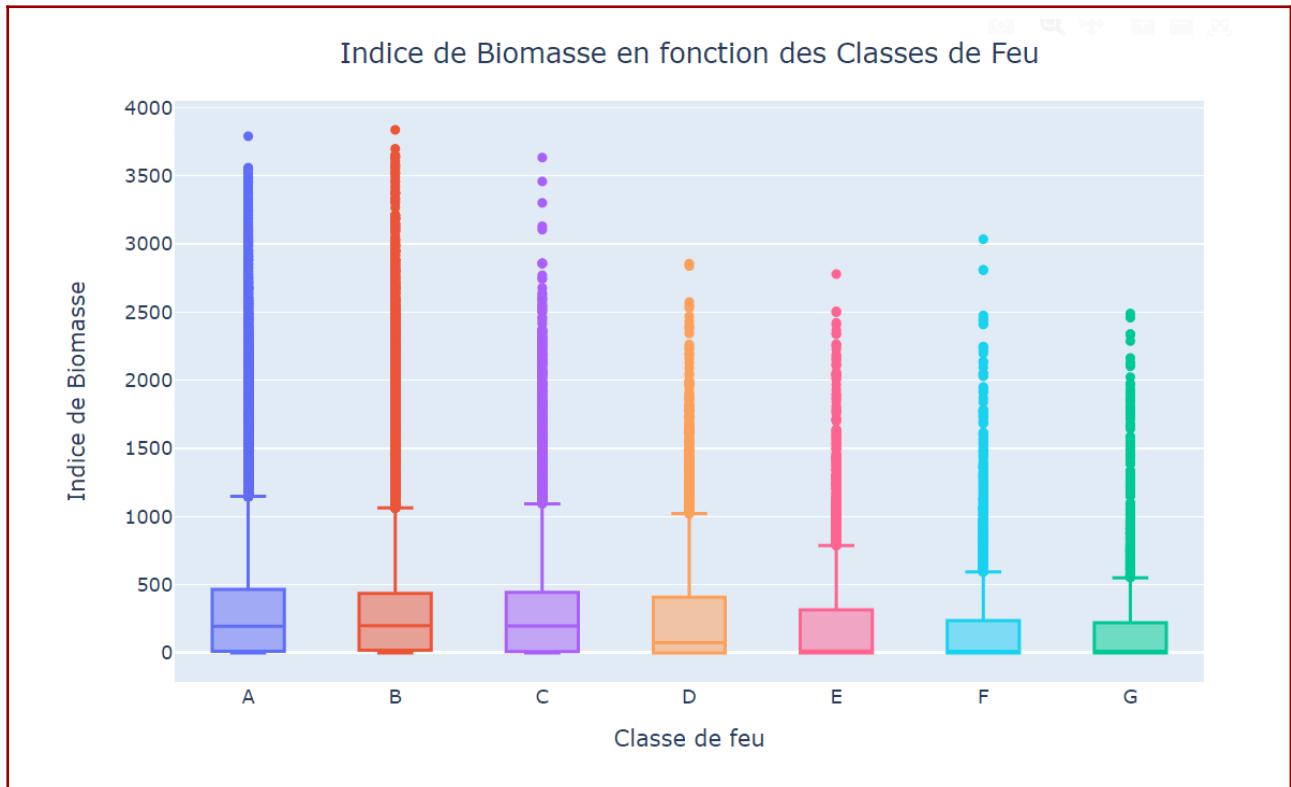
- L'incidence de taux d'humidité du combustible végétal sur les classes de feux



Les incendies les plus importants (F et G) se produisent dans les écorégions où l'humidité moyenne annuelle du combustible est inférieure à 12 %.

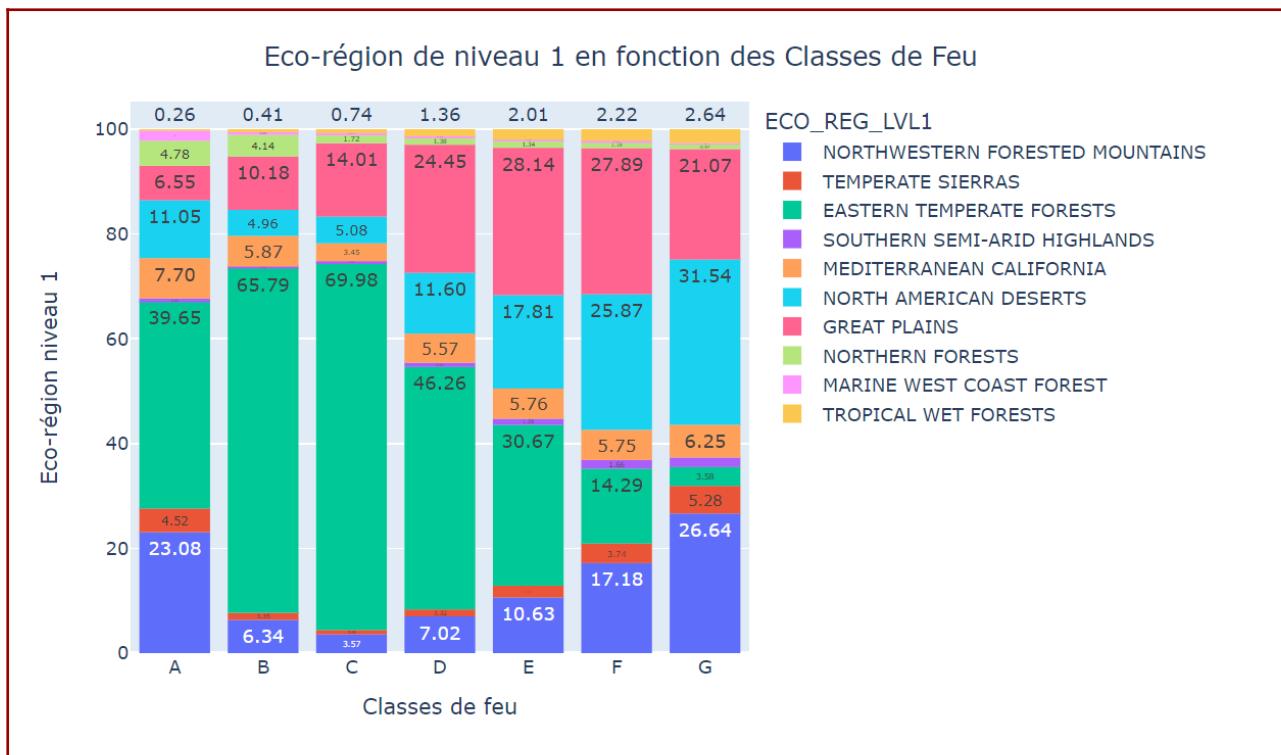
- L'incidence de l'indice de biomasse sur les classes de feux

Cet indice est utile pour évaluer la santé et la vigueur des plantes dans des environnements où la structure de la canopée varie.



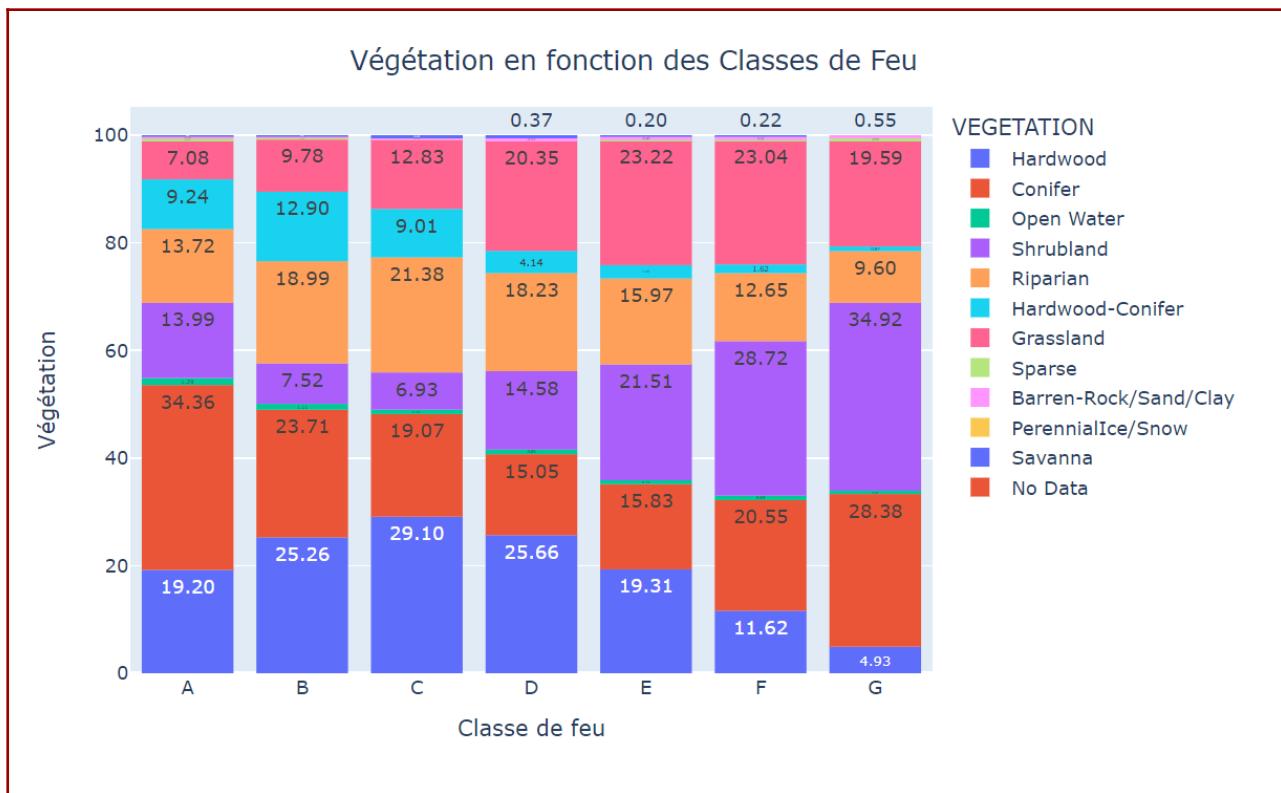
Ici, nous observons une corrélation forte entre un indice de biomasse très faible et la présence de feux, toutes classes confondues. avec un tendance que plus cet indice est bas, plus le feu est grand.

- L'incidence de l'écorégion (niveau 1) sur les classes de feux



### Annexe 3 : Carte des écorégions de niveau 1

- L'incidence du type de végétation sur les classes de feux



### Annexe 4 : Répartition géographique des types de végétation

---

## IV. Fusion et préprocessing

---

### IV.1. Jointure

La jointure entre les deux datasets a été effectuée grâce à l'identifiant fonctionnel (FPA\_ID), l'année et le jour de l'année de découverte du feu.

### IV.2. Suppression des colonnes

Les colonnes d'identifiants, les colonnes temporelles de maîtrise de feu et la colonne de surface de feu (directement corrélée à la classe de feu par définition) ont été supprimées.

### IV.3. Réduction de la taille du dataset

Le dataset est très volumineux (près de 2 millions de lignes, plus d'une quarantaine de colonnes).

L'intuition est que la durée va jouer un rôle prépondérant dans la modélisation. Par conséquent, seuls les enregistrements comportant une durée ont été conservés. Cela représente un peu moins de 50 % du dataset initial.

### IV.4. Encodage des variables cycliques

Afin de conserver la périodicité des variables cycliques, telles que le jour de l'année, un encodage trigonométrique a été effectué : la variable est alors rapprochée d'un point sur un cercle trigonométrique, ce qui débouche sur un doublet de coordonnées (cosinus,sinus).

Ce traitement a été appliqué sur :

- Les coordonnées géographiques : latitude, longitude
- Le jour de l'année

### IV.5. Encodage des variables catégorielles

Toutes les variables catégorielles d'intérêt ont été encodées avec un OneHotEncoder :

- Les causes du feu (13 valeurs)
- Les types de propriétaires des terrains où sont localisés les feux (16 valeurs)
- Les Etats américains (49 valeurs)
- Les écorégions de niveau 1 (10 valeurs)
- Les écorégions de niveau 3 (84 valeurs)
- Les types de végétation (12 valeurs)
- La variable cible (7 valeurs), pour des traitements annexes nécessitant un type numérique

## IV.6. Séparation des variables

Le dataset fusionné a été séparé en deux datasets d'entraînement et de test, avec une proportion réservée au test de 20 %.

## IV.7. Scaling

Le dataset présente sur certaines variables, notamment la durée, des disparités très importantes : il est impossible d'appliquer dans ces conditions une standardisation ou une normalisation.

Est alors utilisé un RobustScaler, insensible aux outliers.

---

## V. Modélisation et optimisation

---

Nous avons utilisé différents algorithmes afin d'évaluer au mieux la classe du feu :

- Arbre de décision : modèle simple mais facilement interprétable
- Régression logistique
- SVC : malheureusement, trop gourmand en ressources, nous n'avons pas pu l'entraîner dans des délais raisonnables
- Forêt aléatoire : plus robuste qu'un arbre de décision mais moins interprétable

Les modèles ont été entraînés avec une Grid Search Cross Validation, afin de voir si la sélection de meilleurs hyperparamètres permettait d'améliorer significativement les résultats de modélisation.

Des algorithmes de Boosting ont été testés.

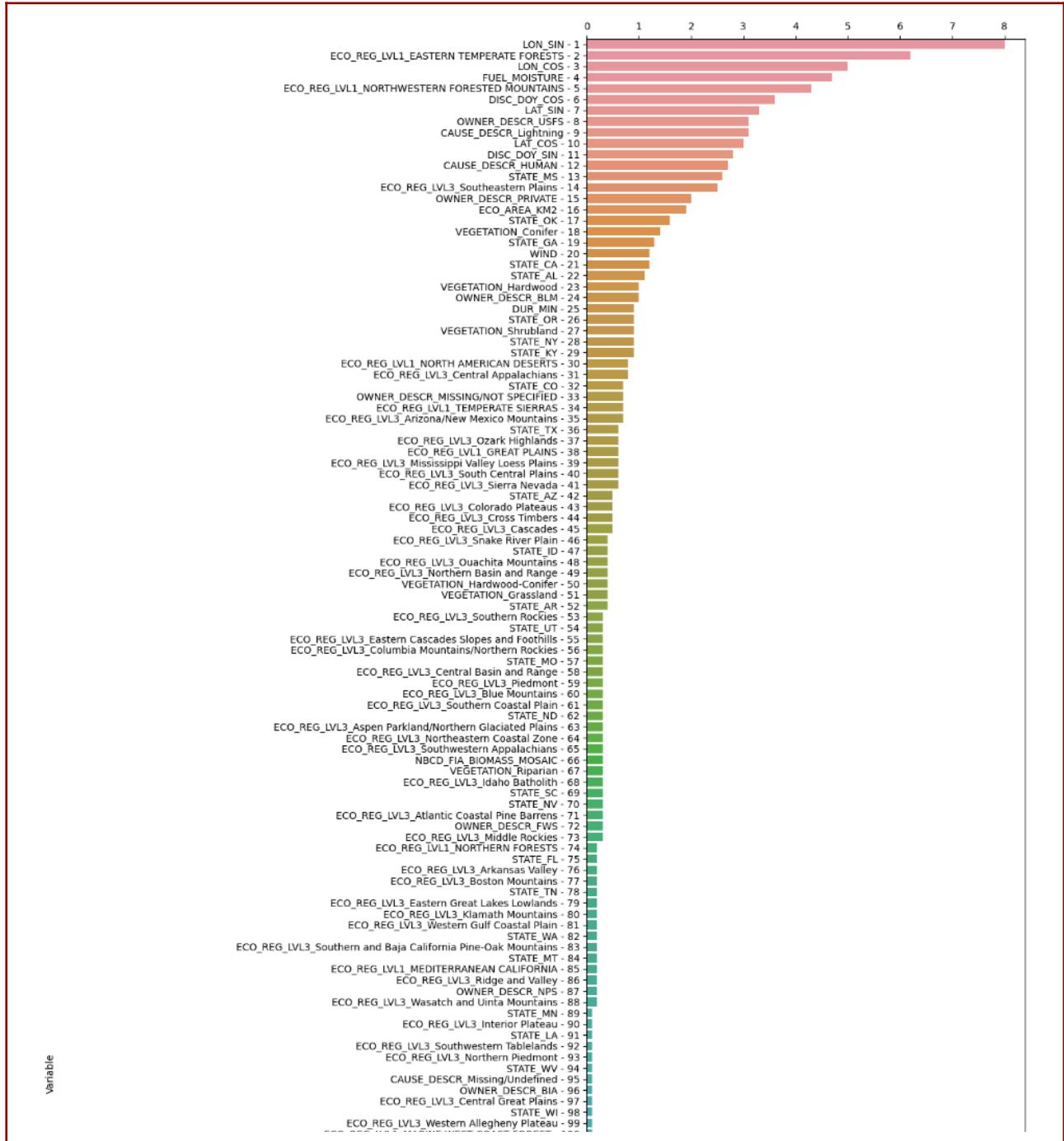
Des techniques d'over et d'undersampling ont été utilisées afin de rééquilibrer le dataset.

## V.1. Sélection des features : panorama

### V.1.a. KBest avec fonction “f\_classif”

La fonction f\_classif associée à la statistique F-test capture uniquement les dépendances linéaires.

Voici les résultats obtenus :

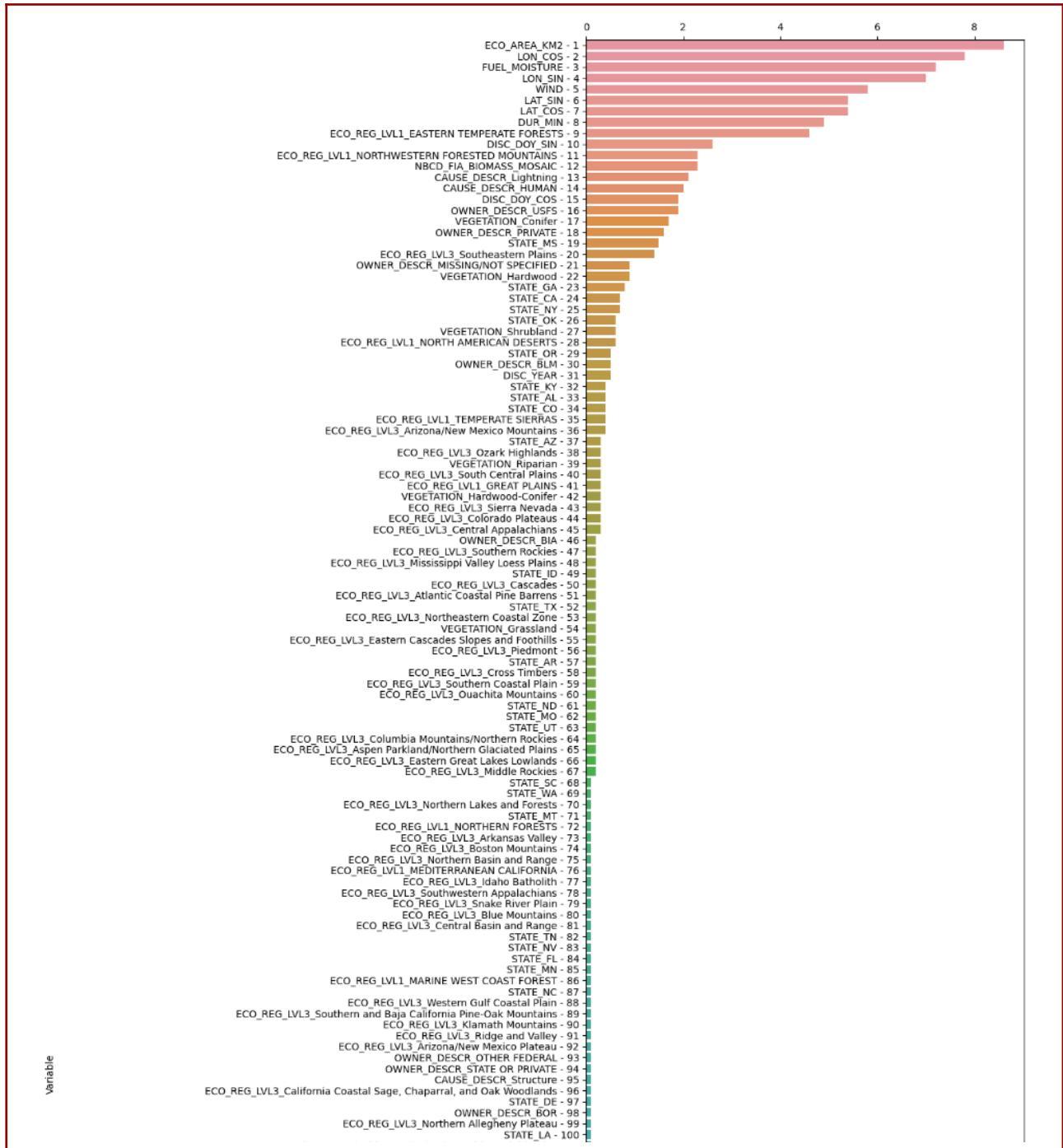


L'image a été volontairement coupée car trop grande, du fait des 200 variables à afficher.

## V.1.b. KBest avec fonction “mutual\_info\_classif”

La fonction mutual\_info\_classif associée à “l’information mutuelle” capture plutôt des patterns périodiques.

Voici les résultats obtenus :



L'image a été volontairement coupée car trop grande, du fait des 200 variables à afficher.

### V.1.c. KBest : premières constatations

Certaines colonnes sont repérées par les deux KBest : la longitude, la latitude, le jour de l'année, l'humidité du combustible.

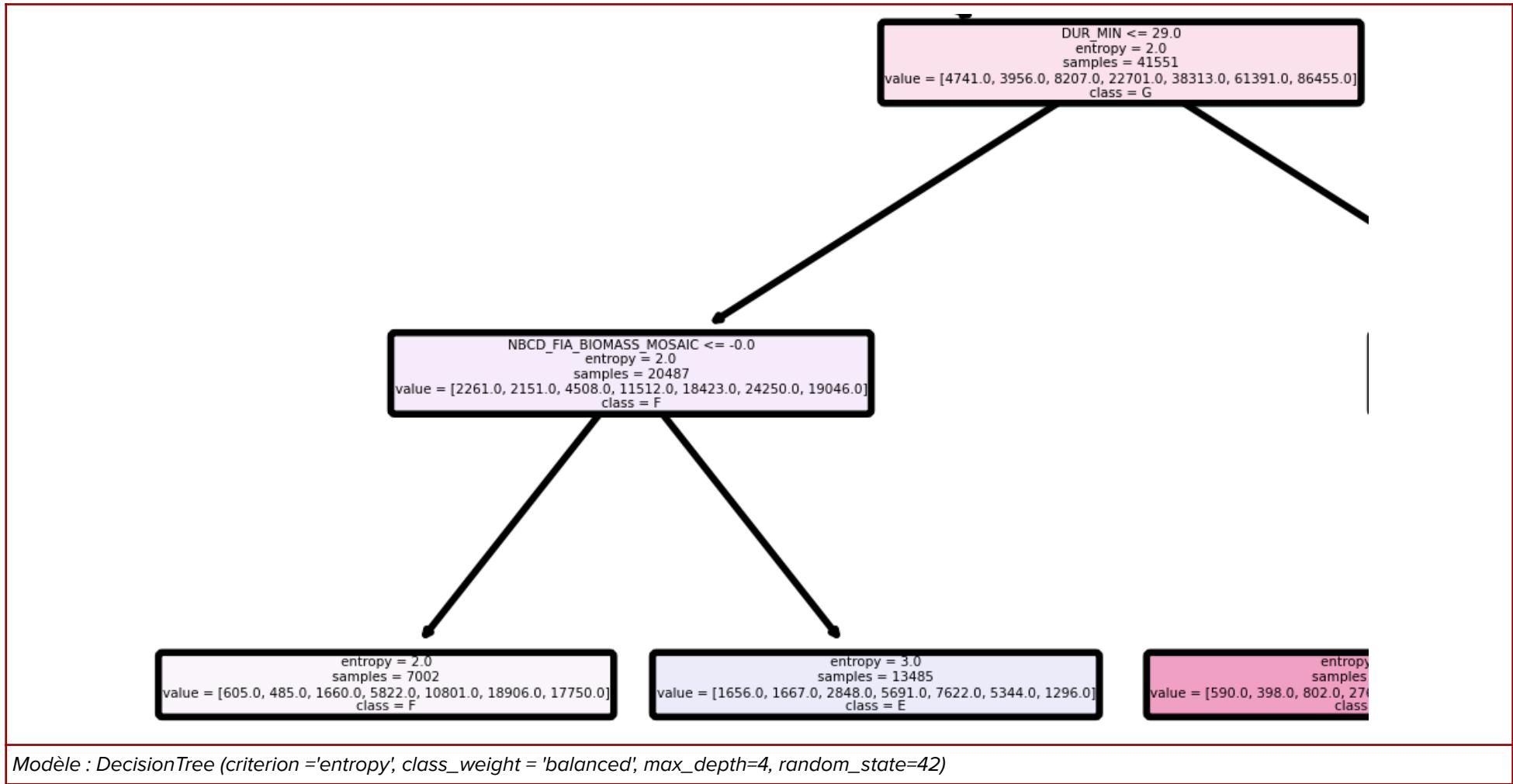
Cependant, étant basés sur des fonctions ne captant pas le même type de dépendance, on remarque quelques différences notables dans l'ordre et le poids des features. Le vent, la durée, l'indice de biomasse sont des exemples de features qui ont un poids plus important dans le deuxième KBest (fonction d'information mutuelle).

Bien que ce soit une première source d'informations, il conviendra d'adapter cette liste aux modèles utilisés.

## V.2. Arbre de décision

### V.2.a. Première approche avec plot\_tree

Comme attendu, la durée intervient à de nombreuses reprises dans cette version simpliste d'arbre de décision à 4 étages.



## V.2.b. Paramètre “class\_weight”

Le dataset étant très déséquilibré, il semblerait pertinent d'utiliser le paramètre “class\_weight” dans l'arbre de décision : cela pénalise plus fortement les erreurs de classification pour les classes minoritaires, ce qui a pour effet d'améliorer, normalement, l'apprentissage sur ces classes.

Matrice de confusion :										
Classe prédictive	A			B						
	A	B	C							
Classe réelle										
A	60807	13316	921	A	0.62	0.81	0.70	75044		
B	29362	41979	5667	B	0.67	0.55	0.60	77008		
C	4624	7215	6329	C	0.42	0.35	0.38	18168		
D	1220	342	1186	D	0.00	0.00	0.00	2748		
E	948	134	566	E	0.00	0.00	0.00	1648		
F	743	54	210	F	0.00	0.00	0.00	1007		
G	467	28	47	G	0.00	0.00	0.00	542		
accuracy							0.62	176165		
macro avg							0.24	0.24	0.24	176165
weighted avg							0.60	0.62	0.60	176165

Modèle : *DecisionTree (criterion =’gini’, max\_depth=5, min\_samples\_leaf= 10)*

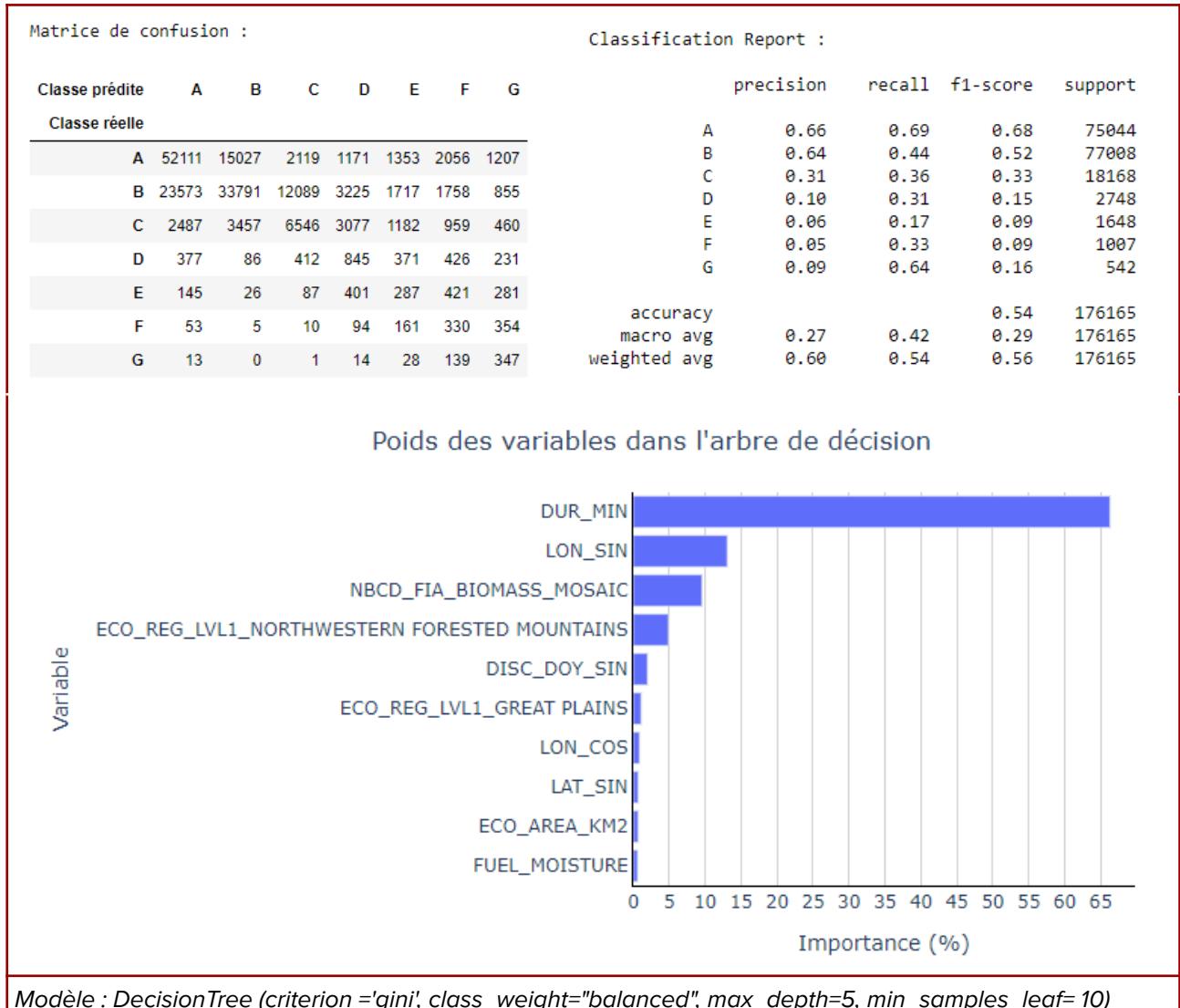
Matrice de confusion :												
Classe prédictive	A B C D E F G											
	A	B	C	D	E	F	G					
Classe réelle												
A	52111	15027	2119	1171	1353	2056	1207	A	0.66	0.69	0.68	75044
B	23573	33791	12089	3225	1717	1758	855	B	0.64	0.44	0.52	77008
C	2487	3457	6546	3077	1182	959	460	C	0.31	0.36	0.33	18168
D	377	86	412	845	371	426	231	D	0.10	0.31	0.15	2748
E	145	26	87	401	287	421	281	E	0.06	0.17	0.09	1648
F	53	5	10	94	161	330	354	F	0.05	0.33	0.09	1007
G	13	0	1	14	28	139	347	G	0.09	0.64	0.16	542
accuracy							0.54	176165				
macro avg							0.27	0.42	0.29	176165		
weighted avg							0.60	0.54	0.56	176165		

Modèle : *DecisionTree (criterion =’gini’, class\_weight=“balanced”, max\_depth=5, min\_samples\_leaf= 10)*

En effet, on constate ci-dessus que l'ajout du paramètre “class\_weight” permet à l'algorithme de détecter les classes de grands feux, ce qui était impossible auparavant.

## V.2.c. Causes humaines : regroupées ou non ?

Afin de réduire la taille du dataset et améliorer le temps de traitement, une colonne a été créée afin de regrouper les causes humaines sous une seule bannière.

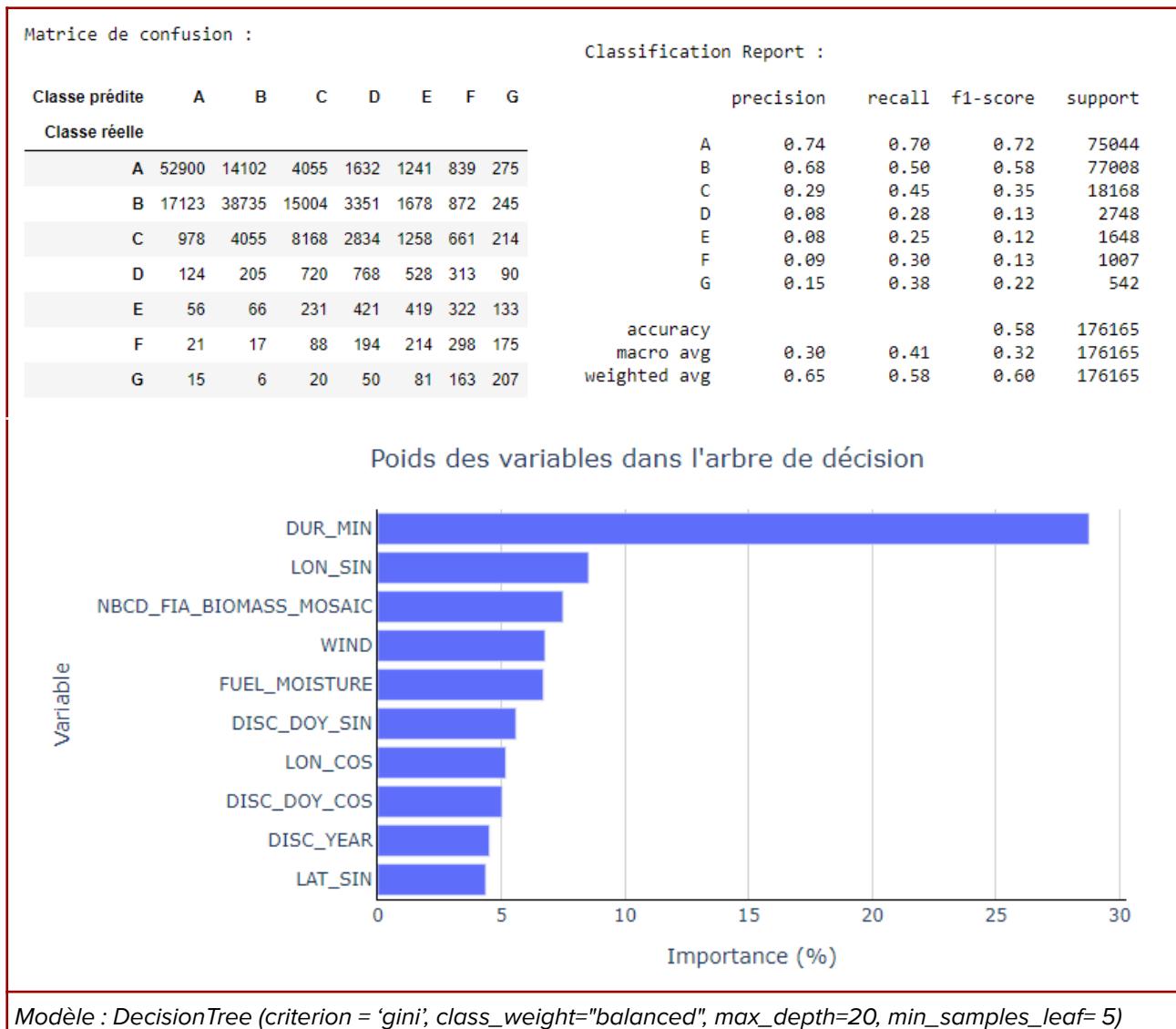


La séparation ou le regroupement des causes humaines ne semble pas avoir d'impact. On a donc décidé de les garder regroupées pour limiter le nombre de colonnes dans le dataset et donc alléger l'entraînement.

## V.2.d. Les features les plus importantes

On entraîne le modèle avec une profondeur plus importante et un nombre de records plus faible par feuille afin d'identifier les features ayant le plus de poids.

De ce fait, l'apprentissage du modèle est plus long que précédemment : 37 s contre 11 s.

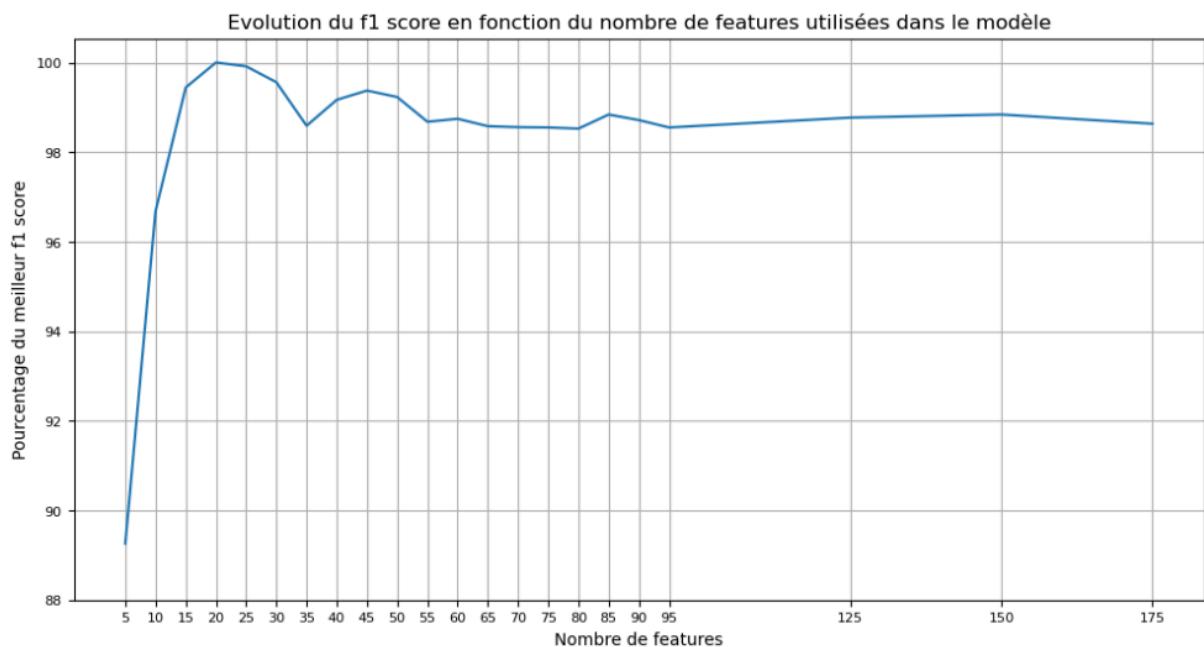


On remarque que le poids des premières features (durée du feu, sinus de la longitude, indice de biomasse) diminue par rapport à précédemment : l'arbre a réparti de manière un peu plus homogène les poids entre les features.

De plus, on voit apparaître de nouvelles features comme le vent ou l'humidité du combustible végétal.

Afin d'avoir une idée du nombre adéquat de features à garder, on trace le pourcentage du F1-score obtenu pour un modèle entraîné avec un nombre croissant de features.

La valeur "100 %" correspond au modèle ayant le F1-score le plus élevé.



Modèle : `DecisionTree (criterion = 'entropy', class_weight="balanced", max_depth=20, min_samples_leaf= 5)`

Le F1-score atteint un maximum de 0,329 pour 20 features. Dans la suite, on se limitera donc à ce nombre de features.

### V.2.e. GridSearch CV

Une précédente GridSearch a montré que le critère d'évaluation “entropy” était le plus pertinent

On applique la GridSearch CV suivante afin de trouver la meilleure combinaison d'hyperparamètres :

- 'criterion' : ['entropy']
- 'max\_depth' : [10,15,20,25]
- 'min\_samples\_leaf' : [1,2,5]
- 'min\_samples\_split' : [2,5]
- 'max\_features' : [2,5,10]

Le meilleur F1-score est de 0,328 et obtenu avec : {'criterion': 'entropy', 'max\_depth': 20, 'max\_features': 10, 'min\_samples\_leaf': 1, 'min\_samples\_split': 2}.

C'est un score assez faible mais est à l'effigie de la complexité du dataset et l'entremêlement des valeurs obtenues par chaque classe pour chacune des caractéristiques.

Matrice de confusion :							Classification Report :					
Classe prédictive	A	B	C	D	E	F	G		precision	recall	f1-score	support
Classe réelle								A	0.72	0.72	0.72	75044
A	53984	15699	3422	952	500	340	147	B	0.65	0.53	0.59	77008
B	18438	41048	14323	1873	824	366	136	C	0.30	0.46	0.37	18168
C	1520	5110	8382	1864	809	369	114	D	0.10	0.20	0.13	2748
D	291	471	861	549	330	189	57	E	0.11	0.19	0.14	1648
E	149	207	354	325	320	195	98	F	0.12	0.21	0.15	1007
F	81	109	168	131	187	211	120	accuracy			0.59	176165
G	43	63	46	46	74	121	149	macro avg	0.31	0.37	0.33	176165
								weighted avg	0.63	0.59	0.61	176165

Poids des variables dans l'arbre de décision

Variable	Importance (%)
DUR_MIN	~32
LON_COS	~9
LON_SIN	~8
NBCD_FIA_BIOMASS_MOSAIC	~6
FUEL_MOISTURE	~5.5
WIND	~5.5
LAT_COS	~5.5
LAT_SIN	~5
DISC_DOY_SIN	~5
DISC_DOY_COS	~5

Modèle : DecisionTree (criterion='entropy', max\_depth=20, max\_features=10, min\_samples\_leaf=1, min\_samples\_split=2)

### En conclusion :

- La prépondérance de la durée : plutôt logique car intuitivement, plus le feu s'étend, plus il dure longtemps
- Les variables de localisation : longitude, latitude
- Les variables “physiques” : vent, humidité de la végétation, indice de biomasse
- La périodicité dans l’année avec le jour de l’année
- Les 10 premières features sont toutes des variables continues
- On espère qu’un modèle plus robuste, comme une random forest, aura de meilleurs résultats

## V.3. Régression logistique

### V.3.a. Paramètres de la Régression Logistique

Le dataset étant très déséquilibré, il semble pertinent d'utiliser le paramètre “class\_weight” dans la régression logistique : cela pénalise plus fortement les erreurs de classification pour les classes minoritaires, ce qui a pour effet d'améliorer, normalement, l'apprentissage sur ces classes.

- Classification Report sans le “class\_weight”

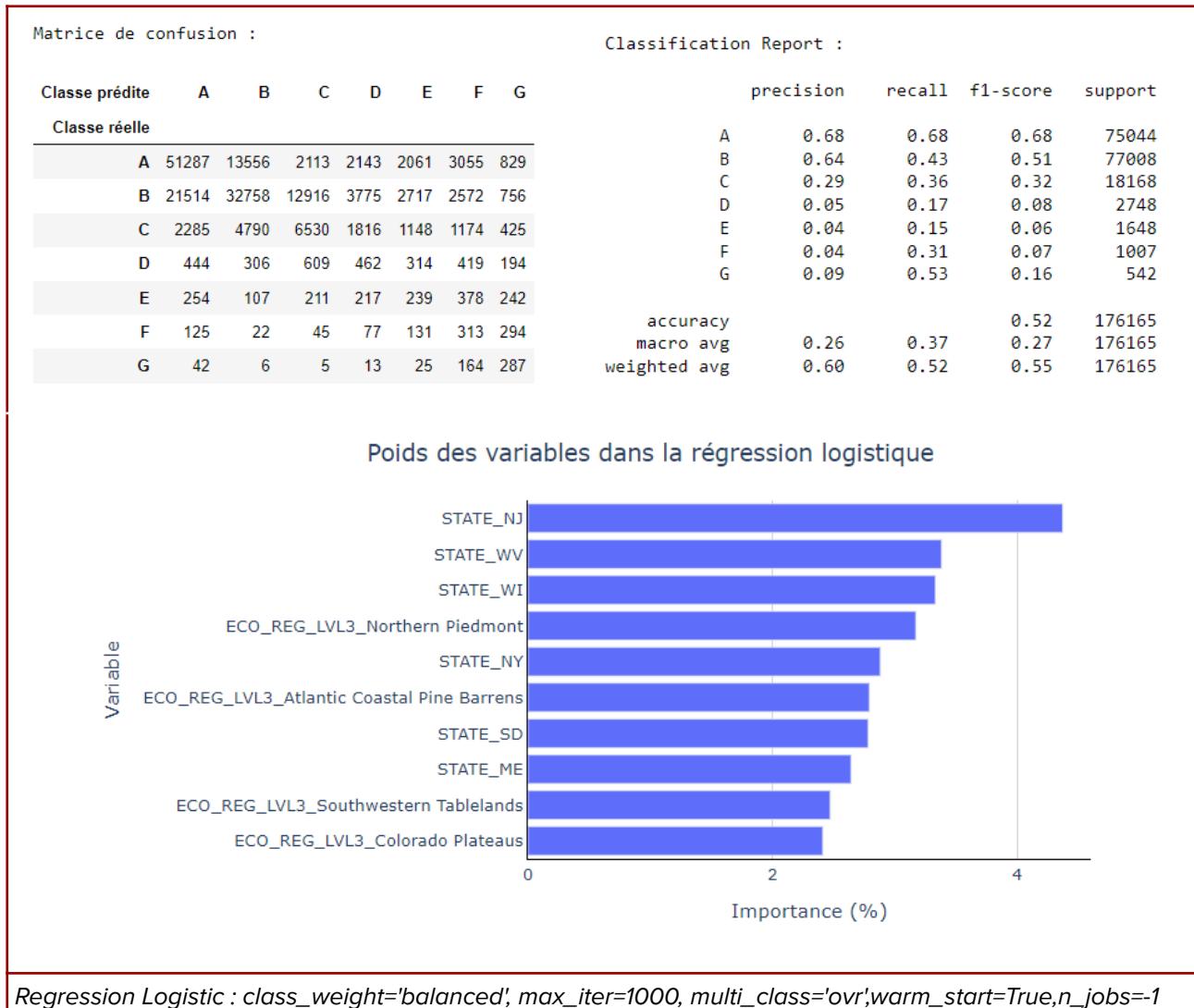
On constate qu'un modèle de régression logistique “basique” (sans ajustement de paramètre) n'est pas du tout concluant, en effet, certaines classes (F ou G par exemple) ne sont pas tout prédites :

Matrice de confusion :						precision	recall	f1-score	support		
Classe prédictive	A	B	C	D	G	A	0.64	0.69	0.66	75044	
Classe réelle						B	0.57	0.70	0.63	77008	
A	51751	23179	49	64	1	C	0.22	0.00	0.00	18168	
B	23025	53878	67	37	1	D	0.04	0.00	0.00	2748	
C	3606	14511	43	8	0	E	0.00	0.00	0.00	1648	
D	1006	1725	12	5	0	F	0.00	0.00	0.00	1007	
E	791	835	14	8	0	G	0.00	0.00	0.00	542	
F	599	393	11	4	0	accuracy		0.60	176165		
G	383	149	3	7	0	macro avg		0.21	0.20	0.19	176165
						weighted avg		0.54	0.60	0.56	176165
Variable Importance Cumul											
0						VEGETATION_Conifer	1.67	1.67			
1						LON_SIN	1.65	3.32			
2						ECO_REG_LVL1_EASTERN TEMPERATE FORESTS	1.50	4.82			
3						OWNER_DESCR_USFS	1.48	6.30			
4						STATE_NY	1.48	7.78			
5						CAUSE_DESCR_Miscellaneous	1.39	9.17			
6						OWNER_DESCR_MISSING/NOT SPECIFIED	1.38	10.55			
7						ECO_REG_LVL1_NORTHWESTERN FORESTED MOUNTAINS	1.37	11.92			
8						CAUSE_DESCR_Lightning	1.35	13.27			
9						STATE_GA	1.35	14.62			
10						CAUSE_DESCR_Campfire	1.30	15.92			
11						LAT_SIN	1.27	17.19			
12						STATE_AZ	1.24	18.43			
13						VEGETATION_Hardwood	1.22	19.65			
14						STATE_CO	1.21	20.86			
15						ECO_REG_LVL3_Arizona/New Mexico Mountains	1.19	22.05			
16						ECO_REG_LVL1_TEMPERATE SIERRAS	1.19	23.24			
17						VEGETATION_Riparian	1.19	24.43			
18						STATE_CA	1.19	25.62			
19						ECO_REG_LVL1_NORTH AMERICAN DESERTS	1.17	26.79			
20						OWNER_DESCR_PRIVATE	1.17	27.96			

Regression Logistic Classification Report sans le “class\_weight”

- Classification Report avec le “class\_weight”

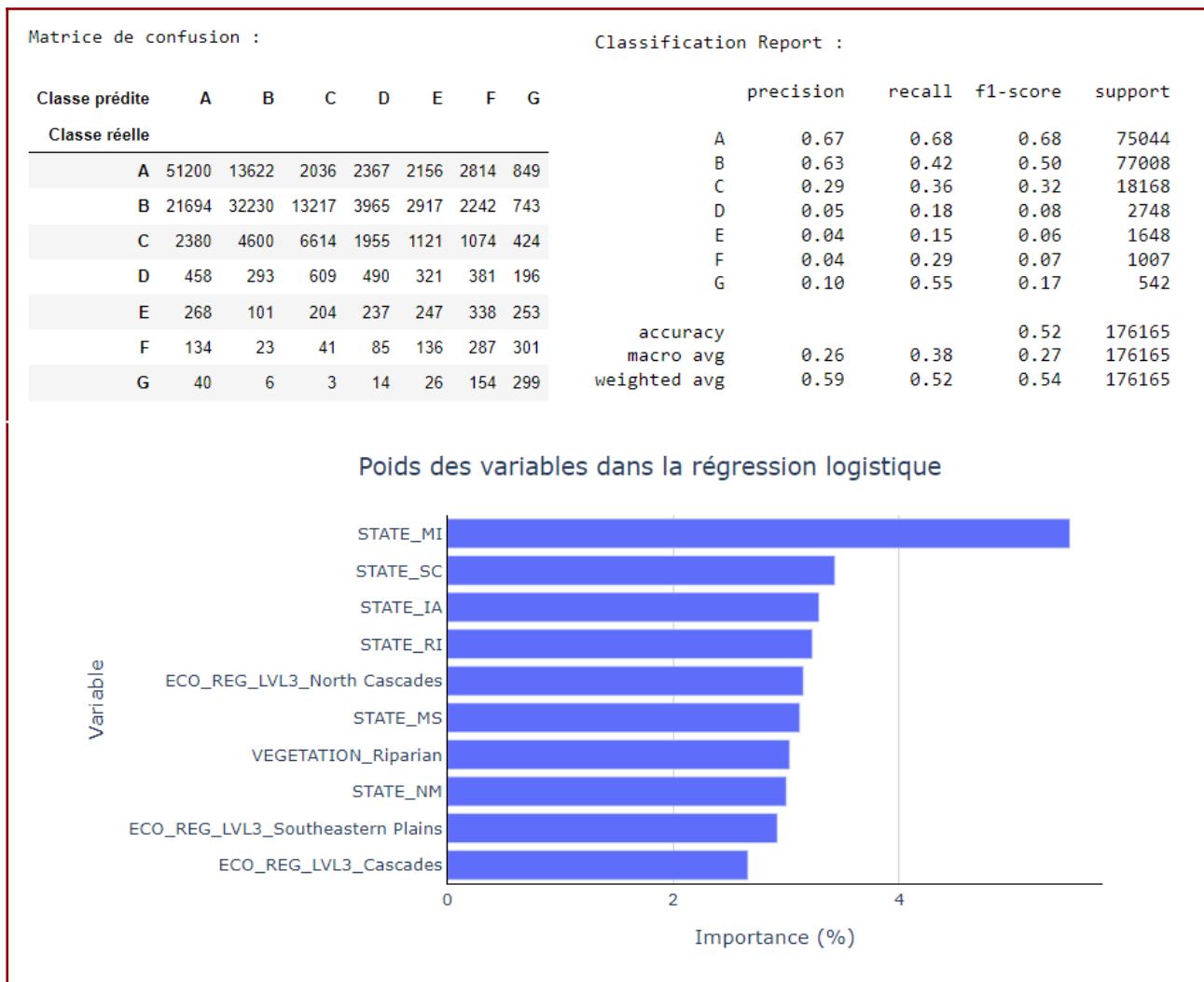
On constate que, grâce à la pénalisation des prédictions sur les classes minoritaires, toutes les classes de feu sont représentées.



### V.3.b. Causes humaines : regroupées ou non ?

Afin de réduire la taille du dataset et améliorer le temps de traitement, une colonne a été créée afin de regrouper les causes humaines sous une seule bannière.

Testons le modèle sur ce dataset réduit, avec pénalisation de rééquilibrage :



On décide de garder les causes humaines regroupées pour alléger le dataset et donc le temps de traitement.

### V.3.c. Les features les plus importantes

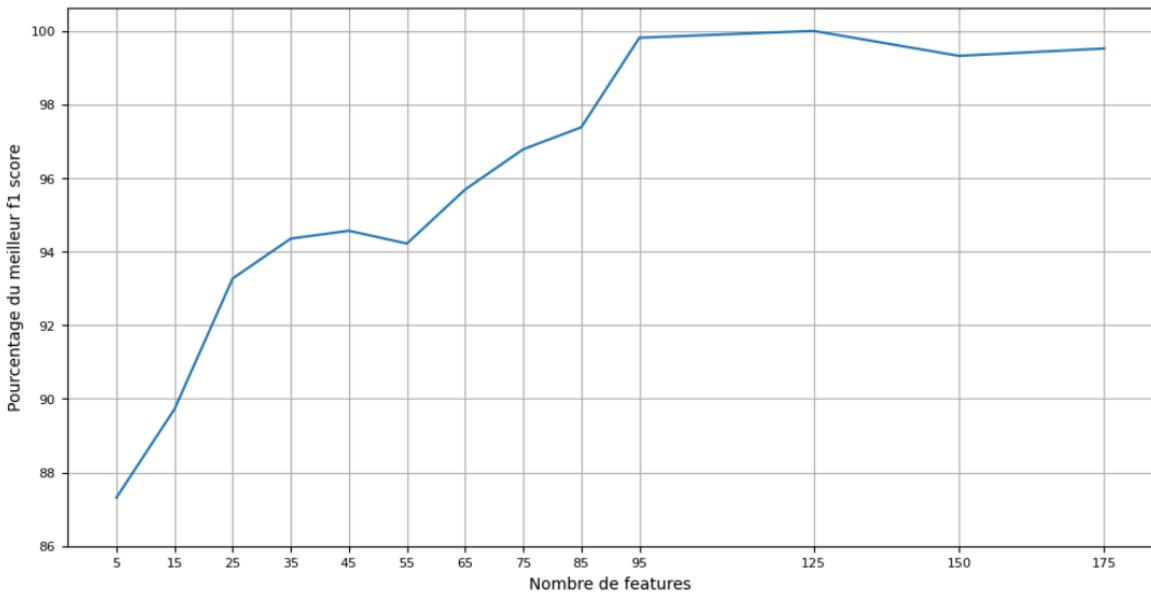
Évaluons l'importance de features d'après leur F1-score sur le modèle suivant.

Matrice de confusion :							Classification Report :					
Classe prédictive	A	B	C	D	E	F	G		precision	recall	f1-score	support
Classe réelle								A	0.67	0.68	0.68	75044
A	51224	13659	2048	2433	2164	2893	623	B	0.63	0.42	0.51	77008
B	21507	32543	13180	4082	2792	2364	540	C	0.29	0.36	0.32	18168
C	2361	4669	6622	1892	1146	1145	333	D	0.05	0.17	0.08	2748
D	456	300	608	466	331	428	159	E	0.03	0.14	0.06	1648
E	262	102	207	231	237	402	207	F	0.04	0.34	0.08	1007
F	127	24	39	77	134	344	262	accuracy			0.52	176165
G	39	6	4	13	22	180	278	macro avg	0.26	0.38	0.27	176165
								weighted avg	0.60	0.52	0.55	176165
Importance des features :												
	Variable	Importance	Cumul									
0	STATE_ME	9.000360	9.000360									
1	STATE_NJ	7.266401	16.266761									
2	STATE_NY	5.066092	21.332852									
3	LAT_COS	3.921490	25.254342									
4	STATE_RI	3.572737	28.827079									
5	STATE_CT	3.417473	32.244552									
6	STATE_NH	3.350413	35.594965									
7	STATE_WV	3.319739	38.914704									
8	STATE_SD	3.276093	42.190797									
9	ECO_REG_LVL3_Southern Michigan/Northern Indiana Drift Plains							VEGETATION_Savanna	2.343955	58.184124		
10	ECO_REG_LVL3_Atlantic Coastal Pine Barrens							OWNER_DESCR_NPS	2.335484	60.519608		
11	STATE_WI							ECO_REG_LVL3_Colorado Plateaus	2.332763	62.852371		
12	ECO_REG_LVL3_Northern Piedmont							ECO_REG_LVL3_Southwestern Tablelands	2.270062	65.122433		
13	STATE_PA							ECO_REG_LVL3_Northern Lakes and Forests	2.260955	67.383389		
14	VEGETATION_Savanna							ECO_REG_LVL3_North Central Hardwood Forests	2.188367	69.571755		
15	OWNER_DESCR_NPS							ECO_REG_LVL3_Arizona/New Mexico Plateau	2.018644	71.590399		
16	ECO_REG_LVL3_Colorado Plateaus											
17	ECO_REG_LVL3_Southwestern Tablelands											
18	ECO_REG_LVL3_Northern Lakes and Forests											
19	ECO_REG_LVL3_North Central Hardwood Forests											
20	ECO_REG_LVL3_Arizona/New Mexico Plateau											

LogisticRegression : class\_weight='balanced', max\_iter=1000, multi\_class='ovr', warm\_start=True, n\_jobs=-1, random\_state=42

Le meilleur score est 0.265, atteint pour 125.

Evolution du f1 score en fonction du nombre de features utilisées dans le modèle



A la lecture du graphique, on décide de tester le modèle sur 2 groupes de features :

- Jusqu'à 45
- Jusqu'à 95

### Réduction du nombre de features dans notre modèle pour l'optimiser :

Classe prédictive	A	B	C	D	E	F	G	Classification Report :				
Classe réelle								precision	recall	f1-score	support	
A	37841	9673	7901	3036	2917	2099	11577	A	0.65	0.50	0.57	75044
B	17580	20550	22088	3788	1845	3606	7551	B	0.58	0.27	0.37	77008
C	1895	4707	7504	1201	438	723	1700	C	0.19	0.41	0.26	18168
D	396	375	868	279	113	209	508	D	0.03	0.10	0.05	2748
E	287	154	408	136	87	153	423	E	0.02	0.05	0.02	1648
F	243	70	133	48	58	115	340	F	0.02	0.11	0.03	1007
G	152	23	23	19	18	47	260	G	0.01	0.48	0.02	542
							accuracy			0.38	176165	
							macro avg			0.19	176165	
							weighted avg			0.43	176165	

Réduction à 45 features

Classe prédictive	A	B	C	D	E	F	G	Classification Report :				
Classe réelle								precision	recall	f1-score	support	
A	49903	13285	3207	2257	2772	2737	883	A	0.67	0.66	0.67	75044
B	21393	29845	16290	3693	2803	2198	786	B	0.62	0.39	0.48	77008
C	2355	4558	7548	1400	924	963	420	C	0.27	0.42	0.33	18168
D	455	309	791	365	280	358	190	D	0.05	0.13	0.07	2748
E	270	115	312	177	202	330	242	E	0.03	0.12	0.05	1648
F	139	41	91	60	107	306	263	F	0.04	0.30	0.08	1007
G	49	7	13	8	23	163	279	G	0.09	0.51	0.15	542
							accuracy			0.50	176165	
							macro avg			0.26	176165	
							weighted avg			0.53	176165	

Réduction à 95 features

On constate donc que de réduire le nombre de features n'a pas d'impact positif, au contraire, cela détériore les résultats du modèle.

### V.3.d. GridSearch CV

'C'	'penalty'	'multi_class'	'solver'	résultats																											
[1e-4, 1]	['l1', 'elasticnet']	['ovr']	['sag', 'saga']	<table border="1"> <thead> <tr> <th></th><th>params</th><th>mean_test_score</th></tr> </thead> <tbody> <tr> <td>5</td><td>{'C': 1, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'saga'}</td><td>0.376115</td></tr> <tr> <td>1</td><td>{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'saga'}</td><td>0.367575</td></tr> <tr> <td>0</td><td>{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'sag'}</td><td>NaN</td></tr> <tr> <td>2</td><td>{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'elasticnet', 'solver': 'sag'}</td><td>NaN</td></tr> <tr> <td>3</td><td>{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'elasticnet', 'solver': 'saga'}</td><td>NaN</td></tr> <tr> <td>4</td><td>{'C': 1, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'sag'}</td><td>NaN</td></tr> <tr> <td>6</td><td>{'C': 1, 'multi_class': 'ovr', 'penalty': 'elasticnet', 'solver': 'sag'}</td><td>NaN</td></tr> <tr> <td>7</td><td>{'C': 1, 'multi_class': 'ovr', 'penalty': 'elasticnet', 'solver': 'saga'}</td><td>NaN</td></tr> </tbody> </table>		params	mean_test_score	5	{'C': 1, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'saga'}	0.376115	1	{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'saga'}	0.367575	0	{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'sag'}	NaN	2	{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'elasticnet', 'solver': 'sag'}	NaN	3	{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'elasticnet', 'solver': 'saga'}	NaN	4	{'C': 1, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'sag'}	NaN	6	{'C': 1, 'multi_class': 'ovr', 'penalty': 'elasticnet', 'solver': 'sag'}	NaN	7	{'C': 1, 'multi_class': 'ovr', 'penalty': 'elasticnet', 'solver': 'saga'}	NaN
	params	mean_test_score																													
5	{'C': 1, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'saga'}	0.376115																													
1	{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'saga'}	0.367575																													
0	{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'sag'}	NaN																													
2	{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'elasticnet', 'solver': 'sag'}	NaN																													
3	{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'elasticnet', 'solver': 'saga'}	NaN																													
4	{'C': 1, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'sag'}	NaN																													
6	{'C': 1, 'multi_class': 'ovr', 'penalty': 'elasticnet', 'solver': 'sag'}	NaN																													
7	{'C': 1, 'multi_class': 'ovr', 'penalty': 'elasticnet', 'solver': 'saga'}	NaN																													
[1e-4, 1]	['l1', 'l2']	['ovr']	['lbfgs', 'newton-cg']	<table border="1"> <thead> <tr> <th></th><th>params</th><th>mean_test_score</th></tr> </thead> <tbody> <tr> <td>2</td><td>{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'l2', 'solver': 'lbfgs'}</td><td>0.504833</td></tr> <tr> <td>3</td><td>{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'l2', 'solver': 'newton-cg'}</td><td>0.504695</td></tr> <tr> <td>7</td><td>{'C': 1, 'multi_class': 'ovr', 'penalty': 'l2', 'solver': 'newton-cg'}</td><td>0.502023</td></tr> <tr> <td>6</td><td>{'C': 1, 'multi_class': 'ovr', 'penalty': 'l2', 'solver': 'lbfgs'}</td><td>0.489657</td></tr> <tr> <td>0</td><td>{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'lbfgs'}</td><td>NaN</td></tr> <tr> <td>1</td><td>{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'newton-cg'}</td><td>NaN</td></tr> <tr> <td>4</td><td>{'C': 1, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'lbfgs'}</td><td>NaN</td></tr> <tr> <td>5</td><td>{'C': 1, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'newton-cg'}</td><td>NaN</td></tr> </tbody> </table>		params	mean_test_score	2	{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'l2', 'solver': 'lbfgs'}	0.504833	3	{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'l2', 'solver': 'newton-cg'}	0.504695	7	{'C': 1, 'multi_class': 'ovr', 'penalty': 'l2', 'solver': 'newton-cg'}	0.502023	6	{'C': 1, 'multi_class': 'ovr', 'penalty': 'l2', 'solver': 'lbfgs'}	0.489657	0	{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'lbfgs'}	NaN	1	{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'newton-cg'}	NaN	4	{'C': 1, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'lbfgs'}	NaN	5	{'C': 1, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'newton-cg'}	NaN
	params	mean_test_score																													
2	{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'l2', 'solver': 'lbfgs'}	0.504833																													
3	{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'l2', 'solver': 'newton-cg'}	0.504695																													
7	{'C': 1, 'multi_class': 'ovr', 'penalty': 'l2', 'solver': 'newton-cg'}	0.502023																													
6	{'C': 1, 'multi_class': 'ovr', 'penalty': 'l2', 'solver': 'lbfgs'}	0.489657																													
0	{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'lbfgs'}	NaN																													
1	{'C': 0.0001, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'newton-cg'}	NaN																													
4	{'C': 1, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'lbfgs'}	NaN																													
5	{'C': 1, 'multi_class': 'ovr', 'penalty': 'l1', 'solver': 'newton-cg'}	NaN																													
[1, 1e4]	['l1','l2']	['multinomial']	['sag', 'saga']	<table border="1"> <thead> <tr> <th></th><th>params</th><th>mean_test_score</th></tr> </thead> <tbody> <tr> <td>2</td><td>{'C': 1, 'multi_class': 'multinomial', 'penalty': 'l2', 'solver': 'sag'}</td><td>0.345005</td></tr> <tr> <td>6</td><td>{'C': 10000.0, 'multi_class': 'multinomial', 'penalty': 'l2', 'solver': 'sag'}</td><td>0.345005</td></tr> <tr> <td>1</td><td>{'C': 1, 'multi_class': 'multinomial', 'penalty': 'l1', 'solver': 'saga'}</td><td>0.330138</td></tr> <tr> <td>3</td><td>{'C': 1, 'multi_class': 'multinomial', 'penalty': 'l2', 'solver': 'saga'}</td><td>0.330137</td></tr> <tr> <td>5</td><td>{'C': 10000.0, 'multi_class': 'multinomial', 'penalty': 'l1', 'solver': 'saga'}</td><td>0.330137</td></tr> <tr> <td>7</td><td>{'C': 10000.0, 'multi_class': 'multinomial', 'penalty': 'l2', 'solver': 'saga'}</td><td>0.330137</td></tr> <tr> <td>0</td><td>{'C': 1, 'multi_class': 'multinomial', 'penalty': 'l1', 'solver': 'sag'}</td><td>NaN</td></tr> <tr> <td>4</td><td>{'C': 10000.0, 'multi_class': 'multinomial', 'penalty': 'l1', 'solver': 'sag'}</td><td>NaN</td></tr> </tbody> </table>		params	mean_test_score	2	{'C': 1, 'multi_class': 'multinomial', 'penalty': 'l2', 'solver': 'sag'}	0.345005	6	{'C': 10000.0, 'multi_class': 'multinomial', 'penalty': 'l2', 'solver': 'sag'}	0.345005	1	{'C': 1, 'multi_class': 'multinomial', 'penalty': 'l1', 'solver': 'saga'}	0.330138	3	{'C': 1, 'multi_class': 'multinomial', 'penalty': 'l2', 'solver': 'saga'}	0.330137	5	{'C': 10000.0, 'multi_class': 'multinomial', 'penalty': 'l1', 'solver': 'saga'}	0.330137	7	{'C': 10000.0, 'multi_class': 'multinomial', 'penalty': 'l2', 'solver': 'saga'}	0.330137	0	{'C': 1, 'multi_class': 'multinomial', 'penalty': 'l1', 'solver': 'sag'}	NaN	4	{'C': 10000.0, 'multi_class': 'multinomial', 'penalty': 'l1', 'solver': 'sag'}	NaN
	params	mean_test_score																													
2	{'C': 1, 'multi_class': 'multinomial', 'penalty': 'l2', 'solver': 'sag'}	0.345005																													
6	{'C': 10000.0, 'multi_class': 'multinomial', 'penalty': 'l2', 'solver': 'sag'}	0.345005																													
1	{'C': 1, 'multi_class': 'multinomial', 'penalty': 'l1', 'solver': 'saga'}	0.330138																													
3	{'C': 1, 'multi_class': 'multinomial', 'penalty': 'l2', 'solver': 'saga'}	0.330137																													
5	{'C': 10000.0, 'multi_class': 'multinomial', 'penalty': 'l1', 'solver': 'saga'}	0.330137																													
7	{'C': 10000.0, 'multi_class': 'multinomial', 'penalty': 'l2', 'solver': 'saga'}	0.330137																													
0	{'C': 1, 'multi_class': 'multinomial', 'penalty': 'l1', 'solver': 'sag'}	NaN																													
4	{'C': 10000.0, 'multi_class': 'multinomial', 'penalty': 'l1', 'solver': 'sag'}	NaN																													
[1, 1e4]	['l1','l2']	['multinomial']	['lbfgs', 'newton-cg']	Killed - trop long																											

[1]	['l1','l2']	['multinomial']	['lbfgs']	
				params mean_test_score
				1 {C: 1, 'multi_class': 'multinomial', 'penalty': 'l2', 'solver': 'lbfgs'} 0.417253 0 {C: 1, 'multi_class': 'multinomial', 'penalty': 'l1', 'solver': 'lbfgs'} NaN

[1]	['l1','l2']	['multinomial']	['newton-cg']	
				params mean_test_score
				1 {C: 1, 'multi_class': 'multinomial', 'penalty': 'l2', 'solver': 'lbfgs'} 0.417253 0 {C: 1, 'multi_class': 'multinomial', 'penalty': 'l1', 'solver': 'lbfgs'} NaN

Après ces différents essais en jouant sur les hyperparamètres, nous pouvons déduire que l'hyperparamètre 'C' n'a pas vraiment d'influence sur les résultats, ainsi que les paramètres 'solver':['sag'], 'penalty':['elasticnet']

Mais si nous devons tirer de ces résultats les meilleurs hyperparamètres, nous mettrons en exergue :

- 'C' = [1e-4]
- 'Penalty' = ['l2']
- 'Multi\_class' = ['ovr']
- 'Solver' = ['lbfgs']

Matrice de confusion :							Classification Report :					
Classe prédictive	A	B	C	D	E	F	G		precision	recall	f1-score	support
Classe réelle								A	0.65	0.62	0.64	75044
A	46845	16393	3604	1314	3299	2565	1024	B	0.59	0.43	0.50	77008
B	21634	33444	15061	1804	2948	1408	709	C	0.26	0.39	0.31	18168
C	2477	6209	7031	718	664	709	360	D	0.06	0.09	0.07	2748
D	479	466	887	239	234	270	173	E	0.02	0.09	0.03	1648
E	294	185	388	143	145	284	209	F	0.04	0.24	0.07	1007
F	161	68	134	50	96	237	261	G	0.09	0.50	0.15	542
G	46	13	27	10	33	142	271	accuracy		0.50	176165	
								macro avg	0.24	0.34	0.25	176165
								weighted avg	0.56	0.50	0.52	176165

	params	mean_test_score
0 {C: 0.0001, 'multi_class': 'ovr', 'penalty': 'l2', 'solver': 'lbfgs'}		0.504833

```
param_grid7 = {'C': [1e-4], 'penalty': ['l2'], 'multi_class': ['ovr'], 'solver': ['lbfgs']}
grid_search7 = GridSearchCV(LogisticRegression(class_weight='balanced', max_iter=100,
                                              warm_start=True, n_jobs=-1, random_state=42), param_grid7, cv=2)
grid_search7.fit(X_train_df_human_cause.loc[:,importances.iloc[:95,0]], y_train)
```

### En conclusion :

Après de nombreux essais, et beaucoup de temps, la GridSearch n'amène aucune plus-value au modèle de Régression Logistique sur ce Dataset, étant donné que les résultats d'apprentissage de la Régression Logistique avec pénalisation de rééquilibrage sont "meilleurs" (de 0.02%), même si ceux-ci ne sont pas probants non plus.

## V.4. Forêt aléatoire

### V.4.a. Paramètres “class\_weight”

Le dataset étant très déséquilibré, il semblerait pertinent d'utiliser le paramètre “class\_weight” dans l'arbre de décision : cela pénalise plus fortement les erreurs de classification pour les classes minoritaires, ce qui a pour effet d'améliorer, normalement, l'apprentissage sur ces classes.

Classe prédictive	Classification Report :										
	A	B	C	D	E	F	G	precision	recall	f1-score	support
<b>Classe réelle</b>											
<b>A</b>	59503	15158	331	18	17	10	7	A	0.74	0.79	0.76
<b>B</b>	18436	55411	3036	65	26	24	10	B	0.66	0.72	0.69
<b>C</b>	1657	10951	5254	184	71	39	12	C	0.50	0.29	0.37
<b>D</b>	348	1096	1011	168	76	42	7	D	0.27	0.06	0.10
<b>E</b>	258	566	507	109	127	63	18	E	0.29	0.08	0.12
<b>F</b>	203	332	220	58	84	84	26	F	0.26	0.08	0.13
<b>G</b>	117	173	74	21	32	62	63	G	0.44	0.12	0.18
accuracy											
macro avg											
weighted avg											
0.45											
0.31											
0.34											
0.68											
176165											
176165											
176165											

Modèle : RandomForestClassifier(random\_state=42)

Classe prédictive	Classification Report :										
	A	B	C	D	E	F	G	precision	recall	f1-score	support
<b>Classe réelle</b>											
<b>A</b>	59272	15413	308	15	10	13	13	A	0.73	0.79	0.76
<b>B</b>	18843	55401	2656	51	22	24	11	B	0.66	0.72	0.69
<b>C</b>	1806	11400	4704	148	63	34	13	C	0.51	0.26	0.34
<b>D</b>	394	1163	917	167	61	39	7	D	0.30	0.06	0.10
<b>E</b>	308	580	448	97	128	61	26	E	0.33	0.08	0.13
<b>F</b>	219	332	185	53	76	101	41	F	0.30	0.10	0.15
<b>G</b>	132	152	60	18	25	60	95	G	0.46	0.18	0.25
accuracy											
macro avg											
0.47											
0.31											
0.35											
0.68											
176165											
176165											
176165											

Modèle : RandomForestClassifier(random\_state=42, class\_weight = "balanced")

Matrice de confusion :

Classification Report :

Classe prédictive	A	B	C	D	E	F	G	precision	recall	f1-score	support
Classe réelle											
<b>A</b>	59269	15410	321	12	10	13	9	<b>A</b>	<b>0.73</b>	<b>0.79</b>	<b>0.76</b>
<b>B</b>	18835	55425	2653	41	23	20	11	<b>B</b>	<b>0.66</b>	<b>0.72</b>	<b>0.69</b>
<b>C</b>	1789	11331	4786	159	57	33	13	<b>C</b>	<b>0.51</b>	<b>0.26</b>	<b>0.35</b>
<b>D</b>	393	1158	926	160	65	39	7	<b>D</b>	<b>0.30</b>	<b>0.06</b>	<b>0.10</b>
<b>E</b>	283	583	471	102	125	62	22	<b>E</b>	<b>0.32</b>	<b>0.08</b>	<b>0.12</b>
<b>F</b>	237	316	197	46	80	95	36	<b>F</b>	<b>0.29</b>	<b>0.09</b>	<b>0.14</b>
<b>G</b>	139	133	50	17	36	65	102	<b>G</b>	<b>0.51</b>	<b>0.19</b>	<b>0.27</b>
								accuracy		<b>0.68</b>	176165
								macro avg	<b>0.47</b>	<b>0.31</b>	176165
								weighted avg	<b>0.66</b>	<b>0.68</b>	176165

Modèle : `RandomForestClassifier(random_state=42, class_weight = "balanced_subsample")`

On remarque que le paramètre `class_weight` influe légèrement sur les F1 scores obtenus (+0,01 point sur la macro average).

Le paramètre renseigné avec “balanced\_subsample” permet d’obtenir un meilleur score sur les classes C et G, mais le F1 score sur les autres classes minoritaires E et F (D reste identique) qu’avec le paramètre renseigné “balanced”.

On remarque la prédominance de deux variables : les classes A et B, majoritaire dans le jeu de données.

Le paramètre `class_weight = “balanced”` sera privilégié pour les prochains modèles.

#### V.4.b. Causes humaines : regroupées ou non ?

Afin de réduire la taille du dataset et améliorer le temps de traitement, une colonne a été créée afin de regrouper les causes humaines sous une seule bannière.

Matrice de confusion :

Classification Report :

Classe prédictive	A	B	C	D	E	F	G	precision	recall	f1-score	support	
Classe réelle								A	0.74	0.79	0.76	75044
<b>A</b>	59378	15352	274	14	6	9	11	B	0.66	0.72	0.69	77008
<b>B</b>	18533	55805	2582	29	29	19	11	C	0.51	0.26	0.34	18168
<b>C</b>	1772	11437	4708	141	67	29	14	D	0.31	0.05	0.09	2748
<b>D</b>	402	1155	938	151	61	33	8	E	0.32	0.07	0.12	1648
<b>E</b>	278	609	473	90	119	56	23	F	0.30	0.09	0.14	1007
<b>F</b>	222	329	200	52	71	91	42	G	0.48	0.19	0.27	542
<b>G</b>	130	143	66	11	21	69	102	accuracy		0.68	176165	
								macro avg	0.47	0.31	0.35	176165
								weighted avg	0.66	0.68	0.67	176165

Dataset avec toutes les causes humaines

Modèle : RandomForestClassifier(random\_state=42, class\_weight = "balanced")

Matrice de confusion :

Classification Report :

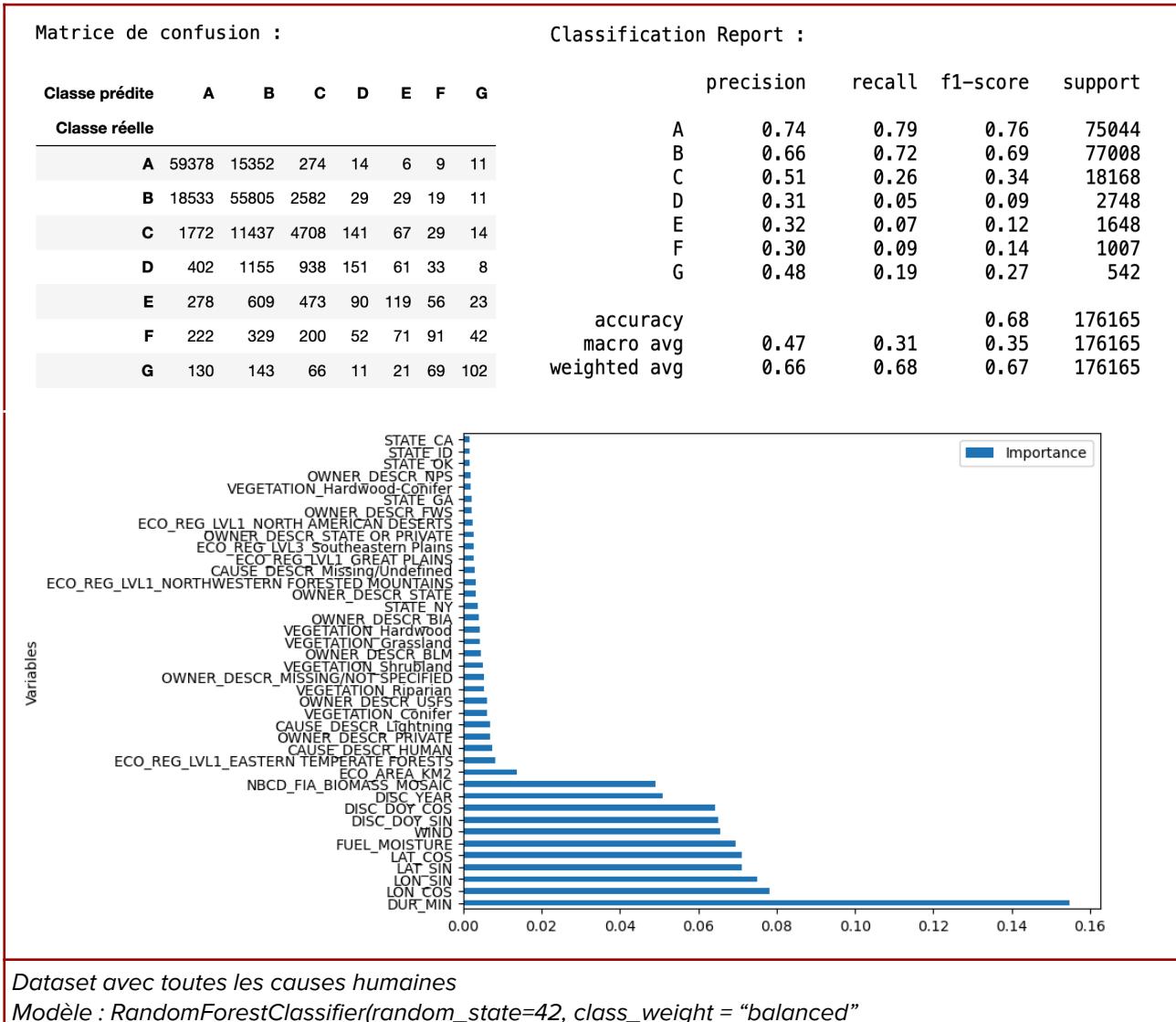
Classe prédictive	A	B	C	D	E	F	G	precision	recall	f1-score	support	
Classe réelle								A	0.73	0.79	0.76	75044
<b>A</b>	59272	15413	308	15	10	13	13	B	0.66	0.72	0.69	77008
<b>B</b>	18843	55401	2656	51	22	24	11	C	0.51	0.26	0.34	18168
<b>C</b>	1806	11400	4704	148	63	34	13	D	0.30	0.06	0.10	2748
<b>D</b>	394	1163	917	167	61	39	7	E	0.33	0.08	0.13	1648
<b>E</b>	308	580	448	97	128	61	26	F	0.30	0.10	0.15	1007
<b>F</b>	219	332	185	53	76	101	41	G	0.46	0.18	0.25	542
<b>G</b>	132	152	60	18	25	60	95	accuracy		0.68	176165	
								macro avg	0.47	0.31	0.35	176165
								weighted avg	0.66	0.68	0.66	176165

Dataset avec les causes humaines regroupées

Modèle : RandomForestClassifier(random\_state=42, class\_weight = "balanced")

La séparation ou le regroupement des causes humaines n'a visiblement pas ou peu d'impact sur le score de notre modèle. On a donc décidé de les garder regroupées pour limiter le nombre de colonnes dans le dataset et donc alléger l'entraînement.

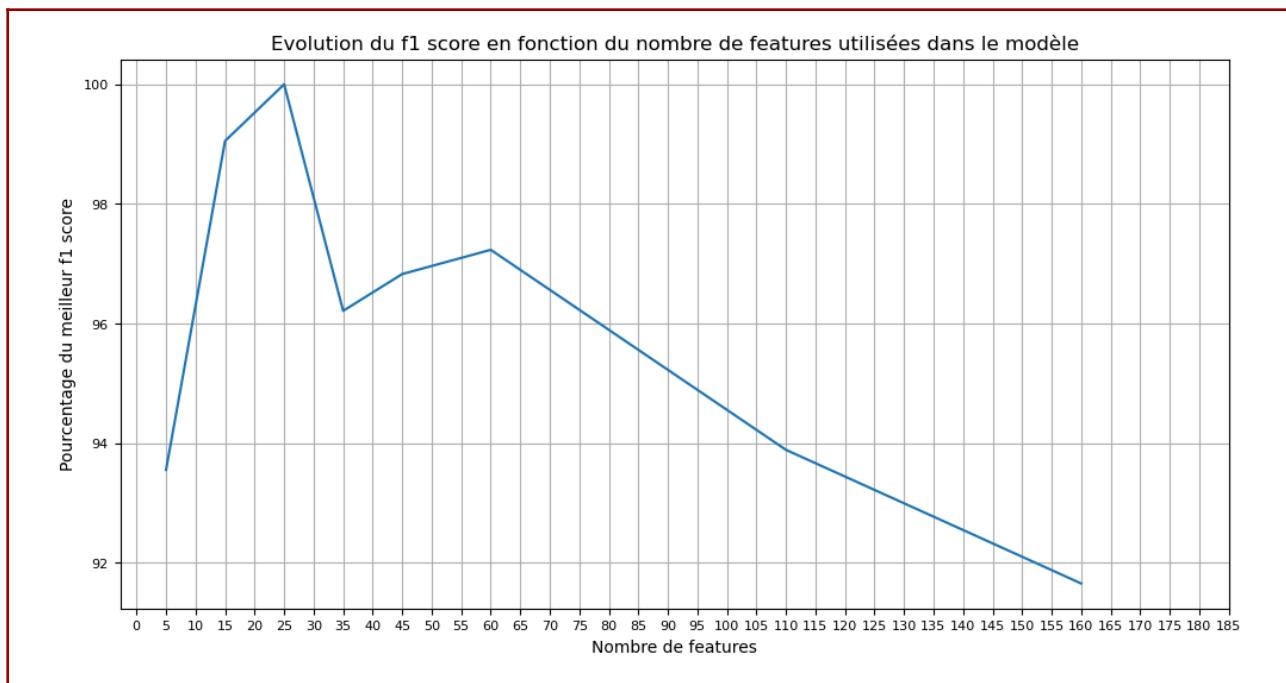
## V.4.c. Les features les plus importantes



Quand nous regardons la figure ci-dessus, illustrant les 40 premières features issues de la feature\_importance\_, nous remarquons qu'il y a une forte baisse à partir de la 12ème (ECO\_AREA\_KM2).

Comme sur les modèles précédents, nous cherchons à connaître le nombre optimal de features à appliquer à notre modèle de classification.

Si nous observons la courbe du f1 score, nous remarquons que le score optimal devrait être trouvé en prenant les 25 premières features les plus importantes.



A partir de cette courbe, nous décidons de faire tourner notre modèle avec 15, 25 et 60 features, qui correspondent aux pics observés.

Pour rappel, nous gardons le paramètre `class_weight = "balanced"` sur nos modèles.

- `RandomForestClassifier` avec 15 features.

Matrice de confusion :							Classification Report :					
Classe prédictive	A	B	C	D	E	F	G		precision	recall	f1-score	support
Classe réelle								A	0.74	0.79	0.77	75044
A	59445	15148	379	20	15	18	19	B	0.67	0.72	0.69	77008
B	17970	55747	3155	53	27	28	28	C	0.50	0.30	0.38	18168
C	1543	10852	5458	158	83	45	29	D	0.31	0.06	0.11	2748
D	345	1031	1029	176	88	65	14	E	0.28	0.08	0.12	1648
E	235	527	525	97	130	107	27	F	0.28	0.14	0.18	1007
F	184	261	217	52	85	137	71	G	0.45	0.29	0.35	542
G	92	94	57	19	37	88	155	accuracy		0.69	176165	
								macro avg	0.46	0.34	0.37	176165
								weighted avg	0.67	0.69	0.67	176165

Avec 15 features  
Modèle : `RandomForestClassifier(random_state=42, class_weight = "balanced")`

Nous obtenons des scores légèrement meilleurs sur chaque classe et une augmentation de +0.02 points sur la métrique macro avg f1 score.

Ces résultats restent insatisfaisants.

- RandomForestClassifier avec 25 features

Matrice de confusion :								Classification Report :				
Classe prédictive	A	B	C	D	E	F	G		precision	recall	f1-score	support
Classe réelle												
<b>A</b>	59377	15237	361	18	14	17	20	A	0.75	0.79	0.77	75044
<b>B</b>	17721	56080	3087	45	24	27	24	B	0.67	0.73	0.70	77008
<b>C</b>	1462	10935	5461	170	75	42	23	C	0.51	0.30	0.38	18168
<b>D</b>	323	1045	1031	187	94	52	16	D	0.32	0.07	0.11	2748
<b>E</b>	232	521	514	98	156	88	39	E	0.32	0.09	0.15	1648
<b>F</b>	163	268	218	57	93	149	59	F	0.31	0.15	0.20	1007
<b>G</b>	91	95	58	17	39	102	140	G	0.44	0.26	0.32	542
								accuracy			0.69	176165
								macro avg	0.47	0.34	0.38	176165
								weighted avg	0.67	0.69	0.68	176165

Avec 25 features  
Modèle : RandomForestClassifier(random\_state=42, class\_weight = "balanced")

A nouveau, nous observons une très légère amélioration du f1 score global. Nous gagnons 0,01 points sur la macro avg f1 score. Remarquons par contre que la classe D reste la classe la moins bien reconnue par nos différents modèles.

Les résultats restent insatisfaisants.

- RandomForestClassifier avec 60 features :

Matrice de confusion :								Classification Report :				
Classe prédictive	A	B	C	D	E	F	G		precision	recall	f1-score	support
Classe réelle												
<b>A</b>	59411	15234	336	18	15	15	15	A	0.75	0.79	0.77	75044
<b>B</b>	17749	56172	2961	48	27	24	27	B	0.67	0.73	0.70	77008
<b>C</b>	1479	10924	5461	168	72	44	20	C	0.52	0.30	0.38	18168
<b>D</b>	316	1065	1014	189	80	69	15	D	0.31	0.07	0.11	2748
<b>E</b>	227	533	520	110	135	95	28	E	0.30	0.08	0.13	1648
<b>F</b>	166	293	218	62	92	117	59	F	0.26	0.12	0.16	1007
<b>G</b>	101	105	59	12	35	91	139	G	0.46	0.26	0.33	542
								accuracy			0.69	176165
								macro avg	0.46	0.34	0.37	176165
								weighted avg	0.67	0.69	0.68	176165

Avec 60 features  
Modèle : RandomForestClassifier(random\_state=42, class\_weight = "balanced")

Sans surprise, les résultats obtenus varient très légèrement à la baisse.

Pour conclure, nos modèles de Random Forest ne performent pas de manière satisfaisante malgré nos différentes tentatives d'optimisation "simples".

Se pose alors la question de l'application d'une Grid Search.

#### V.4.d. GridSearch CV

Pour optimiser les hyperparamètres de notre Random Forest, nous avons fait plusieurs Grid Search se concentrant sur la métrique “macro f1\_score”.

Une Grid Search simple nous a permis de déterminer quel était le meilleur critères d'évaluation de notre modèle :

```
Fitting 3 folds for each of 3 candidates, totalling 9 fits
[CV 1/3] END .....criterion=gini;, score=0.355 total time= 2.1min
[CV 2/3] END .....criterion=gini;, score=0.356 total time= 2.1min
[CV 3/3] END .....criterion=gini;, score=0.362 total time= 2.1min
[CV 1/3] END .....criterion=entropy;, score=0.364 total time= 2.6min
[CV 2/3] END .....criterion=entropy;, score=0.352 total time= 2.6min
[CV 3/3] END .....criterion=entropy;, score=0.363 total time= 2.5min
[CV 1/3] END .....criterion=log_loss;, score=0.364 total time= 2.5min
[CV 2/3] END .....criterion=log_loss;, score=0.352 total time= 2.5min
[CV 3/3] END .....criterion=log_loss;, score=0.363 total time= 2.5min
CPU times: user 24min 47s, sys: 16.7 s, total: 25min 4s
Wall time: 25min 10s
```

Les critères “entropy” et “log\_loss” arrivent aux exacts mêmes scores, comme pour l'arbre de décision, nous garderons le critère “entropy” pour nos modèles.

On applique la GridSearch CV suivante afin de trouver la meilleure combinaison d'hyperparamètres :

- 'Max\_depth' : [5,10],
- 'N\_estimators' : [100],
- 'Max\_features' : [5,7],
- 'Min\_samples\_leaf' : [3,5],
- 'Min\_samples\_split' : [1,2]

Le meilleur F1-score est de 0,325 et obtenu avec : {'max\_depth': 10, 'max\_features': 7, 'min\_samples\_leaf': 3, 'n\_estimators': 100}

C'est un score assez faible mais est à l'effigie de la complexité du dataset et l'entremêlement des valeurs obtenues par chaque classe pour chacune des caractéristiques.

Observons aussi que ce score est inférieur au score obtenu avec les paramètres par défaut (hors class\_weight), dont le meilleur F1 score est de 0,38.

Ci-dessous, comparons les résultats d'un modèle avec les hyperparamètres issus du Grid Search et notre modèle “par défaut” avec l'ajout du critère d'évaluation “entropy”.

Matrice de confusion :							Classification Report :					
Classe prédictive	A	B	C	D	E	F	G		precision	recall	f1-score	support
Classe réelle								A	0.70	0.74	0.72	75044
<b>A</b>	55758	12713	1462	1292	1739	1317	763	B	0.69	0.46	0.55	77008
<b>B</b>	21681	35337	12453	3694	2265	996	582	C	0.34	0.42	0.38	18168
<b>C</b>	1545	3023	7563	3337	1593	730	377	D	0.09	0.33	0.15	2748
<b>D</b>	189	75	409	905	607	385	178	E	0.07	0.29	0.11	1648
<b>E</b>	57	17	78	365	484	403	244	F	0.08	0.35	0.13	1007
<b>F</b>	17	2	7	87	206	349	339	G	0.13	0.69	0.22	542
<b>G</b>	10	0	1	7	32	120	372	accuracy			0.57	176165
								macro avg	0.30	0.47	0.32	176165
								weighted avg	0.64	0.57	0.59	176165

Modèle : RandomForestClassifier(random\_state=42, class\_weight='balanced', criterion = 'entropy', max\_depth=10, max\_features = 7, min\_samples\_leaf = 3, n\_estimators = 100)

Matrice de confusion :							Classification Report :					
Classe prédictive	A	B	C	D	E	F	G		precision	recall	f1-score	support
Classe réelle								A	0.75	0.79	0.77	75044
<b>A</b>	59433	15154	377	18	19	23	20	B	0.67	0.73	0.70	77008
<b>B</b>	17862	55890	3110	58	31	36	21	C	0.51	0.31	0.38	18168
<b>C</b>	1478	10750	5606	174	86	44	30	D	0.30	0.07	0.11	2748
<b>D</b>	314	993	1070	191	97	68	15	E	0.26	0.08	0.12	1648
<b>E</b>	240	488	530	117	135	105	33	F	0.27	0.13	0.18	1007
<b>F</b>	158	264	219	60	107	133	66	G	0.45	0.28	0.35	542
<b>G</b>	101	88	54	18	44	85	152	accuracy			0.69	176165
								macro avg	0.46	0.34	0.37	176165
								weighted avg	0.67	0.69	0.68	176165

Modèle : RandomForestClassifier(random\_state=42, class\_weight='balanced', criterion = 'entropy')

Notre modèle “par défaut” est donc plus performant que celui après application d’une Grid Search.

Surprenant, l’ajout du critère “entropy” a finalement fait perdre 0,01 sur la métrique f1 macro avg.

Nous n’avons exploré qu’une infime partie des possibilités d’hyperparamétrage de notre modèle de Random Forest avec la Grid Search, par manque de temps mais aussi par contrainte technique. En effet, avec une machine très performante, nous aurions pu saisir plus de critères et de paramètres dès le départ.

### En conclusion :

Le dataset étant trop déséquilibré sur la variable cible, cela provoque des difficultés générales sur les classes hors A et B qui sont majoritaires.

On peut aussi se poser la question de la qualité et la corrélation des features sur lesquelles notre modèle agit.

La forêt aléatoire reste toutefois le modèle le plus prometteur à ce stade.

## V.5. Boosting

Le Boosting est un ensemble de méthodes visant essentiellement à réduire le biais de modèles de Machine Learning simples et faibles et les convertir en un modèle stable et puissant.

Le principe général du boosting consiste à construire une famille d'estimateurs "faibles" construits de manière récursive, qui sont ensuite agrégés par un vote à la majorité dans le cas d'un problème de classification.

Chaque estimateur est une version améliorée du précédent, qui vise à donner plus de poids aux observations mal ajustées ou mal prédites lors de l'entraînement de la version suivante.

Enfin les classificateurs sont combinés et pondérés par des coefficients associés à leurs performances prédictives respectives.

### V.5.a. AdaBoost avec arbre de décision

L'AdaBoost, appliquée à un classifieur de type decision tree, améliore sensiblement l'accuracy globale ( $0,60 \rightarrow 0,68$ ) et très légèrement le f1-score ( $0,33 \rightarrow 0,34$ ).

On remarque toutefois que cette amélioration provient surtout d'un gain sur la précision, malheureusement au détriment du recall : l'algorithme ne détecte pas très bien les classes de grands feux mais lorsqu'il prédit ces classes, il est un peu moins emprunté que ne l'est un seul arbre de décision.

Matrice de confusion :							Classification Report :					
Classe prédictive	A	B	C	D	E	F	G		precision	recall	f1-score	support
Classe réelle												
A	58724	16023	258	10	7	10	12	A	0.74	0.78	0.76	75044
B	18286	55861	2791	30	15	13	12	B	0.65	0.73	0.69	77008
C	1638	11146	5199	97	46	26	16	C	0.51	0.29	0.37	18168
D	356	1143	1047	113	51	32	6	D	0.30	0.04	0.07	2748
E	256	632	522	72	76	76	14	E	0.26	0.05	0.08	1648
F	196	334	251	38	71	76	41	F	0.24	0.08	0.12	1007
G	110	145	65	18	25	79	100	G	0.50	0.18	0.27	542
								accuracy			0.68	176165
								macro avg			0.34	176165
								weighted avg			0.67	176165

Modèle :

- `DecisionTree (class_weight="balanced", criterion='entropy', max_depth=20, max_features=10, min_samples_leaf=1, min_samples_split=2)`
- `AdaBoostClassifier (n_estimators=100, random_state=42)`

## V.5.b. HistGradientBoostingClassifier

L'algorithme HistGradientBoostingClassifier est une autre technique de boosting.

Une grid search CV a été appliquée selon les hyperparamètres suivants :

- max\_depth : [15,20,25]
- max\_iter : [100,200]
- max\_bins : [150,255]

Le meilleur F1-score est de 0,342 et obtenu avec : {'max\_bins': 255, 'max\_depth': 20, 'max\_iter': 100}.

Les accuracy et F1-score reste globalement identiques par rapport au meilleur arbre de décision. Cependant, cet algorithme semble mettre l'accent sur de meilleurs recall, malheureusement au détriment de la précision..

Matrice de confusion :							Classification Report :						
Classe prédite	A	B	C	D	E	F	G		precision	recall	f1-score	support	
Classe réelle									A	0.73	0.73	0.73	75044
A	55149	13126	1760	1175	1544	1688	602		B	0.69	0.49	0.58	77008
B	18903	37935	12698	3245	2293	1434	500		C	0.34	0.43	0.38	18168
C	929	3517	7785	2915	1640	1040	342		D	0.10	0.30	0.15	2748
D	113	119	421	827	614	480	174		E	0.07	0.28	0.11	1648
E	30	33	79	326	454	474	252		F	0.07	0.40	0.12	1807
F	10	5	11	68	179	399	335	accuracy			0.58	176165	
G	8	0	1	4	30	114	385	macro avg	0.31	0.48	0.33	176165	
								weighted avg	0.65	0.58	0.61	176165	

Modèle : HistGradientBoostingClassifier (max\_bins: 255, max\_depth: 20, max\_iter: 100)

### En conclusion :

Les algorithmes de boosting performent légèrement mieux que l'arbre de décision seul, sans apporter malheureusement de significatives améliorations.

Ils restent moins performants que la forêt aléatoire.

## V.6. Under et oversampling

### V.6.a. SMOTEN seul

Afin de favoriser l'apprentissage sur les classes minoritaires, on utilise un algorithme d'oversampling (SMOTEN) qui crée des records synthétiques à partir des données du dataset.

Il en résulte l'augmentation du nombre de lignes de dataset d'entraînement :

- Avant : A : 300964, B : 306232, C : 73201, D : 11516, E : 6524, F : 4201, G : 2019
- Après : A : 300964, B : 306232, C : 73201, D : 35000, E : 20000, F : 16000, G : 8000

Le meilleur arbre de décision au sein de l'AdaBoostClassifier et le HistGradientBoostingClassifier sont utilisés.

Encore une fois, l'accuracy est la meilleure pour l'Ada Boost alors que l'HistGradientBoostingClassifier obtient le meilleur F1-score (augmentation de 10 %), même si celui-ci reste très moyen 0,37.

Matrice de confusion :							Classification Report :					
Classe prédite	A	B	C	D	E	F	G		precision	recall	f1-score	support
Classe réelle												
A	57086	12876	3580	170	414	579	339		A	0.75	0.76	0.76
B	17853	40217	17057	313	684	564	320		B	0.71	0.52	0.60
C	938	3515	11910	396	682	474	253		C	0.34	0.66	0.45
D	127	133	1434	233	411	285	125		D	0.18	0.08	0.12
E	42	40	521	129	383	322	211		E	0.14	0.23	0.17
F	18	6	148	41	190	316	288	accuracy			0.63	
G	10	1	13	10	37	115	356	macro avg	0.35	0.46	0.37	
								weighted avg	0.67	0.63	0.64	

Modèle : HistGradientBoostingClassifier (class\_weight='balanced', learning\_rate=0.1, max\_depth=20, max\_iter=200, max\_bins=100)

### V.6.b. SMOTEN + Random Under Sampling

On réitère l'opération précédente d'oversampling mais cette fois-ci, on la fait suivre d'un undersampling aléatoire, destiné à diminuer le nombre de records dans les classes majoritaires (A et B).

La répartition des classes dans le dataset d'entraînement est alors la suivante :

- Avant : A : 300964, B : 306232, C : 73201, D : 11516, E : 6524, F : 4201, G : 2019
- Après : A : 200000, B : 200000, C : 73201, D : 35000, E : 20000, F : 16000, G : 8000

Les résultats n'évoluent pas trop, si ce n'est un gain sur la classe D (apparemment la plus compliquée à prédire) au détriment d'une petite perte sur la classe G.

Matrice de confusion :							Classification Report :					
Classe prédictive	A	B	C	D	E	F	G		precision	recall	f1-score	support
Classe réelle												
A	57205	12739	3538	137	415	633	377	A	0.75	0.76	0.76	75044
B	17962	40184	16845	348	739	571	359	B	0.71	0.52	0.60	77008
C	918	3595	11796	388	703	524	244	C	0.34	0.65	0.45	18168
D	134	122	1425	260	398	271	138	D	0.20	0.09	0.13	2748
E	40	42	508	117	400	325	216	E	0.14	0.24	0.18	1648
F	16	5	132	40	189	326	299	F	0.12	0.32	0.17	1007
G	10	1	11	6	37	124	353	G	0.18	0.65	0.28	542
							accuracy			0.63	176165	
							macro avg	0.35	0.46	0.37	176165	
							weighted avg	0.67	0.63	0.64	176165	

Modèle : *HistGradientBoostingClassifier (class\_weight='balanced', learning\_rate=0.1, max\_depth=20, max\_iter=200, max\_bins=100)*

## V.6.c. SMOTEN + Edited Nearest Neighbours

L'undersampling est cette-fois-ci assurée par l'algorithme Edited Nearest Neighbours.

La répartition des classes dans le dataset d'entraînement est alors la suivante :

- Avant : A : 300964, B : 306232, C : 73201, D : 11516, E : 6524, F : 4201, G : 2019
- Après : A : 127109, B : 87563, C : 73201, D : 35000, E : 20000, F : 16000, G : 8000

Les résultats sont un peu moins bons qu'avec l'undersampling aléatoire : un F1-score de 0,36 contre 0,37 précédemment.

Matrice de confusion :							Classification Report :					
Classe prédictive	A	B	C	D	E	F	G		precision	recall	f1-score	support
Classe réelle												
A	54289	12985	5968	182	507	707	406	A	0.76	0.72	0.74	75044
B	16053	40454	18423	388	695	615	380	B	0.71	0.53	0.60	77008
C	727	3562	12024	414	668	520	253	C	0.31	0.66	0.42	18168
D	99	80	1515	235	407	279	133	D	0.17	0.09	0.11	2748
E	28	26	546	134	375	334	205	E	0.13	0.23	0.17	1648
F	8	6	157	43	184	317	292	F	0.11	0.31	0.16	1007
G	9	1	12	8	37	117	358	G	0.18	0.66	0.28	542
							accuracy			0.61	176165	
							macro avg	0.34	0.46	0.36	176165	
							weighted avg	0.67	0.61	0.63	176165	

Modèle : *HistGradientBoostingClassifier (class\_weight='balanced', learning\_rate=0.1, max\_depth=20, max\_iter=200, max\_bins=100)*

## En conclusion :

Un algorithme de boosting, combiné à un resampling du dataset initial, amène un gain de 10 % sur l'accuracy et le F1-score, par rapport à l'arbre de décision seul. Nous sommes sur la bonne voie mais il reste encore des progrès à réaliser.

# VI. Interprétation des résultats

## VI.1. Features principales

### VI.1.a. Durée

Le calcul de la durée du feu, grâce aux datetimes du début et de fin de feu, semble déterminant au vu des résultats : cette variable a de loin le poids le plus important de toutes les variables.

La prépondérance de la durée dans la distinction des classes s'explique par la tendance à laquelle on pouvait s'attendre : plus le feu grossit, plus il a de chances de durer longtemps.

Cela se confirme avec l'évolution de la médiane ci-contre.

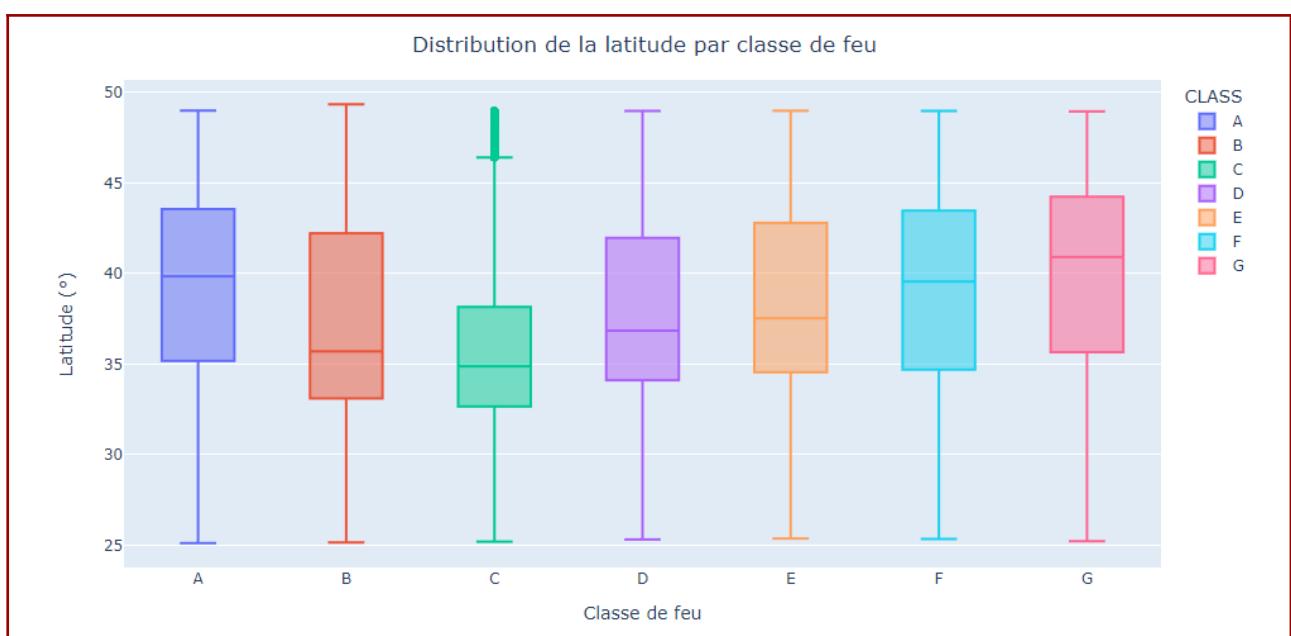
CLASS	median	mean
A	45	92
B	60	79
C	135	179
D	460	1206
E	1440	2551
F	3240	5885
G	10564	23342

### VI.1.b. Coordonnées géographiques

Avant d'aborder le prochain point, rappelons que le gros du territoire américain est compris entre entre -35° et -45° pour la latitude et entre -75° et -120° pour la longitude.

Bien que cela n'ait pas de réalité physique, le barycentre des coordonnées géographiques des feux par classe montre que les feux de classe G et les feux de classe A sont plutôt à l'ouest des USA.

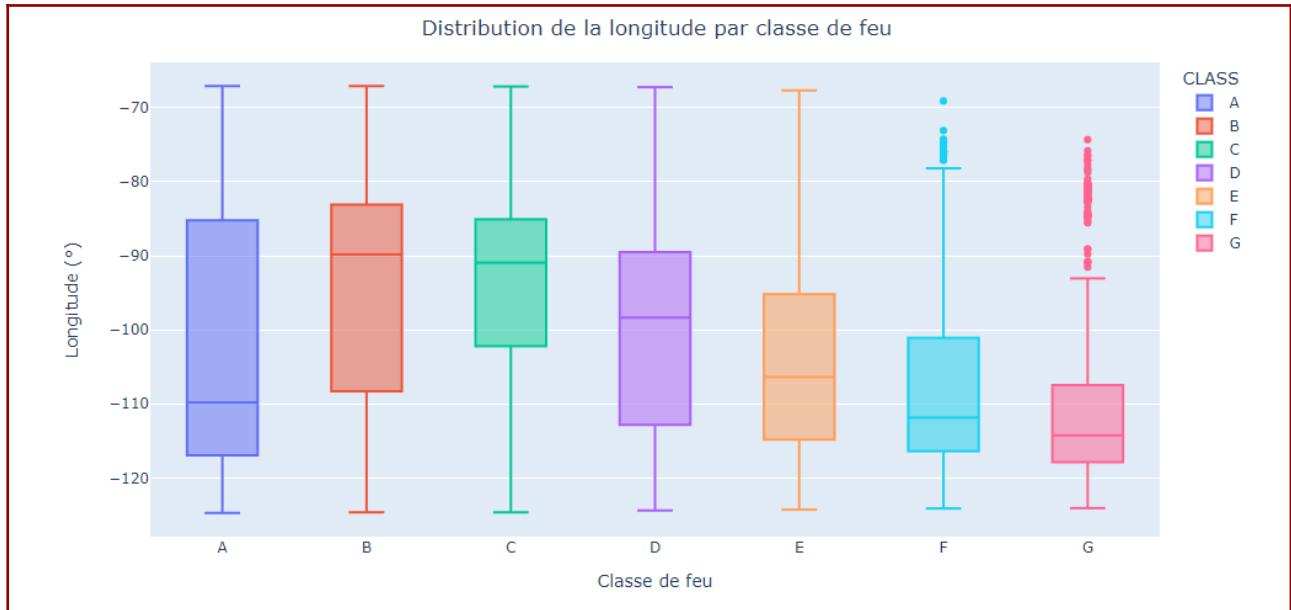
Là encore, la variation de la latitude, d'un ordre de grandeur d'une dizaine de pourcents, semble suffisante pour aider à distinguer les classes de feux.



Le résultat est encore plus criant en ce qui concerne la longitude. En plus d'être caractérisée par la médiane la plus faible, la classe G a aussi la dispersion la plus faible, ce

qui met en évidence la concentration de ces grands feux sur la côte ouest et l'Alaska.

La classe A partage un point commun : une médiane relativement faible, ce qui dénote un surplus de feux à l'ouest des USA. Toutefois, une différence : la classe A possède la dispersion la plus grande avec un écart interquartile qui couvre une grosse partie des USA en termes de longitude. Preuve que ces petits feux touchent l'intégralité du territoire américain.



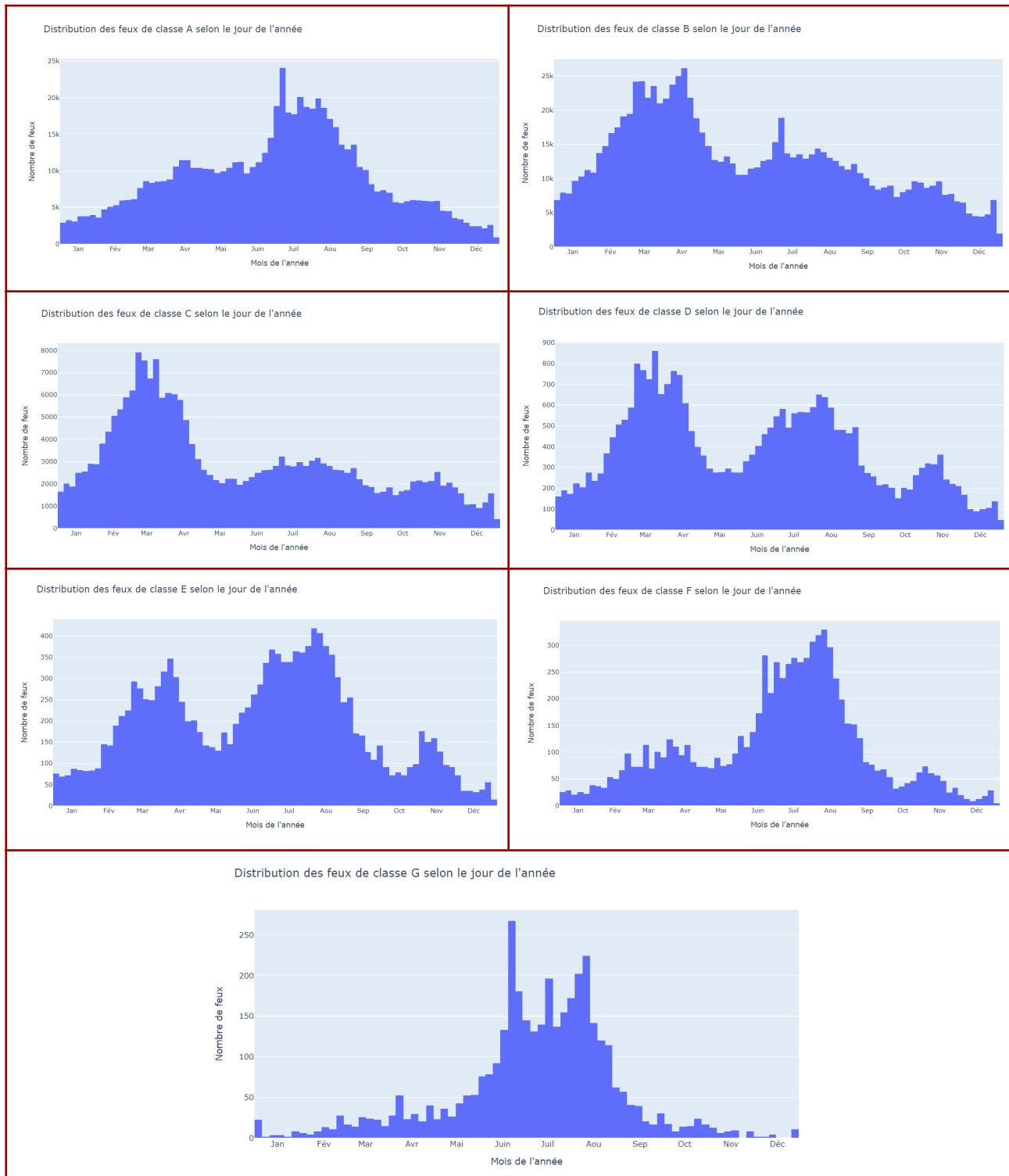
Ce résultat est confirmé par le poids des Etats dans la répartition des feux par classe. On distingue bien à droite la présence importante de l'Alaska (G : 17,23 %) ainsi que de la Californie (G : 10,44 %) dans les feux de grande taille.

Note : l'histogramme normalisé ci-dessous est plus lisible dans le notebook grâce à Plotly grâce à la fonction hover qui permet de s'émanciper de la légende.



## VI.1.c. Saisonnalité

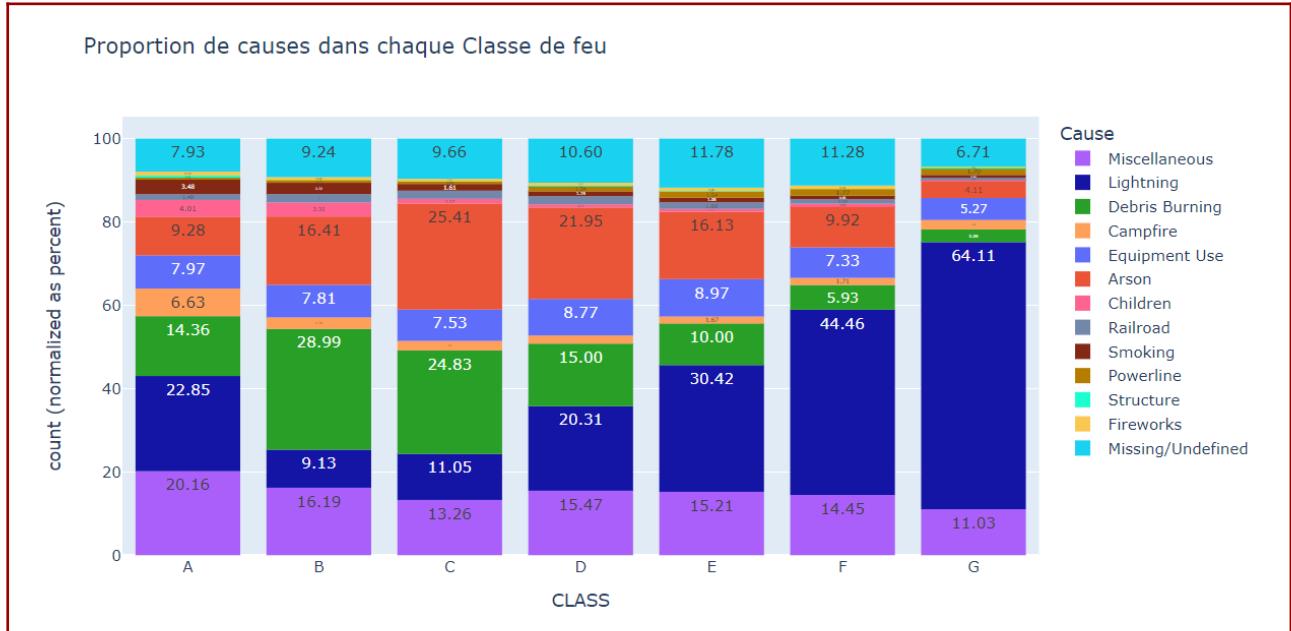
La saisonnalité des feux, représentée par le jour de l'année, contribue aussi à la distinction des classes. Là où les feux de petite taille s'étalent sur toute l'année avec une concentration au début du printemps (excepté la classe A, où la part des feux de foudre en été est plus élevée que pour B et C), les feux de grande taille interviennent plutôt en saison chaude.



On se remémore alors la saisonnalité des deux grandes catégories de feux : cause humaine en début de printemps, foudre plutôt l'été.

On retrouve cette tendance dans l'histogramme normalisé ci-dessous. Le poids des incendies dûs à la foudre (en saison chaude) est plus important pour les classes A, D, E, F et G, comparé aux classes B et C (qui présentent un maximum de feux au début du printemps, notamment lié au brûlage des débris végétaux).

On constate une augmentation de la proportion des feux de foudre à partir de la classe C. Cela se perçoit dans l'enchaînement des histogrammes ci-dessus : au fur et à mesure que l'on avance dans les classes de feu, le nombre de feux au printemps diminue au profit du nombre de feux en été.



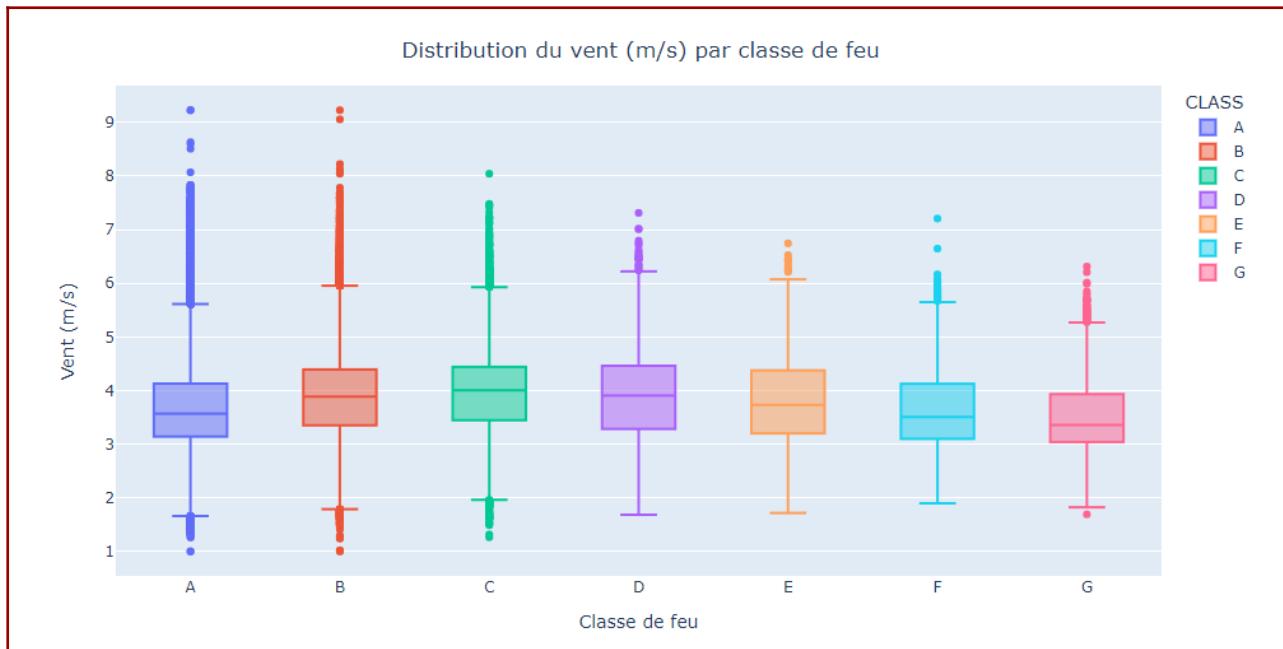
### VI.1.d. Vent

L'intuition est de penser que plus le vent est fort, plus le feu est attisé, et plus il a de chances de s'étendre.

Ce n'est pas ce que montre le graphique suivant : en comparant les deux extrêmes (A et G), on constate que la médiane du vent moyen mensuel pendant les feux de classe G n'est pas plus élevée que celle des feux de classe A, voire même moindre.

On peut relativiser cette observation avec le choix de la variable en elle-même :

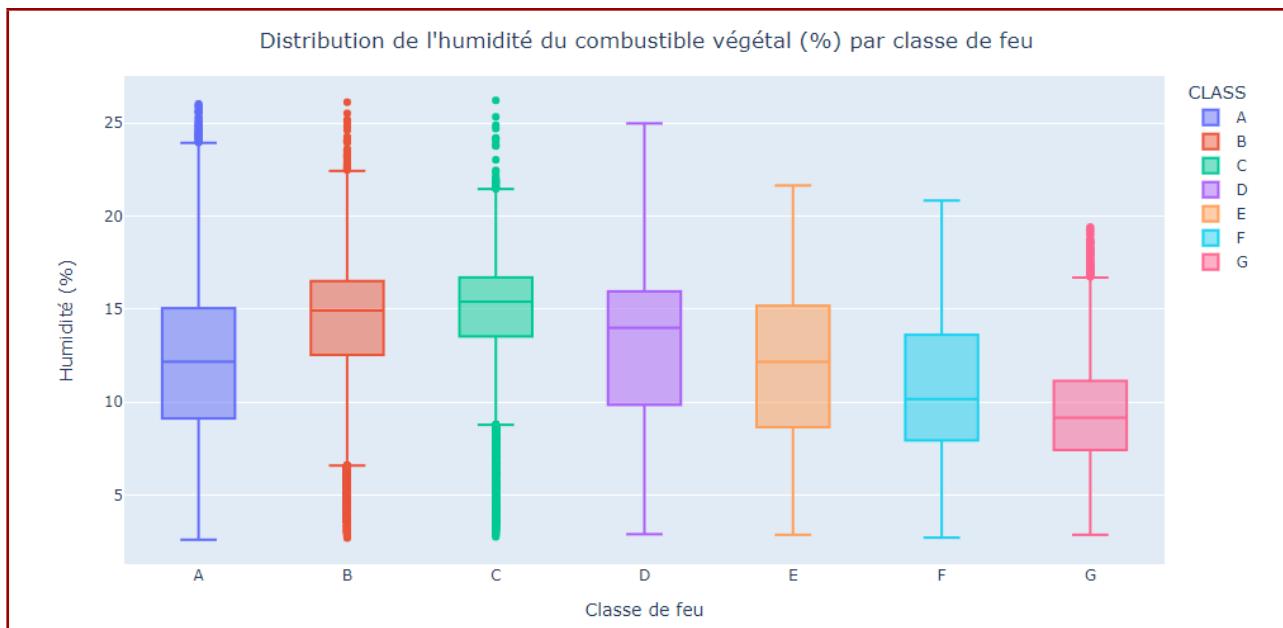
- Pour décrire un feu, cela a-t-il un sens d'utiliser un vent moyen sur un mois ? Ne serait-il pas plus pertinent d'utiliser des mesures bien plus rapprochées sur l'événement en lui-même ?
- Le grand déséquilibre entre les classes de feu s'observe par la présence de nombreux outliers sur les petites classes : la dispersion du vent est bien plus importante. Comme le nombre de feux de petite classe est bien plus grand (300 000 contre quelques milliers, il y a statistiquement bien plus de chances de voir apparaître des phénomènes "extrêmes" de valeur élevée, ce qui a tendance à augmenter la médiane.



### VI.1.e. Humidité de la végétation

Autre hypothèse intuitive : plus l'humidité du combustible végétal est basse, plus le feu a des chances de se propager.

Cet a priori semblerait plutôt confirmé par le boxplot suivant : la tendance est à la baisse entre les feux de classe C et G de près de 40 %, en ce qui concerne la médiane.



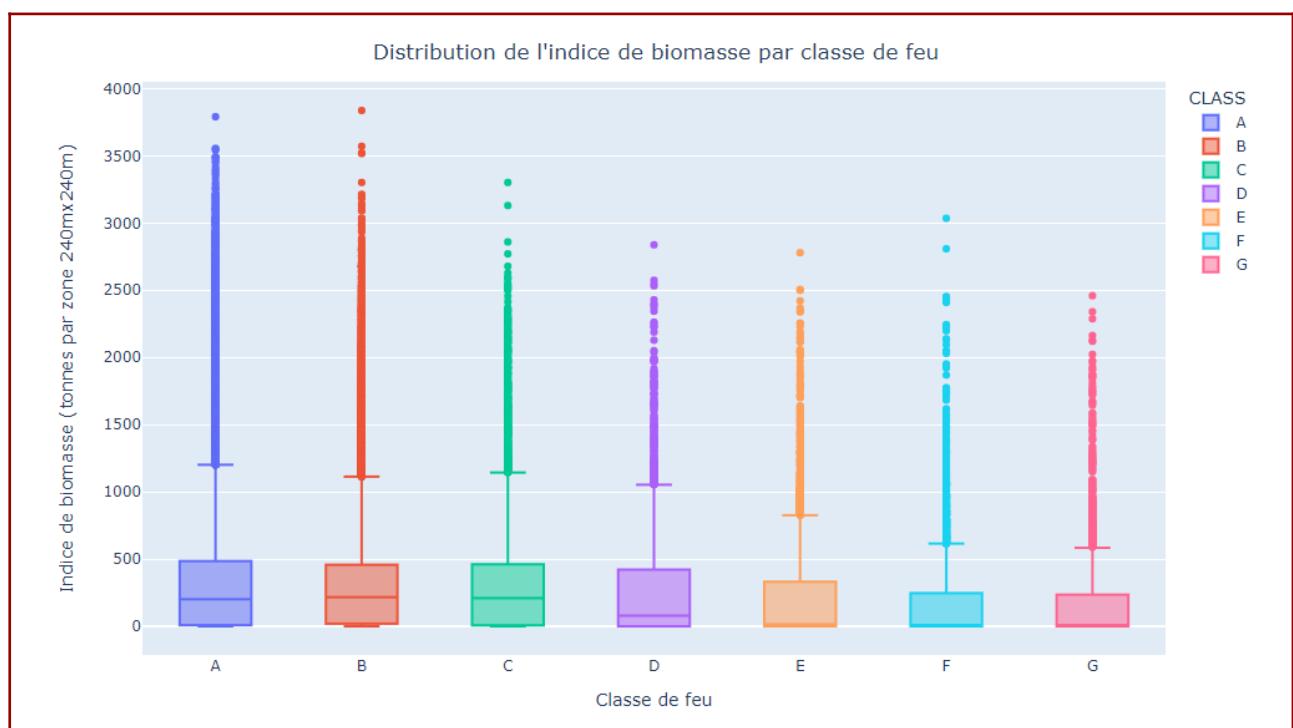
## VI.1.f. Indice de biomasse

Autre a priori : plus l'indice de biomasse d'une zone est élevé, plus il y a de la matière végétale à brûler, plus le feu a de ressources pour se perpétuer et donc là aussi passer à une classe de feu supérieure. C'est le point noir de la fusion des deux datasets. Thibault a bien analysé la présence de "NaN" mais par souci de temps, n'a pas plus approfondi l'étude de la colonne. Malheureusement, après tracé du boxplot ci-dessous et calcul du tableau ci-contre, on constate qu'il y a de nombreuses valeurs égales à 0 pour les classes supérieures (près de 50 %) : sont-ce les vraies valeurs ? Ou bien est-ce 0 car il n'y avait pas l'information ?

On peut aussi légitimement s'interroger sur tous ces outliers.

Toujours est-il que les modèles semblent en avoir tiré un petit indice pour classer les feux... Une tentative de modélisation sans cette colonne montre que les résultats n'évoluent quasiment pas, plutôt logique vu le poids relatif de cette variable.

	Total	Valeurs 0
CLASS		
A	376008	86062
B	383240	78063
C	91369	21495
D	14264	5541
E	8172	3831
F	5208	2770
G	2561	1400



## VI.2. Pistes d'amélioration

La piste la plus intéressante pour améliorer nos résultats de prédiction serait de regrouper les classes de feux en trois catégories, celles-ci restant à définir. En effet, ne vaut-il mieux pas détecter de manière précise un groupe plus large de types de feux plutôt que d'avoir une précision plus faible sur une classe plus détaillée ?

Une autre piste d'amélioration serait de regrouper en une seule information latitude et longitude, à l'aide d'un KMeans par exemple, afin de traiter la localisation d'un seul tenant et non selon deux axes indépendants.

Cette piste n'a pas été explorée, faute de ressources machine et aussi de temps.

Une piste d'amélioration, peu prometteuse vu le poids de la feature, serait de procéder au regroupement des propriétaires, tel qu'on l'a fait pour les causes.

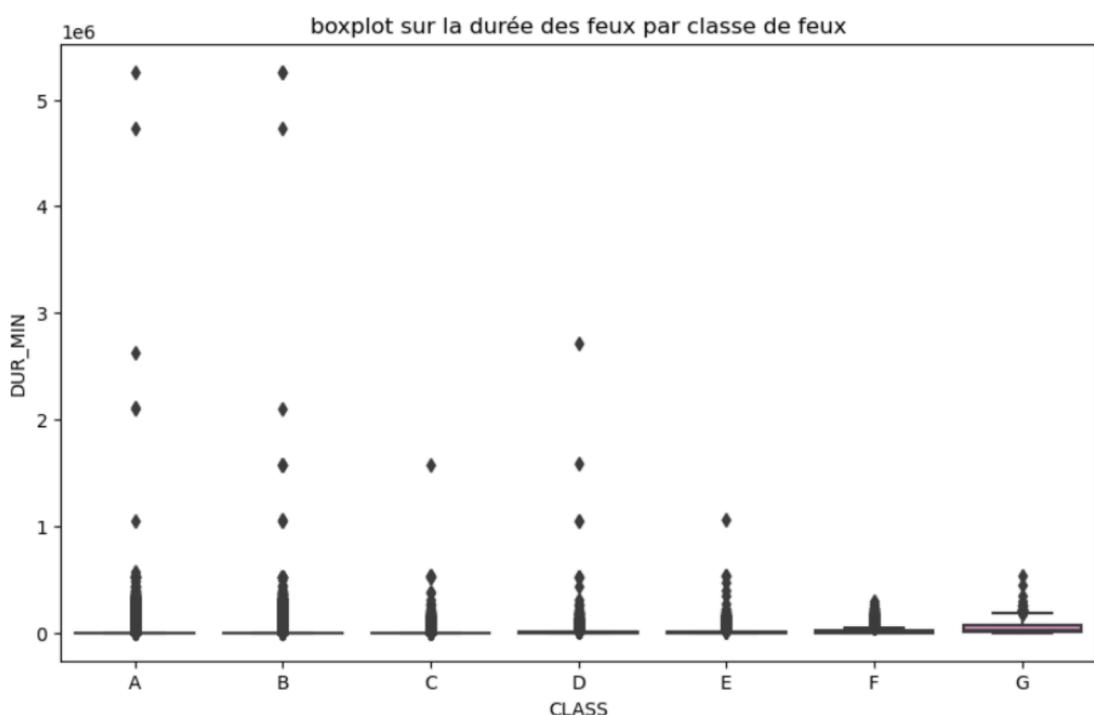
## VII. Dataset corrigé et samplé

### VII.1. Problèmes identifiés

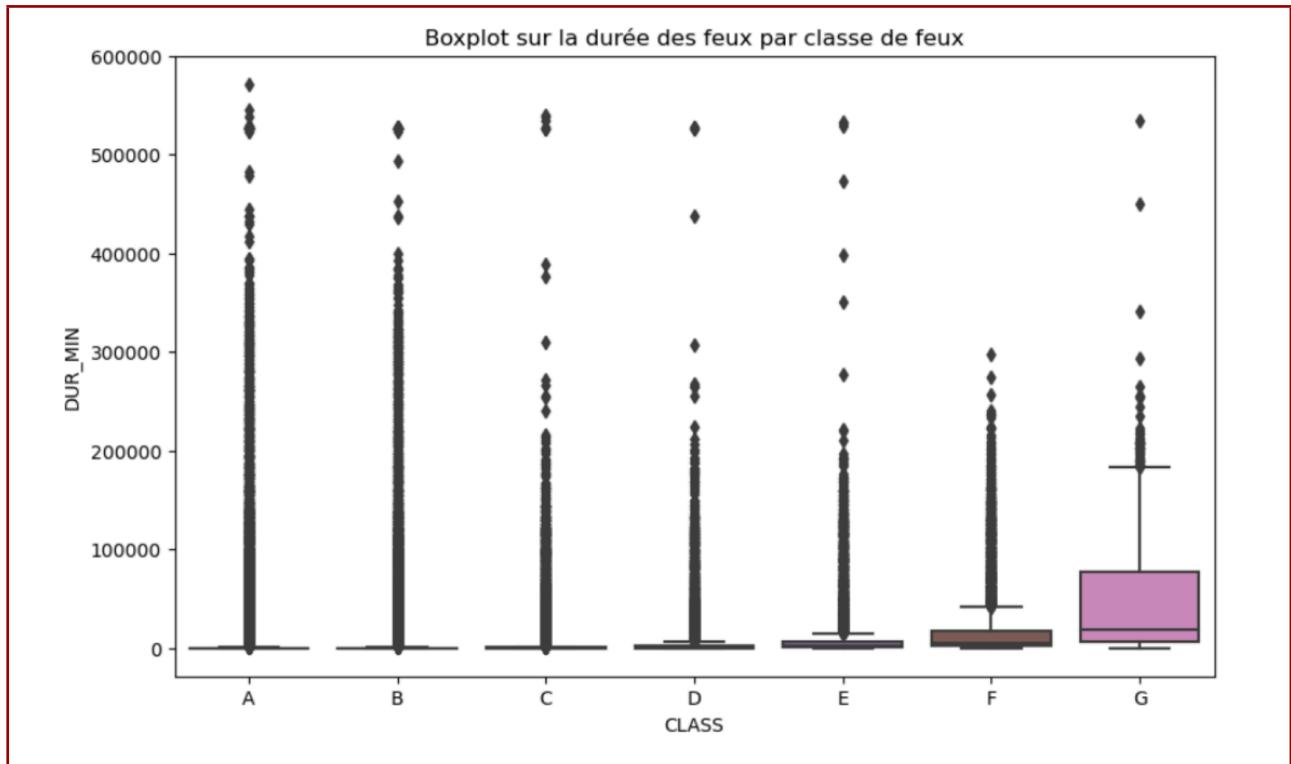
#### VII.1.a. Outliers : durée de feu

Un boxplot de la variable “durée” en fonction de la classe de feu a permis de mettre en évidence un problème d’outliers, qui en fait découle d’un problème de saisie et/ou de conversion de donnée sur la date de fin du feu.

Ainsi, on constate ci-dessous qu’il y a dans les classes autres que F et G des outliers correspondant à des durées de plusieurs années, ce qui est aberrant au regard de la classe de feu. En effet, il n’est pas normal de voir des “petits” feux durer plus que quelques heures.



En supprimant les outliers supérieurs à une durée de 6e5 min, on commence à mieux distinguer les boxplots des classes F et G.



Pour le notebook complémentaire de modélisation, on a donc décidé de retirer du dataset tous les outliers, c'est-à-dire les points en dehors des moustaches pour chacune des classes (au-delà de 1,5 l'écart interquartile).

### VII.1.b. Erreurs sur les dates

Le dataframe ci-dessous, issu du dataset initial, met en évidence un écart de plusieurs années au jour près. Les heures (“discovery datetime”, “contained datetime”) quant à elles semblent cohérentes en regard de la classe de feu.

Nous aurions donc des feux qui ont duré une dizaine d'années, ce qui est fondamentalement impossible.

La date de fin est calculée à partir d'une colonne DISC\_DATE du dataset initial. Cette colonne a un format particulier avec des nombres très grands et semble se comporter comme un compteur de jours. Il semblerait donc qu'il y ait eu une erreur de saisie ou de conversion, ce qui entraîne ce décalage de plusieurs années.

	FPA_ID	DUR_MIN	SIZE	CLASS	DISC_YEAR	DISC_DATETIME	CONT_YEAR	CONT_DATETIME	CAUSE_DESCR	STATE
362576	FWS-1999CAGRRY269	5260410.0	0.5	B	1999	1999-06-16 13:00:00	2009.0	2009-06-16 14:30:00	Debris Burning	CA
362492	FWS-1999CAGRRX345	5260380.0	0.5	B	1999	1999-07-19 11:00:00	2009.0	2009-07-19 12:00:00	Debris Burning	CA
362655	FWS-1999CAPLRY974	5260335.0	0.5	B	1999	1999-07-11 18:15:00	2009.0	2009-07-11 18:30:00	Miscellaneous	CA
362642	FWS-1999CAPLRW160	5260335.0	0.1	A	1999	1999-08-11 14:00:00	2009.0	2009-08-11 14:15:00	Miscellaneous	CA
1317621	SFO-WV-2001-20554	4733280.0	4.0	B	2001	2001-05-14 22:00:00	2010.0	2010-05-14 22:00:00	Equipment Use	WV
1351259	SFO-NY-NY0822-2000-030003	4733280.0	0.1	A	2000	2000-03-21 11:22:00	2009.0	2009-03-21 11:22:00	Miscellaneous	NY
356156	W-513441	2708760.0	120.0	D	2000	2000-08-07 16:00:00	2005.0	2005-10-01 18:00:00	Lightning	CA
365708	FWS-2002CAGRRY287	2629470.0	0.1	A	2002	2002-09-12 06:45:00	2007.0	2007-09-12 07:15:00	Smoking	CA
325491	W-507314	2108054.0	0.1	A	2005	2005-08-25 20:45:00	2009.0	2009-08-28 18:59:00	Lightning	CO
368029	FWS-2004CATNREPTB	2103885.0	0.4	B	2004	2004-12-04 14:30:00	2008.0	2008-12-04 15:15:00	Debris Burning	CA
186216	W-384883	2103872.0	0.1	A	1998	1998-07-27 12:05:00	2002.0	2002-07-27 12:37:00	Arson	WI
305237	W-121410	1585269.0	100.0	D	2001	2001-06-11 20:10:00	2004.0	2004-06-16 17:19:00	Lightning	AK

## VII.1.c. Perte de données

La jointure entre notre dataset initial et le dataset contenant des données “Météo & Végétation”, a été réalisée avec le paramètre : “inner join”. Malheureusement, l’absence de données sur les États de l’Alaska, Porto Rico et Hawaï dans ce second dataset a causé la suppression des lignes correspondantes dans notre dataset final.

Se pose donc la question : vaut-il mieux éviter cette jointure et perdre les informations météorologiques et de végétation ? Ou bien perdre les données relatives à l’Alaska qui est pourtant un gros contributeur de grands feux et sur la cause naturelle “Lightning”.

## VII.2. Entraînement

### VII.2.a. Random Forest et jeu de données corrigé

On réutilise une Random Forest performante sur un dataset dont on a purgé les outliers de durée par classe.

Matrice de confusion :							Classification Report :					
Classe prédictive	A	B	C	D	E	F	G		precision	recall	f1-score	support
Classe réelle												
<b>A</b>	40022	10249	568	16	1	1	0	A	0.78	0.79	0.79	50857
<b>B</b>	10547	37374	5550	36	9	1	0	B	0.69	0.77	0.73	53517
<b>C</b>	704	4619	6713	164	36	2	0	C	0.57	0.35	0.43	12238
<b>D</b>	112	295	813	636	238	115	14	D	0.47	0.30	0.36	2223
<b>E</b>	40	100	252	358	282	199	29	E	0.36	0.22	0.27	1260
<b>F</b>	17	18	69	166	170	272	87	F	0.40	0.33	0.36	799
<b>G</b>	4	0	9	29	45	138	214	G	0.65	0.49	0.56	439
							accuracy			0.72	121333	
							macro avg		0.56	0.46	0.50	121333
							weighted avg		0.71	0.72	0.71	121333

Modèle : `RandomForestClassifier(class_weight='balanced', max_depth=20, n_estimators=200, n_jobs=-1, random_state=42)`

Nous observons une amélioration significative des résultats. Un gain de 0,03 point sur l’accuracy et un gain de 0,12 point sur le F1 score macro.

Si on supprime la colonne d'indice de biomasse (qui comportait un nombre non négligeable de valeurs 0), les performances sont très légèrement dégradées. Malgré les valeurs 0, la Random Forest parvient donc apparemment à capturer une partie d'information dans cette colonne.

Matrice de confusion :							Classification Report :					
Classe prédictive	A	B	C	D	E	F	G		precision	recall	f1-score	support
Classe réelle												
<b>A</b>	40378	10206	267	5	1	0	0	A	0.78	0.79	0.79	50857
<b>B</b>	10550	41030	1917	13	6	1	0	B	0.70	0.77	0.73	53517
<b>C</b>	729	7103	4232	137	34	2	1	C	0.58	0.35	0.43	12238
<b>D</b>	120	478	652	642	235	85	11	D	0.46	0.29	0.36	2223
<b>E</b>	56	149	207	369	278	175	26	E	0.36	0.22	0.27	1260
<b>F</b>	18	28	67	182	166	262	76	F	0.40	0.33	0.36	799
<b>G</b>	4	1	10	37	57	135	195	G	0.63	0.44	0.52	439
							accuracy				0.72	121333
							macro avg		0.56	0.46	0.49	121333
							weighted avg		0.71	0.72	0.71	121333

Modèle : `RandomForestClassifier(class_weight='balanced', n_jobs=-1, random_state=42)`

## VII.2.b. Random Forest et jeu de données corrigé et samplié

Le dataset précédent peut être rééquilibré grâce à des techniques d'over/under sampling. On présente ici le meilleur résultat sur un dataset préparé avec un SMOTE (over) et un random under sampler.

Matrice de confusion :							Classification Report :					
Classe prédictive	A	B	C	D	E	F	G		precision	recall	f1-score	support
Classe réelle												
<b>A</b>	39668	10368	818	2	1	0	0	A	0.75	0.78	0.77	50857
<b>B</b>	11684	34391	7431	5	3	3	0	B	0.71	0.64	0.67	53517
<b>C</b>	985	3441	7626	110	66	10	0	C	0.44	0.62	0.52	12238
<b>D</b>	173	186	906	487	297	152	22	D	0.47	0.22	0.30	2223
<b>E</b>	69	59	287	277	311	202	55	E	0.34	0.25	0.29	1260
<b>F</b>	22	11	86	126	180	260	114	F	0.35	0.33	0.34	799
<b>G</b>	5	0	5	32	49	114	234	G	0.55	0.53	0.54	439
							accuracy				0.68	121333
							macro avg		0.52	0.48	0.49	121333
							weighted avg		0.69	0.68	0.68	121333

Modèle :

- `SMOTEN(sampling_strategy={3.0: 35000, 4.0: 20000, 5.0: 16000, 6.0: 8000}, random_state=42)`
- `RandomUnderSampler(sampling_strategy={0.0: 200000, 1.0: 200000, 2.0: proportions[2], 3.0: 35000, 4.0: 20000, 5.0: 16000, 6.0: 8000}, random_state=42)`
- `RandomForestClassifier(class_weight='balanced', max_depth=20, n_estimators=200, n_jobs=-1, random_state=42)`

On constate que le sampling a très légèrement dégradé les résultats. Toutefois, il est probablement possible de “tuner” les proportions d’over et d’under sampling afin d’obtenir un petit gain par rapport au dataset non samplé.

### VII.2.c. Random Forest et jeu de données initial avec outliers

On souhaite avoir une idée de l’impact de la suppression de la jointure et donc des données de végétation et de météo et donc de la conservation des données liées à l’Alaska.

Matrice de confusion :							Classification Report :					
Classe prédictive	A	B	C	D	E	F	G		precision	recall	f1-score	support
Classe réelle												
<b>A</b>	60327	15140	388	23	13	20	19	A	0.75	0.79	0.77	75930
<b>B</b>	17491	56562	3397	75	51	31	18	B	0.67	0.73	0.70	77625
<b>C</b>	1625	10792	5657	226	109	60	30	C	0.50	0.31	0.38	18499
<b>D</b>	315	1048	1089	213	89	76	17	D	0.28	0.07	0.12	2847
<b>E</b>	222	525	519	142	172	98	41	E	0.29	0.10	0.15	1719
<b>F</b>	181	269	243	61	109	161	89	accuracy			0.69	178401
<b>G</b>	98	103	74	32	44	110	207	macro avg	0.47	0.35	0.38	178401
								weighted avg	0.67	0.69	0.68	178401

Modèle : `RandomForestClassifier(class_weight='balanced', n_jobs=-1, random_state=42)`

La perte des données du dataset complémentaire semble compensée par la présence des données de l’Alaska.

Il pourrait donc être intéressant de conserver les données du deuxième dataset et de procéder à une imputation des données de végétation/météo relatives à l’Alaska.

# Annexes

## Annexe 1 : Traitement du jeu de données principal

Suppression de colonnes		
Nan trop nombreux	Entre 43 et 99 % de valeurs manquantes, non remplaçables car valeurs uniques de rapports officiels	'ICS_209 INCIDENT_NUMBER', 'ICS_209_NAME', 'MTBS_ID', 'MTBS_FIRE_NAME', 'COMPLEX_NAME', 'LOCAL_FIRE_REPORT_ID', 'LOCAL INCIDENT_ID', 'FIRE_CODE', 'FIRE_NAME'
Valeurs uniques non exploitables	Valeurs type “Identifiant d'unité NWCG actif”, “Code de l'unité de l'agence”, “Nom de l'unité de l'agence d'évaluation” > non pertinent pour notre étude  Colonne très “sale” en termes de données : mélange de strings, codes alphanumériques, nombres, avec des espaces	'OBJECTID', 'SOURCE_SYSTEM_TYPE', 'SOURCE_SYSTEM', 'NWCG_REPORTING_AGENCY', 'NWCG_REPORTING_UNIT_ID', 'NWCG_REPORTING_UNIT_NAME', 'SOURCE_REPORTING_UNIT', 'SOURCE_REPORTING_UNIT_NAME', 'COUNTY', 'FIPS_CODE', 'FIPS_NAME'
Modification et enrichissement du dataset		
Gain de mémoire	Changement du type de certaines colonnes de “object” à “category” afin de gagner de l'espace mémoire. On passe en effet de 540 MB au départ à 188 MB avec la suppression des colonnes et le changement de type.	'STAT_CAUSE_DESCR', 'FIRE_SIZE_CLASS', 'OWNER_DESCR', 'STATE', 'COUNTY', 'STAT_CAUSE_CODE', 'OWNER_CODE', 'FIRE_YEAR', 'DISCOVERY_DOY'
Renommer les colonnes	Pour faciliter les différentes manipulation du dataset	'FIRE_YEAR':'DISC_YEAR', 'DISCOVERY_DATE':'DISC_DATE', 'DISCOVERY_DOY':'DISC_DOY', 'DISCOVERY_TIME':'DISC_TIME',

		'STAT_CAUSE_CODE':'CAUSE_CODE, 'STAT_CAUSE_DESCR':'CAUSE_DESCR' 'FIRE_SIZE':'SIZE', 'FIRE_SIZE_CLASS':'CLASS', 'LATITUDE':'LAT', 'LONGITUDE':'LON'
Recaler et renommer des colonnes "XX_DATE"	Les deux colonnes DISCOVERY_DATE et CONT_DATE sont en fait des sortes de compteurs de jour, dont la plage correspond à la période temporelle étudiée en jours.	'DISC_DATE':'DISC_DAYS' (integer), 'CONT_DATE':'CONT_DAYS' (integer)
Nouvelles colonnes	Création d'une colonne de date de début du feu et une de date de fin de feu	'DISC_DATE' , 'CONT_DATE' (format date : yyyy-mm-dd, type datetime)
	Par souci d'homogénéité, création d'une colonne "CONT_YEAR" afin d'avoir l'année de fin du feu, pour les lignes disposant de l'information de la date de feu (sinon NaN)	'CONT_YEAR' (float64)
	Création de 2 colonnes pour l'heure et les minutes des horaires de départ et de fin de feu pour une utilisation potentielle plus tard dans l'imputing ou les analyses. Les informations manquantes sont transformées en NaN. On supprime les deux colonnes de départ.	'CONT_HOUR', 'CONT_MIN', (float64) drop des colonnes : 'DISC_TIME','CONT_TIME'
	Création de deux colonnes datetime pour les dates et horaires de départ de feu et de fin de feu. Cela permettra d'affiner, pour les lignes complètes, la durée du feu.	'DISC_DATETIME' 'CONT_DATETIME' (format datetime : yyyy-mm-dd hh:mm:ss, type datetime)
	Création d'une colonne de durée de feu en minutes, ce qui enrichit le dataset d'une nouvelle variable. Cette variable servira aussi pour l'imputation sur les durées manquantes.	'DUR_MIN' (float64)

## Annexe 2 : Description des Causes des Incendies

- **Lightning : foudre/Causes non humaines**

La plupart des incendies qui ne sont pas d'origine humaine sont provoqués par la foudre. D'autres causes naturelles d'incendie comprennent les incendies de lave et de charbon. Contrairement aux incendies d'origine humaine qui se propagent raisonnablement également de juin à septembre, 78 % des incendies provoqués par la foudre se produisent historiquement en été (juin-août).

- **Powerline : production/transmission/distribution d'électricité**

Les incendies de forêt causés par les lignes électriques sont souvent dus à des vents violents, au contact avec la végétation, à une panne d'équipement ou au contact humain ou animal avec une ligne électrique (fil conducteur). Parfois, plusieurs de ces facteurs peuvent contribuer à provoquer un incendie, comme le vent poussant la végétation au contact de l'équipement électrique.

- **Equipment use : utilisation d'équipement domestiques et industriels**

Les équipements domestiques et industriels créent des étincelles en raison d'une mauvaise utilisation, ce qui peut provoquer un incendie. Les tracteurs, tondeuses à gazon, scies à chaîne, soudeurs, désherbeurs et autres équipements courants constituent tous des risques d'incendie possibles. Un incendie se produit souvent lorsque les lames entrent en contact avec des roches cachées dans la végétation. Les voitures peuvent également être responsables d'un incendie si un pot d'échappement ou un silencieux chaud entre en contact avec des broussailles sèches sur le bord de la route.

- **Structure : incendies de structure de maison**

Incendie qui a pris naissance dans une structure et s'est propagé vers la nature en raison de pannes ou d'activités associées. Il peut y avoir de la fumée ou des flammes provenant de la cheminée, une panne de courant dans la structure ou dans la zone voisine, ou une activité humaine dans la zone.

- **Campfire : feux de camp**

Cette catégorie comprend les feux de camp mal construits, laissés sans surveillance, mal éteints ou abandonnés, des barbecues.

- **Miscellaneous :**

- **Clôtures électriques :** les systèmes de clôture électrique de type coupe-herbe sont les plus susceptibles de déclencher des incendies de forêt. La végétation en croissance ou la végétation qui vient de se dessécher peut entrer en contact avec le fil de clôture et être chauffée jusqu'à sa température d'inflammation, provoquant un incendie
- **Réfraction :** les incendies provoqués par la réfraction du verre sont très rares

- **Chauffage spontané** : cela inclut la combustion spontanée, c'est-à-dire l'auto-échauffement et l'inflammation de chiffons huileux, de meules de foin et/ou de tas de compost. Certains carburants s'auto-échauffent et s'enflamment spontanément lorsque les conditions favorisent les processus biologiques et/ou chimiques. Cette action est plus susceptible de se produire après des périodes de journées chaudes et humides dans des tas de matières organiques en décomposition telles que du foin, des céréales, des aliments pour animaux, du fumier, de la sciure de bois, des tas de copeaux de bois et de la mousse de tourbe empilée.
  - **Utilisation d'armes à feu et d'explosifs** : tout projectile d'arme à feu ainsi que les cibles explosives doivent être considérés comme une source potentielle d'inflammation. La poudre noire et les projectiles tels que le noyau d'acier, la gaine d'acier, les composants en acier, le cuivre, le plomb, la gaine de cuivre à noyau de plomb, les perforants (AP), les incendiaires et les traceurs, font partie des types de munitions qui peuvent enflammer la végétation sauvage à cause de la chaleur.
- **Arson : incendie criminel**
- L'incendie criminel est l'acte criminel consistant à incendier délibérément ou malicieusement une propriété, y compris des terrains publics, dans l'intention de l'endommager. Les appareils et les « ensembles chauds » sont couramment utilisés pour allumer des incendies.
- Les incendies criminels peuvent représenter plus de 20 % de tous les incendies de forêt d'origine humaine, et jusqu'à 70 % ou plus des incendies dans certaines juridictions.
- **Firework : feux d'artifice**
- Les feux d'artifice brûlent à des températures extrêmement élevées, ce qui rend tous les feux d'artifice compétents comme sources d'allumage, en particulier ceux de type aéroporté (c'est-à-dire les fusées-bouteilles et les bougies romaines). On sait que les feux d'artifice causent chaque année d'importants dégâts matériels, notamment des incendies de terres sauvages et de structures. Utilisés de manière dangereuse, les feux d'artifice peuvent rejeter des matières en feu dans la végétation inflammable. L'utilisation des feux d'artifice augmente avant et après les périodes de vacances, entraînant une augmentation des incendies liés aux feux d'artifice.
- **Railroad : exploitation et entretien ferroviaire**
- Les travaux d'entretien de la voie (travaux à chaud, meulage des rails, remplacement de la voie, débroussaillage de l'emprise, etc.) et les opérations (échappement, freins, déraillement et autres défaillances mécaniques) sont associés à cette catégorie. Les incendies causés par les opérations ferroviaires, le personnel et le matériel roulant peuvent inclure l'entretien des voies et des emprises.

- **Smoking : fumeur / tabagisme**

Le tabagisme négligent est à l'origine d'environ 2 % de tous les incendies de forêt aux États-Unis. Ce type d'inflammation est plus courant à proximité des autoroutes interétatiques lorsque les gens jettent des cigarettes hors des véhicules en mouvement. Les cigarettes provoquent beaucoup moins d'incendies qu'avant en raison de la réduction du taux de tabagisme aux États-Unis et des cigarettes auto-extinguibles.

- **Debris burning : débris et brûlage à l'air libre**

Le brûlage inapproprié des débris de jardin est la principale source d'incendie d'origine humaine. Les brûlages dirigés se produisent lorsque les propriétaires fonciers ou les agences gouvernementales brûlent les débris de jardin ou les excès de combustible du sol forestier dans le but de réduire le risque d'incendie. Il est important d'obtenir un permis de brûlage pour ce type d'activité.

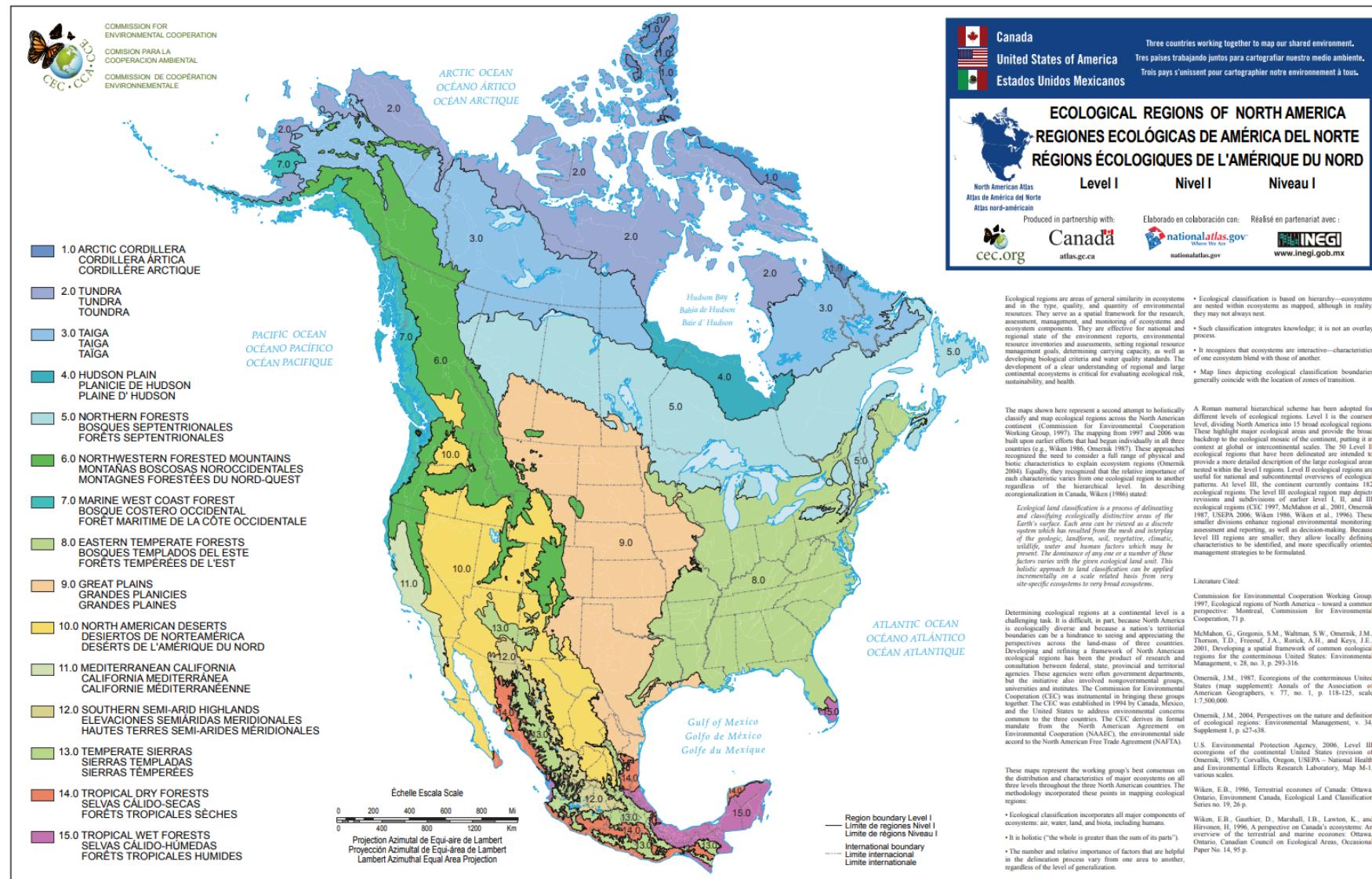
Ces incendies sont causés par des débris et des activités de brûlage à l'air libre, y compris les tas de brûlis, les débris de cour, les barils de brûlage, le brûlage de fossés/de clôtures, la lutte antiparasitaire, le brûlage de déchets à l'air libre, le brûlage d'objets personnels, les feux de détresse/de signalisation, le défrichement, les risques liés à l'emprise, réduction ou tout autre brûlage contrôlé échappé

- **Children : abus de feu par un mineur**

Les incendies causés par des mineurs de 17 ans et moins ont leur propre catégorie. Les jeunes enfants, âgés de 12 ans ou moins, motivés par une curiosité normale, peuvent utiliser le feu de manière expérimentale ou ludique, ce que l'on appelle « jouer avec des allumettes ». Ils recherchent des dispositifs d'allumage facilement accessibles et utilisent fréquemment des allumettes en papier et en bois, des briquets, des feux d'artifice ou des loupes pour allumer des incendies. Les adolescents âgés de 13 à 17 ans qui allument un incendie sont souvent en colère contre quelque chose ou envers quelqu'un. D'autres peuvent avoir de la curiosité ou un caractère destructeur et déclencher des incendies pourrait en être une extension.

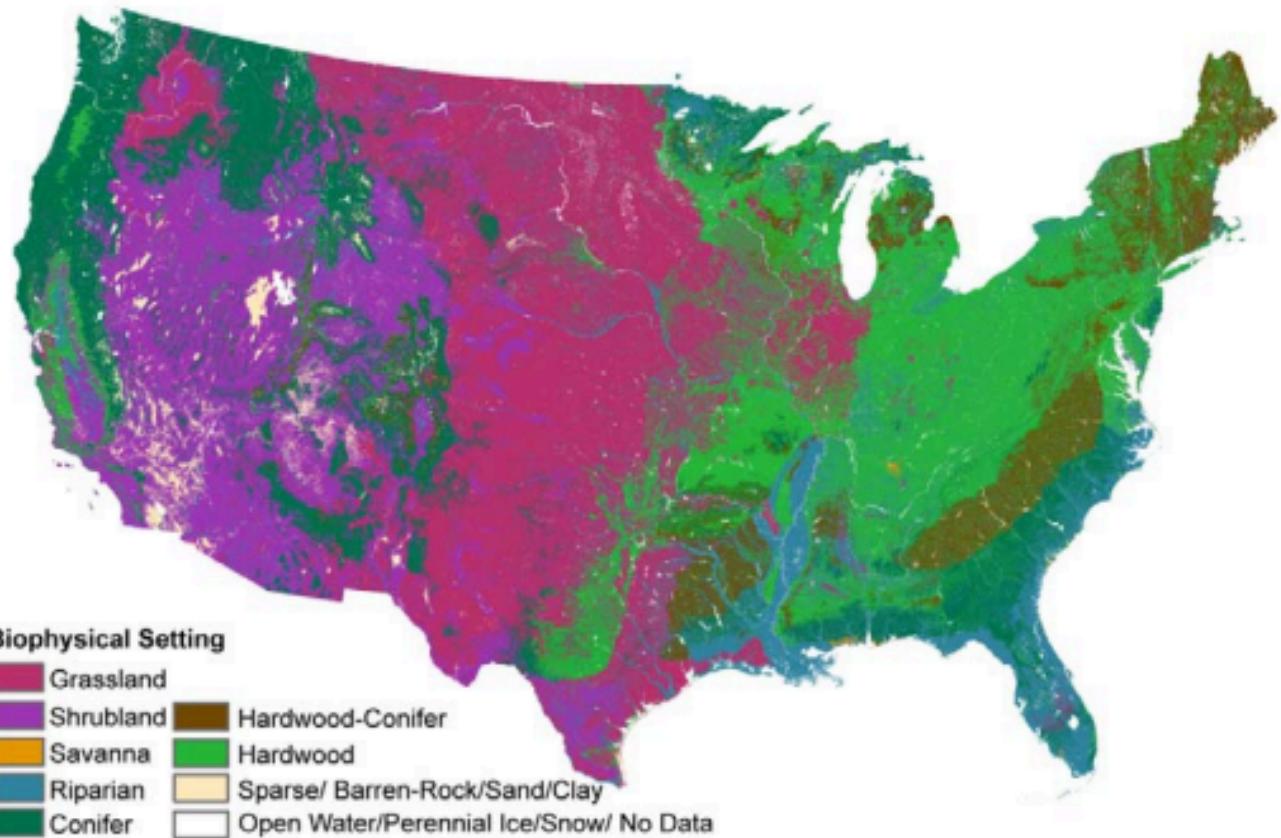
- **Missing / undefined : manquant / non défini**

## Annexe 3 : Carte des écorégions de niveau 1



source : <https://www.epa.gov/eco-research/ecoregions-north-america>

## Annexe 4 : Répartition géographique des types de végétation



Traduction des termes relatifs à la végétation :

- **Grassland** : Prairie
- **Shrubland** = Arbustes
- **Savanna** = Savane
- **Riparian** = zone plus ou moins large longeant un cours d'eau et recouverte de végétation appelée ripisylve, forêt galerie ou bande enherbée selon la nature de celle-ci
- **Conifer** = Conifère
- **Hardwood - Conifer** = Bois dur / Conifère
- **Harwood** = Bois dur
- **Sparce / Barren-Rock / Sand / Clay** = Paysage clairsemé / Roche stérile / Sable / Argile
- **Open Water / Perennial Ice / Snow / No Data** = Etendue d'eau / Glace éternelle / Neige / pas de data