INSTITUT FRANCOPHONE INTERNATIONAL

MASTER I RSC

PROMOTION 23

MODULE :FOUILLE DES DONNEES

Sujet : Application de Machine Learning pour l'estimation de la superficie de forêt en feux.

Rédigé par :

KIOMBA KAMBILO Eddy

TSHIBANGU MUABILA Jean

Professeur *:*

Nguyen Thi Minh Huyen

Hanoi, Novembre 2019

Table des matières

Table des figures

1 Introduction du Problème

Les incendies de forêt sont un problème environnemental critique pouvant causer de graves dommages. Une détection rapide et une estimation précise de la superficie brûlée par un feu de forêt peuvent aider les pompiers à contrôler efficacement les dommages. Par conséquent, l'objectif de ce travail est d'appliquer une méthode de modélisation de données de pointe pour estimer la superficie brûlée par un incendie de forêt à l'aide des techniques de machine Learning, particulièrement l'apprentissage supervisé. L'ensemble de données est constitué de données réelles sur les incendies de forêt du parc naturel de Montesinho, dans la région nord-est du Portugal. L'ensemble de données d'origine comprend 517 enregistrements avec 13 attributs. Lien vers le dataset **https ://archive.ics.uci.edu/ml/datasets/forest+fires**

L'ensemble de données **"forest fires"** comporte 12 attributs d'entrées et 1 attribut de sortie. Les attributs d'entrée de notre DataSet sont décrits de la manière suivante :

1. X : coordonnée en abscisse sur la carte du parc Montesinho avec des valeurs de 1 à 9 ;
2. Y : coordonnée en ordonnée sur la carte du parc Montesinho avec des valeurs de 2 à 9 ;
3. month : mois de l'année avec des valeurs de "jan" à "dec" ;
4. day : jour de la semaine avec des valeurs de "mon" à "sun" ;
5. FFMC : l'indice FFMC provenant du système FWI avec des valeurs de 18.7 à 96.20 ;
6. DMC : L'indice DMC provenant du système FWI avec des valeurs de 1.1 à 291.3 ;
7. DC : L'indice DC provenant du système FWI avec des valeurs de 7.9 à 860.6 ;
8. ISI : l'indice ISI provenant du système FWI avec des valeurs de 0.0 à 56.10 ;
9. Temp : température en degrés Celsius avec des valeurs de 2.2 à 33.30 ;
10. RH : humidité relative avec des valeurs de 15.0 à 100 ;
11. wind : vitesse du vent en km/h avec des valeurs de 0.40 à 9.40 ;
12. rain : pluie à l'extérieur en mm/m2 avec des valeurs de 0.0 à 6.4 ;
13. area : surface brûlée de la forêt (en ha) avec des valeurs de 0.00 à 1090.84. (cette variable de sortie est très faussée vers 0.0, donc elle peut faire sens de modéliser avec la transformation logarithmique).

Parmi ces attributs, deux (2) sont qualitatifs (month et day) et les autres sont quantitatifs. La figure ci-dessous nous montre les détails sur les différentes observations pour l'ensemble des attributs par rapport aux données automobiles recueillies sur le site.

2 Présentation de la Méthode étudiée(SVR)

S'agissant d'un problème de régression, plusieurs méthodes nous ont traversé l'esprit. Dans le cadre de notre travail nous allons utilisé le support vector Machine for régression, autrement dit support vector régression, méthode que nous allons présenter dans les lignes qui suivent.

SUPPORT VECTOR REGRESSION(SVR)

2.1 Généralités

La régression vectorielle de support (SVR) utilise les mêmes principes que la SVM pour la classification, avec seulement quelques différences mineures. Tout d'abord, étant donné que la sortie est un nombre réel, il devient très difficile de prévoir les informations disponibles, ce qui offre des possibilités infinies. En cas de régression, une marge de tolérance (epsilon) est définie comme une approximation du SVM qui l'aurait déjà demandé au problème. Mais outre ce fait, il y a aussi une raison plus compliquée, l'algorithme est donc plus compliqué à prendre en compte. Cependant, l'idée principale est toujours la même : minimiser l'erreur, individualiser l'hyperplan qui maximise la marge.

2.2 Principes

1. **Noyau :** Le noyau est une fonction utilisée pour mapper des points de données de dimension inférieure en points de données de dimension supérieure. Étant donné que SVR effectue une régression linéaire dans une dimension supérieure, cette fonction est cruciale. Il existe de nombreux types de noyau tels que le noyau polynomial, le noyau Gaussien, le noyau sigmoïde, etc.
2. **Hyper-plan :**Dans Support Vector Machine, un hyperplan est une ligne utilisée pour séparer deux classes de données dans une dimension supérieure à la dimension réelle. En SVR, l'hyperplan est la ligne utilisée pour prédire la valeur continue.
3. **Ligne de délimitation :**Deux lignes parallèles tracées des deux côtés du vecteur de support avec la valeur de seuil d'erreur, $\boldsymbol{\epsilon}$ sont appelées lignes de délimitation. Cette ligne crée une marge entre les points de données.
4. **Vecteur de support :** : Ligne à partir de laquelle la distance est minimale ou minimale à partir de deux points de données limites.

Les fonctions du noyau transforment les données en un espace de fonctions de dimension supérieure pour permettre la séparation linéaire. La régression vectorielle de support (SVR) est assez différente des autres modèles de régression. Il utilise l'algorithme SVM, un algorithme de classification, pour prédire une variable continue. Tandis que d'autres modèles de régression linéaire tentent de minimiser l'erreur entre la valeur prédite et la valeur réelle, Régression vectorielle de support essaie de faire correspondre la meilleure ligne à une valeur d'erreur prédéfinie ou à un seuil. Ce que SVR fait en ce sens, il essaye de classer toutes les lignes de prédiction en deux types, ceux qui traversent la limite d'erreur (espace séparé par deux lignes parallèles) et ceux qui ne le font pas. Les lignes qui ne dépassent pas la limite d'erreur ne sont pas considérées comme étant la différence entre la valeur prédite et la valeur réelle ayant dépassé le seuil d'erreur $\boldsymbol{\epsilon}$. Les lignes qui passent sont considérées comme un vecteur de support potentiel pour prédire la valeur d'un inconnu.
La figure 1 présente l'illustration de l'algorithme SVR :

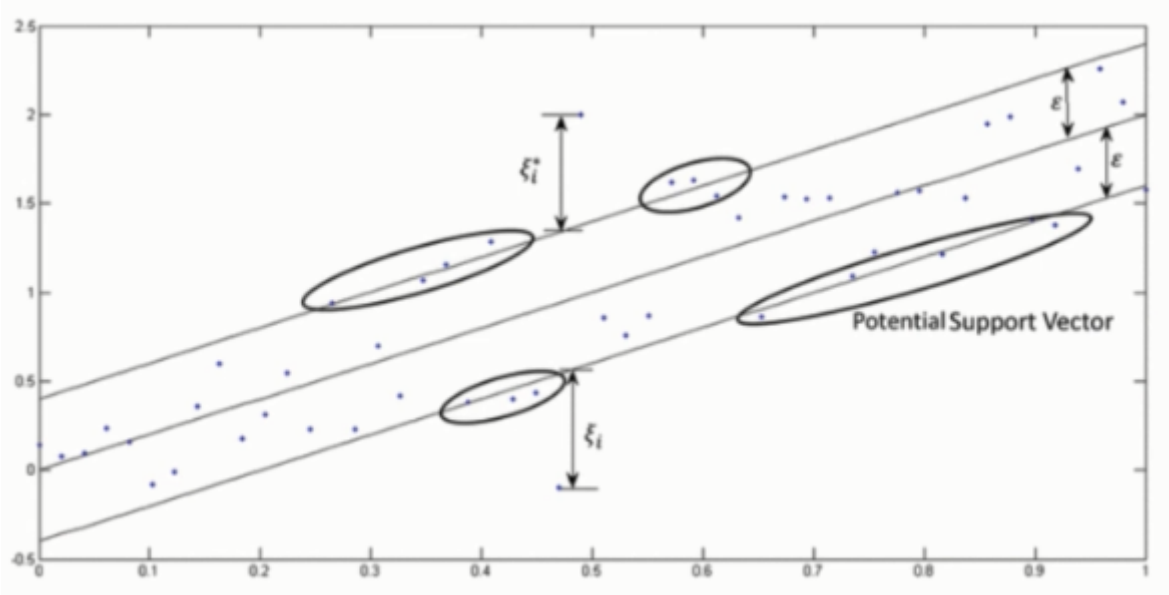

FIGURE 1 – Illustration SVR

Les limites tentent d'adapter autant d'instances que possible sans violer les marges. La largeur de la limite est contrôlée par le seuil d'erreur $\boldsymbol{\epsilon}$. Dans la classification, le vecteur de support X est utilisé pour définir l'hyperplan qui sépare les deux classes différentes. Ici, ces vecteurs sont utilisés pour effectuer une régression linéaire.

2.3 Fonction Noyau

Il existe de nombreuses fonctions de noyau possibles pour transformer les données afin qu'elles se trouvent dans un espace de fonctions plus élevé, ce qui peut aider la SVM à séparer les données de manière linéaire. Parmi les nombreuses fonctions existantes, la plus applicable est la fonction de base radiale. Son calcul (Cristianini et Shawe-Taylor, 2000) est montré dans (1) et (2). Dans notre travail, nous considérons également une fonction de noyau plus simple, appelée polynôme, comme indiqué dans (3).

$$f\left(X_i, X_j\right) = exp\left(-\gamma\left(X_i - X_j\right)^2\right) \qquad (1)$$

$$\gamma = -\frac{1}{2\sigma^2} \qquad (2)$$

$$f\left(X_i, X_j\right) = \left(\gamma X_i, X_j + \theta\right)^q \qquad (3)$$

— Xi est un vecteur de variables d'entrée.
— Xj est la variable cible,
— $\boldsymbol{\gamma}$ est un paramètre gamma,
— $\boldsymbol{\epsilon}$ est une variable libre, q est le degré de fonction polynomiale,
— $\boldsymbol{\theta}$ est le biais.

2.4 Étape de mise en place l'algorithme SVR

Les étapes ci-après décrits la mise en place de l'algorithme SVR :

1. Recueillir un jeu d'entraînement
2. Choisissez un noyau et ses paramètres ainsi que toute régularisation nécessaire.
3. Former la matrice de corrélation, K
4. Entraînez la machine, exactement ou approximativement, pour obtenir les coefficients de contraction, = $\{i\}$

Utiliser ces coefficients pour créer un estimateur f(X , x*) = y*.
Nous arrivons maintenant à la matrice de corrélation, défini par la formule :

Correlation Matrix

$$K_{i,j} = \exp\left(\sum_k \theta_k \left|x_k^i - x_k^j\right|^2\right) + \epsilon\delta_{i,j}$$

Dans l'équation ci-dessus, nous évaluons le noyau pour toutes les paires de points de l'ensemble d'apprentissage et ajoutons le régulariser résultant dans la matrice.
La partie principale de l'algorithme est, K = y.
Ici, y est le vecteur de valeurs correspondant à votre jeu de formation, K est la matrice de corrélation Et, est l'ensemble des inconnus que nous devons résoudre. Sa valeur est obtenue à partir de l'équation suivante.

$$K^{-1}\mathbf{y}=\mathbf{1}$$

Maintenant que le paramètre est connu, nous pouvons former l'estimateur. Nous utilisons tous les coefficients trouvés lors du processus d'optimisation et le noyau avec lequel nous avons commencé. Pour estimer la valeur inconnue, y*, pour un point de test x *, nous avons besoin du produit intérieur de et de la matrice de corrélation K. y* = K. Ensuite, nous estimons les éléments de la matrice de coefficients comme suit :

$$k_i = \exp\left(\sum_k \theta_k \left|x_k^i - x_k^\star\right|^2\right)$$

Globalement, après toutes ces étapes, le modèle SVR est maintenant prêt à prédire les valeurs inconnues.

2.5 Indicateurs pour valider un modèle prédiction

Une fois que le modèle a été créé avec le jeu de données d'apprentissage, il est nécessaire de calculer des indicateurs objectifs pour évaluer si le modèle a généré des prédictions pertinentes pour la variable étudiée. Les valeurs « vraies » de cette variable sont censées être connues pour l'ensemble des jeux de données d'apprentissage et de validation. Intuitivement, pour chaque

échantillon dans le jeu de données de validation, on cherche à savoir si les valeurs prédites par le modèle sont proches des vraies valeurs du jeu de données de validation. Les indicateurs suivants sont les plus souvent reportés :

2.5.1 L'erreur moyenne absolue : MAE

L'erreur moyenne absolue est donnée par la relation :

$$MAE = \frac{\sum_{i=1}^{n} \left| Y_i - \widehat{Y}_i \right|}{n} \qquad (5)$$

où Y_i est la valeur réelle de la surface brûlée, $\hat{Y}_i$ est la superficie brûlée estimée, et n est le nombre de enregistrements de données.
La seule différence entre le MAE et le biais est la valeur absolue des différences entre les valeurs réelles et prédites. Un des avantages de l'indicateur MAE est qu'il donne une meilleure idée de la qualité de prédiction. Par contre, il n'est pas possible de savoir si le modèle a tendance à sous ou sur-estimer les prédictions.
N.B : L'erreur opté pour notre cas.

2.5.2 L'erreur quadratique moyenne : RMSE

L'erreur quadratique moyenne est donnée par la relation :

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} \left(Y_i - \widehat{Y}_i \right)^2}{n}} \qquad (6)$$

Un dernier indicateur pertinent est le RMSE. Cet indice fournit une indication par rapport à la dispersion ou la variabilité de la qualité de la prédiction. Le RMSE peut être relié à la variance du modèle. Souvent, les valeurs de RMSE sont difficiles à interpréter parce que l'on n'est pas en mesure de dire si une valeur de variance est faible ou forte. Pour pallier à cet effet, il est plus intéressant de normaliser le RMSE pour que cet indicateur soit exprimé comme un pourcentage de la valeur moyenne des observations. Cela peut être utilisé pour donner plus de

sens à l'indicateur.
Par exemple, un RMSE de 10 est relativement faible si la moyenne des observations est de 500. Pourtant, un modèle a une variance forte s'il conduit à un RMSE de 10 alors que la moyenne des observations est de 15. En effet, dans le premier cas, la variance du modèle correspond à seulement 5% de la moyenne des observations alors que dans le second cas, la variance atteint plus de 65% de la moyenne des observations.

2.5.3 Le coefficient de détermination : R^2

Le coefficient de détermination est donné par la relation :

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y_i})^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2}$$

- Où n est le nombre de mesures,
- y_i est la valeur de la $i^{ème}$ observation du jeu de données de validation,
- $\bar{y}$ est la moyenne des valeurs du jeu de données de validation,
- $\hat{Y_i}$ est la valeur prédite pour la $i^{ème}$ observation

A noter que dans la précédente équation, la fraction est le ratio entre la somme des écarts résiduels et la somme des écarts totaux. Les résidus représentent les différences entre la prédiction et la réalité. Plus R2 est proche de 1, meilleure est la prédiction. Dans un graphique à deux dimensions représentant les valeurs réelles en abscisse et les valeurs prédites en ordonnées, on cherche à ajuster une régression linéaire à l'ensemble des données.

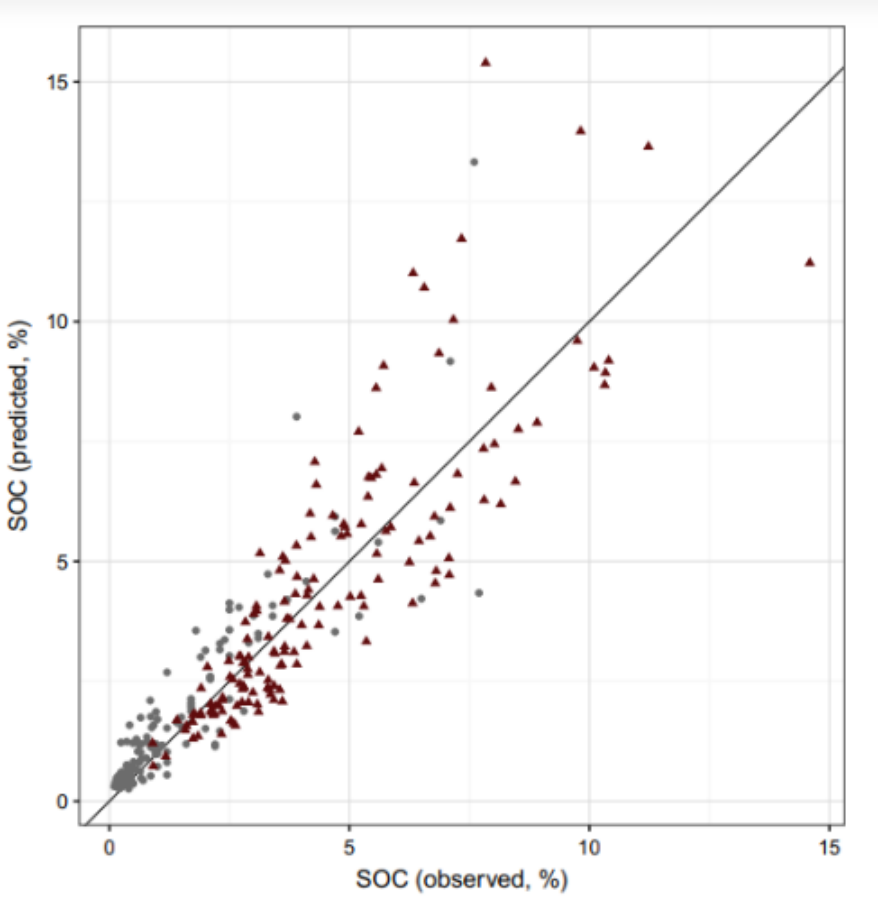

FIGURE 2 – Ajustement de qualité entre les valeurs réelles et prédites

Cependant, il est nécessaire de faire attention lorsque l'on calcule ce coefficient de détermination parce qu'il peut conduire à des conclusions erronées. En effet, certains points d'influence peuvent particulièrement augmenter la valeur du coefficient de détermination, ce qui peut parfois laisser penser que les prédictions sont assez précises. Ici, le modèle est relativement bien ajusté aux données. Les points gris proches de l'origine du graphique aident néanmoins à augmenter la valeur du coefficient de détermination

N.B : Aucun des indicateurs précédemment cités n'est clairement meilleur que les autres. Au contraire, toutes ces métriques sont à utiliser ensemble pour mieux comprendre et caractériser la qualité d'un modèle de prédiction.

3 Préparation des données

3.1 Description sur les Attributs

Les attributs FFMC, DMC, DC, ISI font partie des principaux composants permettant de calculer les échelles d'évaluation du danger des incendies de forêt (Taylor et Alexander, 2006) selon le système météorologique canadien.
La FFMC détermine l'influence des litières sur l'inflammation et la propagation du feu.
Le DMC et le DC identifient l'intensité du feu, tandis que l'ISI est corrélé à la propagation de la vitesse du feu. Les quatre autres attributs (température, humidité relative, vent, pluie) sont des données météorologiques pouvant également influer sur la propagation du feu. La cible de notre modélisation est le dernier attribut, la zone. La figure 2 illustrent les attributs principaux de notre dataset. Nous signalons que la variable X et Y qui représente ici la géolocalisation n'as pas été utiliser dans nos attributs principaux dans le cadre de ce travail.

Attribute name	Description	Unit
FFMC	Fine Fuel Moisture Code	--
DMC	Duff Moisture Code	--
DC	Drought Code	--
ISI	Initial Spread Index	--
Temp	Temperature	^{o}C
RH	Relative Humidity	%
Wind	Wind speed	km/h
Rain	Rain volume	mm/m^2
Area	Total burned area	ha

FIGURE 3 – Attributs principaux du DataSet

Dans les lignes suivantes, nous allons faire le pré-traitement des données. Pour notre travail, nous allons travailler Avec Python et Tanagra pour nos différents analyses.

3.1.1 Data Preprocessing

Avec quelques codes Python, nous allons afficher les informations pour nos données.

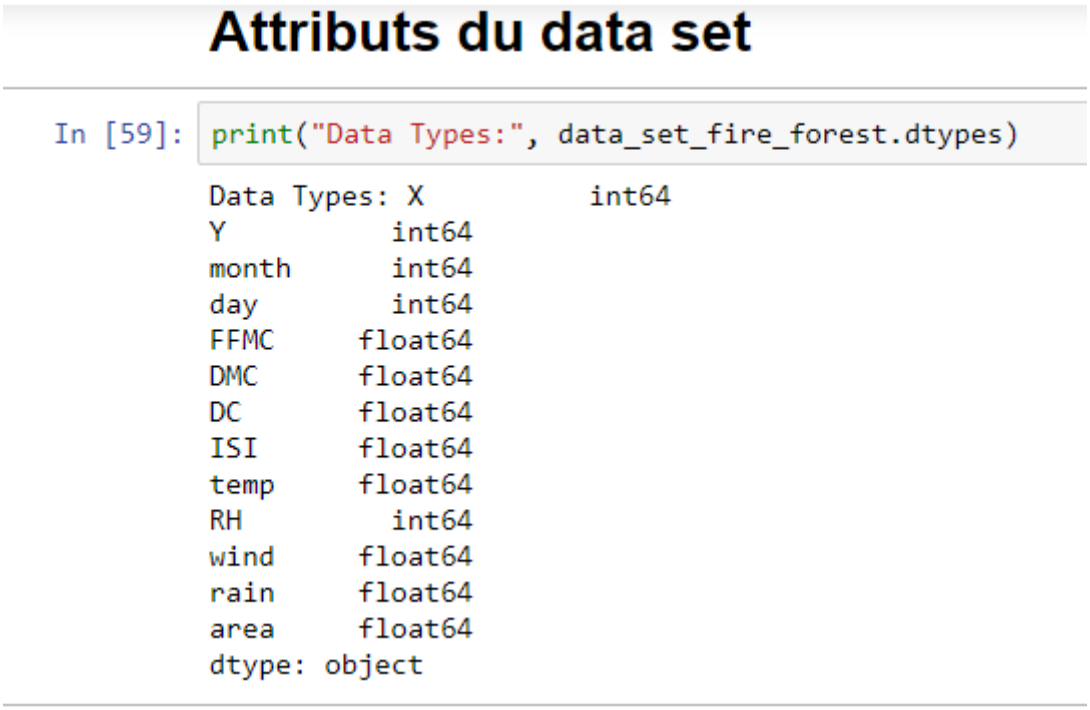

FIGURE 4 – Affichage des différents Attributs

Pour faire nos différentes Analyses, nous avons codifier nos différentes variables nominales.

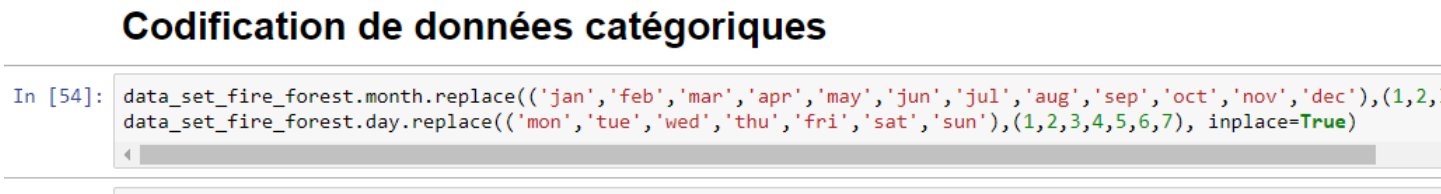

FIGURE 5 – Codification des Variables nominales

En ce qui concerne la codification des variables nominales, nous avons fait l'ANOVA sur nos différentes variables pour déterminer de l'influence de ce variable nominale avant de décider de leurs codification. ceci n'influence en rien nos résultats sur la corrélation des variables.

Affichage des déscriptions statistiques

```
In [90]: print(data_set_fire_forest.describe())
```

	X	Y	month	day	FFMC	DMC
count	517.000000	517.000000	517.000000	517.000000	517.000000	517.000000
mean	4.669246	4.299807	7.475822	4.259188	90.644681	110.872340
std	2.313778	1.229900	2.275990	2.072929	5.520111	64.046482
min	1.000000	2.000000	1.000000	1.000000	18.700000	1.100000
25%	3.000000	4.000000	7.000000	2.000000	90.200000	68.600000
50%	4.000000	4.000000	8.000000	5.000000	91.600000	108.300000
75%	7.000000	5.000000	9.000000	6.000000	92.900000	142.400000
max	9.000000	9.000000	12.000000	7.000000	96.200000	291.300000

	DC	ISI	temp	RH	wind	rain
count	517.000000	517.000000	517.000000	517.000000	517.000000	517.000000
mean	547.940039	9.021663	18.889168	44.288201	4.017602	0.021663
std	248.066192	4.559477	5.806625	16.317469	1.791653	0.295959
min	7.900000	0.000000	2.200000	15.000000	0.400000	0.000000
25%	437.700000	6.500000	15.500000	33.000000	2.700000	0.000000
50%	664.200000	8.400000	19.300000	42.000000	4.000000	0.000000
75%	713.900000	10.800000	22.800000	53.000000	4.900000	0.000000
max	860.600000	56.100000	33.300000	100.000000	9.400000	6.400000

FIGURE 6 – Affichage statistiques des Différentes variables

En ce qui concerne la corrélation des variables, nous avons utiliser la corrélation de 'pearson'

Correlation entre attributs area

```
In [60]: print("Correlation:", data_set_fire_forest.corr(method='pearson'))
```

Correlation:	X	Y	month	day	FFMC	DMC	DC	\
X	1.000000	0.539548	-0.065003	-0.024922	-0.021039	-0.048384	-0.085916	
Y	0.539548	1.000000	-0.066292	-0.005453	-0.046308	0.007782	-0.101178	
month	-0.065003	-0.066292	1.000000	-0.050837	0.291477	0.466645	0.868698	
day	-0.024922	-0.005453	-0.050837	1.000000	-0.041068	0.062870	0.000105	
FFMC	-0.021039	-0.046308	0.291477	-0.041068	1.000000	0.382619	0.330512	
DMC	-0.048384	0.007782	0.466645	0.062870	0.382619	1.000000	0.682192	
DC	-0.085916	-0.101178	0.868698	0.000105	0.330512	0.682192	1.000000	
ISI	0.006210	-0.024488	0.186597	0.032909	0.531805	0.305128	0.229154	
temp	-0.051258	-0.024103	0.368842	0.052190	0.431532	0.469594	0.496208	
RH	0.085223	0.062221	-0.095280	0.092151	-0.300995	0.073795	-0.039192	
wind	0.018798	-0.020341	-0.086368	0.032478	-0.028485	-0.105342	-0.203466	
rain	0.065387	0.033234	0.013438	-0.048340	0.056702	0.074790	0.035861	
area	0.063385	0.044873	0.056496	0.023226	0.040122	0.072994	0.049383	

FIGURE 7 – Corrélation entre chaque paire des variables(1)

	ISI	temp	RH	wind	rain	area
X	0.006210	-0.051258	0.085223	0.018798	0.065387	0.063385
Y	-0.024488	-0.024103	0.062221	-0.020341	0.033234	0.044873
month	0.186597	0.368842	-0.095280	-0.086368	0.013438	0.056496
day	0.032909	0.052190	0.092151	0.032478	-0.048340	0.023226
FFMC	0.531805	0.431532	-0.300995	-0.028485	0.056702	0.040122
DMC	0.305128	0.469594	0.073795	-0.105342	0.074790	0.072994
DC	0.229154	0.496208	-0.039192	-0.203466	0.035861	0.049383
ISI	1.000000	0.394287	-0.132517	0.106826	0.067668	0.008258
temp	0.394287	1.000000	-0.527390	-0.227116	0.069491	0.097844
RH	-0.132517	-0.527390	1.000000	0.069410	0.099751	-0.075519
wind	0.106826	-0.227116	0.069410	1.000000	0.061119	0.012317
rain	0.067668	0.069491	0.099751	0.061119	1.000000	-0.007366
area	0.008258	0.097844	-0.075519	0.012317	-0.007366	1.000000

FIGURE 8 – Corrélation entre chaque paire des variables(2)

3.1.2 Prétraitement de données manquantes

Au cours de cette section il s'agira de vérifier si notre dataset dispose des variables manquantes, au besoin comment traiter notre dataset pour compenser ces variables manquantes. Avec Python, après test, nous avons le résultat suivant :

Vérification de données manquantes

```
In [91]: print(data_set_fire_forest.isnull().sum)
```

```
<bound method DataFrame.sum of          X      Y  month    day   FFMC    DMC
0    False  False  False  False  False  False  False  False  False  False
1    False  False  False  False  False  False  False  False  False  False
2    False  False  False  False  False  False  False  False  False  False
3    False  False  False  False  False  False  False  False  False  False
4    False  False  False  False  False  False  False  False  False  False
5    False  False  False  False  False  False  False  False  False  False
6    False  False  False  False  False  False  False  False  False  False
7    False  False  False  False  False  False  False  False  False  False
8    False  False  False  False  False  False  False  False  False  False
9    False  False  False  False  False  False  False  False  False  False
10   False  False  False  False  False  False  False  False  False  False
11   False  False  False  False  False  False  False  False  False  False
12   False  False  False  False  False  False  False  False  False  False
13   False  False  False  False  False  False  False  False  False  False
14   False  False  False  False  False  False  False  False  False  False
15   False  False  False  False  False  False  False  False  False  False
16   False  False  False  False  False  False  False  False  False  False
17   False  False  False  False  False  False  False  False  False  False
```

FIGURE 9 – Vérification de valeurs Manquantes

Interprétation : Après check, comme nous le montre le tableau précédent, nous voyons que toutes nos valeurs sont à False, ce qui nous pousse à dire que nous avons un dataset équilibré et qu'il n'y a aucune valeur manquante.

3.2 Données d'entraînement et validation

La version la plus simple de la validation croisée consiste à diviser le jeu de données en deux sous-ensembles : L'ensemble d'entraînement et l'ensemble de test.

FIGURE 10 – Division d'un ensemble de données en un ensemble d'apprentissage et un ensemble d'évaluations

On entraîne notre algorithme sur le premier sous-ensemble de données. Ensuite, on compare les indicateurs de performances en appliquant l'algorithme sur le premier et le second ensemble de données.
Si les indicateurs trouvés pour l'ensemble d'entraînement sont bien supérieurs à ceux trouvés sur l'ensemble de test, notre algorithme sur-apprend et il peut valoir le coup de retoucher le modèle pour améliorer les performances sur l'ensemble de test.
Sinon, il faut penser à augmenter la complexité du modèle afin d'obtenir de meilleures performances.

3.2.1 Découpage de données d'entraînements et validations de Jeu de données.

Lors de notre travail, nous avons procédé à 3 découpages de DataSet ci-après :
Découpage 1 : 70% de données d'apprentissage et 30% de données de validation.
Extrait du code python :

```
In [142]: X_train,X_test,Y_train,Y_test = train_test_split(X,Y, test_size = 0.3, random_state = 42)
```

Découpage 2 : 80% de données d'apprentissage et 20% de données de validation.
Extrait du code python :

```
In [142]: X_train,X_test,Y_train,Y_test = train_test_split(X,Y, test_size = 0.2, random_state = 42)
```

Découpage 3 : 90% de données d'apprentissage et 10% de données de validation
Extrait du code python :

```
In [142]: X_train,X_test,Y_train,Y_test = train_test_split(X,Y, test_size = 0.1, random_state = 42)
```

4 Choix des paramètres et application de la méthode étudiée au jeu de données et évaluation

Dans la section précédente, nous avons eu a faire quelques tests qui s'avèrent être très important pour la détermination de notre scénario le plus probable à utiliser mais aussi pour le choix des paramètres influant sur notre variable à expliquer qui est Area.

Par rapport au Tableau de corrélation que nous avons présenté précédemment, quatre variable donne des corrélations par rapport à notre variable de sortie à savoir :

- RH : avec une corrélation négative, car plus le solide n'est pas humide, plus il y'a une forte chance.
- temp : corrélation positive, car plus la température est élevé, plus il y'a risque de propagation d'incendie.
- wind : Corrélation positive car avec un vent fort, il y'a aussi risque que l'incendie évolue rapidement.
- DMC

```
In [66]: results = []
         names = []
         scoring = []
         for name, model in models:
             # Fit the model
             model.fit(attributs_explicatifs_bis, attribut_tag)
             predictions = model.predict(attributs_explicatifs_bis)
             # Evaluate the model
             score = explained_variance_score(attribut_tag, predictions)
             mae = mean_absolute_error(predictions, attribut_tag)
             # print("%s: %.2f%%" % (model.metrics_names[1], scores[1]*100))
             results.append(mae)
             names.append(name)
             msg = "%s: %f (%f)" % (name, score, mae)
             print(msg)

         SVM: 0.005588 (12.085837)
```

FIGURE 11 – Vérification de valeurs Manquantes

Nota : Comme abordé au point 3, notre jeu de données a été entraîné suivant différents modèle de répartition de dataset(20-80, 30-70 et 10 - 90). et exécuté a 10 itérations suivant chaque modèle de répartition.

Nous ne saurons représenté tous ces résultats sur le rapport vu la quantité des captures a y mettre, mais pour les 10 premiers itérations du modèle 20-80 par exemple, nous avons les résultats suivants pour le Mae (12.08583, 11.90897, 12.08237, 12.085120,6745523, 12.09789, 12.04563, 12.08362, 12.08237, 12.10983, 12.09281)

La moyenne de notre Mae est donc égale à 12,067455

Interprétation : Nous remarquons comme nous l'avons dit ci-haut en se basant sur le mean absolute error(Mae) qui est un indicateur de mesure de performance.

Notre MAE = 12%, donc notre seuil de confiance est égale à 1-MAE, donc seuil de 88%, soit autrement nous pouvons dire que avec nos variables d'entrée, nous avons un modèle à 80% fiable.

5 Comparaison avec le résultat d'application d'une autre méthode d'apprentissage supervisé

Au vu de la méthode que nous avons utilisé, le SVR, pour la comparaison, nous avons choisi deux autres méthodes de régression à savoir.

— Régression Linéaire.

— Random Forest Regressor.

Régression Linéaire

Comme notre problème, s'avère être un problème de régression, nous allons utiliser la régression linéaire. Nous allons faire trois répartition de dataset en 70-30, 80-20, 10-90 que nous présenterons dans la section suivante pour l'évaluation globale du modèle.

5.1 Évaluation Globale Du Modèle 70-30

5.1.1 Tableau d'analyse de variance

Premier outil pour l'évaluation de la régression, le tableau d'analyse de variance décompose la variabilité totale de l'endogène (SCT) en variabilité expliquée par le modèle (SCE) et variabilité résiduelle (SCR), non prise en compte par le modèle.
Le ratio (SCE / SCT) indique la part de variabilité expliquée, c'est le coefficient de détermination R^2, il varie entre 0 et 1. Lorsque le R^2 est égal à 1, le modèle permet de prévoir avec exactitude les valeurs prises par l'endogène Y c.-à-d. SCR = 0.

Global results

Endogenous attribute	area
Examples	361
R^2	0,011128
Adjusted-R^2	0,002818
Sigma error	61,948898
F-Test (3,357)	1,3392 (0,261393)

Analysis of variance

Source	xSS	d.f.	xMS	F	p-value
Regression	15417,8819	3	5139,2940	1,3392	0,2614
Residual	1370046,7270	357	3837,6659		
Total	1385464,6088	360			

FIGURE 12 – Tableau Analyse de variance

Notre variable d'ajustement R et R^2 s'approchent de 0 plutôt que de 1, ce qui nous pousse à conclure que dans l'ensemble, le modèle n'est pas significatif.

5.1.2 Test de significativité globale

La régression est-elle globalement statistiquement pertinente ? Pour répondre à cette question, nous introduisons le test de significativité globale. Il s'agit de vérifier si aucune des exogènes, prises simultanément, n'apporte d'information sur l'endogène.
p = Pr(>= Fisher)

Lorsque la (p-value) est inférieure au seuil de significativité $\boldsymbol{\alpha}$ (généralement 5%), on décide le rejet de l'hypothèse nulle c.-à-d. on conclut à la significativité globale du modèle.
Nota : Dans notre modèle le p-value = 0.26 > 0.05, le modèle n'est donc pas significatif

5.1.3 Évaluation des coefficients

L'étape suivante consiste à évaluer l'apport individuel des variables dans l'explication de Y. Nous utilisons pour ce faire le test de significativité individuelle des coefficients.
Dans le tableau « COEFFICIENTS », Tanagra fournit directement ˆaˆ j et t j . Sous l'hypothèse nulle H0, la statistique suit une loi de Student à (n – p – 1 = 24) degrés de libertés.

Dans la dernière colonne, nous obtenons la p-value du test. La variable est significative si elle est inférieure au risque $\boldsymbol{\alpha} = 5\%$. Dans notre cas, la variable WEIGHT semble être la seule significative avec une p-value égale à 0.00014

Nota : Aucune donnée n'est significative

Coefficients

Attribute	Coef.	std	t(357)	p-value
Intercept	14,366688	22,195257	0,647286	0,517863
DMC	0,101003	0,101002	1,000011	0,317983
temp	0,005978	0,979483	0,006103	0,995134
RH	-0,300261	0,268357	-1,118887	0,263941

FIGURE 13 – Tableau Analyse de variance-coefficient

5.2 Évaluation Globale Du Modèle 80-20

5.2.1 Tableau d'analyse de variance

En se basant sur le même principe que nous avons soulevé ci-haut, Notre variable d'ajustement R et R^2 s'approchent de 0 plutôt que de 1, ce qui nous pousse à conclure que dans l'ensemble, le modèle n'est pas significatif.

5.2.2 Test de significativité

Lorsque la (p-value) est inférieure au seuil de significativité $\boldsymbol{\alpha}$ (généralement 5%), on décide le rejet de l'hypothèse nulle c.-à-d. on conclut à la significativité globale du modèle.

Nota : Dans notre modèle le p-value = 0.21 > 0.05, le modèle n'est donc pas significatif dans son ensemble

Global results

Endogenous attribute	area
Examples	413
R^2	0,010792
Adjusted-R^2	0,003537
Sigma error	58,675665
F-Test (3,409)	1,4874 (0,217386)

Analysis of variance

Source	xSS	d.f.	xMS	F	p-value
Regression	15362,7882	3	5120,9294	1,4874	0,2174
Residual	1408118,9691	409	3442,8337		
Total	1423481,7573	412			

Coefficients

Attribute	Coef.	std	t(409)	p-value
Intercept	12,757159	20,007513	0,637618	0,524079
DMC	0,080024	0,066597	1,201615	0,230208
temp	0,072869	0,811359	0,089811	0,928481
RH	-0,247244	0,238032	-1,038702	0,299557

FIGURE 14 – Tableau Analyse de variance

5.2.3 Évaluation des coefficients

Nota : En se basant sur student, tous nos coefficients sont supérieurs au seuil de student, 5%, donc aucun de coefficient n'est significatif.

Coefficients

Attribute	Coef.	std	t(409)	p-value
Intercept	12,757159	20,007513	0,637618	0,524079
DMC	0,080024	0,066597	1,201615	0,230208
temp	0,072869	0,811359	0,089811	0,928481
RH	-0,247244	0,238032	-1,038702	0,299557

FIGURE 15 – Tableau Analyse de variance-coefficient

5.3 Évaluation Globale Du Modèle 90-10

5.3.1 Tableau d'analyse de variance

Global results

Endogenous attribute	area
Examples	464
R^2	0,016665
Adjusted-R^2	0,010252
Sigma error	65,437183
F-Test (3,460)	2,5986 (0,051729)

Analysis of variance

Source	xSS	d.f.	xMS	F	p-value
Regression	33381,7434	3	11127,2478	2,5986	0,0517
Residual	1969731,4844	460	4282,0250		
Total	2003113,2278	463			

Coefficients

Attribute	Coef.	std	t(460)	p-value
Intercept	11,318191	21,298091	0,531418	0,595386
DMC	0,077652	0,059321	1,309019	0,191181
temp	0,368555	0,842017	0,437705	0,661805
RH	-0,319862	0,251024	-1,274228	0,203226

FIGURE 16 – Tableau Analyse de variance

Notre variable d'ajustement R et R^2 s'approchent de 0 plutôt que de 1, ce qui nous pousse à conclure que dans l'ensemble, le modèle n'est pas significatif globalement.

5.3.2 Test de significativité globale

Lorsque la (p-value) est inférieure au seuil de significativité $\boldsymbol{\alpha}$ (généralement 5%), on décide le rejet de l'hypothèse nulle c.-à-d. on conclut à la significativité globale du modèle.

Nota : Dans notre modèle le p-value = 0.05 >= 0.05(seuil), le modèle n'est donc pas significatif dans son ensemble.

5.4 Random Forest Regressor

Une forêt aléatoire est une technique d'ensemble capable d'effectuer à la fois des tâches de régression et de classification avec l'utilisation de plusieurs arbres de décision et une technique appelée agrégation de Bootstrap, connue sous le nom de mise en sac.
L'idée de base est de combiner plusieurs arbres de décision pour déterminer le résultat final plutôt que de s'appuyer sur des arbres de décision individuels. Dans le cas de notre travail, nous appliquerons cela aussi à trois de nos datasets toujours avec le même proportion que dans nos méthodes linéaires.
Les caractéristiques sont toujours permutées au hasard à chaque scission. Par conséquent, la meilleure répartition trouvée peut varier, même avec les mêmes données d'apprentissage, max_features = n_features et bootstrap = False, si l'amélioration du critère est identique pour plusieurs scissions énumérées lors de la recherche de la scission optimale.
Comme il s'agit de régression, nous nous baserons plus sur **le mean absolute error** qui est considéré dans le problème de régression comme un indicateur de mesure des erreurs sur la globalité du modèle.

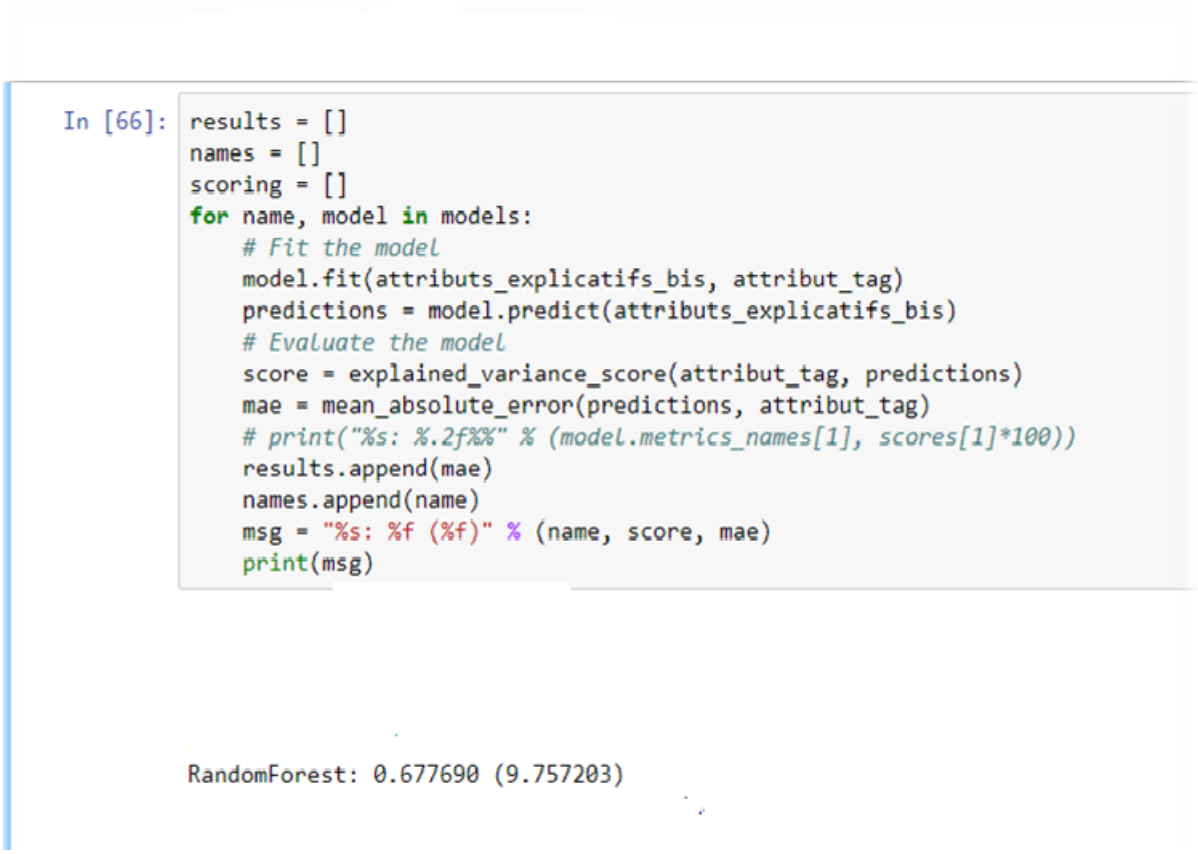

FIGURE 17 – Prédiction avec Random Forest

Interprétation : Random forest souvent évité d'être utilisé dans la pratique à cause de sa capacité a faire de la sur-apprentissage, nous a semblé être la méthode idéale pour l'interprétation de nos différents résultats.

Avec une erreur Mae (erreur sur la précision) = 9.7%, nous avons donc une précision sur la prédiction qui est égale a 100-9.7 = 90.3%.

La Méthode de random Forest nous donne donc une précision de 90% d'un point de vue fiabilité.

6 Conclusion

Ce travail concerné de définir un modèle capable de prédire les surfaces de forêt brûlées dans le parc de Montesinho situé dans la région nord-est du Portugal. Pour y arriver, plusieurs étapes ont été cruciale dans notre démarche, le prétaitement des données qui nous a permis via des tests de corrélations de pouvoir dégager des attributs pertinents par rapport à notre variable à expliquer et aussi les méthodes de régression utilisé pour évaluer nos différents modèles et prédiction. Trois Méthodes de régression ont été utilisés à savoir, le Support Vector Régression qui a été notre méthode de base avec une fiabilité de 88%, puis la régression Linéaire qui s'est avéré être moins efficace avec une précision de 82%, puis la dernière méthode a été le random forest Regressor avec une précision de 90%. De nos trois méthodes utilisés, le Random Forest a été la pus efficace grâce a son taux de précision.

Bibliographie