Année Académique : 2023-2024

MASTER II

MODELISATION, CALCUL ET AIDE A LA DECISION

Fouille de Données et Extractions de Connaissances

Projet :

La Fouille de Données au Service du Développement Durable

08/01/2024

Enseignant :	*Etudiants* :
Sabeur **Aridhi**	Vanga Gustave Hermann MOULO
Laura Zanella **Calzada**	Pascaline KOUDA

Table des matières

1 Introduction **3**

2 Analyse exploratoire des données **3**

3 Prétraitement des données **4**
3.1 Nettoyage des données 4
3.2 Transformation des données 4

4 Modélisation - Défi 1 **4**
4.1 Classification uni-label 4
4.2 Classification multi-label 5

5 Conclusion **6**

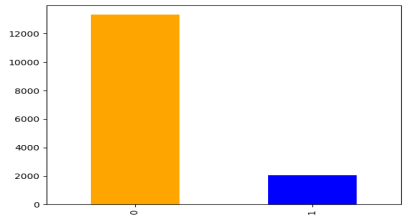
1 Introduction

Notre étude se porte sur la prédiction du défaut des arbres de Grenoble à partir des données comme l'adresse du secteur, le diamètre du tronc, le stade de développement ... A cet effet, Le langage de programmation Python, offrant une implémentation des principaux algorithmes de classification, sera utilisé dans ce travail.

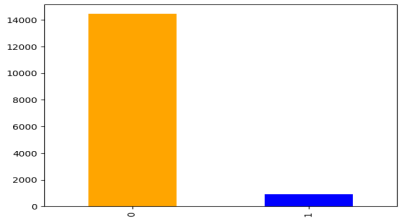
2 Analyse exploratoire des données

L'analyse descriptive du jeu de données nous a donné les informations suivantes :

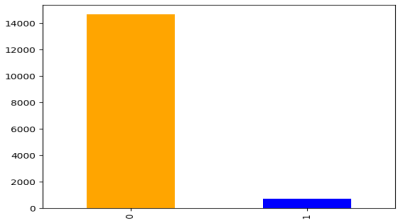
- 15375 instances
- 34 attributs dont certains décrivent l'arbre (Code, DiamètreArbreÀUnMètre, AnnéeDePlantation, Espèce...), son emplacement (Adr_Secteur, Trottoir...), des informations établies à l'occasion de diagnostics (AnnéeRéalisationDiagnostic, AnnéeTravauxPréconisésDiag...)
- Un total de 20,9 % des données manquantes dans l'ensemble du jeu de données
- 5 variables cibles qui sont **DEFAUT**, COLLET, HOUPPIER, RACINE et TRONC qui sont distribués comme suit :



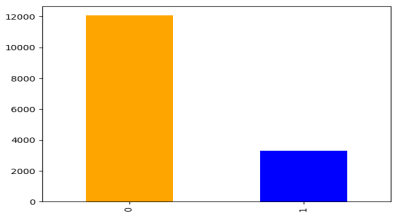
(a) Distribution de Tronc



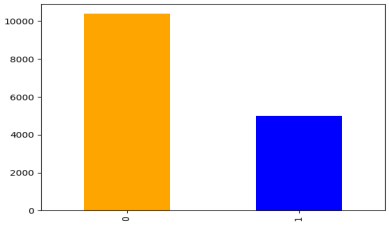
(b) Distribution de Collet



(c) Distribution de Racine



(d) Distribution de Houppier



(e) Distribution de Defaut

FIGURE 1 – Distribution des variables cibles

3 Prétraitement des données

3.1 Nettoyage des données

Les attributs VARIETE , TYPEIMPLANTATIONPLU , TRAITEMENTCHENILLES , RAISONDEPLANTATION , INTITULEPROTECTIONPLU , IDENTIFIANTPLU et REMARQUES ont été supprimés du jeu de donnée car les pourcentages de valeurs manquantes sont supérieurs à 70%.

Les attributs suivants ont été supprimé à cause de leur redondance :
- CODE est supprimé parce que c'est l'identifiant unique donné à chaque plante recensée
- CODE_PARENT_DESC est supprimée car elle décrit CODE_PARENT
- SOUS_CATEGORIE_DESC est supprimée car elle décrit SOUS_CATEGORIE
- CODE_PARENT est supprimée car elle est redondante avec ADR_SECTEUR, Coord_x, Coord_y et elle contient beaucoup de variables uniques près de 1000

Nous avons ensuite fait une imputation par le mode pour gérer le reste des valeurs manquantes présent dans le jeu de données.

3.2 Transformation des données

Les attributs ESPECE, GENRE_BOTA, TRAVAUXPRECONISESDIAG, DIAMETREARBREAUNMETRE qui contiennent respectivement 226, 107, 15, 19 modalités ont été transformés en attributs binaires suivant le nombre de leurs modalités.
Les attributs VIGUEUR, STADEDEDEVELOPPEMENT, STADEDEVELOPPEMENTDIAG, SOUS_CATEGORIE, PRIORITEDERENOUVELLEMENT, NOTEDIAGNOSTIC, FREQUENTATIONCIBLE, TROTTOIR ont été transformés en attributs numériques.
Nous nous retrouvons à présent avec 386 attributs.

4 Modélisation - Défi 1

Nous nous consacrerons uniquement au défi 1 qui consiste en deux tâches de prédiction, une classification unilabel visant à déterminer, à partir des données disponibles, si l'arbre a ou non un défaut et une classification multilabel visant à determiner à quel(s) niveaux se présentent ces défauts sachant qu'un arbre peut présenter un défaut à différents endroits : racine, tronc, collet, houppier.

4.1 Classification uni-label

Étant donné que nous sommes en face d'un problème de classification, nous utilisons donc les algorithmes d'apprentissage qui y sont adaptés.

Pour cela, nous considérons les modèles suivants :
— Régression logistique (LR)
— Forêt aléatoire (Random Forest)
— Gradient Boosting
— XGBoost (XGB)
— Arbre de décision (Decision Tree)

Pour chaque modèle, nous divisons notre jeu de données en données d'entraînement et données de test qui servira à évaluer le modèle à l'aide de train_test_split de Sckit-learn.

On remarque qu'il y a un déséquilibre au niveau de la répartition de nos variables cibles.
Pour rééquilibrer notre ensemble de données, on utilise la méthode d'oversampling SMOTE
A la suite de cela, on effectue une validation croisée à l'aide de StratifiedKFold et on calcule la précision moyenne sur les plis.
On obtient les résultats suivants pour chaque modèle :

Methode	precision	Rappel	f1_score	Exactitude	Oversampling
LR	0.000000	0.000000	0.000000	0.673315	None
LR	0.326685	1.000000	0.492484	0.326685	SMOTE()
RandomForest	0.815217	0.746516	0.779356	0.861912	None
RandomForest	0.801527	0.766423	0.783582	0.861695	SMOTE()
GradientBoosting	0.854149	0.676178	0.754815	0.856493	None
GradientBoosting	0.828006	0.721964	0.771358	0.860178	SMOTE()
XGB	0.849811	0.747180	0.795198	0.874268	None
XGB	0.847345	0.762442	0.802655	0.877520	SMOTE()
DecisionTree	0.760678	0.744526	0.752515	0.840017	None
DecisionTree	0.750000	0.752488	0.751242	0.837199	SMOTE()

FIGURE 2 – Resultats métriques

Modele	Crossvalidation stratifié
LR	0.6747316940690535
RandomForest	0.8719999864630827
GradientBoosting	0.8600982864800883
XGB	0.8820167976222404
DecisionTree	0.8489749169383216

FIGURE 3 – Crossvalidation stratifié

XGB semble être le meilleur modèle avec la validation croisée stratifiée. Les performances de Random Forest sont assez bonnes, tant avec ou sans sur-échantillonnage (SMOTE). L'exactitude et le rappel sont élevés, faisant de Random Forest un bon candidat. De plus, le modèle XGB affiche aussi une performance élevée en termes d'exactitude et de rappel. les performances sont assez similaires avec ou sans sur-échantillonnage et aussi avec la validation croisée stratifiée.

On retient donc qu'en termes d'exactitude et du rappel sur l'ensemble de test, Random Forest et XGBoost sont les meilleurs modèles.

4.2 Classification multi-label

On s'intéresse à présent à l'endroit où se trouve le défaut de l'arbre. Nous allons considérer le problème de classification multi-label comme quatre problèmes de classification uni-label et on effectue la même tache que précédemment. Par la suite, on calcule la moyenne des résultats des métriques obtenus lors de chaque classification unilabel de Tronc, Collet, Houppier et Racine

Les résultats sont récensés dans le tableau suivant :

Methode	precision	Rappel	f1_score	Exactitude	Oversampling
LR	0.0	0.00000	0.00000	0.8847728	None
LR	0.08515925	0.64100350	0.145745525	0.41973775	SMOTE()
RandomForest	0.63037451	0.3977781575	0.476444	0.91746125	None
RandomForest	0.5869905	0.44065675	0.49858175	0.914318	SMOTE()
GradientBoosting	0.68093725	0.300352	0.3387425	0.913451	None
GradientBoosting	0.45739075	0.49874849	0.48008025	0.89448299	SMOTE()
XGB	0.6567255	0.4287875	0.47569474	0.9220677	None
XGB	0.5785312499	0.471372	0.5106745	0.91616075	SMOTE()
DecisionTree	0.474796	0.481547	0.4773065	0.895838	None
DecisionTree	0.44366545	0.482801	0.46227324	0.88922625	SMOTE()

FIGURE 4 – Resultats métriques

Modele	Crossvalidation stratifié
LR	0.885756
RandomForest	0.920325
GradientBoosting	0.9158215
XGB	0.925615
DecisionTree	0.89895975

FIGURE 5 – Crossvalidation stratifié

Random Forest affiche de bonnes performances en termes d'exactitude et de rappel, avec ou sans sur-échantillonnage (SMOTE).
XGB est identifié comme le meilleur modèle lors de la validation croisée stratifiée avec une précision moyenne de 0.9256. Ses performances en termes d'exactitude et de rappel sur l'ensemble de test sont également élevées.
On conclut donc que XGB est probablement le meilleur modèle, mais Random Forest est reste également une bonne alternative.

5 Conclusion

Dans notre travail, nous avons répondu au défi 1 en explorant les données fournies. Plusieurs algorithmes de classification que nous avons jugées pertinentes ont par la suite été expérimentés dans le cadre des deux tâches du défi 1. Les performances ont été évaluées par crossvalidation stratifié. Cela nous a permis de sélectionner les meilleurs modèles uni-label et multi-label qui sont XGBoost et Random Forest.

Les résultats de notre analyse offrent aux botanistes une perspective détaillée sur les facteurs associés aux défauts des arbres, les aidant ainsi à prendre des décisions éclairées en matière de soins et de gestion. Pour les décideurs urbains, ces résultats fournissent une base solide pour la planification stratégique des espaces verts et la réduction des risques pour la sécurité publique. En contribuant à une gestion plus précise des arbres en milieu urbain, notre travail peut jouer un rôle essentiel dans la création d'environnements urbains durables et sécurisés.