Projet : Classification avec Arbre de Décision et Forêt Aléatoire (version provisoire)

SAE Algorithmiques et Programmation 3

November 11, 2024

Contents

1 Introduction

Dans ce ce projet, vous êtes amener à proposer une implémentation de deux algorithmes de classification en intelligence artificielle, à savoir arbre de décision et forêt aléatoire. Vous allez coder l'algorithme CART et RF, mais aussi construire une interface graphique, permettant de charger une donnée et de choisir l'algorithme à exécuter et montrer le résultat en fonction de l'exemple choisi.

2 Classification: CART et RF

2.1 Algorithme CART

L'algorithme CART (Classification and Regression Trees) est une méthode de classification basée sur des arbres de décision binaires. Chaque nœud effectue une division selon la mesure d'impureté Gini ou l'entropie, permettant de séparer les classes. Vous pouvez trouver plus d'explications sur le fonctionnement de cet algorithme sur wikipédia.

L'algorithme CART (Classification and Regression Trees) est une méthode d'apprentissage supervisé dédié aux tâches de classification et de régression. Créé par Breiman en 1984, cet algorithme construit des arbres de décision en segmentant les données en sous-ensembles de plus en plus homogènes. À chaque étape de l'arbre, une variable et un seuil sont choisis pour diviser les données, ce qui permet de minimiser l'erreur d'estimation.

Pour la classification, l'algorithme classe les données dans des catégories spécifiques en formant des "feuilles" qui représentent les classes cibles. Dans le cas de la régression, il prévoit une valeur continue pour chaque feuille. CART utilise des mesures comme le Gini (pour la classification) ou l'erreur quadratique (pour la régression) afin de choisir les meilleures divisions.

Cet algorithme est populaire pour sa simplicité et sa capacité à produire des modèles explicables, mais il peut être sensible au surapprentissage si l'arbre est trop profond.

Pour illustrer l'utilisation de l'algorithme CART avec un exemple de données sur les prêts bancaires, prenons un petit ensemble de données hypothétique. L'objectif ici est de déterminer si un prêt bancaire sera approuvé ou non, en fonction de certains critères. Voici un exemple de données et l'arbre de décision associé :

À partir de cet ensemble de données, l'algorithme CART va construire un arbre de décision. Voici à quoi pourrait ressembler un arbre simplifié basé sur ces données :

ID	**Revenu (€)** Mensuel	**Montant** du Prêt (€)	**Durée de l'Emploi** (années)	**Historique** de Crédit	**Prêt** Approuvé
1	3,500	10,000	5	Bon	Oui
2	2,000	5,000	3	Mauvais	Non
3	4,000	20,000	10	Bon	Oui
4	1,500	2,000	2	Mauvais	Non
5	3,200	7,000	7	Bon	Oui
6	1,800	3,000	1	Mauvais	Non
7	3,500	12,000	8	Bon	Oui
8	2,500	6,000	4	Mauvais	Non

Table 1: Exemple de données pour l'approbation de prêt bancaire

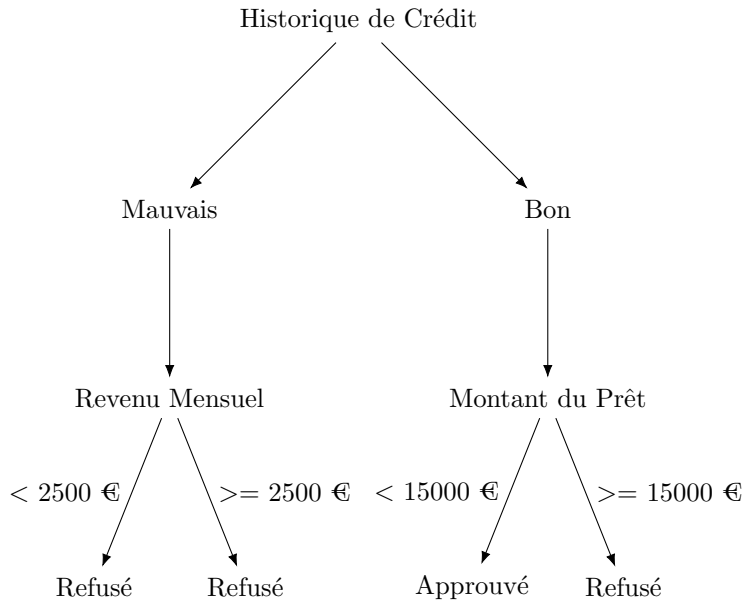

Figure 1: Arbre de décision pour l'approbation de prêt bancaire

2.2 Algorithme Random Forest

Le Random Forest (RF), ou forêt aléatoire, est un algorithme d'apprentissage supervisé qui combine plusieurs arbres de décision pour améliorer la précision des prédictions et réduire le risque de sur-apprentissage (ou overfitting). Développé par Leo Breiman, cet algorithme repose sur le principe de l'ensemble learning : il utilise un ensemble de modèles faibles (arbres de décision) et agrège leurs prédictions pour obtenir un modèle global plus robuste et précis.

Dans une forêt aléatoire, chaque arbre est construit en sélectionnant un

sous-ensemble aléatoire des données d'entraînement et des caractéristiques. Lors de la prédiction, chaque arbre de la forêt vote pour une classe (pour les problèmes de classification) ou prédit une valeur numérique (pour les problèmes de régression), et le résultat final est obtenu par la majorité des votes ou la moyenne des prédictions des arbres.

Le Random Forest est largement utilisé en raison de sa précision, de sa capacité à gérer de grandes quantités de données et de caractéristiques, ainsi que de sa résilience aux valeurs aberrantes et aux données manquantes. C'est un choix courant pour les tâches de classification et de régression dans divers domaines comme la finance, la biologie, et le marketing.

3 Création de l'Interface Graphique (GUI)

L'interface qui vous est demandé doit permettre de charger un fichier contenant la donnée au format CSV. Si le fichier est mal formé, vous devez afficher un message d'erreur. Votre interface doit permettre aussi de faire un choix entre CART et RF à lancer. Une fois le modèle construit vous devez permettre à l'utilisateur de donner un exemple et de pouvoir dire si le modèle prédit correctement la classe. Un message doit apparaître pour dire si la classe est correct ou pas.

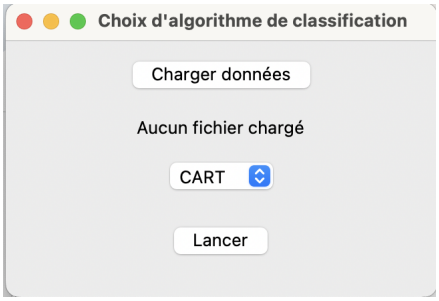

Figure 2: Exemple très simplifiée de l'application graphique demandée

4 Votre travail

Votre travail consiste à

1. Comprendre le fonctionnement de CART et RF
2. Proposer votre prpore implémentation pour chacun deux algorithmes

3. Proposer une application graphique permettant de charger la donnée et de choisir l'algorithme à exécuter parmi les deux algorithmes.

4. Ajouter des fonctionalités comme le temps, des mesures comme la précision, etc.. (optionnel)

Le projet est à rendre le 20 décembre 23h59 dernier délai.