Arbres de décision

Arbres de décision

Algorithmes ID3 et CART

Random Forests

Objectif du chapitre

- Comprendre les arbres de décisions
- Construire un arbre de décision
- Connaitre les différents algorithmes d’arbres de décision

Plan

- Arbres de décisions
- Algorithme ID3
- Algorithme CART

Algorithme CART

Classification and regression trees

Introduction

- L'algorithme CART (Classification and Regression Trees) est une méthode d'apprentissage supervisé qui permet de construire des arbres de décision pour la classification et la régression. Il repose sur un principe récursif de division de l’espace des caractéristiques afin de minimiser un critère d'impureté (pour la classification) ou une mesure d’erreur (pour la régression).
- Les arbres CART sont des arbres binaires, c'est-à-dire que chaque nœud interne a exactement deux enfants. L'algorithme sélectionne les divisions en fonction d'un critère de séparation optimal.

Arbre de Classification et de Régression

- Implémentation alternative des arbres de décision
- Il divise les nœuds de façon binaire (binary split)
- CART peut être utiliser pour la classification mais aussi la régression
- CART cherche tous les attributs et tous les seuils pour trouver celui qui donne la meilleure homogénéité (pureté) du découpage.

Arbre de Classification et de Régression

- Quand un nœud interne S est coupé sur l'attribut j avec un seuil a_j, il donne naissance à deux nœuds descendants:
  - Sous-nœud gauche S_g $\left(p_g \approx \frac{|S_g|}{|S|}\right)$ qui contient tous les éléments qui ont les valeurs de l'attributs $v_j < a_j$
  - Sous-nœud droit S_d $\left(p_d \approx \frac{|S_d|}{|S|}\right)$ qui contient tous les éléments qui ont les valeurs de l'attributs $v_j > a_j$

Arbre de Classification et de Régression

- Soit $I(S)$ une fonction qui mesure l'impureté de S par rapport à la classe cible.
- CART étudie le changement de l'impureté par rapport au seuil et pour tous les attributs:
- $E\left[I\left(S_{gd}\right)\right] = p_g I\left(S_g\right) + p_d I(S_d)$
- $\Delta I(S) = I(S) - E\left[I\left(S_{gd}\right)\right] = I(S) - p_g I\left(S_g\right) - p_d I(S_d)$
- Donc CART cherche l'attribut et le seuil qui maximise la décroissance de l'impureté du nœud par rapport à la cible
- Plus un couple (attribut / seuil) maximise la décroissance de $I(S)$, plus de chance il a d'être choisi

Arbre de Classification et de Régression

Le problème d'optimisation est le suivant

$$\arg\max_{j;a_j} \Delta\, I(S)$$

CART pour classification

En **classification**, la mesure de l'impureté utilisée est l'**index (ou impureté) de Gini**

Il représente la vraisemblance qu'un élément du nœud soit incorrectement étiquette par un tirage aléatoire qui respecte la loi statistique de la cible estimée dans le nœud.

L'index de Gini $I_G(S)$ pour un nœud S est calculé comme suit:

- Partitionner S sur les valeurs de la cible en n groupes : $C_1, \dots, C_n$
- Calculer p_i : probabilité estimée qu'un élément de S se retrouve dans C_i $\left(p_i \approx \frac{|C_i|}{|S|}\right)$
- $I_G(S) = \sum_{i=1}^{m} p_i(1-p_i) = \sum_{i=1}^{m}\left(p_i - p_i^2\right) = 1 - \sum_{i=1}^{m} p_i^2$
- $I_G(S) = 0$ si S est homogène (pure) (tous les éléments sont de la même classe)

CART pour Régression

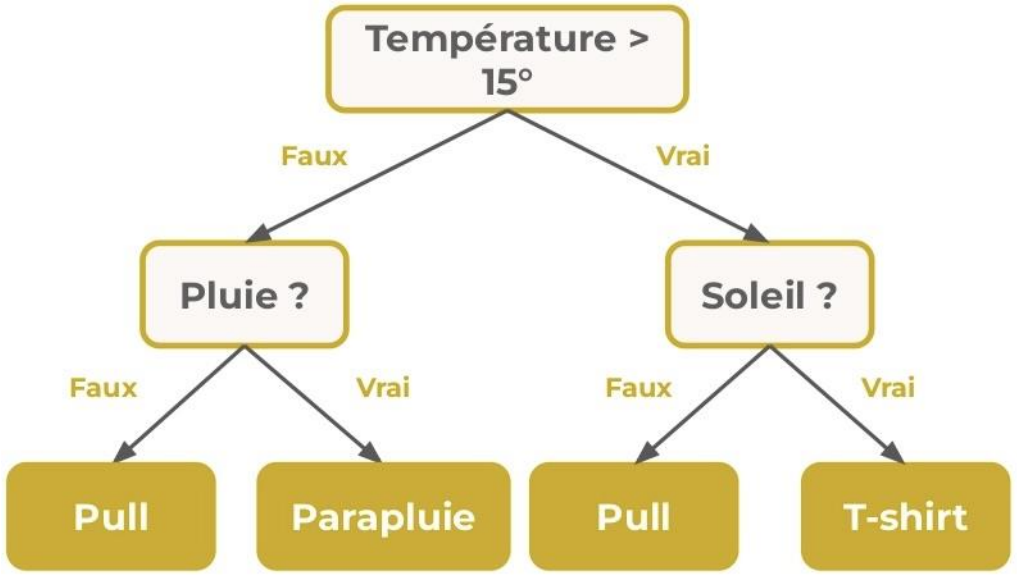
En **Régression,** CART essaye de minimiser la variance moyenne des groupes

Ainsi le critère utilisé est l'erreur quadratique moyenne

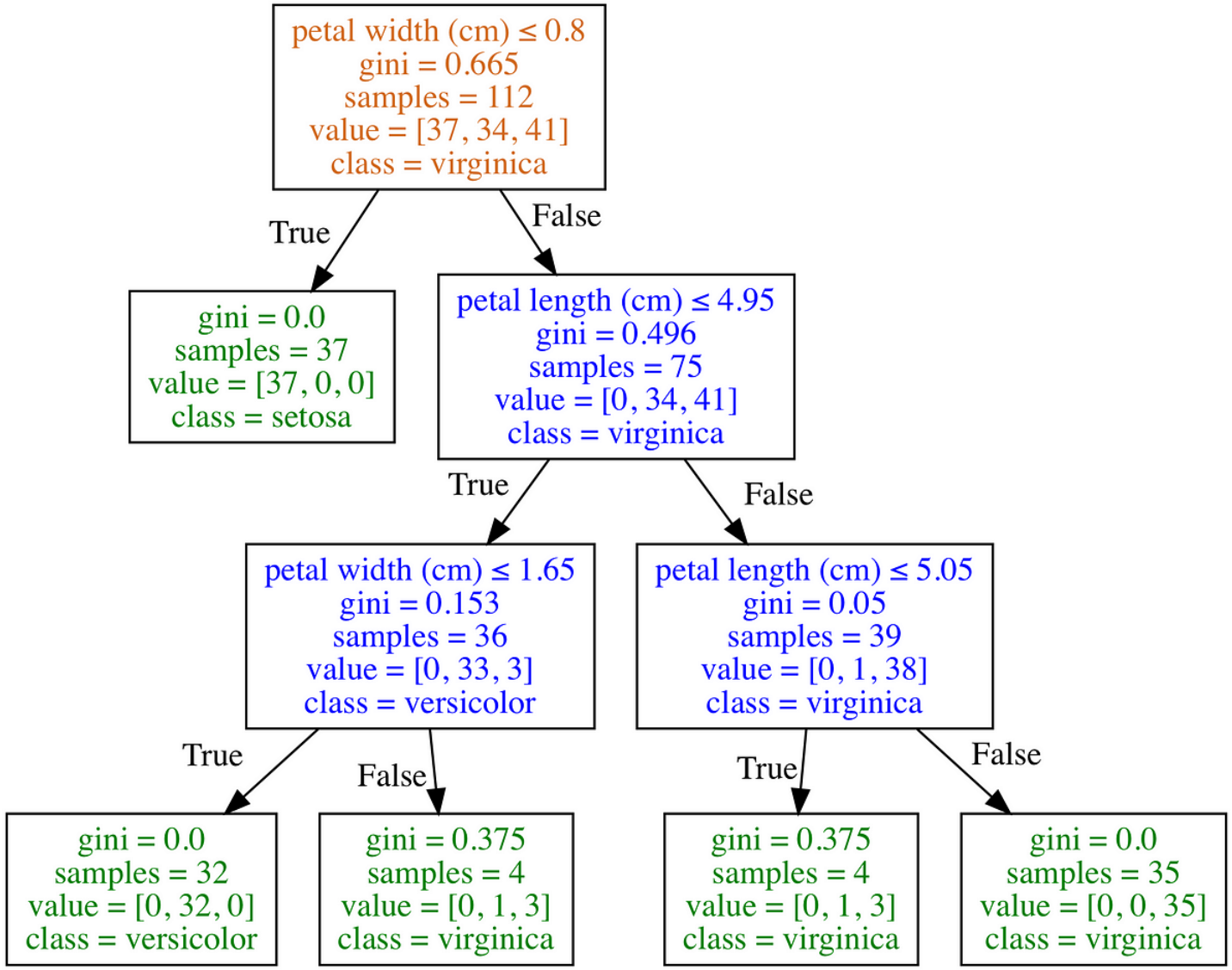
Le problème d'optimisation est le suivant

$$\arg\min_{j;a_j} p_g Var(S_g) + p_d Var(S_d)$$

Arbre de Classification et de Régression



Arbre de Classification et de Régression



Arbre de Classification et de Régression

Classification de nouvelles données :

- Parcours de l'arbre pour arriver dans une feuille.
- Pour la classification : on considère la classe dominante (majoritaire) dans la feuille.
- Pour la régression : on considère les valeurs dominantes dans la feuilles.

Avantages CART :

- Forme non paramétrique.
- Pas de sélection de variables nécessaire.
- Invariable aux transformation monotones des attributs.
- Bonne gestion des outliers.

Fonctionnement de l'algorithme

- Afin de construire l'arbre il faut tester les seuils pour chaque variables (caractéristiques)
- Cependant, il faudra aussi itérer sur toutes les variables du dataset

- Donc on aura une boucle imbriqué:
  - Pour chaque variable dans le dataset:
    - Calculer les seuils
    - Pour chaque seuil:
      - Trouver la valeur de l'indice gini pour le coté droit du seuil
      - Trouver la valeur de l'indice gini pour le coté gauche du seuil
      - Calculer le gini pondéré

- Calculer la valeur du gini index pour tous les seuil d'une variable et pour toutes les variables dans le dataset
- Choisir le couple (Variable / Seuil) avec l'indice gini minimum
  - **Même chose pour la régression, à la différence qu'on utilise la variance (MSE)**