IHDCM037 - Machine Learning
Travaux pratiques
Séance 2 : Arbres de décision

Remarque : Pour réaliser les travaux pratiques de ce cours, nous travaillerons avec le langage `Python` sur des `Jupyter Notebook`. Ceux-ci peuvent être utilisés en ouvrant la plateforme `Anaconda Navigator` et en lançant `Jupyter Notebook`.

Première partie

La première partie de cette séance est dédiée à la découverte d'un deuxième modèle de Machine Learning : les arbres de décision. Le jeu de données utilisé pour cette séance est très connu pour effectuer une classification binaire. Il contient des informations sur des tumeurs et prédit si celles-ci sont bénignes ou malignes. Voici les différentes étapes à suivre :

1. Importez le *dataset* `breast_cancer` de la librairie `sklearn`.
   - Combien de tumeurs sont reprises dans ce *dataset* ? .....
   - Combien d'entre elles sont malignes et bénignes ? .....
   - Combien de *features* sont considérées pour les classifier comme bénignes ou malignes ?

     Citez également les 3 premières *features* : .....

     .....
2. Sauvegardez la partie *data* du *dataset* dans un premier tableau et la partie *target* dans un deuxième.
3. Formez un ensemble d'entraînement et un ensemble de test. Pour avoir les mêmes résultats, formez un ensemble d'entraînement qui contient 67% des données initiales et utilisez l'hyperparamètre `random_state = 42`.
4. Entraînez un arbre de décision sur l'ensemble d'entraînement. Utilisez l'hyperparamètre `max_leaf_nodes = 2`.
   - Quelle fonction utilisez-vous pour créer le type de modèle arbre de décision ?

     .....
   - Qu'induit l'hyperparamètre `max_leaf_nodes = 2` ? .....

     .....
5. Affichez l'arbre de décision généré.
   - Quelle fonction utilisez-vous pour afficher l'arbre de décision ? .....
   - Que fait l'arbre de décision ? Sur quel critère se base-t-il pour classifier les données ?

     .....

     .....

     .....
6. Évaluez la performance de cet arbre de décision.
   - Quelles sont les métriques utilisées pour calculer la performance de cet arbre de décision ?

     Que calculent-elles ? .....

.....

.....

.....

.....

- Grâce à l'*accuracy* et à l'affichage de l'arbre, que pensez-vous de ce premier arbre de décision ? Quel concept de Machine Learning reconnaissez-vous ?

  .....

  .....

  .....

Deuxième partie

Pour la deuxième partie de cette séance, générez un deuxième arbre de décision avec, cette fois, l'hyperparamètre `max_leaf_nodes = 30`. Comme pour la première partie, affichez l'arbre de décision et évaluez la performance de cet arbre de décision.

- Quel est l'impact du changement de la valeur de l'hyperparamètre `max_leaf_nodes` ? .....

  .....

- Grâce à l'*accuracy* et à l'affichage de l'arbre, que pensez-vous de ce deuxième arbre de décision ? Quel concept de Machine Learning reconnaissez-vous ?

  .....

  .....

  .....

Troisième partie

Le but de la troisième partie de cette séance est de faire un choix concernant la valeur de l'hyperparamètre `max_leaf_nodes`. En effet, dans la première et deuxième partie, vous avez observé que la valeur des paramètres ont un impact significatif sur la performance d'un modèle. Il est donc crucial de trouver les meilleures valeurs possibles pour les hyperparamètres d'un modèle afin que celui-ci produise des résultats compétitifs.

Générez des arbres de décision avec l'hyperparamètres `max_leaf_nodes` prenant des valeurs entre 2 et 30 (29 arbres de décisions seront donc générés et entraînés). Pour chaque arbre, calculez *l'accuracy* sur l'ensemble d'entraînement et de test, puis affichez ces valeurs sur un même graphique. Ce graphique doit représenter les *accuracy* en fonction de la valeur de l'hyperparamètre `max_leaf_nodes`.

- Pour choisir la valeur de l'hyperparamètre, l'ensemble d'entraînement et de test sont-ils

  suffisants ? Pourquoi ? .....

  .....

- Est-ce que ce graphe correspond à la théorie ? Pourquoi ? .....

  .....

- Pouvez-vous repérer les cas *d'underfitting* et *d'overfitting* dans le graphique ?

  .....

  .....

- Quelle valeur choisissez-vous pour l'hyperparamètre `max_leaf_nodes` ? Pourquoi ?

  .....

  .....