Jeu de données "Mushroom"

Mushroom est un ensemble de données sur les champignons. Il est constitué de 8124 exemples pour lesquels diverses descriptions sont disponibles comme la surface, l'odeur, la couleur, etc, ainsi que l'information : cosmétible ou vénéneux. Cette dernière information conduit naturellement à se poser la question : peut-on inférer, à partir des exemples, un modèle prédictif capable de différencier les champignons comestibles des non-cosmestibles.

Enterprise Miner met à disposition diverses méthodes de classification. Evidemment, toutes ces méthodes ne sont pas équivalentes et il est nécessaire de choisir la ou les méthodes à lancer sur les données en fonction de leur nature. Par exemple, les données du jeu Mushroom sont telles que la méthode des arbres de décision est plus adaptée et fourni de meilleurs résultats que la méthode de régression logistique. Les deux objectifs de ce TP sont :

- comprendre quelles sont les particularités de ces données qui privilégient la méthode des arbres de décision,
- étudier les différents paramètres de la méthode et déterminer leur impact sur les performances de l'algorithme.

Etape 1 : Récupération des données.

La disquette qui vous sera distribuée contient deux fichiers : *agaricus-lepiota.data* et *agaricus-lepiota.names*. Le premier fichier (.data) contient les données du jeu Mushroom, le second un historique et une description de ces données. En particulier ce fichier vous renseigne sur ce que représente chacun des attributs ainsi que l'ensemble des valeurs qu'ils peuvent prendre.

Importez le fichier *agaricus-lepiota.data* à partir de la base SAS et mettez le fichier de données dans la librairie que vous avez créé au TP précédent. Pour l'importation du fichier, suivez la procédure d'importation de données disponible à partir du menu "Fichier" de la barre d'outils, vérifiez bien les options proposées ainsi que le format du fichier. Une fois les données importées dans votre librairie, vous avez la possibilité de les visualiser sans passer par Enterprise Miner. Vérifiez par exemple le nombre de données et le nombre de variables par donnée récupérée.

Créez un nouveau projet dans Enterprise Miner que vous appelerez *Mushroom*. Sur un diagramme de processus vide, importez vos données comme vous l'avez fait aux TPs précédents et définissez la variable cible (la première variable).

Etape 2 : Analyse et traitement des données.

Par l'intermédiaire du noeud *Insight*, étudiez la distribution des attributs, en particulier des attributs 12 et 17. Que remarquez-vous ? Quelle solution a adopté Enterprise Miner pour l'attribut 17 ? Pourquoi ? Quelle solution devez-vous adopter pour l'attribut 12 ?

Avec Enterprise Miner, vous devez vous assurer que les valeurs manquantes ont bien été détectées. En effet, si le caractère mis pour indiqué une valeur manquante dans la donnée n'est pas celui attendu ('.'), il peut alors être traité comme une valeur d'attribut comme c'est le cas pour Mushroom.

Une fois le problème des valeurs manquantes réglé, vous passerez directement à la partition des données, gardez les valeurs par défaut du logiciel. Il n'est pas indispensable de créer, comme pour les TP précédents, une matrice de coût. En effet, le critère d'optimisation par défaut des méthodes est le taux d'erreur sur les données. C'est le critère que vous étudierez.

Etape 3 : Arbres de décision.

Lancez une première fois la méthode des arbres de décision sur les données sans modifier aucun paramètre ni du partitionnement ni de la méthode et observez les résultats obtenus. En analysant soigneusement les arbres de décision proposés (tailles 2 à 6), faites une première synthèse des attributs sélectionnés par la méthode comme étant les plus pertinents (les plus corrélés à la cible) pour la tâche de prédiction. Comparez ces résultats avec les règles de décision données dans le fichier *agaricus-lepiota.names*. Que constatez-vous ?

Etape 4 : Impact du partitionnement des données.

Modifiez à plusieurs reprises les taux de données d'apprentissage, de validation et de test de telle manière à tester des taux de données d'apprentissage et de validation très bas comme très haut. Que se passe-t-il lorsque le nombre de données d'apprentissage est très bas ?

Etape 5 : Critère de sélection de variables.

Remettez les paramètres par défaut pour le partitionnement des données (40-30-30) et tester les trois critères de sélection de variable pour la construction de l'arbre (Gini, Entropie, Chi-square test). Pour cela, supprimez le noeud *Tree* déjà présent et insérez trois nouveaux noeuds du même type en paramétrant chacun avec une méthode de sélection différente. Comparez les résultats sur les données test en utilisant le noeud *Assessment*. Qu'observez-vous ?

Vérifiez également qu'une différence est visible entre les performances de l'algorithme sur les données d'apprentissage et les données test.

Etape 6 : Réglage des paramètres de la méthode et élagage.

Les paramètres par défaut de la méthode des arbres de décision peuvent être incohérents par rapport à vos besoins spécifiques sur un jeu de données. En utilisant le noeud *Tree* qui vous a fournit les meilleurs résultats, modifiez à plusieurs reprises le nombre de données minimal par noeud, le nombre de données minimal pour un essai de construction d'un nouveau test, le nombre maximal de branches pour un noeud, ainsi que la profondeur maximale de l'arbre. Qu'observez-vous ?

Etape 7 : Réglages optimaux.

Une fois votre architecture en place et en considérant les constats faits sur les différents paramètres et leurs impacts sur les performances des arbres de décision, proposez une instance des paramètres qui semble être optimale. Supprimez les autres noeuds du diagramme.

Etape 8 : Comparaison avec la méthode de régression.

Lancez la méthode de régression logistique pas à pas avec les paramètres par défaut de la méthode et comparez les résultats avec ceux de la méthode des arbres de régression. Qu'en concluez-vous ? Pourquoi ?

Etape 9 : Réflexion sur la tâche d'apprentissage.

Jusqu'ici, vous avez configuré la méthode pour faire le moins d'erreurs possible. En regardant bien votre arbre optimal, observez que quelques champignons vénéneux seront classés comme cosmétibles par l'arbre. Si on prend en compte le fait que se tromper en prédisant qu'un champignon est comestible alors qu'il est vénéneux n'a pas du tout le même impact que prédire qu'un champignon cosmestible est vénéneux, proposez divers moyens de modifier l'objectif de la méthode pour tenter de ne plus faire d'erreurs sur les champignons vénéneux, au risque de jeter un peu plus de champignons comestibles...