

Nous utilisons Python (package « scikit-learn »).

Tutoriel sur le schéma « apprentissage – test » pour la construction et l'évaluation des classifieurs :

<http://tutoriels-data-mining.blogspot.com/2015/09/python-machine-learning-avec-scikit.html>

Outils « scikit-learn » :

- pour les arbres de décision ([DecisionTreeClassifier](#)),
- pour l'analyse discriminante ([LinearDiscriminantAnalysis](#)).

## 1. Données

Nous utilisons les données « [segmentation.xlsx](#) ».

Il s'agit de classer automatiquement des images à partir de leur description (<http://archive.ics.uci.edu/ml/datasets/image+segmentation>). La variable cible est « classe », les autres sont les explicatives potentielles.

## 2. Comparaison de méthodes

**A rendre :** un rapport commenté PDF issu d'un projet Notebook JUPYTER.

L'objectif est de déterminer expérimentalement lequel de l'arbre de décision et de l'analyse discriminante linéaire est le plus performant sur ce jeu de données.

Voici quelques repères importants pour votre étude :

1. Scindez les données en échantillon d'apprentissage et de test pour obtenir une vision objective des performances de vos modèles (ex. 70% vs. 30%).
2. Réalisez une première confrontation avec les paramètres par défaut des différents algorithmes (Arbre vs. Analyse discriminante).
3. Essayez ensuite de jouer sur les caractéristiques des approches pour améliorer leurs performances (ex. sélection de variables pour l'analyse discriminante, paramètres pesant sur la profondeur pour l'arbre).
4. Faites un bilan global de votre travail avec un tableau récapitulatif.