# La Prédiction de Notes Attribués étant donné les Attributs d'une Section de Cours

J'avais commencé avec des données sur les notes issues de l'Université de Wisconsin à Madison entre le printemps de 2007 et celui de 2018. Pour chaque section de cours, j'avais aussi accès à plusieurs autres attributs, comme l'horaire, la salle de classe, le format du cours, l'enseignant, et le numéro du cours. Mon but était de prédire la distribution des notes pour une classe dans le futur en utilisant seulement les données que l'on pourrait connaître au jour d'inscription pour le semestre.

Dès le début, il était clair que j'avais des variables indépendantes qu'on ne connait le jour de l'inscription. Par exemple, les emplois de temps des enseignants sont souvent établis au dernier moment. De plus, comme le temps de l'inscription peut durer jusqu'à la troisième semaine du semestre, le nombre d'inscriptions est inconnu jusqu'à ce date. D'autres variables étaient catégorielles avec tellement de valeurs possibles qu'elles étaient presque inutilisables.

Après réflexion, j'ai sélectionné les attributs suivants pour utiliser dans mes modèles:
- `section_type` : le format du cours en considération (cours magistral, travaux dirigé, etc.)
- `start_time` : l'heure auquel commence une séance de cours (en minutes après minuit)
- `class_length` : la durée d'une séance de cours (en minutes)
- `term_code` : un code de quatre chiffres qui indique l'année et le semestre du cours
- `class_level` : cela depend de niveau de connaissances et de nombres des cours au même sujet (ou des sujets proches) qu'un étudiant est censé completer avant d'inscrire dans ce cours. Les cours de niveau 0 sont pour les étudiants qui manque même quelques connaissances de bac tandis que les cours de niveau 9 sont pour les doctorants.
- `class_meetings` : le nombre de fois par semaine qu'un cours avait lieu (de 1 à 6)
- `median_enrollment` : le médiane des inscriptions de toutes les classes qui se réunissaient dans la même salle
- `number_of_sections` : le nombre de sections de ce même cours données dans le même semestre.

J'ai décidé d'éliminer quelques attributs : `end_time` (l'heure de la fin du cours), `mon`, `tues`, `wed`, `thurs`, `fri`, `sat`, `sun` (si une séance de cours a lieu ou pas ce jour), `subject_code` (le sujet du cours), `max_enrollment`, `min_enrollment`, `third_quartile` (des descripteurs des inscriptions de la salle de classe), et `section_number` (le numéro de section s'il y en a plusieurs dans ce même cours).

Ensuite j'ai estimé une variété de types de modèles différents afin de prédire le pourcentage d'étudiants ayant obtenu la note « A », la note « AB », etc. Pour chaque type de modèle, j'ai d'abord réglé les paramètres en utilisant les premiers 16 semestres de données pour estimer les modèles, puis les 3 semestres suivants pour en choisir les meilleurs. Après j'ai trouvé les meilleurs paramètres pour chaque type de modèle, j'ai estimé chaque modèle en utilisant les premiers 19 semestres de données. Cela m'a laissé 3 semestres de données pour faire la comparaison entre les modèles de types différents.

J'ai créé des pipelines pour les modèles suivants:
- Un modèle de regression linéaire (modèle de base)
- Deux modèles de k plus proches voisins différents:
  - Avec 16 plus proches voisins
  - Avec 7 plus proches voisins
- Un modèle de régression de type forêt d'arbres décisionnels

- Un modèle de régression de type AdaBoost
- Un modèle de régression de type machine à vecteurs de support
- Un modèle de régression de type elastic net
- Un modèle de régression de type perceptron multicouche

J'ai ensuite évalué les performances des modèles sur les données de trois derniers semestres en employant les indicateurs suivants:
- L'erreur quadratique moyenne (MSE)
- L'erreur absolue moyenne (MAE)
- L'erreur absolue médiane (Med AE)
- Le coefficient de détermination (R2)

J'ai obtenu les résultats suivants:

| | MSE | MAE | Med AE | R2 |
|---|---|---|---|---|
| **Machine à vecteurs de support** | 0,0175 | 0,0749 | 0,0254 | 0,0279 |
| **Régression linéaire (base)** | 0,0170 | 0,0794 | 0,0370 | 0,1166 |
| **ElasticNet** | 0,0168 | 0,0788 | 0,0361 | 0,1210 |
| **AdaBoost** | 0,0164 | 0,0783 | 0,0372 | 0,1226 |
| **K plus proches voisins 7** | 0,0146 | 0,0695 | 0,0290 | 0,1427 |
| **K plus proches voisins 16** | 0,0141 | 0,0695 | 0,0300 | 0,1871 |
| **Perceptron multicouche** | 0,0138 | 0,0688 | 0,0296 | 0,2193 |
| **Forêt d'arbres décisionnels** | 0,0135 | 0,0694 | 0,0307 | 0,2281 |

On peut observer qu'aucun de ces modèles est spectaculaire, ce qui suggère qu'il y aurait d'autres éléments influençant des distributions de notes. Parmi les modèles que nous avions trouvés en utilisant nos données, les modèles *forêt d'arbres décisionnels* et *perceptron multicouche* sont similaires, mais le modèle d'arbres est le meilleure de peu.