seance_3

October 1, 2020

1 Séance 3 : Apprentissage supervisé

Pour cette dernière séance, on va utiliser toujours le même corpus pour faire cette fois de l'apprentissage supervisé. On s'interessera à de la classification binaire et multi-classe, mais pas à de la classification multi-label (un exemple n'aura qu'une seule classe associée).

Pour chaque approche, vous pouvez considérer comme représentation des articles soit la version brute tf-idf, soit la décomposition par ACP, soit celle par NMF. Attention cependant, si vous faites une ACP ou une NMF, celle-ci doit être apprise uniquement sur l'ensemble d'apprentissage, pas sur celui de test (puis ensuite utilisé sur l'ensemble test).

Vous pouvez reprendre le code de la séance précédente pour charger et traiter les données.

1.1 Classification binaire

Pour toute cette partie, choisir un couple de labels appartenant à la même catégorie générale (par exemple cs), et un couple de labels de différente catégorie. Chaque couple nous procure un problème de classification binaire. Consistuer vos 2 jeux de données en fonction des labels choisis.

(Préambule) **Exercice numpy** : dans un premier temps, coder une classe Perceptron avec une méthode fit(X,Y) qui permet d'apprendre le modèle avec une hinge-loss en mini-batche, et une méthode predict(X) pour scorer les données. Tester votre fonction.

Tester la performance de quelques algorithmes usuels de classification : SVM, forêt aléaoitre, k-plus proches voisins et votre perceptron. Tracer les courbes d'erreur en apprentissage et en test en fonction des hyperparamètres. Comparer les résultats entre tf-idf, ACP et NMF. En considérant une des deux classes positives et l'autre négative, on aimerait tracer une courbe ROC des classifieurs. Pour cela, nous avons besoin d'un score de positivité (ou d'une probabilité) plutôt que d'une décision binaire. Pour quel(s) algo(s) est-il possible d'obtenir un tel score ? Tracer les courbes ROCs pour ces algos.

2 Classification multi-classe

Comment peut-on faire de la classification multi-classe en utilisant des algorithmes de classification binaire ? Certains sont-ils adaptables directement ?

Refaire les mêmes expériences que ci-dessus mais dans une configuration multi-classe, en étudiant le problème pour 5 classes, puis 10 classes que vous choisirez et enfin toutes les classes d'une catégorie générique (type cs). Représenter la matrice de confusion.