esiea
ECOLE D'INGENIEURS
DU MONDE NUMERIQUE

5A : Machine Learning 1

TP – Support Vector Machines & Decision trees (Part 1)

Ce TP sera réalisé en Python après avoir installé Anaconda. Les instructions principales sont données dans les slides de cours. Vous pouvez aussi travailler en R si vous le souhaitez.

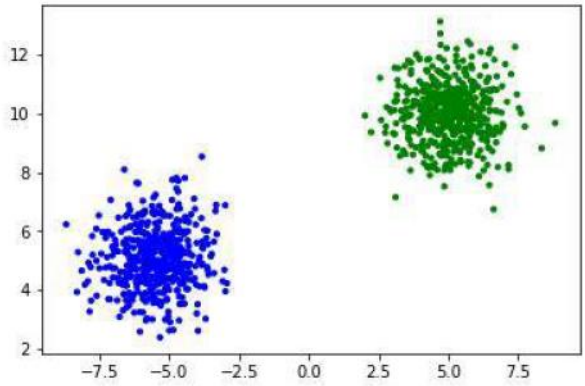
A. Génération des données

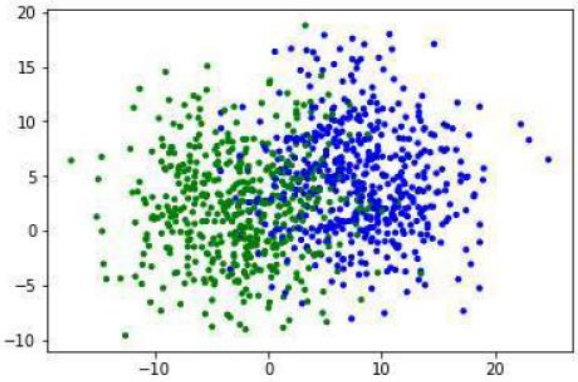
On utilisera la fonction suivante pour générer une première base de données synthétiques (B1) :

X,y = datasets.make_blobs(n_samples=1000, n_features=2, centers=2, cluster_std=1.0, center_box=(-10.0, 10.0))

Le parameter *std* permet de contrôler la dispersion des données (et la non linéarité des frontières) :

std = 1 *std = 5*





Les autres paramètres sont faciles à comprendre …

Normaliser (*standardize*) les données.

Remarque :
Ce n'est pas indispensable ici mais, pensez, si nécessaire, à diminuer la dimension des données à l'aide d'une PCA (en conservant assez de composantes pour que l'inertie cumulée soit « suffisante » : 90% par exemple).

B. SVM linéaire

1) Données linéairement séparables :

Utiliser d'abord l'ensemble de la base de données B1 pour entraîner le modèle puis l'évaluer.

a) Définir un SVM à marge stricte en fixant le paramètre de pénalisation C à une valeur élevée.

b) Calculer l'erreur d'apprentissage et la matrice de confusion. Afficher les frontières de décision et les analyser.

c) Répéter l'expérience une fois. Comparer les résultats des deux expériences. Conclure sur les qualités des SVM par rapport aux réseaux de neurones.

d) Conclure.

2) Données non linéairement séparables :

a) Générer une base de données B2 à l'aide de la fonction *make_blobs*, en augmentant la valeur du paramètre *std*.

b) Séparer cette base de données B2 en deux : apprentissage et test (échantillonnage stratifié sans remise[1]). Pourquoi ?

c) Relaxer les contraintes (SVM à marge souple) en faisant varier C sur une échelle logarithmique. Calculer les erreurs en apprentissage et en test pour chaque valeur de C. Rassembler les résultats dans un tableau.

d) Conclure :
- sur la capacité d'un SVM linéaire à séparer des données non linéairement séparables.
- sur les performances en test.

Remarque :
Jusqu'ici, vous observez l'influence du paramètre C. par la suite, il conviendra de l'optimiser sur une base de validation (voir ci-après).

[1] Les deux bases sont identiquement distribuées : le ratio entre les classes reste le même.