Université d'Avignon – Master 1
Intelligence Artificielle – Apprentissage supervisé – TP 3
Juan-Manuel Torres - 2025

Travaillez individuellement ou en binôme. Langages acceptés : C/C++ ; python ; awk, rust, perl ; ruby – Exclus : R, java, javascript
Interdit d'utiliser des fonctions/librairies des RN natives (R, python, matlab, etc)
IA interdite, IN à volonté

Théorie

L'algorithme d'apprentissage Minimerror utilise une fonction de coût qui dépend d'un hyper-paramètre appelé température T>0; qui peut être écrit comme **β**=1/T. Il a les propriétés de trouver un hyperplan séparateur de stabilité maximale si l'ensemble d'apprentissage est LS ou s'il ne l'est pas, de trouver le nombre minimum de fautes ayant des stabilités maximales. Utilisez la fonction de coût, sa dérivée et une stratégie de recuit déterministe pour programmer l'algorithme minimerror.

Pour ce TP, en particulier pour les paramètres (initialisation, delta de températures, delta du recuit, etc) consulter thèse de M *Torres et les slides Minimerror,* disponibles sur ENT.
Il s'agit d'un algorithme difficile : Il n'y a pas UNE seule implantation de cette algorithme : impléméntez votre propre version de l'implémentation

Travail pratique. Données au choix : ET, XOR, sonar, machine UCI...

PARTIE I

Minimerror avec 1 température β=1/T. Programmer l'algorithme minimerror pour apprendre sur l'ensemble « train », puis tester sur l'ensemble de « test ». Les données à tester NE DOIVENT PAS CONTENIR LEUR CLASSE.

a) Calculer les erreurs d'apprentissage Ea et de généralisation Eg ;
b) Afficher les N+1 poids **W** du perceptron ;
c) Calculer les stabilités des P exemples de « test » (distance a l'hyperplan séparateur avec les poids normés)
d) Graphique des stabilités

Note : Il est possible que Ea>0 en fonction des paramètres

PARTIE II

Minimerror avec 2 températures β+ / β- = rapport de températures. Modifier l'algorithme minimerror pour apprendre avec 2 températures sur l'ensemble « train », puis tester sur l'ensemble de « test ». Les données à tester NE DOIVENT PAS CONTENIR LEUR CLASSE.

a) Calculer les erreurs d'apprentissage Ea et de généralisation Eg ;
b) Stocker les N+1 poids **W** du perceptron dans un fichier;
c) Calculer les stabilités des P exemples de « test » (distance a l'hyperplan séparateur avec les poids normés)
d) Graphique des stabilités en fonction de **β = 1,2,...10**

Rapport. Pour chaque partie : Vos programmes, Ea, Eg, P stabilités, N+1 poids. Rendu PDF et codes sources compressés sur ENT