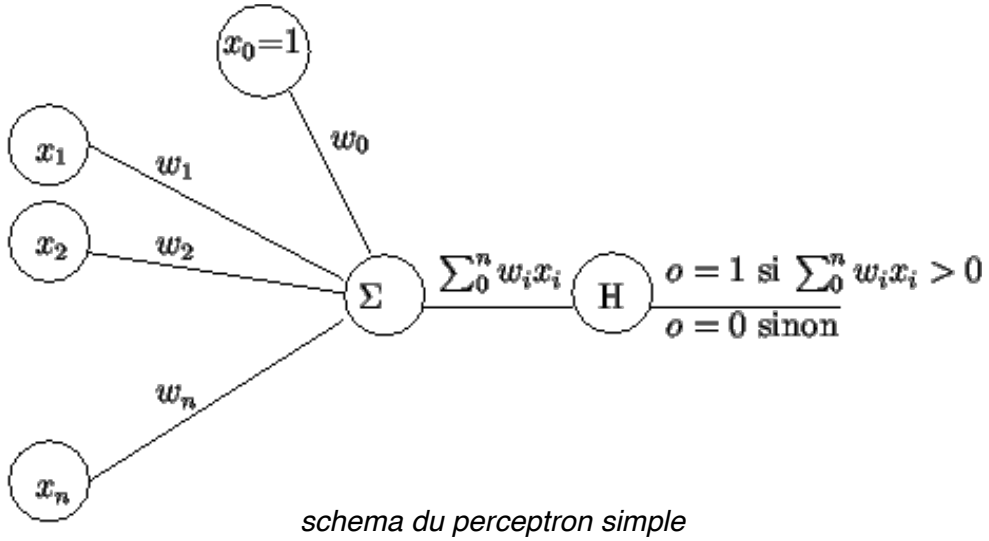
Rapport de TP sur le Perceptron

Introduction



schema du perceptron simple

Le perceptron peut être considéré comme le réseau de neurones le plus simple qu'on a vu cette année en IA. Il se situent parmi les algos d'apprentissage supervisé. Son rôle est de classifier un groupe d'échantillons.

Exemples de domaines d'utilisation : Reconnaissance de caractères, classification des données d'un pays ou des villes, classification d'animaux, etc...

Fonctionnement

Pour chaque entrée Xi, on calcule la sortie du réseau en appliquant la fonction d'activation. Si on ne tombe pas sur la sortie désirée on applique la règle d'apprentissage (détaillée ci-après), sinon en passe l'itération suivante.
À la fin de l'apprentissage le réseau devra avoir appris. Ainsi, si on lui présente une entrée, il doit être capable de décider à quelle classe appartient cette entrée.

Fonction d'activation

La fonction d'activation utilisée pour le perceptron est la fonction Heavyside ou la fonction signe qui est définit comme :

F(x) = -1 si $x < 0$
1 si $x \geq 0$

Cette fonction nous permet de discrétiser notre sortie (-1 ou 1).

Règle d'apprentissage

La règle d'apprentissage utilisée par le perceptron est la règle de Hebb.
« **Neurons that fire together wire together** »
Ce qui signifie qu'on va renforcer le lien unissant deux neurones s'il sont stimulés en même temps.

Wi (t+1) = Wi (t) + ε . (d – y) . Xi(version prenant en compte l'erreur locale)

où

ε correspond au pas d'apprentissage
d représente la sortie désirée et y la sortie actuelle
Wi (t) est le poids à l'instant t et Wi (t+1) le poids à l'instant d'après
Xi l'entrée courante

Avantages et Inconvénients

Inconvénients
Converge seulement pour des données linéaires
Pas très efficace dès qu'il y a trop de descripteurs

Avantages
Simple et efficace
Garantie d'apprendre un problème linéairement séparable

Resultats

Mauvaise solution

Dans ce TP, pour tester que le réseau a appris, j'ai d'abord fait l'apprentissage comme suit : Apprendre A 100 000 fois, puis Apprendre C 100 000 fois. Ceci est une solution qui marche pas, parce qu'à force de présenter la même lettre autant de fois, le réseau va finir par se spécialiser à la reconnaissance de celle-ci (au lieu de faire une généralisation).

Bonne solution

J'ai ensuite fait l'apprentissage comme suit :
Apprendre A une fois, puis Apprendre C une fois; et faire cela 100 000 fois. Après cet apprentissage, j'ai bien constaté que le réseau avait apprit à reconnaitre les deux lettres.

Extension Possible du TP

Une extension des fonctionnalités possible pour ce TP est la prise en charge des 26 lettres de l'alphabet.

Idée

On pourrait imaginer faire ça avec 26 neurones (au lieu d'un seul dans ce TP). Chaque neurone va donc se spécialiser à la reconnaissance d'un seul caractère. On peut dire par exemple que si c'est la lettre A qui est reconnue, le neurone qui code A va avoir 1 comme sortie et tous les autres neurones auront 0 comme sortie.

NB : Code source disponible en pièce jointe