

Perceptron simple

Exercice 1

Créez un nouveau projet Java sous Eclipse.

Exercice 2

Créez une nouvelle classe intitulée `Perceptron`. Cette classe a 3 attributs:

- une liste de `double` représentant les entrées (vous pouvez utiliser la classe `java.util.ArrayList` pour cela)
- une liste de `double` représentant les poids associés à chaque
- un `double` représentant la valeur de sortie du perceptron

Exercice 3

Écrivez des méthodes pour modifier la valeur des entrées et des poids et lire la valeur de sortie. Les signatures des méthodes doivent être:

- `public void setPoids(ArrayList<Double> p)`
- `public void setInputs(ArrayList<Double> i)`
- `public double getSortie()`

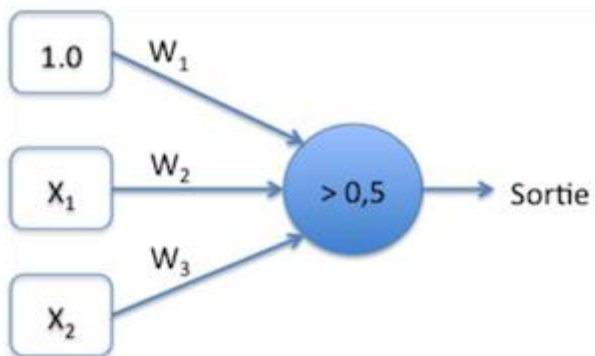
Exercice 4

Écrivez enfin une méthode `calcul` qui réalise la somme des produits `entrée * poids`. Si cette somme est supérieure à un seuil (disons `0.5`) la valeur de sortie est fixée à `1`, sinon elle vaut `0`.

Classe d'exécution

Exercice 5

Créez une classe avec une méthode exécutable `main`. Dans cette méthode, créez une nouvelle instance de perceptron disposant de 3 entrées tel que représenté ci-dessous.



La première entrée est une valeur constante qui servira à ajuster le seuil de la fonction d'activation. Dans votre méthode `main` demandez une saisie au clavier pour les deux autres entrées. Fixez les poids à des valeurs quelconques, exécutez la méthode `calcul` et affichez la valeur de sortie du perceptron pour vérifier son bon fonctionnement.

Apprentissage

Exercice 6

Créez une nouvelle méthode `train(double expected)` qui implémentera une étape d'apprentissage supervisé sur le perceptron. Cette méthode doit se comporter comme suit:

1. Calculer la sortie du perceptron
2. si la sortie est différente de celle attendue (la valeur `expected` passée en paramètre), modifiez le poids de chaque entrée tel que $w_{i+1} = w_i + \text{learningRate} * \text{erreur} * x_i$. Le paramètre `learningRate` définit la vitesse de l'apprentissage. Nous le fixerons à 0.1. L'erreur est la différence entre la valeur attendue et la valeur réelle de sortie.

Exercice 7

Modifiez votre méthode `main` pour entraîner le perceptron à effectuer l'opération NOT AND. L'échantillon d'apprentissage est :

Entrées	Sortie attendue
(1,1)	0
(0,1)	1
(1,0)	1
(0,0)	1

L'entraînement du perceptron nécessite une boucle lors de laquelle chaque itération appelle la méthode `train` avec chaque exemple de l'échantillon. L'entraînement se termine quand le perceptron renvoie la valeur attendue pour tous les exemples de l'échantillon.