

TP : Le perceptron

1 Exercice 1

Tout d'abord écrivez un programme qui génère des points aléatoires dans $[0, 1]^2$. Pour chaque point, le tagger 1 si $x_1 + x_2 - 1 > 0$ et -1 sinon. Vous stockerez ces points dans deux fichiers : un pour l'ensemble d'apprentissage, et un pour l'ensemble de validation.

Utilisez le format suivant :
 x_1 x_2 étiquette.

Par exemple :

```
0.840188 0.394383 1
0.798440 0.911647 1
0.335223 0.768230 1
0.553970 0.477397 1
0.364784 0.513401 -1
0.141603 0.606969 -1
0.242887 0.137232 -1
0.108809 0.998925 1
0.764871 0.699075 1
...
```

Vous pouvez utiliser le script python suivant <http://www-lisic.univ-littoral.fr/~teytaud/files/Cours/Apprentissage/plot.txt> (à renommer en .py) pour visualiser vos données.

2 Exercice 2

Construire un perceptron capable de séparer les données de l'exercice 1. Pour cela :

2.1 La structure neurone

Créer une structure Neurone, contenant 3 champs : le biais (réel), la sortie (entier) et le tableau de poids (réels).

2.2 Le pas d'apprentissage

Définir une constante correspondant au pas d'apprentissage à 0.01.

2.3 Initialisation d'un neurone

Faire une fonction initialisant un neurone :

- La sortie est initialisée à 0.
- Le biais est initialisé à 0.5.
- Les poids (w) sont initialisés aléatoirement entre 0 et 1.

2.4 Sortie d'un neurone

Faire une fonction qui calcule la valeur interne d'un neurone :

- $\sigma = \left(\sum_{\text{pour chaque attribut}} w_i * x_i \right) - \text{biais}$
- avec w_i le poids du i-ème attribut et x_i la valeur du i-ème attribut pour l'exemple courant.
- Si $\sigma > 0$ la sortie y du neurone est 1 sinon -1.

La fonction prend donc en paramètre un exemple.

2.5 Mise à jour d'un neurone

Faire une fonction de mise du neurone (mise à jour du biais et du vecteur de poids). La fonction prend également en paramètre un exemple. Pour la mise à jour nous allons utiliser une descente de gradient.

- $\text{biais} = \text{biais} + \text{Pas d'apprentissage} * (\text{étiquette} - \text{sortie}) * -0.5$.
- pour chaque attribut :
 - $w_i = w_i + \text{Pas d'apprentissage} * (\text{étiquette} - \text{sortie}) * x_i$

2.6 Boucle principale

- Initialiser le neurone.
- Pour 100 itérations :
 - $\text{NombreDerreurs} = 0$
 - Pour chaque exemple :
 - Calculer la sortie du neurone.
 - Si la sortie est différente de l'étiquette : $\text{NombreDerreurs}++$ et mettre à jour le neurone.
- Afficher le nombre d'erreur.

3 Exercice 3

1. Regarder la courbe d'évolution du nombre d'erreur.
2. Faire varier le nombre d'exemples et comparer.
3. Regarder l'impact du pas d'apprentissage (en le faisant varier).
4. Changer les données : pour chaque point généré, rajouter une condition : x_2 doit en plus être supérieur à 0.5 et refaire les exercices 1 et 2.
5. Regarder également le nombre d'erreurs en généralisation.