

PERCEPTION, MANIPULATION ET PROTECTION D'IMAGE

Marie Guénon / Jean-Dominique Favreau / Arnaud Tanguy

COMPTE RENDU
DE TP3

Table des matières

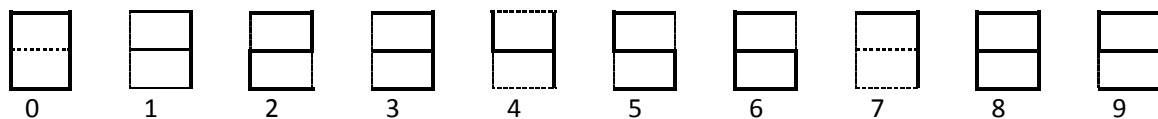
Description du sujet	2
1. Affichage des chiffres.....	2
2. Travail demandé	3
Travail réalisé	5
1. Initialisation des x_i	5
2. Constantes	5
3. Initialisation des w_i	5
4. Descente de gradient.....	5
5. Résultats	5

Description du sujet

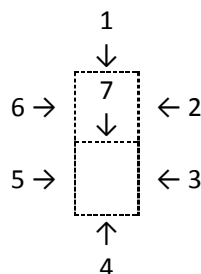
Le but de ce projet est de créer un algorithme automatique basé sur une descente de gradient et qui saurait dire si un chiffre affiché d'une manière particulière est pair ou impair.

1. Affichage des chiffres

Les chiffres sur lesquels nous allons travailler suivent un affichage et une mise en forme bien particulière. Globalement, on peut dire que l'affichage des chiffres que nous allons utiliser suit l'affichage classique des horloges numérique. Ce qui nous donne :



Nous pouvons donc dire que chaque chiffre est constitué de 7 segments qui sont "allumé" (ici en gras) ou non et qui définit le chiffre affiché. Chaque segment a été numéroté comme suit :



A partir de cette numérotation, on définit l'état de chaque segment de manière binaire : le segment est à 1 si il est "allumé", à 0 sinon. Puis concatène dans l'ordre l'état binaire de tous les segments. Ce qui nous donne :

0	1	2	3	4	5	6	7	8	9
1111110	0110000	1101101	1111001	0110011	1011011	1011111	1110000	1111111	1111011

2. Travail demandé

Le but est de détecter de manière automatique si les chiffres formatés comme décrit ci-dessus sont pairs ou impairs grâce à un algorithme de perceptron. Celui-ci sera de la forme suivante :



PERCEPTION, MANIPULATION ET PROTECTION D'IMAGE

Où les x_i sont les entrées correspondant à l'état des segments et x_0 est fixé à 1.

Les w_i sont les poids attribués à chaque entrée et w_0 est le coefficient synaptique.

La sortie o est calculée selon la formule suivante :

$$o = \sum_{i=0}^7 w_i x_i$$

Soit c la sortie qui était attendue. A chaque itération, les w_i sont recalculés suivant la formule :

$$w_i = w_i + \varepsilon(c - o)x_i$$

Où ε est une constante influant la vitesse de convergence de notre algorithme.

Travail réalisé

1. Initialisation des x_i

Binaire -1/1 à cause de la sigmoïde ($s(0)=0$ fait buger notre algo)

2. Constantes

3. Initialisation des w_i

4. Descente de gradient

5. Résultats