

Classification par réseau de neurones et extraction des caractéristiques

2.1. Neurone biologique

Un neurone est une cellule nerveuse constituant la base du système nerveux, spécialisée dans le traitement des signaux électriques. C'est en 1881 que le terme de neurone fut inséré dans le vocabulaire médical par l'anatomiste allemand Heinrich Wilhelm Waldeyer. En biologie, le cerveau humain contient un grand nombre de neurones fortement interconnectés constituant des réseaux de neurones.

Le système nerveux compte plus de 1000 milliards de neurones interconnectés. Bien que les neurones ne soient pas tous identiques, leur forme et certaines caractéristiques permettent de les répartir en quelques grandes classes. En effet, il est aussi important de savoir que les neurones n'ont pas tous un comportement similaire en fonction de leur position dans le cerveau.

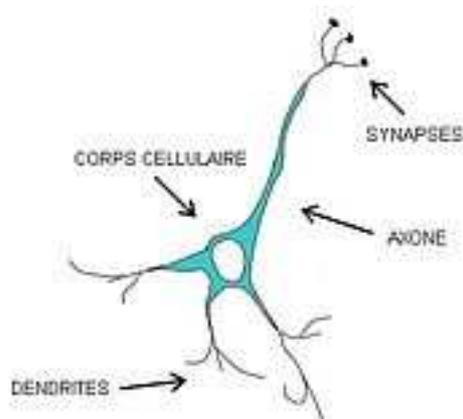


Figure 2.1 : Neurone biologique

Comme illustré sur la Figure 2.1, un neurone contient:

- Le corps cellulaire : il contient le noyau du neurone ainsi que la machine biochimique nécessaire à la synthèse des enzymes. Ce corps cellulaire, de forme sphérique ou

pyramidale, contient aussi les autres molécules essentielles à la vie de la cellule. Sa taille est de quelques microns de diamètre.

- Les dendrites : ce sont de fines extensions tubulaires qui se ramifient autour du neurone et forment une sorte de vaste arborescence. Les signaux envoyés au neurone sont captés par les dendrites. Leur taille est de quelques dizaines de microns de longueur.
- L'axone: c'est le long de l'axone que les signaux partent du neurone. Contrairement aux dendrites qui se ramifient autour du neurone, l'axone est plus long et se ramifie à son extrémité où il se connecte aux dendrites des autres neurones. Sa taille peut varier entre quelques millimètres à plusieurs mètres.
- La synapse: une synapse est une jonction entre deux neurones, et généralement entre l'axone d'un neurone et une dendrite d'un autre neurone.

Le traitement de l'information par chaque neurone montre que celle-ci n'est pas stockée dans les neurones, mais est bien le résultat du comportement de tout le réseau interconnecté. L'information est donc principalement dans l'architecture et dans la force des connexions neuronales.

Les neurones biologiques ont deux propriétés qui peuvent être appliquées au niveau des neurones artificiels :

- l'excitabilité : c'est-à-dire la capacité de réagir sous l'effet de stimulations extérieures et de convertir celles-ci en impulsions
- la conductivité : c'est-à-dire la capacité de transmettre les impulsions nerveuses.

2.2. Réseau de neurone artificiel

Les réseaux de neurones biologiques qui constituent le cerveau humain réalisent simplement de nombreuses applications telles que la reconnaissance de formes, le traitement du signal, la mémorisation, la généralisation, l'apprentissage, etc. Les réseaux de neurones artificiels sont donc un moyen de modéliser le mécanisme d'apprentissage et de traitement de l'information qui se produit dans le cerveau humain.

2.2.1. Modélisation d'un neurone

Le neurone artificiel reçoit un certain nombre de variables d'entrées provenant de neurones en amont. À chacune de ces entrées, est associé un poids qui représente la force de la connexion entre les deux neurones. Chaque neurone est doté d'une unique sortie qui se ramifie ensuite pour représenter les entrées qui alimenteront d'autres neurones en aval. Pour résumer, chaque neurone calcule une sortie unique en se basant sur les informations qui lui sont données.

2.2.2. Le neurone formel

Le passage des observations neurophysiologiques et anatomiques au neurone formel a été proposé par W. McCulloch et W. Pitts. Malgré la simplicité de cette modélisation, le neurone formel dit de «McCulloch et Pitts» reste aujourd'hui un élément de base des réseaux de neurones artificiels.

Le neurone de McCulloch et Pitts est un dispositif binaire, qui reçoit des stimulations par des entrées, et les pondère grâce à des valeurs réelles appelées coefficients synaptiques, poids synaptiques ou simplement synapses. Ces coefficients peuvent être positifs, et l'on parle alors de synapses excitatrices, ou négatifs pour des synapses inhibitrices. Le neurone calcule ainsi une somme de ses entrées pondérées par les coefficients et prend une décision en la comparant à un seuil fixé : si la somme pondérée des entrées dépasse le seuil, la sortie produite vaut +1, sinon la sortie vaut -1. Un schéma du fonctionnement d'un neurone formel est proposé dans la Figure 2.2.

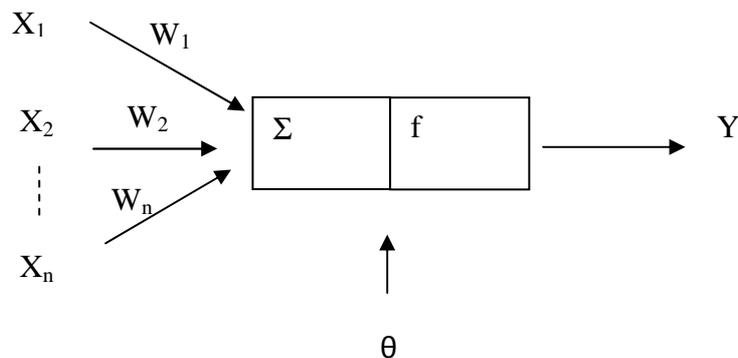


Figure 2.2 : Neurone formel

Le neurone reçoit les entrées X_1, X_2, \dots, X_n et calcule le potentiel v en sommant les entrées pondérées avec selon l'Eq. 2.1.

$$v = \sum_{i=1}^n W_i * X_i + \theta \quad (2.1)$$

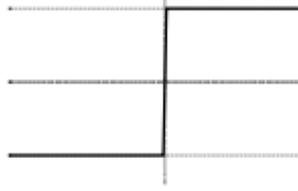
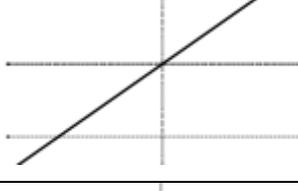
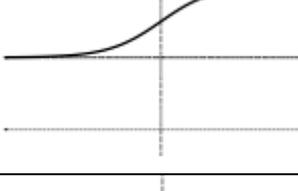
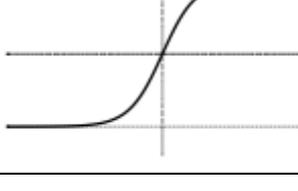
Une décision est ensuite prise pour calculer la sortie s en fonction du seuil θ (Eq.2.1)

$$\begin{cases} \text{Si } v > \theta, \text{ alors } Y = +1 \\ \text{Sinon } Y = -1 \end{cases} \quad (2.2)$$

Les variations possibles de ce modèle viendront du choix du nombre d'entrées, de la valeur des poids, ou de la fonction de décision. Le nombre d'entrées dépend essentiellement des problèmes abordés : en reconnaissance de formes par exemple, on pourra avoir autant d'entrées que de pixels dans l'image, en classification de données on aura autant d'entrées que de mesures du processus observé, etc. La fonction de décision f est un élément du modèle qui dépend en partie du problème, si la sortie du neurone doit délivrer une valeur binaire, il faut utiliser une fonction à échelon mais si la sortie doit être une valeur réelle, une fonction à échelon ne convient plus. Dans ce cas on prend $Y = f(v)$.

On peut voir des exemples de fonction de décision au Tableau I.

Tableau I: Fonction de décision

Nom de la fonction	Valeur	Représentation
Seuil	$f(x) = 0$ si $x \leq 0$ $f(x) = 1$ si $x > 0$	
Seuil symétrique	$f(x) = -1$ si $x < 0$ $f(0) = 0$ si $x = 0$ $f(x) = 1$ si $x > 0$	
Linéaire saturé	$f(x) = 0$ si $x < 0$ $f(x) = x$ si $0 \leq x \leq 1$ $f(x) = 1$ si $x > 1$	
Linéaire	$f(x) = x$	
Sigmoïde	$f(x) = \frac{1}{1+e^{-x}}$	
Tangente hyperbolique	$f(x) = \frac{e^x - e^{-x}}{e^{-x} + e^x}$	

2.2.3. Connectivité des neurones

La connectivité des réseaux est la manière dont les neurones sont connectés entre eux. Elle peut-être totale quand tous les neurones sont connectés entre eux ou organisée par couche

quand les neurones d'une couche ne sont connectés qu'à la couche précédente en entrée et à la couche suivante en sortie.

2.2.4. Architectures des réseaux de neurones

On associe plusieurs neurones pour établir un réseau. Il existe plusieurs types de réseaux suivant la façon dont sont connectés les neurones entre eux. Plusieurs architectures des réseaux existent. Les deux architectures principales sont les « réseaux à une couche » et les « réseaux multicouches ».

- Réseaux à une seule couche :

Un réseau à une seule , illustré à la Figure 2.3, ne permet de séparer que des classes linéairement séparables. On appelle classes linéairement séparables des classes qui peuvent être correctement séparées par un hyperplan ou une fonction linéaire.

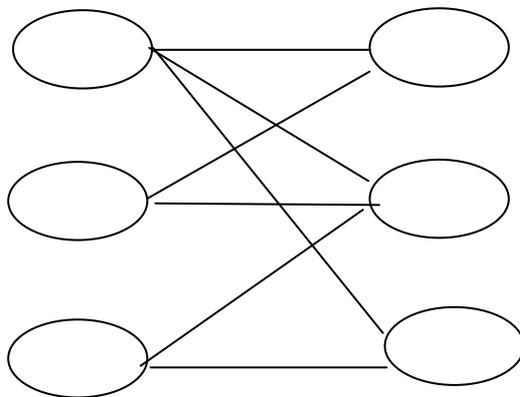
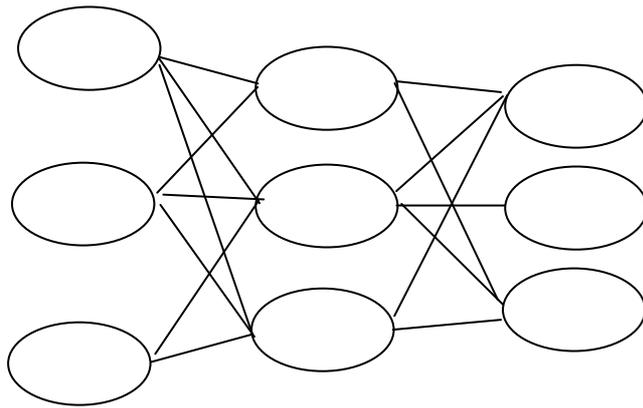


Figure 2.3 : Réseau monocouche

- Réseaux multicouches :

Notons que le nombre de couches et le nombre de neurones par couche dépendent principalement du problème étudié et des résultats de l'expérimentation sur différentes combinaisons des valeurs de ces paramètres. Les réseaux de neurones à une seule couche sont limités par le fait qu'il ne peut pas séparer des classes non linéairement séparables.



Couche d'entrée Couche cachée Couche de sortie

Figure 2.4 : Réseau multicouche

2.2.5. Apprentissage

Une caractéristique des réseaux de neurones est leur capacité à apprendre à reconnaître une forme, un son, etc. Mais cette connaissance n'est pas acquise dès le départ. L'apprentissage d'un réseau de neurones artificiels passe par les trois étapes:

- Acquisition des données formant la base d'apprentissage.
- Prétraitement : il consiste à localiser, segmenter et normaliser les représentations.
- Choix des attributs : après le prétraitement, on doit extraire des attributs qui définissent les données. Ces attributs servent comme des entrées au réseau de neurones.

Avant le traitement des données, nous avons donc à effectuer le choix des objets, la définition des attributs caractérisant les objets et la construction de la base d'apprentissage. A la fin de cette phase, on obtient un tableau (Tableau II) à deux entrées : les données et les attributs les caractérisant .

Tableau II : Base d'apprentissage

		Données			
		X_1	X_2	...	X_n
Attributs	1	X_{11}	X_{12}	...	X_{1n}
	2	X_{21}	X_{22}	...	X_{2n}

	n	X_{n1}	X_{n2}	...	X_{nn}

Ayant une nouvelle observation, il s'agit de décider à quelle classe l'affecter. Pour que le système soit performant, il faut qu'il se comporte bien sur une base de données autre que sa base d'apprentissage, cette base s'appelle la base de généralisation.

La plupart des réseaux de neurones apprennent par l'exemple en suivant un algorithme d'apprentissage. Il y a deux algorithmes principaux : l'apprentissage supervisé et l'apprentissage non supervisé [12].

- Lors d'un apprentissage supervisé, les résultats corrects sont fournis au réseau, si bien que celui-ci peut ajuster ses poids de connexion pour les obtenir. Après l'apprentissage, le réseau est testé en lui donnant seulement les valeurs d'entrée mais pas les sorties désirées, et en regardant si le résultat obtenu est proche du résultat désiré. Un exemple de cet apprentissage est qu'on apprend au réseau à reconnaître un carré, un triangle et un cercle. Comme base d'apprentissage, on pourra définir que s'il y a quatre côtés ce sera un carré, s'il y a trois côtés ce sera un triangle et que s'il n'y a pas de côté ce sera un cercle.
- Lors d'un apprentissage non supervisé, on ne fournit pas au réseau les sorties que l'on désire obtenir. On dispose d'une base de données sans connaître leur classe d'appartenance. Il s'agit de découvrir la structure sous-jacente aux données sans imposer aucun modèle. Un exemple de cet apprentissage est d'avoir en entrée un ensemble de données dont le type est inconnu, le système va alors classifier ces données.

2.3. Réseau de neurone existant

2.3.1. Perceptron

C'est un des premiers réseaux de neurones, conçu en 1958 par Rosenblatt. Il est linéaire et monocouche. Il est inspiré du système visuel. La première couche d'entrée représente la rétine, les neurones de la couche sont les cellules d'association et la couche finale est composée des cellules de décision. Les sorties des neurones ne peuvent prendre que deux états (-1 et 1 ou 0 et 1).

Seuls les poids des liaisons entre la couche d'association et la couche finale peuvent être modifiés. La règle de modification des poids utilisés est la règle de Widrow-Hoff : si la sortie du réseau est égale à la sortie désirée, le poids de la connexion entre ce neurone et le neurone d'association qui lui est connecté n'est pas modifié. Dans le cas contraire, le poids est modifié proportionnellement à la différence entre la sortie obtenue et la sortie désirée :

$$w \leftarrow w + k (d - s)$$

s est la sortie obtenue, d la sortie désirée et k une constante positive.

En 1969, Papert et Minsky ont démontré les limites du perceptron classique, incapable, par exemple de simuler la fonction ou exclusif (xor).

2.3.2. Les perceptrons multicouches (PMC)

Ils sont une amélioration du perceptron comprenant une ou plusieurs couches intermédiaires dites, couches cachées, dans le sens où elles n'ont qu'une utilité intrinsèque pour le réseau de neurones et pas de contact direct avec l'extérieur. Chaque neurone n'est relié qu'aux neurones des couches directement précédente et suivante, mais à tous les neurones de ces couches.

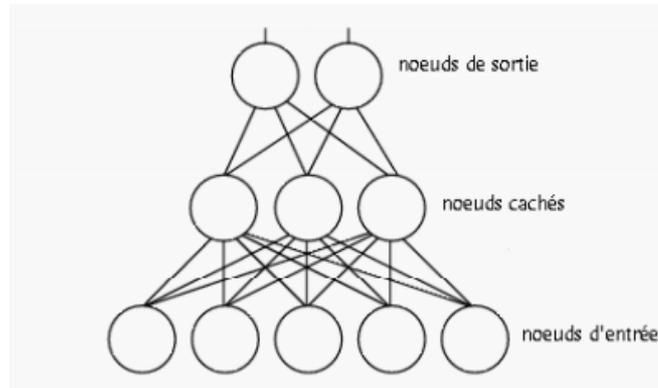


Figure 2.5 : Perceptron multicouche

Les PMC utilisent, pour modifier leurs poids, un algorithme de « rétropropagation du gradient », qui est une généralisation de la règle de Widrow-Hoff. Il s'agit toujours de minimiser l'erreur quadratique. On propage la modification des poids de la couche de sortie jusqu'à la couche d'entrée. Les PMC agissent comme un séparateur non linéaire et peuvent être utilisés pour la classification, le traitement de l'image ou l'aide à la décision.

2.3.3. Les réseaux de Hopfield

Il s'agit d'un réseau constitué de neurones à deux états (-1 et 1, ou 0 et 1), dont la loi d'apprentissage est la règle de Hebb (1949), qui veut qu'une synapse améliore son activité si et seulement si l'activité de ses deux neurones est corrélée, c'est-à-dire que le poids d'une connexion entre deux neurones augmente quand les deux neurones sont activés en même temps. Le modèle a la particularité d'être dynamique, contrairement au perceptron simple et multi-couche, dont les sorties sont calculées en une passe, après présentation des entrées. Le réseau de Hopfield réinjecte les sorties calculées à chaque itération dans le réseau, afin d'initier une itération supplémentaire, jusqu'à ce que l'état interne du réseau devienne stable.

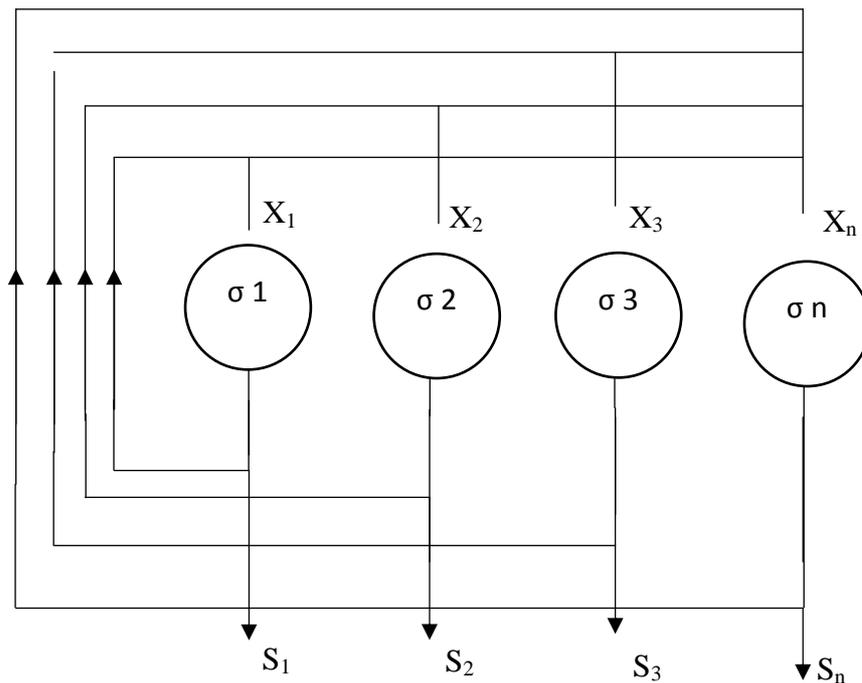


Figure 2.6 Réseau de Hopfield

Le modèle de Hopfield comporte : N entrées x_i , N sorties s_i et N neurones internes x_i . L'état du système dépend des valeurs de X_i comme illustré sur la Figure 2.6.

2.3.4. Les réseaux de Kohonen

Contrairement aux réseaux de Hopfield où les neurones sont modélisés de la façon la plus simple possible, on recherche ici un modèle de neurone plus proche de la réalité. Ces réseaux sont inspirés des observations biologiques du fonctionnement des systèmes nerveux de perception des mammifères.

Une loi de Hebb modifiée qui tient compte de l'oubli est utilisée pour l'apprentissage. La connexion est renforcée dans le cas où les neurones reliés ont une activité simultanée et diminuée dans le cas contraire alors qu'il ne se passait précédemment rien dans ce cas. Ceci se résume par la formule :

(2.3)

$$d w / dt = k S e - B(S) w$$

où w est le poids associé à une certaine connexion, e la valeur que le neurone reçoit en entrée par cette connexion, S la valeur qu'il renvoie en sortie qui est toujours positive, $B(S)$ la fonction d'oubli et k une constante positive.

Une loi d'interaction latérale est aussi modélisée. Les neurones très proches physiquement interagissent positivement, le poids des connexions est augmenté autour d'une certaine zone quand une synapse est activée et négativement pour les neurones un peu plus loin et pas du tout pour les neurones éloignés. Ceci crée un "amas" de neurones activés et contribue à spécialiser certains neurones. Pour une entrée donnée, une sortie particulière sera activée alors que les autres resteront inertes. On utilise aussi parfois des lois de concurrence entre les neurones. Ceci permet de résoudre certains problèmes, dits NP complets, par exemple : comment relier n villes par le chemin le plus court.

Les réseaux de Kohonen ont des applications dans la classification, le traitement de l'image, l'aide à la décision et l'optimisation.

2.4. Extraction des caractéristiques

Tout système de reconnaissance ou de classification, qui traite des données réelles comme des images ou encore des signaux acoustiques, doit passer par une phase d'extraction de primitives ou caractéristique. Cette phase critique, est considérée comme l'un des problèmes posés lors de la construction d'un système de reconnaissance, parce que la majorité des techniques d'extraction des caractéristiques s'accompagne d'une perte d'information irrémédiable. Le principe de l'extraction de caractéristiques est donc d'extraire de l'image du caractère segmenté des valeurs numériques qui caractérisent la classe du caractère et le discriminent idéalement de tous les caractères des autres classes. N'avoir qu'un petit nombre de valeurs numériques extraites facilite d'autant l'apprentissage du classifieur.

2.4.1. Types de caractéristiques

On peut distinguer deux types de caractéristiques :

- Caractéristiques non reconstructives : à partir des caractéristiques non reconstructives, il n'est pas possible de retrouver la forme de départ.

- **Caractéristiques reconstructives :** Les caractéristiques reconstructives permettent de reconstruire la forme initiale, avec un degré d'approximation plus ou moins bon selon que l'on garde plus ou moins de valeurs caractéristiques.

2.4.2. Méthode d'extraction des caractéristiques

a. Méthode des pixels moyennés

Dans cette méthode, la matrice de l'image est divisée en $p \times q$ zones de même taille. Pour chaque zone, on calcule la valeur moyenne de niveaux de gris de tous les pixels dans cette zone et le résultat obtenu sera utilisé comme un élément du vecteur de caractéristique. Donc, le vecteur de caractéristique est de taille $p \times q$ comme illustré sur la Figure 2.7.

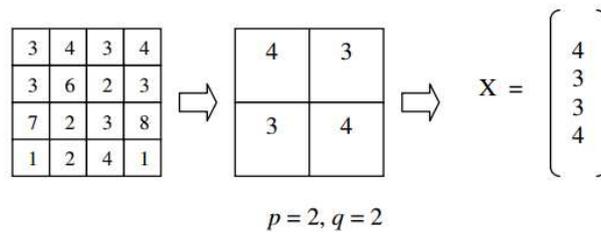


Figure 2.7 : méthode des pixels moyennés

b. Vectorisation

La vectorisation d'un caractère est une procédure qui vise à représenter ce caractère par un ensemble de segments droits plutôt que par un ensemble de pixels comme illustré sur la Figure 2.8. Cette représentation est vectorielle. Chaque ensemble de pixels continus est remplacé par un segment. Chaque segment est codé par les 4 coordonnées de 2 extrémités (x_0, y_0) , (x_n, y_n) .

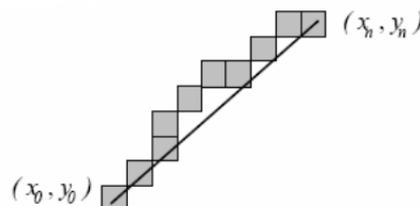


Figure 2.8 : Vectorisation de caractère

Après la vectorisation du caractère, on peut effectuer une étude morphologique pour essayer de détecter, par exemple, la présence éventuelle de trou dans le caractère.

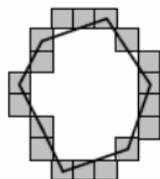


Figure 2.9 : Caractère vectorisé avec un trou

c. Projection horizontale et verticale

Les histogrammes de projection verticale et horizontale servent à déterminer les séparations entre caractères et entre les lignes, mais ils peuvent aussi être utilisés comme descripteurs. On obtient ainsi une approximation de la forme générale de chaque caractère (Figure 2.10).



Figure 2.10 : Projection horizontale et verticale du chiffre 3

d. Concentration de pixel

L'image est divisée horizontalement et verticalement, ainsi on obtient le nombre de pixel formant le caractère à reconnaître pour chaque zone. Pour chaque zone, ce nombre est différent pour chaque caractère comme illustré sur laFigure 2.11.

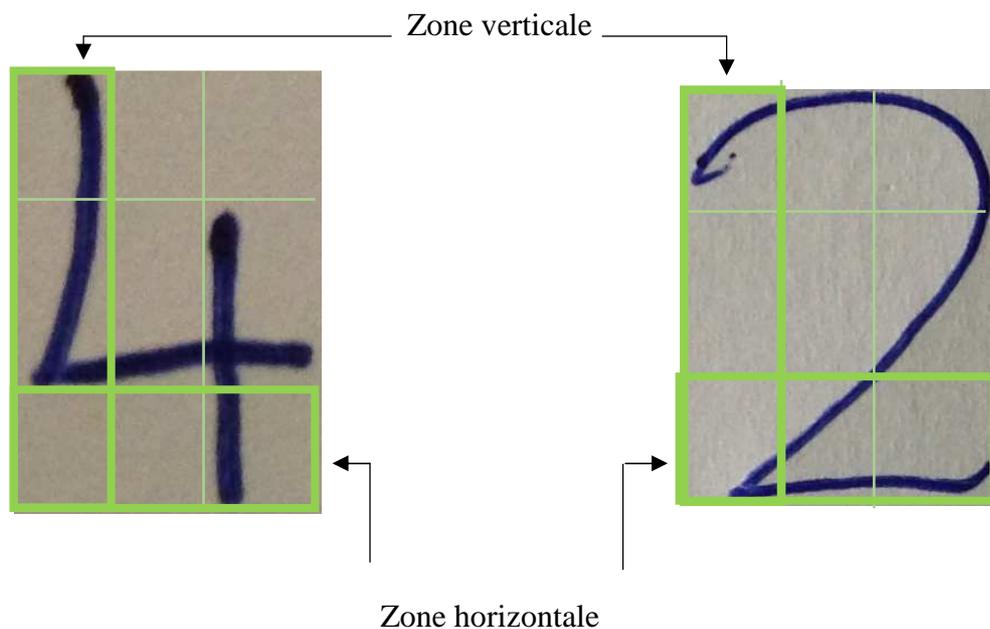


Figure 2.11 : Concentration de pixel

e. Détection de concavité par analyse morphologique

La présence d'une information topologique de plus haut niveau que l'image brute du caractère, dans le vecteur de primitives, peut faciliter grandement la tâche du système de classification. La méthode consiste à envoyer un certain nombre de « sondes » vers le caractère, selon diverses directions (Figure 2.12). La longueur mesurée de chaque sonde est l'ordonnée, selon l'axe de recherche, du premier pixel d'intensité non nulle rencontré.

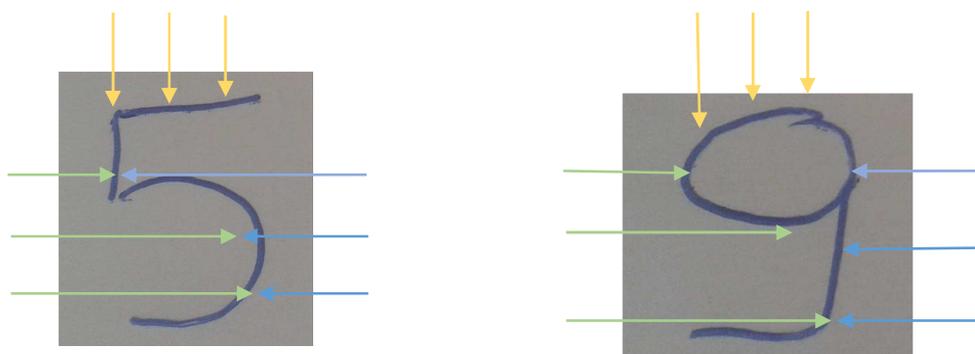


Figure 2.12 : Détection de concavité

Dans le cas des caractères typographiques, les positions et les orientations des caractéristiques topologiques discriminantes demeurent pratiquement fixes dans l'image d'un caractère. L'emplacement ainsi que le nombre de sondes à utiliser peuvent alors être déterminés à priori. Cette situation idéale n'est évidemment pas vérifiée dans le cas des caractères manuscrits et les sondes utilisées doivent alors être uniformément réparties le long du pourtour de l'image du caractère. Pour des raisons de facilité de calcul, il est aussi préférable que la détection des concavités ait lieu sur des caractères préalablement ramenés à des dimensions normalisées. Le signal fourni par les sondes peut être ainsi également normalisé, en divisant la valeur mesurée par celle de la dimension standard des caractères.

2.5. Mis en œuvre des réseaux de neurone

L'un des problèmes majeurs en intelligence artificielle est la distinction entre deux classes. Il se pose comme la base de systèmes d'IA beaucoup plus complexes. Par exemple, en traitement automatique du langage, on cherche à distinguer les voyelles des consonnes. En reconnaissance des formes, on distingue un carré d'un cercle ; pour les systèmes experts, il s'agit de reconnaître un cas particulier d'un autre. L'utilisation d'un réseau de neurones permet de résoudre ces problèmes de classification. Un neurone sait faire la différence entre deux classes dès lors qu'il a appris par des exemples successifs à classer correctement.

Pour la mise en œuvre du perceptron, le perceptron peut être vu comme un unique neurone capable de séparer linéairement un ensemble de vecteurs E entrée en deux groupes distincts étiquetés A et B.

Dans son état initial, un perceptron ne sait pas comment séparer les deux groupes, il faut qu'il "apprenne". Nous disposons pour cela d'un échantillon d'apprentissage constitué de vecteurs d'entrée particuliers, dont on connaît le groupe g d'appartenance :

$$g = \begin{cases} 1, & \text{si } E_{\text{learn}} \times A \\ -1, & \text{si } E_{\text{learn}} \times B \end{cases} \quad (2.4)$$

L'algorithme d'apprentissage consiste à donner les vecteurs E_{learn} en entrée du perceptron et comparer la sortie y à la sortie attendue g . Si la sortie réelle et la sortie attendue sont différentes, on va adapter le vecteur de pondération des entrées W_i et le biais b en les incrémentant ou décrémentant d'un pas α d'apprentissage fixé arbitrairement. On répète ces étapes t fois. Plus t est grand, plus le perceptron apprend. L'apprentissage est illustré par l'organigramme en Figure 2.13.

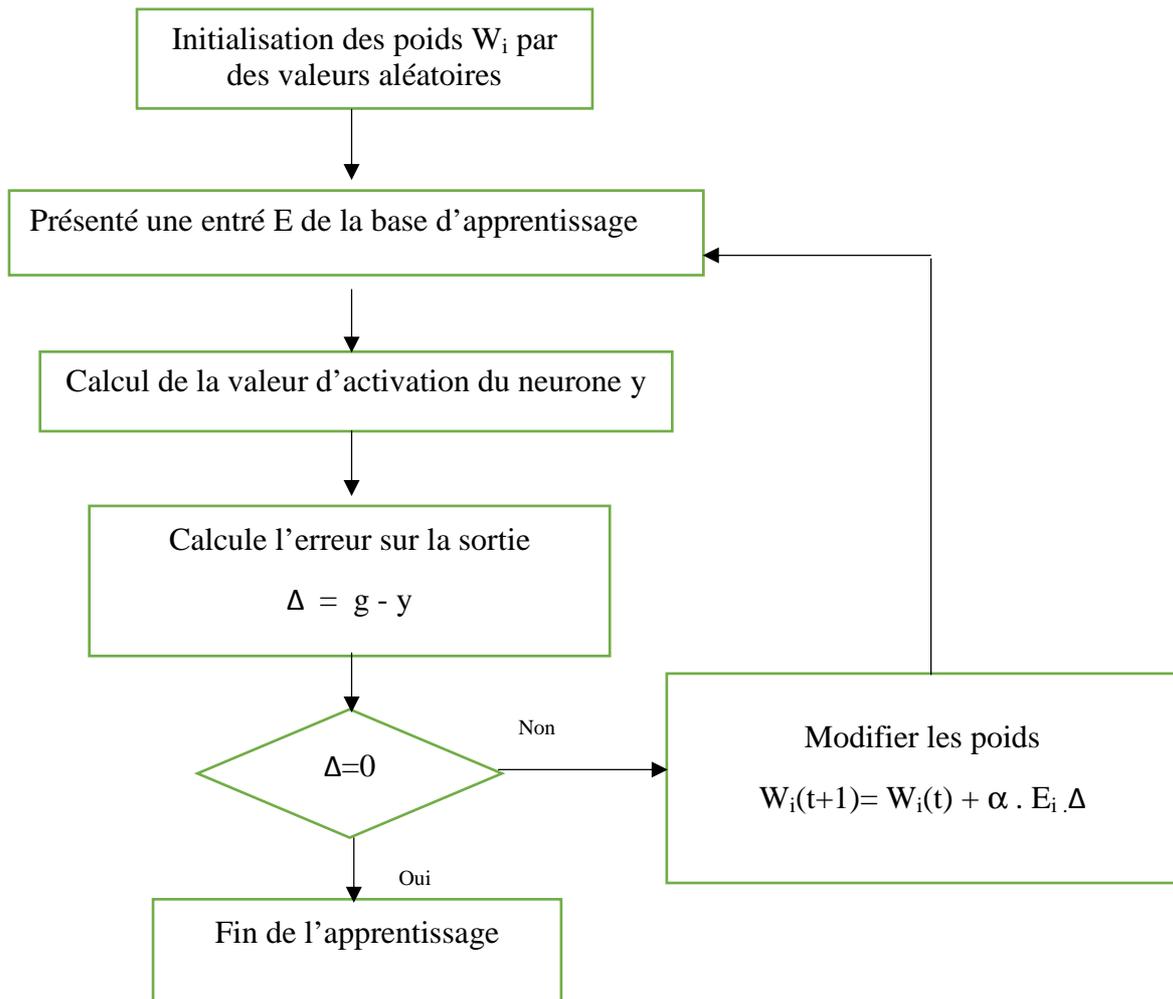


Figure 2.13 : Organigramme d'apprentissage

Après l'apprentissage, les poids et le biais du perceptron sont fixes. On soumet alors des vecteurs E entrée et la sortie du perceptron doit correspondre au groupe g dans lequel ils seraient classés.

2.6. Validation de modèle

Pour pouvoir évaluer les performances d'un modèle, on dispose :

- des performances du modèle sur l'ensemble des données utilisé pour le construire.
- des performances sur de nouvelles données :
 - pour prédire le comportement du modèle.
 - pour pouvoir comparer à d'autre modèle.

Cette dernière étape doit permettre d'estimer la qualité du réseau obtenu en lui présentant des exemples qui ne font pas partie de l'ensemble d'apprentissage. Une validation rigoureuse du modèle développé se traduit par une proportion importante de prédictions exactes sur l'ensemble de la validation.