



**Les journées de
l'interdisciplinarité**

Retour vers le perceptron - fabrication d'un neurone synthétique à base de composants électroniques analogiques simples

Back to the perceptron - building a synthetic neuron from simple analog electronic components

Jean-Christophe ORLIANGES

XLIM

Jean-christophe.orlianges@unilim.fr

Younes EL MOUSTAKIME

XLIM

Aurelian CRUNTEANU STANESCU

XLIM

Ricardo CARRIZALES JUAREZ

XLIM

Oihan ALLEGRET

XLIM

URL : <https://www.unilim.fr/journees-interdisciplinarite/761>

DOI : 10.25965/lji.761

Licence : CC BY-NC-SA 4.0 International

Résumé : Les avancées récentes dans le domaine de l'intelligence artificielle (IA), en particulier dans la reconnaissance d'images et le traitement du langage naturel, ouvrent de nouvelles perspectives qui vont bien au-delà de la recherche académique. L'IA, portée par ces succès populaires, repose sur des algorithmes basés sur des "réseaux de neurones" et elle se nourrit des vastes quantités d'informations accessibles sur Internet, notamment via des ressources telles que l'encyclopédie en ligne Wikipédia, la numérisation de livres et de revues, ainsi que des bibliothèques de photographies. Si l'on en croit les propres dires du programme informatique ChatGPT, son réseau de neurones compte plus de 175 millions de paramètres. Quant à notre cerveau, qui était le modèle initial de cette approche connexionniste, il compte environ 86 milliards de neurones formant un vaste réseau interconnecté... Dans ce travail, nous proposons une approche plus modeste de l'IA en nous contentant de décrire les résultats que l'on peut obtenir avec un seul neurone synthétique isolé, le modèle historique du perceptron (proposé par Frank Rosenblatt dans les années 1950). C'est un "Retour vers le futur" de l'IA qui est entrepris pour fabriquer et tester un neurone artificiel à partir de composants électroniques simples. Celui-ci doit permettre de différencier un chien d'un chat à partir de données anatomiques collectées sur ces animaux.

Mots clés : intelligence artificielle, perceptron, électronique analogique

Abstract: Recent advances in artificial intelligence (AI) in the fields of image recognition and natural language processing are opening up new horizons that extend far beyond the realms of research, to influence our entire society. AI, which is associated with these popular successes, relies on algorithms based on "neural networks" which, using the vast quantities of information accessible via the Internet (information linked to the free encyclopedia Wikipedia, to the digitization of books and magazines, to the libraries of photographs available...), enable these systems to learn automatically. According to the ChatGPT computer program, its neural network contains over 175 million parameters. As for our brain, which was the initial model for this connectionist approach, it comprises some 86 billion neurons forming a vast interconnected network... In this work, we propose a more modest approach to AI, contenting ourselves with describing the results that can be obtained with a single isolated synthetic neuron, the historical perceptron model (proposed by Frank Rosenblatt in the 1950s). It's a "Back to the future" of AI that is undertaken to build and test an artificial neuron from electronic components.

Keywords: artificial intelligence, perceptron, analog electronics

Introduction

Les récents progrès de l'intelligence artificielle (IA) dans les domaines de la reconnaissance d'images, de la traduction de textes et du traitement du langage naturel ouvrent des horizons nouveaux, s'étendant bien au-delà du domaine de la recherche institutionnelle pour exercer une influence sur l'ensemble de notre société. Ces avancées impactent notamment notre façon de communiquer avec les machines, de travailler, de se déplacer, de recevoir des soins médicaux et même de dispenser l'enseignement. L'IA, qui bénéficie du succès croissant de ces réalisations populaires, tire profit d'algorithmes fondés sur des "réseaux de neurones". Ces réseaux exploitent la richesse considérable d'informations offerte par Internet, comprenant des données provenant de sources variées telles que la vaste base de connaissances de Wikipédia, la numérisation de livres et de revues, ainsi que des bibliothèques de photographies... L'apprentissage automatique de ces systèmes repose sur l'utilisation de ces quantités massives d'informations.

Dans cette étude, nous adoptons une démarche plus modeste en matière d'IA, nous nous contentons de décrire les résultats pouvant être obtenus au moyen d'un seul neurone synthétique isolé, en nous inspirant du modèle historique du perceptron, proposé par Frank Rosenblatt dans les années 1950. Une sorte de « Retour vers le futur » de l'IA est entreprise, consistant à concevoir et tester un neurone artificiel en utilisant des composants électroniques analogiques élémentaires. L'objectif de notre perceptron analogique sera de parvenir à distinguer un chien d'un chat à partir de données anatomiques collectées sur des images.

Le travail, présenté ci-après, a été réalisé dans le cadre du stage « Cordée de la recherche » de Younes El Moustakime alors étudiant en Master 1 EUR-IXEO à Limoges.

1. Contexte de l'émergence des réseaux de neurones artificiels

Datant de plus de 60 ans, le concept d'intelligence artificielle fut formalisé à l'initiative de quatre éminents scientifiques : John Mc Carthy, Claude Shannon, Marvin Minski et Nathaniel Rochester (Mc Carthy 1955). Ceux-ci ambitionnaient la réalisation de tâches considérées comme « intelligentes » par des machines de façon plus efficace que l'être humain n'est capable de le faire¹.

La mise en application de ce principe fondateur conduisit au développement de deux approches fondamentalement opposées et rivales : une approche « symbolique » logique totalement conceptualisée et une approche « connexionniste » s'appuyant sur les réseaux de neurones artificiels (RNA) dont le fonctionnement est non explicite, de type « boîte noire » (Larousserie 2018).

L'approche connexionniste, avec ses solutions neuro-inspirées, a suscité beaucoup d'espoir à son lancement mais a rapidement montré ses limites pour les fonctions de reconnaissance et de tri des données (Minsky 1969).

¹ Le terme d'intelligence artificielle est difficile à définir sans ambiguïté, les contours de cette « discipline » étant en constante évolution, ils varient suivant l'efficacité et la nouveauté des solutions algorithmiques proposées (Le Cun 2019).

Le retour en grâce des réseaux de neurones, à la fin des années 80, trouve son origine dans le travail de Yann Le Cun au sein du prestigieux laboratoire américain AT&T Bell (Lecun 1989). Celui-ci entreprend de résoudre le problème de la lecture automatisée des codes postaux apposés sur les colis. En tirant parti des données fournies par l'US Postal Service, il parvient à entraîner un réseau multicouche d'une efficacité remarquable, propulsant ainsi cette approche à une échelle industrielle (Cardon 2018).

Mais ce n'est que depuis 2011, que cette voie de recherche connaît une renaissance éclatante pour le traitement des masses de données avec l'exemple emblématique du triomphe de l'approche connexionniste, portée par une équipe dirigée par Geoffrey Hinton (A Krizhevsky 2012) lors de la conférence ECCV, pour la reconnaissance d'images de la base de données « ImageNet » écrasant les résultats obtenus par les approches « conventionnelles » (Cardon 2018). Le taux d'erreurs de reconnaissance est alors passé de 25 % à 16 % avec ces RNA dits convolutifs.

La figure 1 illustre l'explosion du nombre de paramètres utilisés dans ces algorithmes pour accroître leur capacité d'apprentissage. Le critère retenu dans cette représentation est l'estimation du nombre de neurones modélisés afin de comparer celui-ci au nombre approximatif de neurones dans le système nerveux de divers animaux (allant d'un jusqu'à l'éléphant).

Les premiers algorithmes utilisant des RNA étaient exécutés sur des ordinateurs conventionnels mais l'architecture fondamentalement séquentielle de ces derniers s'avère peu efficace pour implémenter ces réseaux interconnectés et procéder à leur apprentissage, dès lors que leur taille et le nombre de données d'apprentissage augmentent. Parmi ces limites intrinsèques, l'efficacité de ces machines est confrontée au « goulot d'étranglement de Von Neumann », liée à la séparation physique, dans les ordinateurs, de la mémoire et des unités de calcul induisant une latence pour chaque accès mémoire.

Après l'avènement de ces RNA, une recherche très active a alors été initiée pour construire des machines à leur image, d'architecture fondamentalement parallèle, qui seraient à la fois plus efficaces et plus sobres énergétiquement pour effectuer la phase d'apprentissage.

La figure 2 présente une illustration des solutions « hardwares » proposées, en fonction du nombre de neurones accessibles².

² Les algorithmes d'apprentissage profond sont actuellement exécutés grâce à des processeurs graphiques (GPU) plus efficaces pour cette tâche que les microprocesseurs (Lecun 2019)

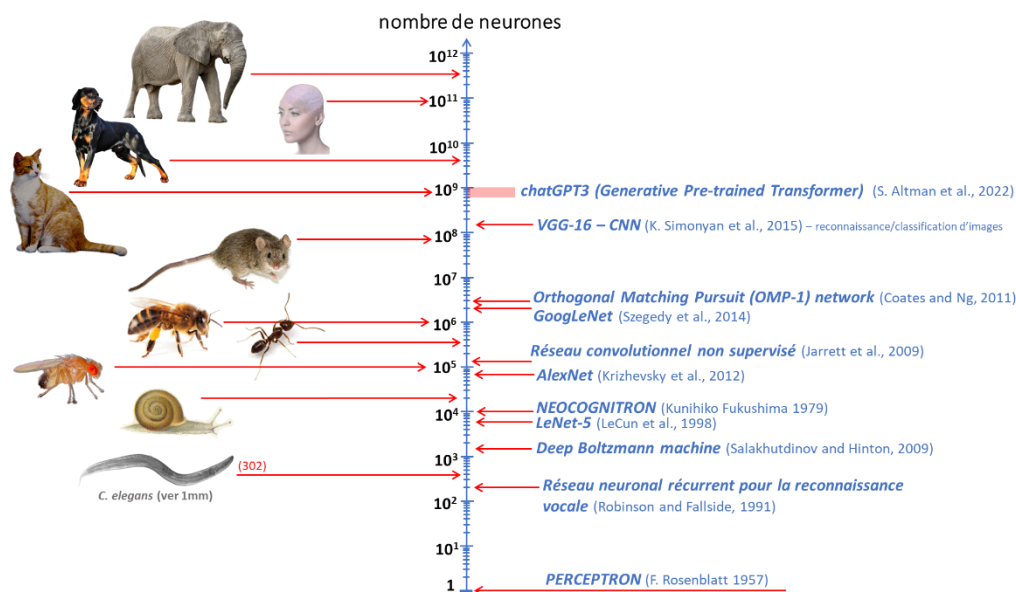


Figure 1 Comparaison du nombre approximatif de neurones dans le système nerveux d'animaux avec le nombre de neurones artificiels utilisés dans différents algorithmes (software) développés depuis les débuts de l'approche « connexionniste » (d'après les données issues de Goodefellow 2016).

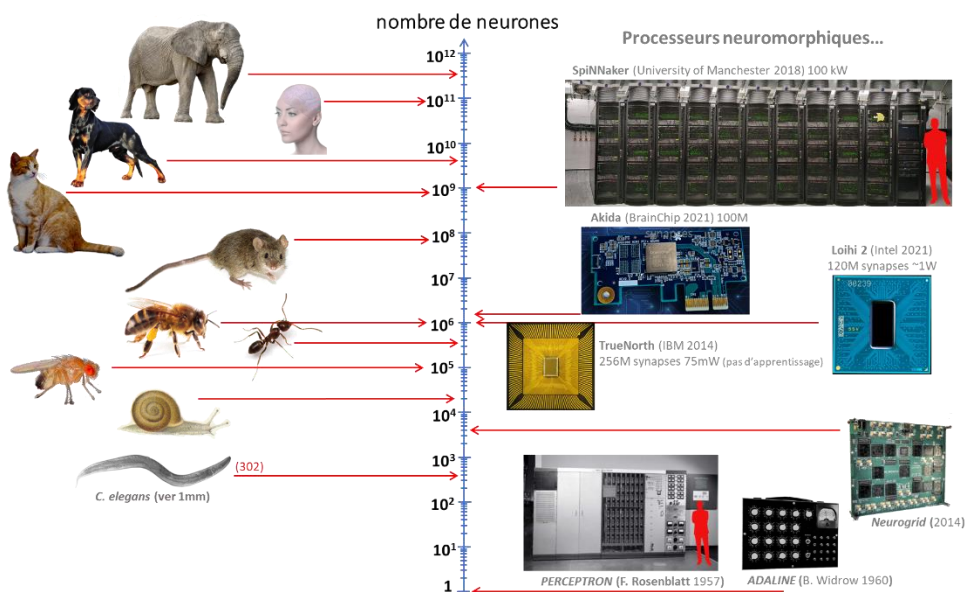


Figure 2 Comparaison du nombre approximatif de neurones dans le système nerveux d'animaux avec le nombre de neurones artificiels dans des dispositifs neuromorphiques physiques (hardware) (d'après les données issues de Wikipedia).

2. Le modèle du perceptron et sa fabrication avec des composants électriques analogiques simples

2.1. Le principe du perceptron

Le perceptron est un modèle (et une machine) créé à la fin des années 1950 par le psychologue américain Frank Rosenblatt de l'Université de Cornell. Il s'appuie sur la compréhension contemporaine des

mécanismes du fonctionnement d'un neurone isolé (figure 3-a) et leur modélisation mathématique proposée quelques années plus tôt par McCulloch et Pitts (McCulloch 1943). Comme illustré sur la figure 3-b, le neurone « mathématique » calcule la somme des signaux amonts (X_1, X_2, X_3, \dots), chacun étant affublé d'un poids spécifique relatif à son importance (W_1, W_2, W_3, \dots). Si la somme de ces activités, une fois pondérées, est inférieure à un certain seuil prédéfini, le neurone reste inactif. En revanche, si cette somme dépasse ce seuil, le neurone s'active (comme illustré sur la figure 3-b).

A ce neurone formel, Rosenblatt ajoute une procédure d'apprentissage à partir de l'estimation de l'erreur commise lors de l'essai ce qui permet l'ajustement des poids synaptiques par correction d'erreur (Rennard 2013).

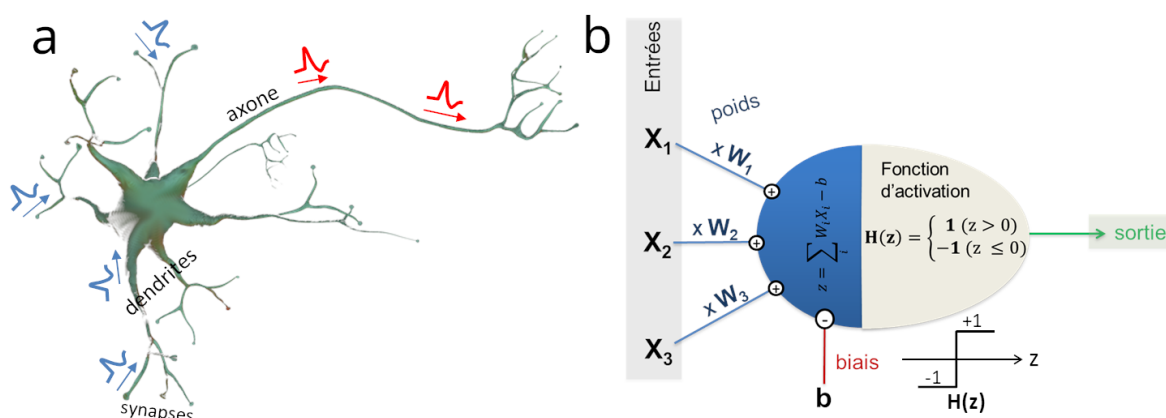


Figure 3 (a) Représentation schématique d'un neurone isolé (b) Shéma de principe d'un perceptron comprenant trois entrées X_1, X_2 et X_3 (soit quatre paramètres : les poids synaptiques W_1, W_2 et W_3 ainsi que le biais b)

Le perceptron a introduit la technique d'apprentissage supervisé en ajustant ses paramètres pour minimiser les erreurs sur un jeu de données dédié. Une fois les poids (W_i) et le biais (b) optimisés, le neurone artificiel devient opérationnel pour sa tâche de classification.

Cependant, la machine apprenante créée par Frank Rosenblatt pour la reconnaissance de formes simples a, malgré les promesses de puissance de son créateur, rapidement révélé ses limites ; elle ne permet de classifier que les données linéairement séparables, ce qui réduit considérablement son champ d'application.

Il faudra attendre l'avènement des réseaux de neurones pour dépasser cette contrainte, ainsi qu'une technique astucieuse pour ajuster en série leurs multiples paramètres (l'algorithme bien connu de la descente de gradient).

2.2. Conception d'un perceptron analogique

Pour réaliser les deux fonctions nécessaires à la fabrication d'un perceptron à partir de composants électroniques simples, il est assez naturel de penser à utiliser un amplificateur opérationnel (AO) en mode additionneur pour la première étape puis, pour la fonction d'activation, il est possible d'utiliser ce même composant en mode comparateur (comme illustré sur la figure 4).

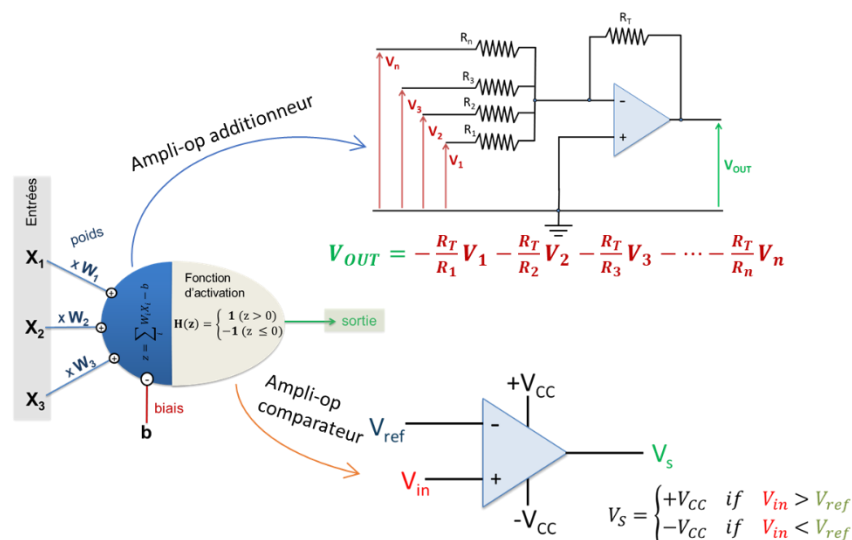


Figure 4 : Conception du perceptron à partir de composants électroniques analogiques simples (résistances ohmiques et amplificateurs opérationnels)

Ainsi, pour la partie additionneur des entrées du perceptron, le poids synaptique appliqué sur chaque entrée sera, au signe près, le rapport $\frac{R_T}{R_i}$. Il suffira donc de modifier les valeurs de ces résistances pour ajuster les poids.

Pour la fonction seuil (fonction d'activation du neurone), si l'on prend la tension $V_{ref} = 0$, donc l'entrée inverseuse de l'AO reliée à la masse, nous obtenons la fonction souhaitée.

Ainsi pour la fabrication de notre perceptron, nous n'utiliserons qu'un jeu de résistances classiques, des potentiomètres (pour ajuster les paramètres à travers la valeur de la résistance) et quelques AO alimentés par des piles 9V.

2.3. Apprentissage du perceptron pour différencier les chiens des chats...

A partir de notre base de données contenant les données morphologiques d'une population de chiens et de chats de races différentes (cf. annexe 1), nous avons choisi deux paramètres sans dimension : les ratios « longueur sur largeur du corps » et « longueur du corps hauteur au garrot » comme illustré sur la figure 5. Le caractère linéairement séparable des données apparaît clairement sur la figure.

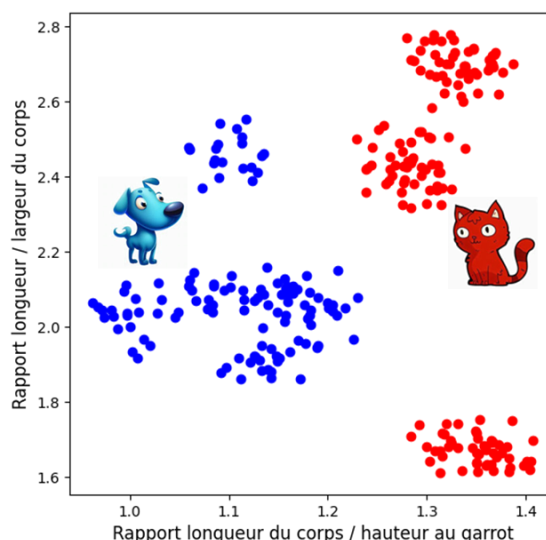


Figure 5 : Répartition des populations de chiens et de chats en fonction des deux paramètres adimensionnels choisis.

Ainsi le perceptron adapté à la résolution de notre problème de classification comporte deux entrées (les couples de valeurs des paramètres morphologiques retenus) et nécessite donc l'optimisation de trois paramètres, deux poids synaptiques (W_1 et W_2) et le biais (b).

Pour réaliser l'apprentissage de notre perceptron, un court programme en langage « python » a été écrit puis exécuté sous la forme d'un « notebook jupyter »³ en travaillant sur une partie des données collectées (données d'apprentissage).

Dans l'algorithme utilisé (présenté en langage python dans l'annexe 2), l'estimation de l'erreur est réalisée en comparant la sortie du perceptron (« -1 » pour un chat et « +1 » pour un chien) avec la réponse connue de l'élément utilisé pendant la phase d'apprentissage. Ainsi de manière élégante, il suffit de s'intéresser au signe du produit de ces deux valeurs pour savoir si la classification chien/chat du perceptron est valable :

$$\begin{array}{ll} \text{entrée} \leftarrow \text{chien} ; \text{perceptron} \rightarrow \text{chien} & \text{entrée} \leftarrow \text{chien} ; \text{perceptron} \rightarrow \text{chat} \\ (+1) \times (+1) > 0 & (+1) \times (-1) < 0 \\ \text{entrée} \leftarrow \text{chat} ; \text{perceptron} \rightarrow \text{chat} & \text{entrée} \leftarrow \text{chat} ; \text{perceptron} \rightarrow \text{chien} \\ (-1) \times (-1) > 0 & (-1) \times (+1) < 0 \end{array}$$

Après la phase d'apprentissage sur l'ensemble des données idoines, nous obtenons les résultats synthétisés sur la figure 6. Avec les paramètres ajustés du perceptron, il est possible de tracer sur le plan de représentation des données la droite de décision (en vert sur la figure 6-b).

³ Le notebook jupyter du perceptron est disponible à l'adresse <https://gitlab.xlim.fr/orlianges/perceptron-analogique.git>

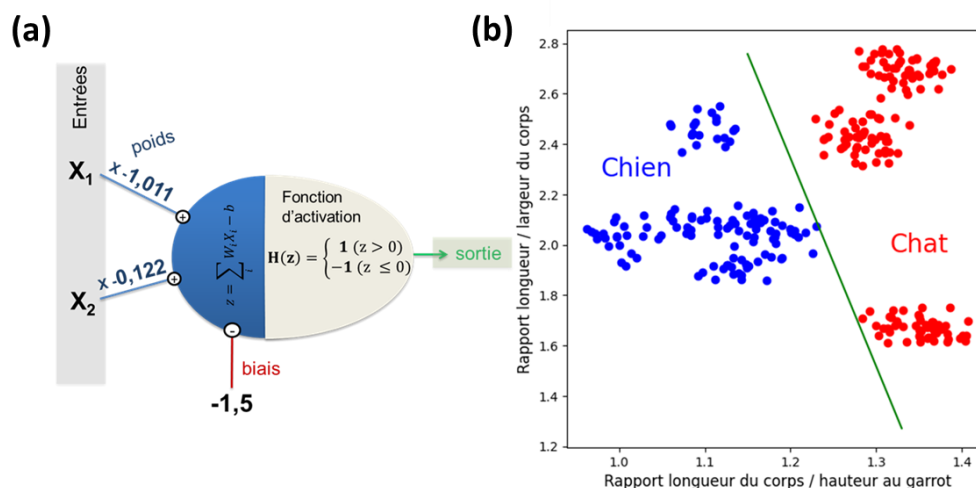


Figure 6-a Schéma du perceptron avec ses paramètres après apprentissage – b Droite de décision associée au perceptron après l'apprentissage, séparant les populations de chiens et de chats dans la même représentation graphique que sur la figure 5.

2.4. Construction et vérification du perceptron « chien / chat »

Connaissant les paramètres du perceptron, il est maintenant possible de choisir et d'ajuster les résistances du circuit à réaliser. Le schéma du circuit est présenté figure 7-a et sa réalisation sur la figure 7-b. Nous avons choisi d'utiliser une résistance $R_a = 10k\Omega$ ainsi les potentiomètres de réglage du perceptron R_1 , R_2 et R_{d2} sont ajustés de façon à obtenir les valeurs suivantes :

$$\frac{R_a}{R_1} = W_1 = 1,011 \Rightarrow R_1 = 9.89k\Omega \quad ; \quad \frac{R_a}{R_2} = W_2 = 0,122 \Rightarrow R_2 = 81,97k\Omega$$

$$\text{et} \quad V_{biases} = +1,5V$$

Le signal à la sortie du dernier AO du circuit, le comparateur, est positif s'il s'agit d'un chat et négatif dans le cas contraire. Deux LED de couleur respective verte et rouge ont donc été branchées sur la sortie en parallèle et dans des sens opposés pour s'allumer l'une ou l'autre suivant le sens du courant. Lorsqu'aux deux entrées de notre perceptron nous imposons des tensions V_1 et V_2 correspondant aux rapports morphologiques d'un chien, la LED verte s'allume (dans le cas d'un chat, la rouge prend le relais).

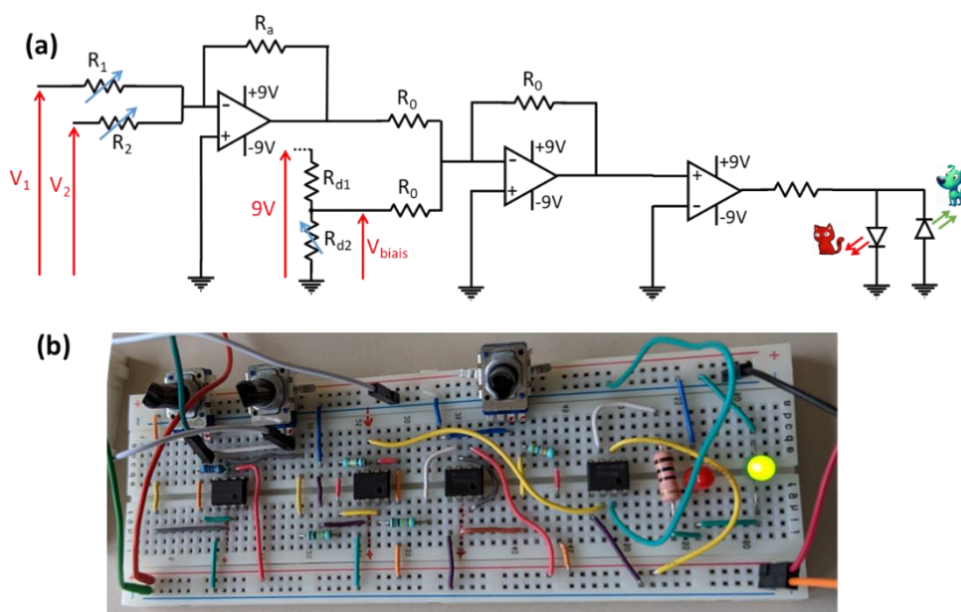


Figure 7-a Scéma du perceptron « chien/chat » - b Circuit réalisé sur une plaque d'étude, alimenté par deux piles de 9V, intégrant des AO, des résistances, des potentiomètres et deux diodes, l'une verte et l'autre rouge.

Après l'apprentissage sur une collection de 270 individus, il a été possible de vérifier la validité des réglages de notre perceptron avec 30 données non encore utilisées sans observer d'erreur.

Conclusion

Dans ce travail d'archéologie scientifique, nous avons pu faire revivre un fossile des début de l'intelligence artificielle, le perceptron monocouche. Nous avons travaillé sur une version analogique rudimentaire de ce dispositif fondateur sur lequel s'appuie encore l'IA moderne avec ses réseaux de neurones profonds comportant des millions d'unités. De même le choix de la nature de la classification de notre perceptron a été inspiré par l'usage historique de ce problème d'identification « chien ou chat », celui-ci étant toutefois directement réalisé sur des bases de données d'images dans ses versions modernes.

L'une des vertus de ce type de travail sur des systèmes physiques simples comme notre perceptron analogique est qu'il permet d'illustrer et de faciliter la compréhension du fonctionnement de systèmes plus complexes ; ainsi, le concept de mémoire sur ce type de dispositif est ici associé au choix et à l'ajustement des différentes valeurs de résistances du circuit (ici la position des différents potentiomètres). Cette mémoire est donc fondamentalement « intriquée » dans ces composants. Dans le cas de réseau de neurones, on appréhende ainsi plus facilement la notion de mémoire associée à un système parallèle distribué.

D'un point de vue pédagogique, ce travail pluridisciplinaire a aussi été conçu pour être proposé à un public de lycéens scientifiques ou techniques en restant assez simple sur la partie électronique analogique et en privilégiant l'usage du langage « python » relativement bien implanté dans ces filières.

Dans les perspectives à ce travail, deux voies de poursuite sont envisagées :

- L'automatisation de l'apprentissage directement sur le perceptron analogique
- La réalisation d'un perceptron multicouche (réseau de perceptrons) ouvrant l'accès à des fonctionnalités plus évoluées

Les auteurs de ce travail remercient le service de diffusion de la culture scientifique, technique et industrielle de l'Université de Limoges ainsi que le personnel administratif de la faculté des sciences et techniques pour le financement de ce travail dans le cadre du projet DEMIA.

Références

CARDON D., COINTET J.-P., MAZIERES A., « La revanche des neurones. L'invention des machines inductives et la controverse de l'intelligence artificielle », *Réseaux*, 2018/5 (n° 211), p. 173, URL : <https://www.cairn.info/revue-reseaux-2018-5-page-173.htm>

GOODFELLOW I., BENGIO Y., et COURVILLE A. (2016) « Deep learning. » MIT press

KRIZHEVSKY A., SUTSKEVER I., HINTON G. (2012), « ImageNet Classification with Deep Convolutional Neural Networks », NIPS 2012, Lake Tahoe, December 3-6.

LAROUSSERIE D. (2018), « Les cinq saisons de l'intelligence artificielle », *Le Monde*, 18 Juillet 2018.

LECUN Y., BOSER B., DENKER J., HENDERSON D., HOWARD R., HUBBARD W., JACKEL L. (1989) « Backpropagation Applied to Handwritten Zip Code Recognition », *Neural Computation*, vol. 1, n° 4, p. 541

LE CUN Y., BRIZARD C. (2019) « Quand la machine apprend : la révolution des neurones artificiels et de l'apprentissage profond », ODILE JACOB

MINSKY M., PAPERT S. (1969) « An introduction to computational geometry. » Cambridge tiass., HIT, vol. 479, no 480, p. 104.

McCARTY J., MINSKY M.L., ROCHESTER N. and SHANNON C.E. (1955) « A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence », August 31, *AI Magazine*, 27(4), p. 12. doi: 10.1609/aimag.v27i4.1904.

McCULLOCH W.S., PITTS W. (1943) « A logical calculus of the ideas immanent in nervous activity », *The Bulletin of Mathematical Biophysics* », 5 (4), p. 115.

RENNARD J.-P. (2013) « Réseaux neuronaux – Une introduction », Paris, Vuibert.

ROSENBLATT F. (1958) « The perceptron: A probabilistic model for information storage and organization in the brain. » *Psychological review*, vol. 65, no 6, p. 386.

Annexes

Annexe 1 : Construction d'une base de données morphologiques sur les chiens et les chats

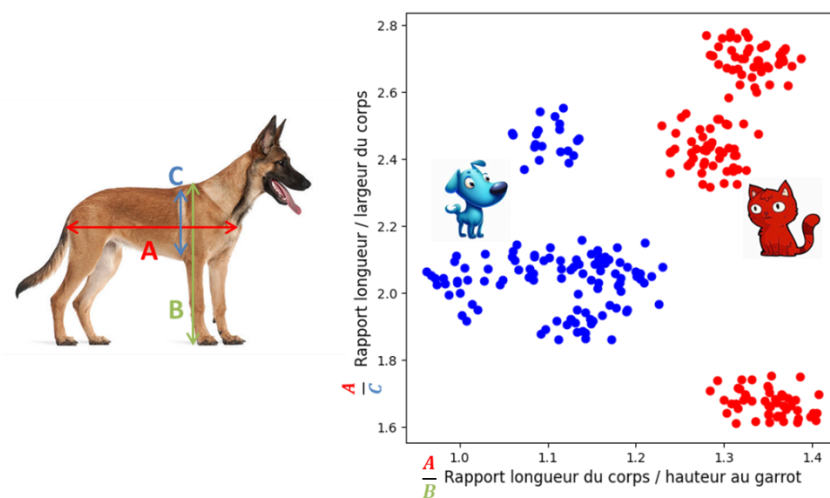
Le web regorge de données concernant les chiens et les chats mais il s'agit principalement de photographies. La capacité de notre perceptron se limitant à quelques paramètres, il nous est impossible de travailler directement avec des images et comme nos recherches ont été infructueuses pour trouver des bases des données numériques concernant la morphologie de ces animaux, nous avons été contraints de créer nos propres données à partir de photographies.

Pour s'affranchir du problème de l'échelle des images récoltées, nous avons choisi d'adimensionner nos données en travaillant sur des rapports de longueurs.

Les races de chien utilisées pour établir notre base de données sont : labrador, doberman, dalmatien, berger allemand, bulldog anglais et chihuahua.

Celles des chats sont les suivantes : siamois, persan, bengal et sacré de Birmanie.

Comme illustré sur la figure ci-après, trois dimensions en pixel ont été relevées sur des images permettant ces mesures : A la longueur du corps, B la hauteur au garrot de l'animal et C la largeur du corps. Comme indiqué dans l'article, le travail sur les ratios A/C et A/B suffit pour obtenir des populations linéairement séparables.



Annexe 2 : Programme du perceptron

Le programme du perceptron donné dans la suite est rédigé dans le langage « python » et a été exécuté dans un notebook jupyter disponible à l'adresse <https://gitlab.xlim.fr/orlianges/perceptron-analogique.git>.

Une première fonction « perceptron » détermine la sortie du perceptron, donc « -1 » pour un chat et « +1 » pour un chien à partir du vecteur d'entrée « X » et de la connaissance des paramètres de celui-ci (vecteur W pour les poids synaptiques et b le biais).

En début du programme poids et biais sont initialisés à la valeur de 0,5.

Comme mentionné dans l'article, c'est le signe du produit « $eti q * activation$ » qui renseigne sur l'erreur commise lors de cette phase d'apprentissage.

Le programme renvoie les paramètres finaux du perceptron lors de son exécution.

```
def perceptron(X, W, b):
    Z = X.dot(W) - b
    return np.where(Z<0, -1, +1)

W = np.array([0.5,0.5]) # initialisation des paramètres
b = 0.5
max_iter = 10
vites_ap = 0.1

for iteration in range(max_iter) :
    test = Data_X[ :,0 :2]
    etiq_test = Data_Etiquette[ :]
    for data, etiq in zip(test, Data_Etiquette) :
        activation = data.dot(W) - b
        if (eti q*activation< =0) :
            W = W + vites_ap*data*eti q
            b - = vites_ap*eti q

W_fin = W
b_fin = b
print("W1 =",W[0]," , W2 =",W[1]," , biais =",b)
```

Il est à noter que le résultat de ce programme n'est pas la valeur optimale des paramètres (qui pourrait être obtenue par une méthode d'ajustement par les moindres carrés par exemple), mais seulement une minimisation « locale » de l'erreur commise lors de l'apprentissage.