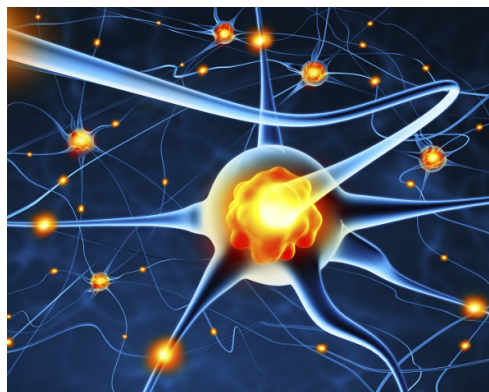


# RAPPORT DE STAGE

## CLASSIFICATION DES CACHALOTS PAR LEURS CLICS

---

AREKION Alexis  
M1 Informatique  
Année universitaire 2019/2020



Université des Antilles  
Laboratoire de Mathématiques Informatique et Application (LAMIA)

# Table des figures

1.1	Figure 1 Schéma de l'organisation interne du LAMIA : <sup>1</sup> Apprentissages Interactions Données. <sup>2</sup> Data analytics and big data gathering with sensors. <sup>3</sup> Mathématiques (analyse variationnelle, analyse numérique, EDP, analyse statistique, mathématiques discrètes). [Pleaseinsertintopreamble]Membres permanents : Suzy Gaucher-Casalis (MCF), Engueran Grandchamp (MCF–HDR), Jean-Luc Henry (MCF), Jimmy Nagau (MCF), Vincent Pagé(MCF), Helene Paugam Moisy (PR), Sébastien Régis (MCF), Céline Rémi (MCF). .	5
2.1	Fonctionnement d'un neurone seul . . . . .	7
2.2	réseau de neurones classique en couches . . . . .	8
2.3	Fonctionnement d'un neurone impulsionnel. <b>à gauche</b> : le neurone $i$ reçoit des impulsions en provenance des entrées $j$ aux instants $s_k$ . <b>à droite</b> : évolution du potentiel de membrane du neurone $i$ en fonction du temps. Ce neurone émettra un spike peu après $s_3$ . . . . .	9
2.4	Architecture de reservoir computing . . . . .	9
2.5	Exemple de rasterplot : On voit que les neurones émettent de façon relativement synchronisée avant de cesser brutalement. . . . .	10
2.6	Nous avons ici un exemple de code exécuté avec colab. . . . .	12
5.1	Architecture scale-free . . . . .	16

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Présentation de la structure d'accueil . . . . .	3
1.1.1	L'université des Antilles . . . . .	3
1.1.2	Le LAMIA . . . . .	4
1.2	Objectifs du groupe Spikes . . . . .	4
1.3	Réseaux de neurones classiques et impulsionnels . . . . .	4
1.4	Problématique : Diffusion de rumeur . . . . .	5
<b>2</b>	<b>Déroulement</b>	<b>6</b>
2.1	Etat de l'art . . . . .	6
2.1.1	Fonctionnement des réseaux de neurones . . . . .	6
2.1.2	Réseaux de neurones impulsionnels . . . . .	7
2.2	Le challenge . . . . .	10
2.3	le travail a distance . . . . .	10
2.3.1	Outils utilisés . . . . .	11
2.3.2	Présentation de GitHub . . . . .	11
2.3.3	Présentation de Google Colab . . . . .	11
2.3.4	Présentation de LaTeX . . . . .	11
2.4	Analyse des données . . . . .	12
2.5	data augmentation et traitement du signal . . . . .	12
2.5.1	Data augmentation . . . . .	12
2.5.2	Traitement du signal . . . . .	12
2.6	difficultés rencontrées . . . . .	12
<b>3</b>	<b>difficultesrencontres</b>	<b>13</b>
3.1	bilan . . . . .	13
<b>4</b>	<b>bilan</b>	<b>14</b>
<b>5</b>	<b>Conclusion</b>	<b>15</b>
5.1	Conclusion de l'étude . . . . .	15
5.2	Piste d'amélioration . . . . .	15
5.2.1	Changer architecture du réseau (Barabasi) . . . . .	15
5.2.2	Réinterprétation de l'expérience . . . . .	15
5.3	Les apports du stage . . . . .	15
5.3.1	les apports au monde . . . . .	15
5.3.2	les apports personnels . . . . .	15

# Chapitre 1

## Introduction

### 1.1 Présentation de la structure d'accueil

Durant la période de mon stage , j'ai été accueilli au **Laboratoire de Mathématiques Informatique et Application (LAMIA)** de l'Université des Antilles (UA).

Pour présenter cette structure, il me faut tout d'abord présenter l'université à laquelle il est rattaché.

#### 1.1.1 L'université des Antilles

Bien que ce soit l'université dans laquelle j'ai fait toutes mes études, voici quelques chiffres que je ne connaissais pas et qui donnent la mesure de sa taille :

L'Université des Antilles s'organise autour deux pôles universitaires régionaux autonomes : le « Pôle Guadeloupe » et le « Pôle Martinique ».

Sur ces pôles, l'Université assure des missions d'*enseignement* et de *recherche*, assistées par des *administratifs et des techniciens*.

#### Administration et personnel technique

L'UA emploie 414 Administratifs et Techniciens (environ 200 personnes pour l'administration centrale et 100 répartis sur chaque pôle)

#### Enseignements

L'UA délivre des diplomes de la licence au doctorat dans de nombreux domaines. Au total, cela représente :

- 484 enseignants-chercheurs (environ 240 pour chaque pôle)
- 12 000 étudiants (environ 7000 pour la Guadeloupe , 5000 pour la Martinique)

Pour l'informatique, cela représente : - autour de 20 enseignants-chercheurs - autour de 120 étudiants

#### Recherche

La recherche est structurée en laboratoires auxquels sont rattachés les enseignants chercheurs qui peuvent former de futurs chercheurs : les doctorants.

L'université compte ainsi au total :

- 17 laboratoires
- 320 doctorants

Pour ma part, comme signalé précédemment, j'ai effectué mon stage dans le laboratoire LAMIA que je vais maintenant présenter.

### 1.1.2 Le LAMIA

Le **Laboratoire de Mathématiques Informatique et Application (LAMIA)**, comme son nom l'indique, se concentre sur les recherches en informatique et mathématiques.

Il compte une soixantaine de membres (Professeurs des Universités, Maîtres de Conférences, ATER, Doctorants) répartis sur deux pôles (Guadeloupe et Martinique) au sein de trois équipes internes :

- Equipe **Mathématiques** (analyse variationnelle, analyse numérique, EDP, analyse statistique, mathématiques discrètes) ;
- Equipe Informatique **DANAIS** : Data analytics and big data gathering with sensors ;
- Equipe Informatique **AID** : Apprentissages Interactions Données ;

De plus, le LAMIA accueille en son sein un groupe de chercheurs associés travaillant en Epidémiologie clinique et médecine.

L'équipe avec laquelle il m'a été donné de travailler principalement est celle d'**Apprentissages Interactions Données** qui développe des méthodes de traitements et d'analyse de données hétérogènes : images (classique, multi-spectrale), séquences vidéos, séries temporelles et spatio-temporelles, dont la responsable est **Mme. Hélène Paugam-Moisy**.

Indépendamment de ces équipes, les travaux de recherche du laboratoire se répartissent en **projets** qui peuvent réunir des membres de plusieurs équipes en **groupes de travail**. Mon stage était en fait plus attaché à un projet et un groupe de travail qu'à une équipe.

Ce projet est nommé de façon informelle projet "**Spikes**" et concerne l'utilisation de **réseaux de neurones impulsionnels** pour l'apprentissage automatique (ces notions seront définies plus loin). Le groupe de travail associé réunit à l'heure actuelle :

- 1 Professeur des Universités
- 2 MCF avec HDR
- 3 MCF
- 1 ingénieur d'études.

Le groupe accueille actuellement deux stagiaires (dont moi) , un Master 2 informatique et un Licence 3 informatique.

C'est avec ces personnes que j'ai travaillé tout au long du stage et mon tuteur de stage était **Mr. Vincent PAGÉ**.

Ci dessous, un schéma présentant la structure du laboratoire mon rattachement à cette structure. (L'équipe de travail **Spikes** étant informelle, elle ne figure pas sur ce schéma.)

La prochaine section sera consacrée à la présentation de la thématique de recherche du groupe "Spikes" et de mon stage.

## 1.2 Objectifs du groupe Spikes

Le groupe **Spikes** s'intéresse aux techniques d'**Intelligence Artificielle**, plus spécifiquement à l'**apprentissage automatique** dont l'objectif est de créer des programmes capable d'apprendre à partir de bases d'exemples.

Actuellement, parmi les techniques permettant l'apprentissage automatique , une se démarque et est très populaire : les **réseaux de neurones artificiels** , notamment dans leur version *profonde* qui sont très utilisés par exemple par **Facebook™** pour sa **reconnaissance faciale** ou encore par **Google™** pour ses **robots** qui apprennent par **répétitions** à jouer (échecs, Go).

## 1.3 Réseaux de neurones classiques et impulsionnels

Ces réseaux de neurones artificiels sont nés dans les années 1960 et les techniques d'apprentissage arrivent à maturité seulement maintenant, avec des résultats très impressionnants.

Bien que basés de façon lointaine sur des neurones biologiques, ils sont en fait très peu plausibles biologiquement (les neurones échangent des chiffres entre eux, les informations se rétropropagent. . . )

A l'inverse, une autre génération de réseaux de neurones, dits **réseaux de neurones impulsionnels** fait l'objet de nombreuses recherches. Nous reviendrons sur le fonctionnement de ces réseaux plus loin. Disons simplement que les neurones n'échangent plus des chiffres mais des impulsions électriques datées, qui se propagent dans le réseau et en déclenchent d'autres.

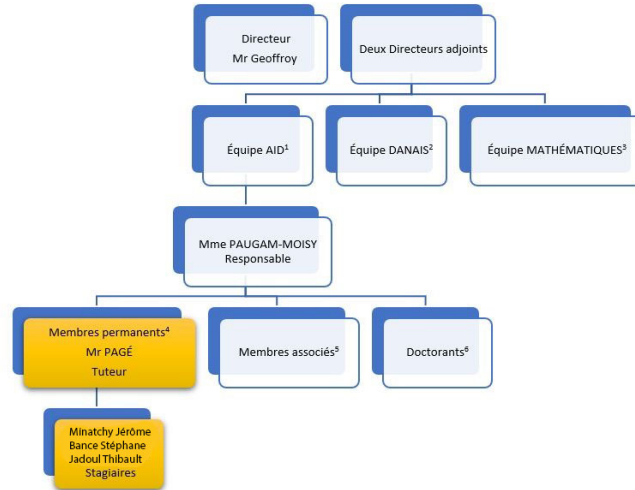


FIGURE 1.1 – Figure 1 Schéma de l'organisation interne du LAMIA : <sup>1</sup>Apprentissages Interactions Données. <sup>2</sup>Data analytics and big data gathering with sensors. <sup>3</sup>Mathématiques (analyse variationnelle, analyse numérique, EDP, analyse statistique, mathématiques discrètes). <sup>4</sup>Membres permanents : Suzy Gaucher-Casalis (MCF), Enguerran Grandchamp (MCF–HDR), Jean-Luc Henry (MCF), Jimmy Nagau (MCF), Vincent Pagé(MCF), Helene Paugam Moisy (PR), Sébastien Régis (MCF), Céline Rémi (MCF).

Ces réseaux de neurones seraient plus proches des neurones biologiques et potentiellement capable de meilleurs résultats que les réseaux classiques. Néanmoins, à l'heure actuelle, les performances en apprentissage de ces réseaux de neurones sont beaucoup plus faibles que celles des réseaux de neurones classiques.

L'activité de recherche du groupe **Spikes** consiste à expérimenter de nouvelles techniques d'apprentissage pour ces **réseaux de neurones impulsionnels** dans le but de contribuer à les améliorer.

## 1.4 Problématique : Diffusion de rumeur

Mon travail consistait dans l'étude de la diffusion de rumeur dans un réseau social. C'est un sujet très actuel et passionnant. En effet, une meilleure connaissance des mécanismes de transmission permettrait par exemple de lutter contre la diffusion de *fake news* ou de promouvoir la propagation d'informations importantes.

Une étude datant de 2015 [?] montre que les réseaux de neurones impulsionnels pourraient permettre de modéliser la diffusion de rumeur.

Cette application des réseaux de neurones impulsionnels a semblé très intéressante au groupe **Spike-Train**. Elle s'accordait en effet bien avec les tests actuels de ce groupe sur le simulateur de réseaux de neurones **Bindsnet**.

Dans ce cadre, mon stage consistait donc à essayer de reproduire les expériences menées dans cet article sur le simulateur *Bindsnet*.

Tous ces concepts m'étant inconnus avant ce stage, il m'a fallu découvrir à la fois les réseaux de neurones impulsionnels, la diffusion de la rumeur ainsi que la façon dont ces réseaux peuvent modéliser la diffusion de rumeur. Ce sont tous ces points qui feront l'objet de la section suivante.

# Chapitre 2

## Déroulement

La première partie sera consacrée à la découverte théorique des réseaux de neurones artificiels classiques, qui me permettra plus facilement de présenter les réseaux de neurones impulsionnels.

### 2.1 Etat de l'art

#### 2.1.1 Fonctionnement des réseaux de neurones

La plupart de mes connaissances sur le sujet vient des présentations que nous ont faites les membres du groupe **SpikeTrain**, ainsi que d'un livre, co-écrit par le Pr. Paugam Moisy [?]. En voici donc un rapide résumé.

##### Neurone artificiel classique

Tout d'abord avant de parler de réseaux de neurone à spike il serait de bon ton d'expliquer le principe du réseau de neurone classique, explication qui commencera par expliquer le le principe du neurone unique.

Un neurone artificiel est pourvu d'un certain nombre d'**entrées**. Dans le cas des neurones classiques, ces entrées sont des nombres réels. Le neurone calculera, en fonction de ces entrées, une unique valeur en **sortie**. Détaillons la façon dont ces calculs sont effectués :

Chacune de ces entrées circule sur une connection, laquelle est caractérisée par un **poids** qui définit l'importance de l'entrée pour le neurone.

Le neurone calcule dans un premier temps la somme de ses entrées, pondérée par leurs poids respectifs, à laquel vient s'ajouter un **biais** spécifique à chaque neurone (cf. equation 2.1).

$$y = \sum_i^n w_i \times x_i + b \quad (2.1)$$

Le résultat de cette somme passe alors dans une **fonction d'activation** qui permet d'introduire une non-linéarité dans les calculs. La sortie  $s$  du neurone est donc calculée conformément à l'équation 2.2

$$s = f\left(\sum_{n=0}^{n_x} x_n w_n + b\right) \quad (2.2)$$

Un schéma reprenant ces explications est présenté dans la figure ?? :

Notre apprentissage ce fera en modifiant les poids de ses différentes connexion (et le biais) de façon à obtenir une sortie proche de celle voulu.

##### Réseaux de neurones classiques

Les neurones présentés précédemment prennent tout leur intérêt lorsqu'ils sont utilisés en groupes, dans des **réseaux de neurones**.

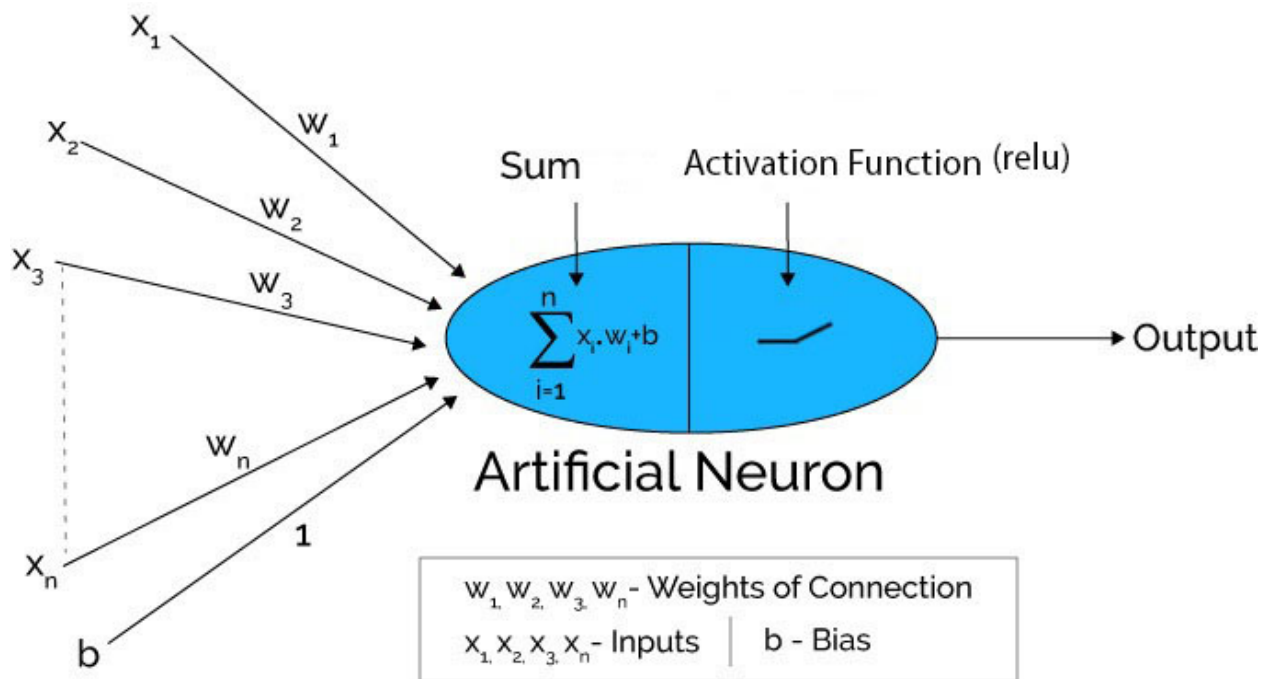


FIGURE 2.1 – Fonctionnement d'un neurone seul

Le premier de ces réseaux, encore utilisé de nos jours, est appelé **perceptron**. Le principe du perceptron n'est pas nouveau et date des années 1960.

Dans ces réseaux, les neurones sont organisés en **couches**. La première couche correspond à celle qui permettra d'introduire des informations dans le réseau (comme la rétine par exemple). Elle est nommée **couche d'entrée**. La dernière couche permettra de lire les décisions du réseau. Elle est appelée **couche de sortie**. Dans les applications classiques, à chaque neurone de la couche de sortie correspond une décision possible et le neurone qui est le plus activé sur la couche de sortie l'emporte. Entre ces couches, on trouve souvent un nombre variable de couches intermédiaires appelées **couches cachées**.

Entre deux couches, on établit le plus souvent un schéma de connexion que nous pouvons qualifier de *full connected*, c-a-d que chaque neurone d'une couche est connecté avec chaque neurone de la couche suivante. Nous allons encore une fois, pour le bien de ce rapport, ne pas épiloguer sur les autres types de connexions existantes.

Nous allons, pour cette partie encore, utiliser une figure (2.2) pour illustrer nos propos :

Voyons maintenant en quoi les réseaux de neurones impulsionnels se distinguent de ces neurones classiques.

### 2.1.2 Réseaux de neurones impulsionnels

Nous allons à présent aborder les neurones à Spikes aussi appelés réseaux de neurones impulsionnel. comme pour les réseaux de neurones classiques, nous allons expliquer le fonctionnement d'un neurone avant de détailler le fonctionnement d'un réseau de neurone impulsionnel.

*Remarque* : Pour le reste de ce rapport nous utiliserons les anglicismes *Spiker*<sup>1</sup>, *Spike*<sup>2</sup>. Ces anglicismes sont en effet couramment utilisés entre chercheurs.

1. Spiker : Émettre une impulsions

2. Spike : l'Impulsion émise par un neurone



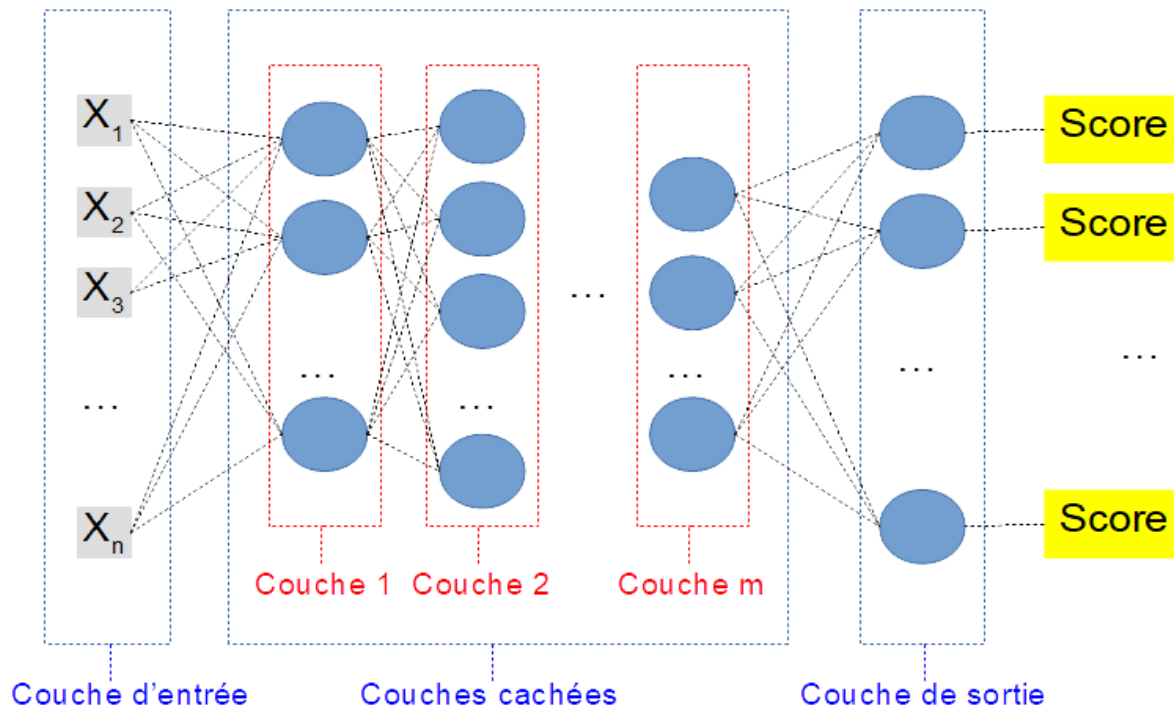


FIGURE 2.2 – réseau de neurones classique en couches

### Neurone à Spikes

Comme les neurones classiques vu en section 2.1.1, un neurone impulsionnel présente des **entrées** pondérées par des **poids** et une **unique sortie**.

En revanche, de façon plus proche du fonctionnement biologique, les neurones n'échangent plus des chiffres mais des **impulsions** (ou **spikes**).

Tous les spikes ont des propriétés similaires et ne se distinguent que par leur instant d'émission. En conséquence, dans un réseau de neurones impulsionnels, l'information n'est portée que par le positionnement temporel des spikes.

*En plus d'être éloignée de la réalité nous avons des calculs lourd et couteux qui sont effectués sur chaque neurone ce qui demande une assez grande puissance de calcul donc une quantité d'énergie proportionnelle.*

Une propriété primordiale du neurone à spike est son **potentiel de membrane** qui représente l'excitation du neurone. En l'absence d'entrées, ce potentiel a une valeur dite **de repos**.

Lorsque qu'un spike arrive au neurone, ce spike déclenche l'augmentation du potentiel de membrane, selon une **équation différentielle**<sup>3</sup>.

Chaque nouveau spike induit ainsi des augmentations du potentiel de membranes proportionnelles aux poids des connections sur lesquelles ils arrivent.

L'équation différentielle induit également qu'en l'absence de nouveau spike, le potentiel tendra à revenir au potentiel de repos.

Si le potentiel de membrane atteint un certain **seuil**, le neurone **émet un spike en sortie**<sup>4</sup>

En conséquence, c'est bien la notion de temps qui détermine si l'on émet un spike ou non. Pour émettre un spike, un neurone doit recevoir un nombre de spike dans un interval de temps suffisamment court pour permettre à son potentiel de membrane d'atteindre le seuil déclenchant l'émission.

3. variable en fonction des modèles de neurones utilisés

4. Après avoir spiké, un neurone devient réfractaire pendant un laps de temps, c-a-d qu'il ignore ses spikes d'entrée.

Mais une fois de plus une image vaut mieux que mille mots :

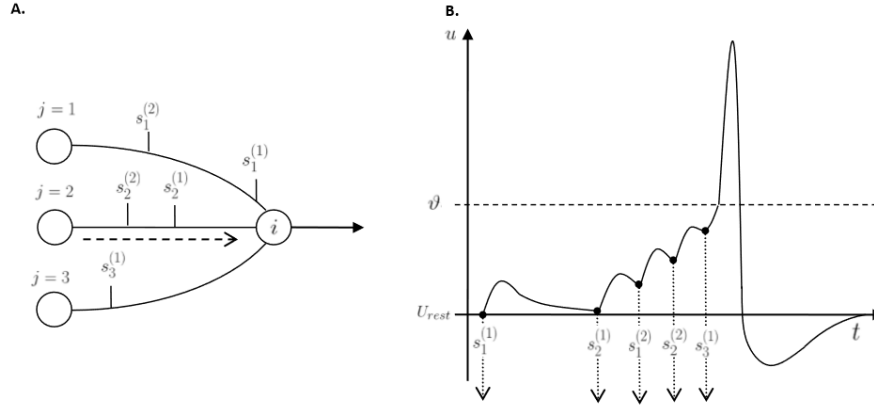


FIGURE 2.3 – Fonctionnement d'un neurone impulsionnel. **à gauche** : le neurone  $i$  reçoit des impulsions en provenance des entrées  $j$  aux instants  $s_k$ . **à droite** : évolution du potentiel de membrane du neurone  $i$  en fonction du temps. Ce neurone émettra un spike peu après  $s_3$

### Reseau de neurones impulsionnels à reservoir

Nous pouvons à présent attaquer les réseaux de neurones à spike.

Comme dans le cas des réseaux de neurones classiques, un réseau impulsionnel comporte le plus souvent une **couche d'entrée**, une **couche de sortie** et une **architecture intermédiaire** entre ces couches.

Le choix de cette architecture intermédiaire varie beaucoup plus que dans le cas des réseaux classiques. Durant ce stage, conformément à l'article de référence [?] qui nous a guidés, nous utiliserons une architecture dite **de Reservoir Computing**, dans laquelle les couches ont disparues et les neurones sont simplement connectés les uns aux autres avec des connections récurrentes, formant un graphe.<sup>5</sup>

Comme pour chaque présentation, nous allons illustrer cette architecture par un exemple sur la figure 2.4

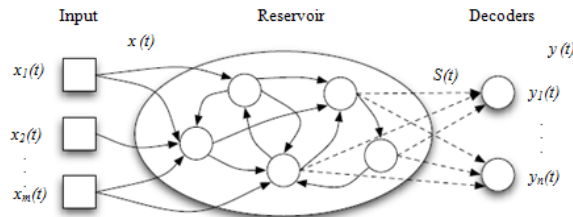


FIGURE 2.4 – Architecture de reservoir computing

Enfin, pour observer l'activité d'un réseau de neurones impulsionnel, on utilise une figure nommée **rasterplot**. Dans un rasterplot, l'axe des abscisses représente le temps et l'axe des ordonnées représente l'indice des neurones observés. Un point dans un rasterplot tel que celui de la figure 2.5 représente un spike émis par un neurone donné, à un instant donné.

5. Pour les lecteurs plus forcenés, les autres architectures sont présentées en annexe.

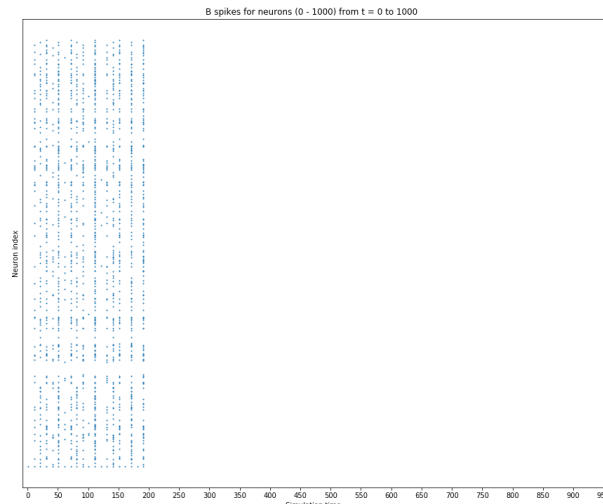


FIGURE 2.5 – Exemple de rasterplot : On voit que les neurones émettent de façon relativement synchronisée avant de cesser brutalement.

## 2.2 Le challenge

Le challenge "Dyini Odontocete Click Classification, 10 species [ DOCC10 ] by Universite de Toulon" (<https://challengedata.ens.fr/challenges/32>) consistait simplement à réaliser un classifieur qui classe les cachalots en une dizaine d'espèces à partir de leurs "Clics" pour cela nous avions à disposition une base d'apprentissage labelisée ainsi qu'une base de test non labelisée sur laquelle nous pouvions évaluer les performances réelles de nos classifieurs. Pour cela nous labelisions les exemples de la base de test puis envoyions nos prédictions sur le site du challenge qui nous renvoyais nos performances ainsi qu'un classement des performances de tous les participants.

Les deux bases sont constituées d'enregistrements audios des clics des différentes espèces de cachalots que je présenterais plus en détail dans la partie analyse des données.

Pour résoudre ce problème j'ai commencé par analyser les données afin -D'avoir une vision claire du problème à résoudre -D'avoir des idées de méthodes de résolution du problème -Comprendre d'éventuelles incohérences dans nos futurs résultats (notamment des différences de performances entre les deux bases)

Ensuite nous avons traité les données de différentes manières notamment en faisant de la data augmentation et du traitement du signal.

Puis M. Clergue se chargeait de tester un multitude de techniques d'apprentissage sur les données ainsi traitées.

M. Page nous a également fournis un grand nombre de petits programmes et de fonctions parfois simples parfois complexes dont nous avons besoin afin de réaliser nos différentes tâches.

## 2.3 le travail à distance

Comme vous le savez certainement durant cette année 2020 nous avons été touchés par la crise du coronavirus qui nous a conduit à être confinés nous forçant à travailler uniquement à distance.

Ces circonstances très particulières ont grandement affecté notre travail particulièrement au début où nous avons du régler de nombreux problèmes techniques et organisationnels. Cependant en nous forçant à nous adapter à ces nouvelles conditions, cette crise nous a permis de grandement augmenter nos compétences en "télétravail".

Ainsi malgré des débuts léthargiques nous avons mis en place une "routine de travail" qui était la suivante : -Des visio-conférences quotidiennes nous permettant d'organiser et de synchroniser notre travail -Un groupe whatsapp dédié à mon stage afin de communiquer le plus efficacement possible -Un Github privé dédié afin de partager l'ensemble du projet -Un partage régulier de google collab via google drive

### 2.3.1 Outils utilisés

### 2.3.2 Présentation de GitHub



Nous pouvons définir GitHub comme une plateforme de développement de projet informatique en groupe. Elle simplifie grandement le développement de projets. Elle permet de versionner ses programmes et d'y apporter des modifications en temps réel à plusieurs.

#### Pourquoi Github

Car cela permet une certaine synergie avec nos autres outils que nous verrons plus tard. Cette plateforme permet une facilité de développement de par sa fonctionnalité de versionnage de notre code à chaque changement ce qui permet une mise à jour dynamique ainsi qu'une relative facilité à retourner à un état antérieur de notre programme ce qui permet une facilité de débogage. Nous pouvons d'ailleurs dire que ce rapport est entreposé sur Github et qu'il peut être récupéré facilement. Cette plateforme est aussi très connue dans le monde de la programmation ce qui sera utile pour notre future profession.

### 2.3.3 Présentation de Google Colab



Colab peut-être défini comme étant une plateforme d'exécution pour notre code il permet du fait que ce soit la puissance de calcul d'ordinateur gérée par Google une vitesse d'exécution ainsi qu'une vitesse de téléchargement de base de données supérieure à celle qui nous est disponible en local.

#### Pourquoi Colab

En premier lieu pour la facilité d'exécution du code car ce n'est pas en local ce qui permet une exécution quasi immédiate du code sans aucune installation. Il est aussi facile de mettre sur github du code produit avec colab car ces deux plateformes sont liées. Il permet de par l'utilisation du format Jupyter de mélanger code et texte (peut aussi comporter des images) dans notre notebook.

Voilà un exemple d'exécution avec colab :

### 2.3.4 Présentation de LaTeX

Nous pouvons dire que LaTeX est un langage de traitement de texte tel que le markdown qui permet de mettre en forme notre texte de manière scientifique cela veut dire que LaTeX permet une facilité d'écriture des équations et de toutes les écritures mathématiques. Permet de par ses nombreux packages une quasi-infinité de possibilités.

#### Pourquoi Github

Cela permet encore une fois une synergie entre nos différents outils car LaTeX peut-être utilisé avec un simple bloc-note c'est donc du texte ce qui permet une interaction facilitée avec GitHub d'ailleurs ce rapport est écrit avec LaTeX et retrouvable sur GitHub.

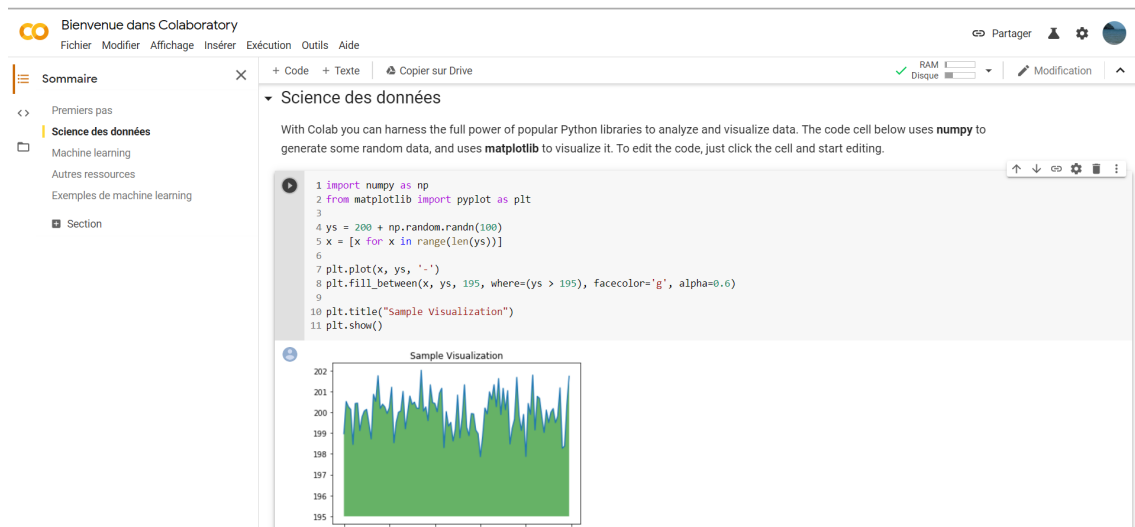


FIGURE 2.6 – Nous avons ici un exemple de code exécuté avec colab.

# L<sup>A</sup>T<sub>E</sub>X

## 2.4 Analyse des données

## 2.5 data augmentation et traitement du signal

### 2.5.1 Data augmentation

### 2.5.2 Traitement du signal

## 2.6 difficultés rencontrées

## Chapitre 3

# difficultesrencontres

test

### 3.1 bilan

## Chapitre 4

### bilan

# Chapitre 5

## Conclusion

### 5.1 Conclusion de l'étude

Nous avons tout d'abord vu avec notre modèle de simulation qu'il fallait tout d'abord limité l'influence de l'aléatoire de notre simulations

### 5.2 Piste d'amélioration

#### 5.2.1 Changer architecture du réseau (Barabasi)

Nous avons envisagé de changer la **topologie**<sup>1</sup>. Pour passer à une topologie dite **scalefree**. Qui est justement celle utilisé dans l'article d'origine. Cette architecture ressemble à ça :

#### 5.2.2 Réinterprétation de l'expérience

Nous pouvons interpréter tout le réseau comme un utilisateur et cet entretient comme le temps qu'il accorde à la rumeur.

Cela permettrait

### 5.3 Les apports du stage

#### 5.3.1 les apports au monde

Grâce à ce stage les chercheurs du laboratoire auront une idée plus précise de la diffusion d'un echo dans un réservoir et comment y créer un entretient ce qui sera utile pour leurs futures expériences. Cela permettra de mieux appréhender certains problèmes.

#### 5.3.2 les apports personnels

Comme dit pendant mon introduction, avant ce stage je n'avais aucune connaissance des réseaux de neurones, ce stage m'a donc permis de m'ouvrir a ce nouveau sujet passionnant, et m'a permis d'obtenir des compétence nécessaire à mon cursus universitaire. Il m'a permis d'affiné mes méthodes de travail grâce à l'apprentissage de l'utilisation de GitHub et Colab. J'ai amélioré ma rédaction grace à l'approfondissement du LaTeX. Il m'a permis d'affiner mon analyse de par l'analyse de mes résultats.

---

1. Topologie : Tel que la topologie d'un graffe il s'agit de la façon dont sont connecté nos neurones entre eux.





FIGURE 5.1 – Architecture scale-free