

Découverte de l'IA

Utilisation d'un réseau de neurones pour classifier des données.



1

Support : Modèles pré-entraînés de Matlab

Didier TOULOUSE

L'IA en quelques dates

1943 : Warren McCullough et Walter Pitts invente le 1^{er} modèle mathématique pour la création d'un réseau de neurones.

1950 : Alan Turing propose le **test de Turing**

1956-1974 (L'âge d'or) : à Dartmouth College, début officiel de la recherche en IA.

1957 : John McCarthy invente le terme "**intelligence artificielle**".

1974-1980 (Le premier hiver de l'IA) : baisse des financements, premiers doutes sur l'IA

1980-1987 (Le boom) : Arrivée des systèmes experts, arrivée des ordinateurs de 5^{ème} génération

1987-1993 (La crise : le second hiver de l'IA) : Chute des financements, la recherche sur l'IA ralentit.

1993 : Regain de l'IA grâce à l'émergence de nouvelles technologies, telles que les réseaux de neurones artificiels et les algorithmes d'apprentissage automatique.

1997 : **Deep Blue d'IBM** a battu le champion du monde d'échecs **Garry Kasparov**

2000 : L'IA dans les applications commerciales, (reconnaissance vocale et l'analyse de données)

2010 : L'IA se développer à la vitesse V, utilisée dans la médecine, la finance et la robotique.

2020 : **Les IA génératives arrivent.** ChatGPT 3.0 est publié. Il s'agit d'un modèle de langage génératif de 175B de paramètres, formé sur un ensemble de données de texte et de code encore plus important.

Sommaire

1

Les types d'IA

p17

2

Les modes d'apprentissage

p18

3

Exemple1 de Machine learning

p19

4

Exemple2 de Machine learning

p 22

5

Le réseau de neurones

p 26

Les 3 types d'intelligence artificielle

IA Faible/Etroite

Assistants vocaux

Systèmes de recommandation

Reconnaissance d'images

Filtres de spam

Systèmes de navigation

IA Générale/Forte

Reconnaissance d'images et de vidéos

Traduction automatique

Reconnaissance vocale et synthèse vocale

Véhicules autonomes

Systèmes de recommandation

Génération de texte

Analyse de données médicales

Jeu vidéo et échecs

Création d'art

Détection de la fraude

Superintelligence artificielle (ASI)

Les modes d'apprentissage

Classification

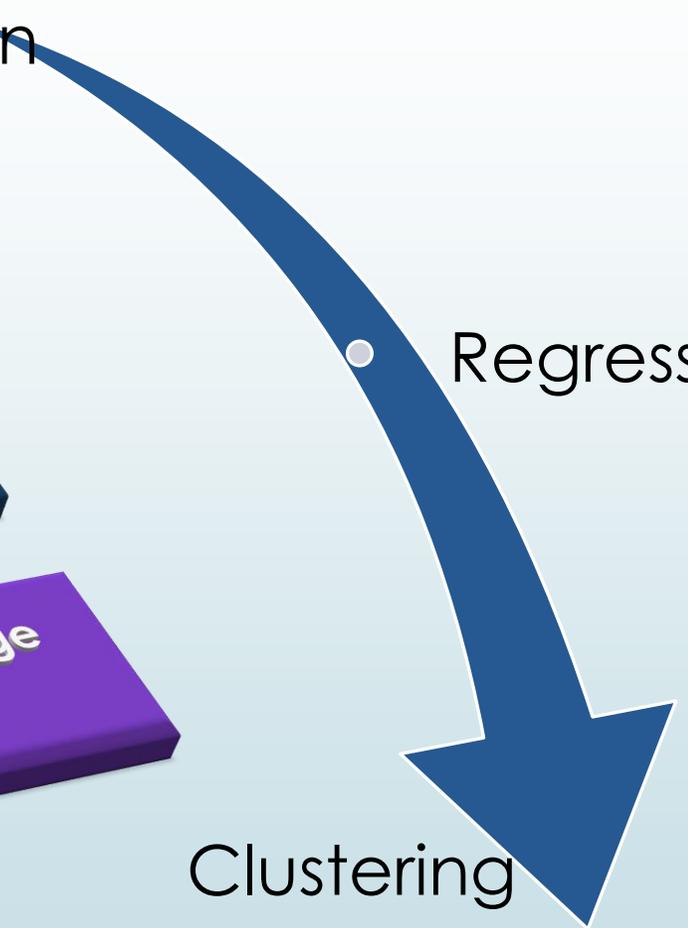
Machine Learning (Apprentissage automatique supervisé ou non)

Deep Learning (apprentissage profond)

Reinforcement learning (apprentissage renforcé)

Regression

Clustering



Apprentissage automatique supervisé



Inputs

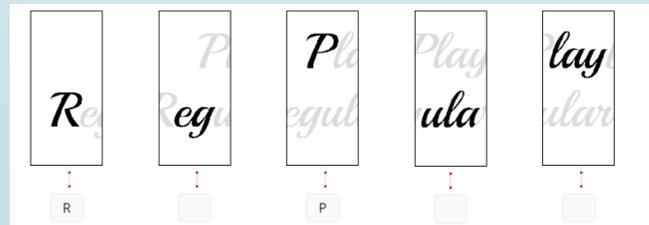
Base de données Images de personnes



- La reconnaissance faciale



- La reconnaissance de caractères



Base de données Images de caractères



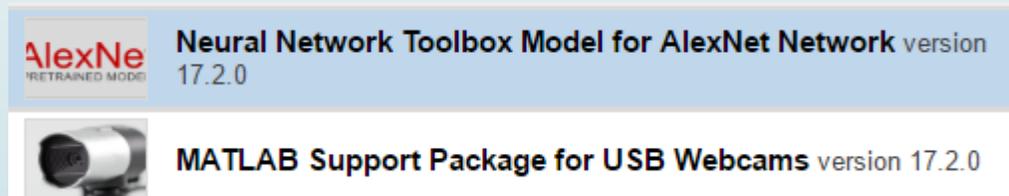
1 Machine learning supervised

Modèle
déjà
élaboré

Étapes
sous
Matlab

Identifier des objets à l'aide d'une webcam et d'une base de données. **AlexNet** est un réseau neuronal de convolution pré-entraîné (CNN) formé sur plus de 60 millions d'images et capable de classer les images en 1 000 catégories d'objets (par exemple, clavier, souris, tasse à café, crayon et de nombreux animaux).

*Installer les supports package:



*Tester les supports package:

Webcamlist

camera=webcam par exemple

puis éditer dans la fenêtre de commandes:

Machine learning supervised

Etapes
sous
Matlab

```
net=alexnet
while true
    im = snapshot(camera);
    image(im);
    im = imresize(im,[227 227]);
    label = classify(net,im);
    title(char(label));
    drawnow
end
```

% Chargement d'un CNN

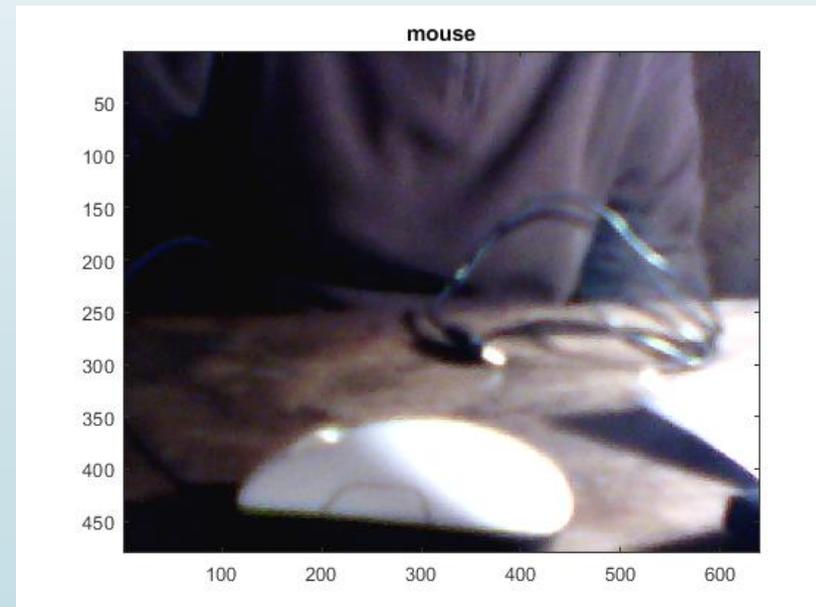
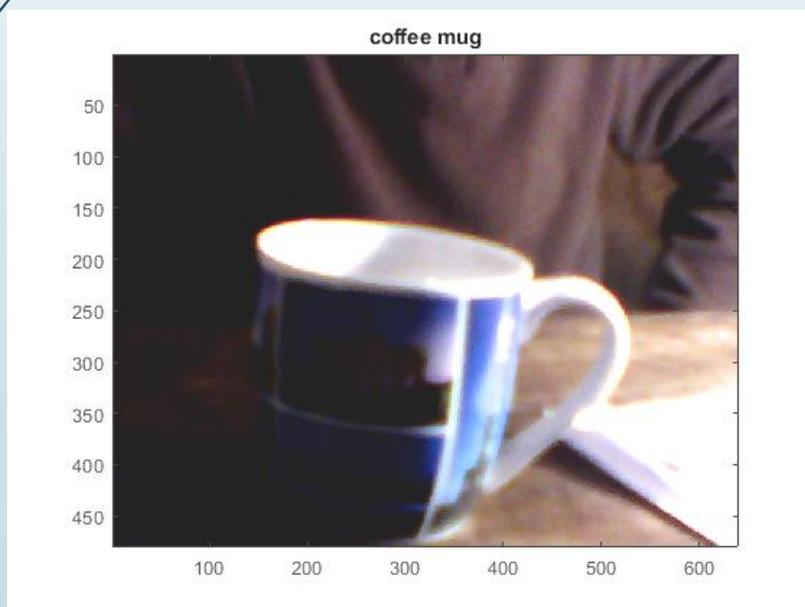
% Acquérir une image

% Afficher l'image

% Redimensionner l'image

% Classifier l'image

% Afficher la classe



2 Classification Learner

Exemple 2
matlab

*Ouvrir classification learner dans APPS

%espace commandes

*fishertable = readtable("fisheriris.csv");

*Vérifier que la variable a bien été créée dans le Workspace

*Lancer une nouvelle session

Dans le menu déroulant Data set variable le nom de votre table apparaît

*Cliquez sur le bouton Start Session

Réponse
(espèce)

Prédicteurs

Data set

Data Set Variable: fishertable (150x5 table)

Response

From data set variable
 From workspace

Species (cell, 3 unique)

Predictors

	Name	Type	Range
<input checked="" type="checkbox"/>	Sepal.Length	double	4.3 .. 7.9
<input checked="" type="checkbox"/>	Sepal.Width	double	2 .. 4.4
<input checked="" type="checkbox"/>	Petal.Length	double	1 .. 6.9
<input checked="" type="checkbox"/>	Petal.Width	double	0.1 .. 2.5
<input type="checkbox"/>	Species	cell	3 unique

Add All Remove All

[How to prepare data](#) Refresh

Validation

Validation Scheme: Cross-Validation

Protects against overfitting. For data not set aside for testing, the app partitions the data into folds and estimates the accuracy on each fold.

Cross-validation folds: 5

[Read about validation](#)

Test

Set aside a test data set

Percent set aside: 10

Use a test set to evaluate model performance after tuning and training models. To import a separate test set instead of partitioning the current data set, use the Test Data button after starting an app session.

[Read about test data](#)

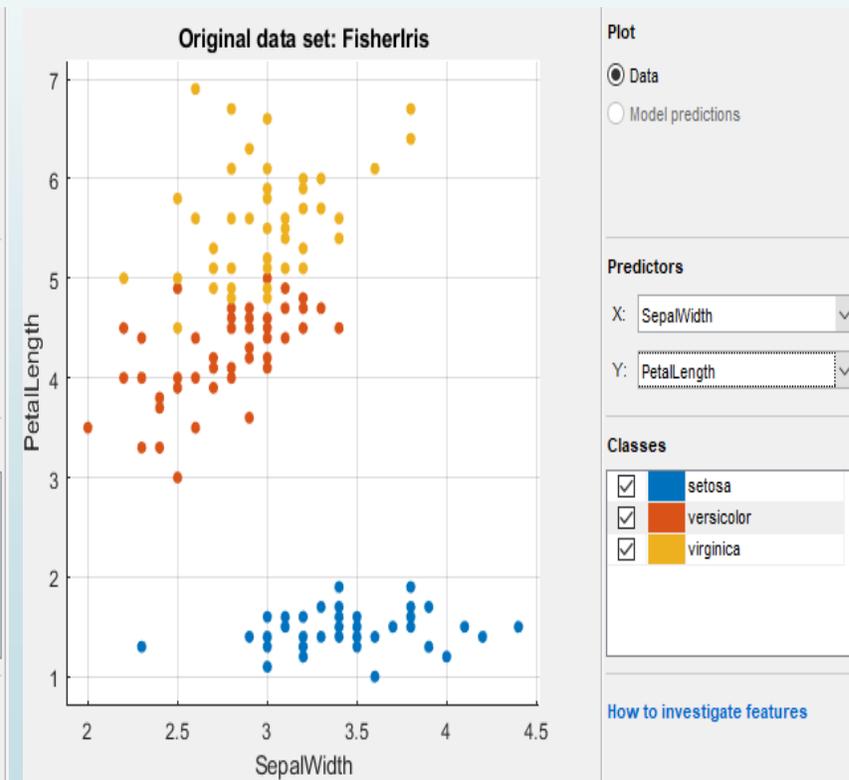
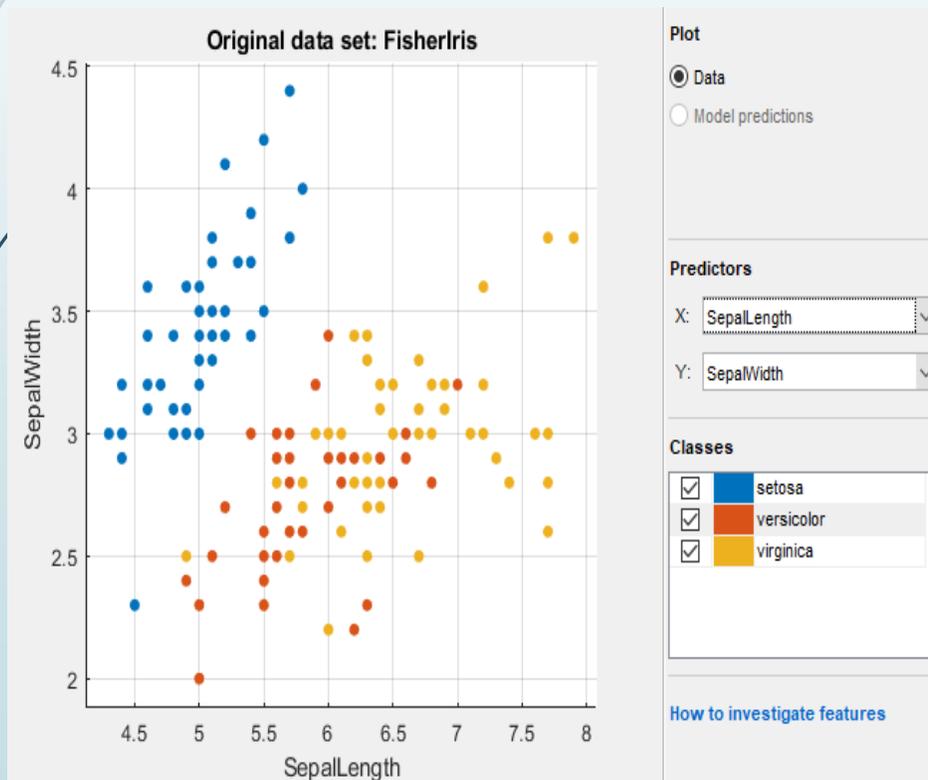
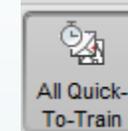
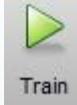
Start Session Cancel

2 Classification Learner

Classifier
une base de
données
existante

*Lancer la session , changer de prédicteur pour séparer les espèces

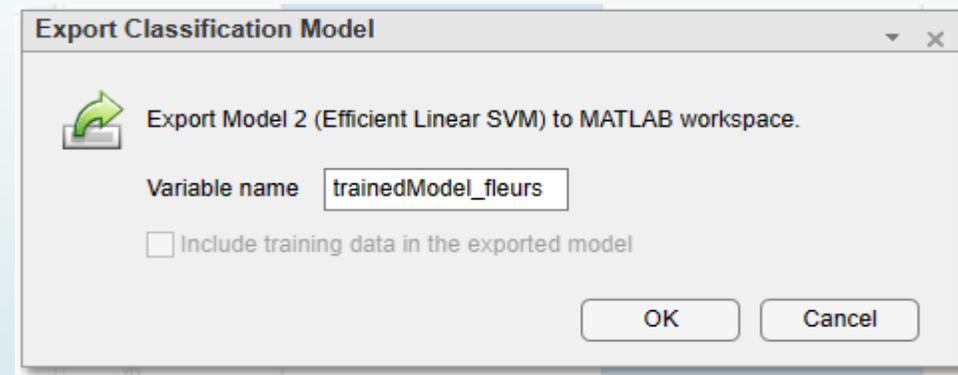
*Faire tourner plusieurs modèles pour en choisir un



2 Classification Learner

Tester le
modèle
choisi

- *Exporter le modèle choisi
- *Choisissez un nom
- *Valider par Ok



*Créer une nouvelle entrée:

```
V2=[5.3 3.6 1.4 0.2]
```

```
T2=array2table(V2,'variableName',{'SepalLength','SepalWidth','Petal  
Length','PetalWidth'})
```

2 Classification Learner

*Tester le modèle :

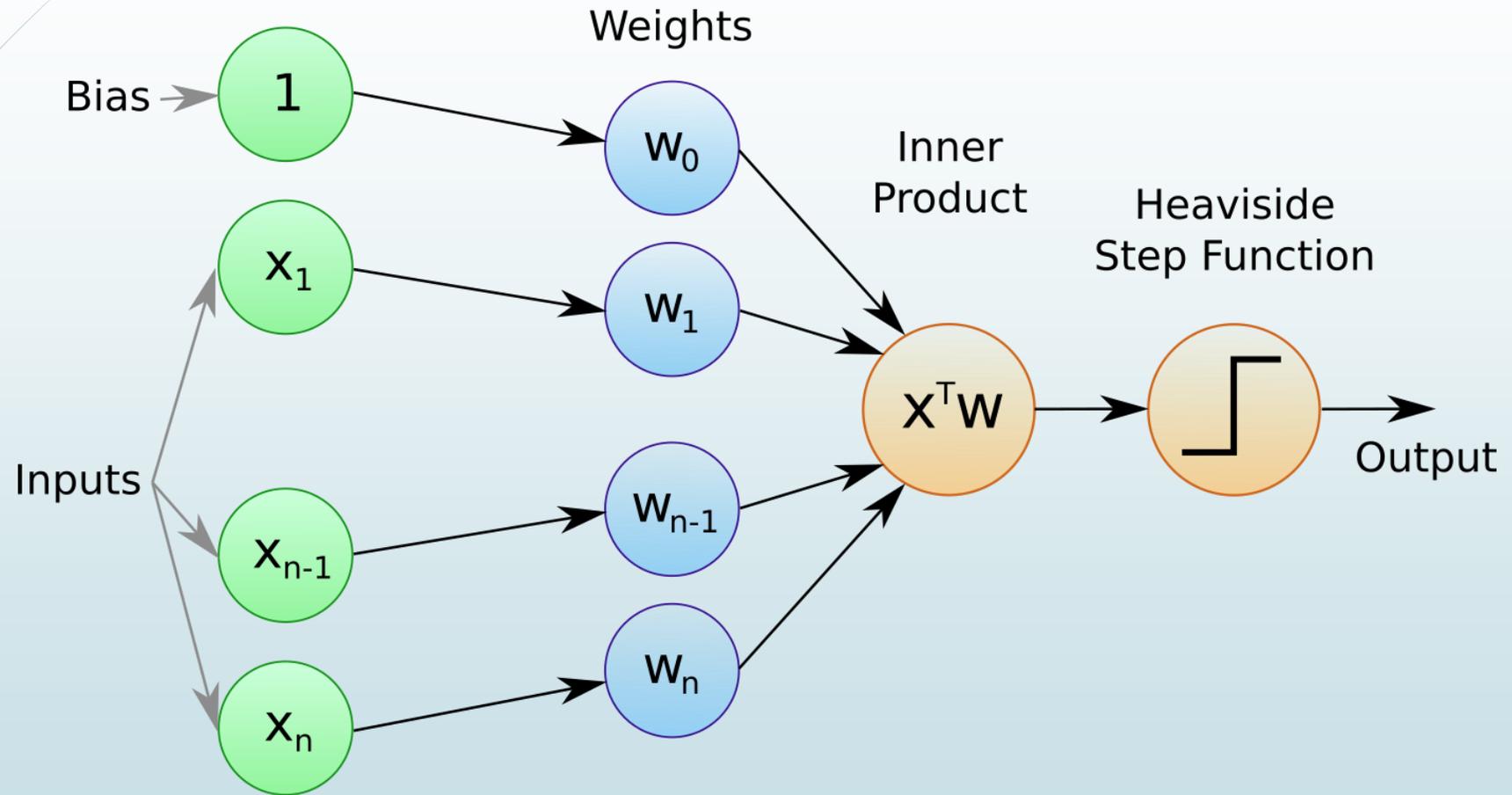
```
yfit=trainedModel_fleurs.predictFcn(T2)
```

*Réponse :

```
categorical : setosa
```

```
V2 =  
    5.3000    3.6000    1.4000    0.2000  
  
T2 =  
  
1x4 table  
  
    SepalLength    SepalWidth    PetalLength    PetalWidth  
    _____    _____    _____    _____  
         5.3         3.6         1.4         0.2  
  
>> yfit=trainedModel_fleurs.predictFcn(T2)  
  
yfit =  
  
1x1 cell array  
  
    {'setosa'}
```

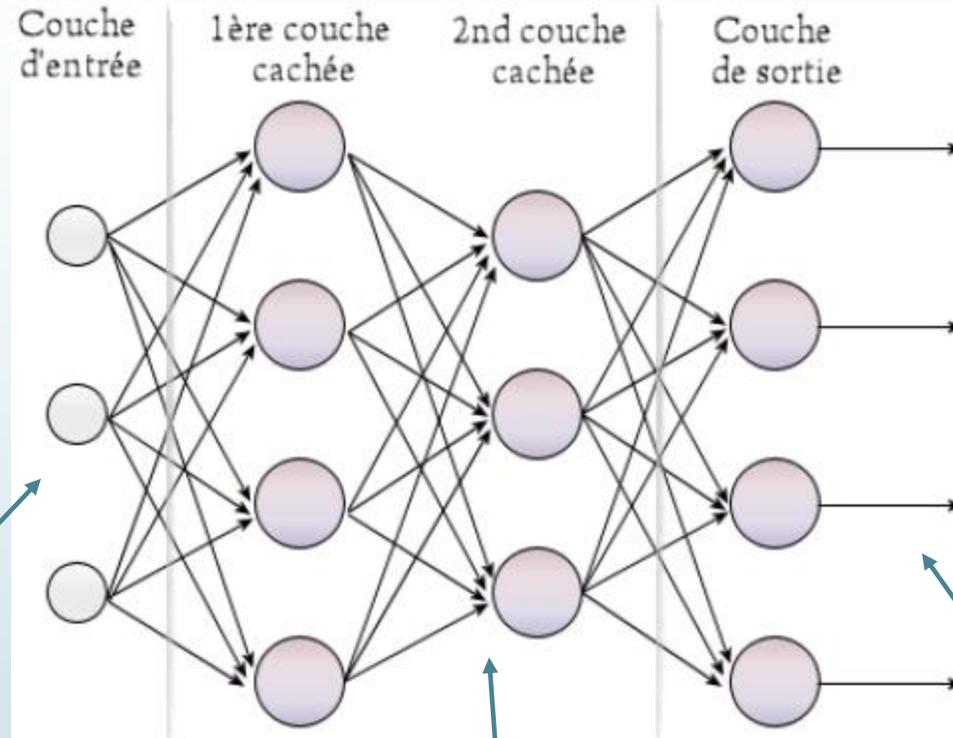
Le perceptron



Les réseaux de neurones

Neuronal Network

Réseau
de
neurones



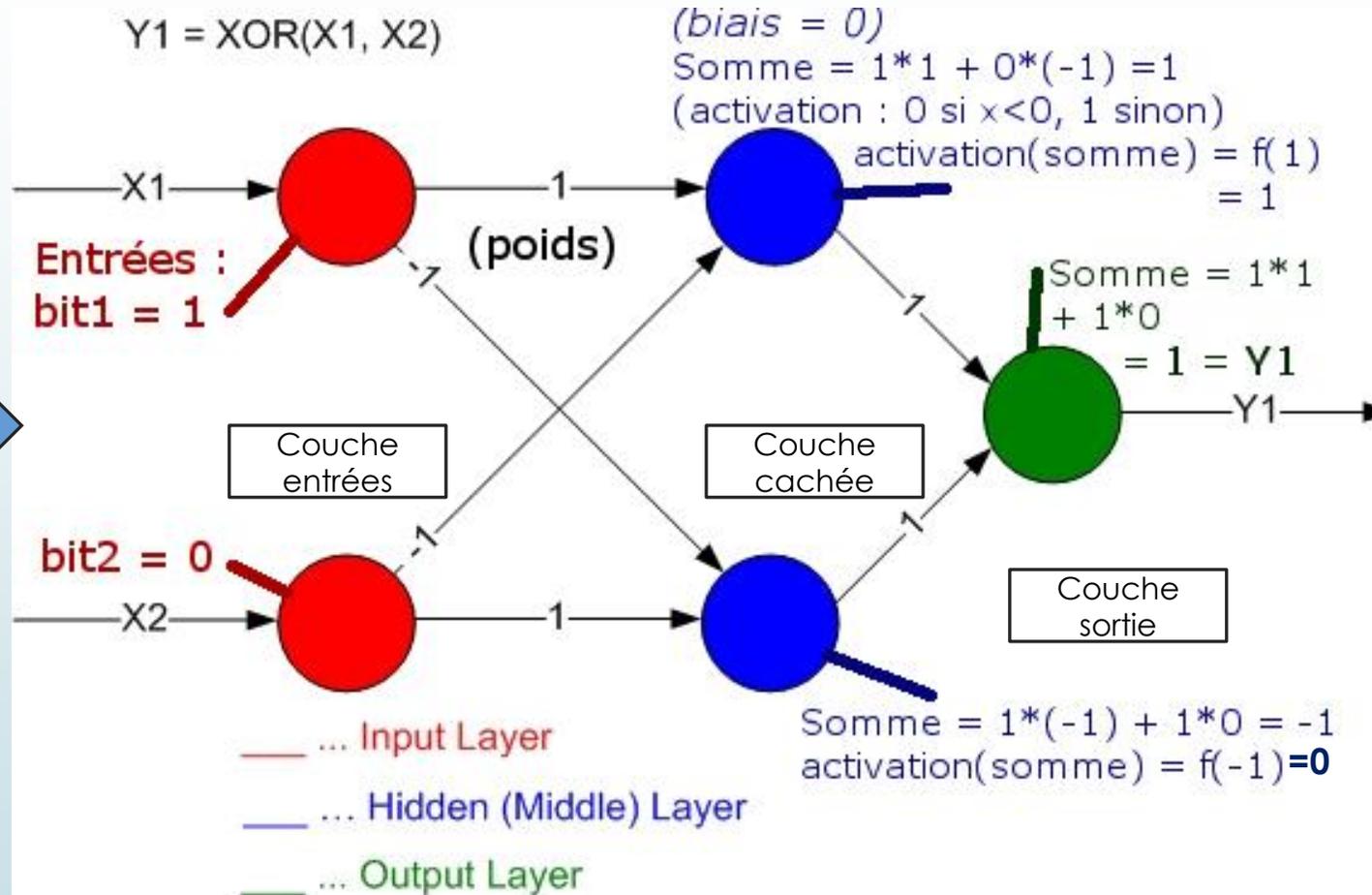
$$y = x$$

$$y = f_{\text{activation}}\left(b + \sum_i w_i \cdot x_i\right)$$

$$y = \sum_i w_i \cdot x_i$$

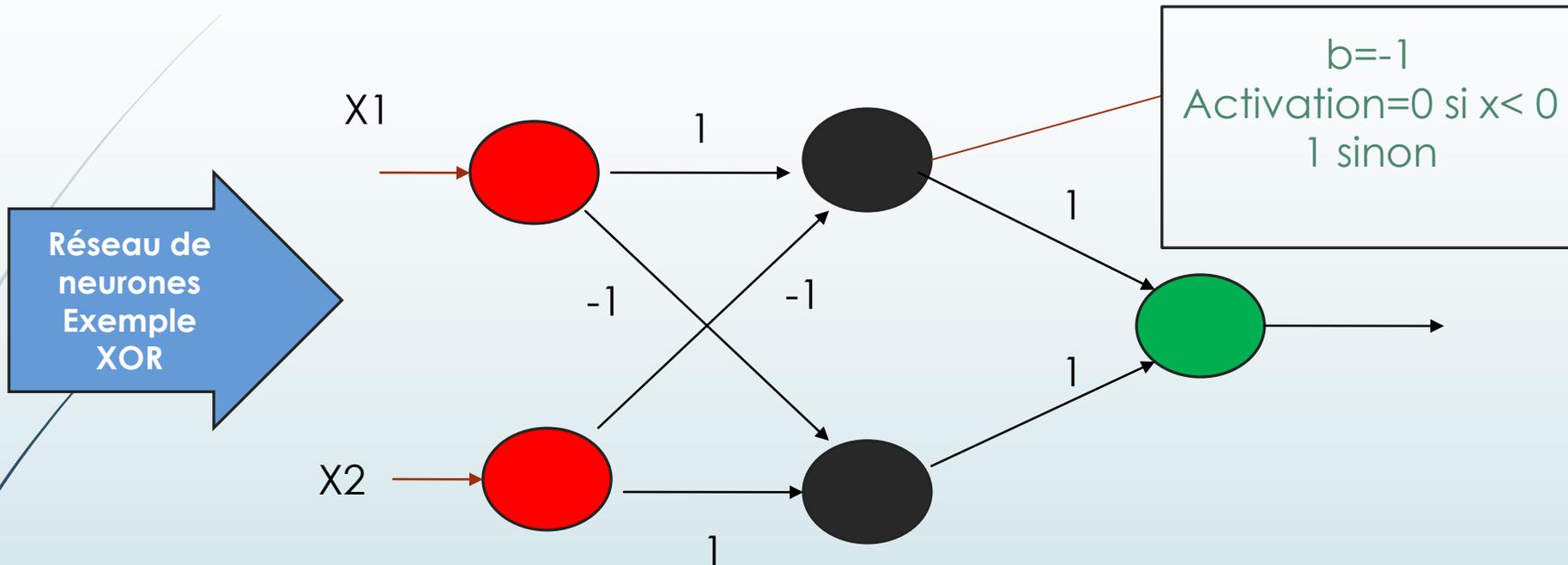
Neuronal Network

Réseau de neurones exemple XOR



Vérifier que pour que cela fonctionne avec toutes les combinaisons de $X1$ et $X2$, il faut que biais = -1.

Neuronal Network



$x_1 = x_2 = 1$ donne $y_{1\text{caché}} = y_{2\text{caché}} = 0$ donc $y_{\text{sortie}} = 0$

Merci pour votre attention