

# Simulation 2D de l'équation du transfert radiatif et reconstruction de la densité par un réseau de neurones

**Stagiaire**  
Roussel Desmond NZOYEM

**Enseignant référent**  
Christophe PRUD'HOMME

**Maîtres de stage**  
Emmanuel FRANCK  
Laurent NAVORET  
Vincent VIGON

Année académique 2019/2020

## Introduction

●○○  
○○○○

## Principe et méthode

○○○○○○○○  
○○○○○○○○

## Le problème en 1D

○○○  
○○○

## Le problème en 2D

○○○  
○○○○

## Conclusion

○○○  
○○○○

L'IRMA

# L'IRMA

# L'équipe MOCO

Plusieurs membres parmi lesquels MM. :

- Emmanuel FRANCK
- Laurent NAVORET

Responsables des séminaires en EDP

- Partenariats internationaux (Portugal, Allemagne, USA, etc.);  
Projets Examag Spexxa, MAToS, EUROFUSION
- Partenariats industriels
- Modélisation des plasmas

# L'équipe MOCO

Plusieurs membres parmi lesquels MM. :

- Emmanuel FRANCK
- Laurent NAVORET

Responsables des séminaires en EDP

Des activités diverses :

- Partenariats internationaux (Portugal, Allemagne, USA, etc.);  
Projets Examag Spexxa, MAToS, EUROFUSION
- Partenariats industriels
- Modélisation des plasmas

# L'équipe MOCO

Plusieurs membres parmi lesquels MM. :

- Emmanuel FRANCK
- Laurent NAVORET

Responsables des séminaires en EDP

Des activités diverses :

- Partenariats internationaux (Portugal, Allemagne, USA, etc.);  
Projets Examag Spexxa, MAToS, EUROFUSION
- Partenariats industriels
- Modélisation des plasmas

# L'équipe MOCO

Plusieurs membres parmi lesquels MM. :

- Emmanuel FRANCK
- Laurent NAVORET

Responsables des séminaires en EDP

Des activités diverses :

- Partenariats internationaux (Portugal, Allemagne, USA, etc.);  
Projets Examag Spexxa, MAToS, EUROFUSION
- Partenariats industriels
- Modélisation des plasmas

# L'équipe Probabilités

Plusieurs membres parmi lesquels M. :

■ Vincent VIGON

- Collaborations internationales (Allemagne, Australie, Chine, etc.) en actuariat, transport optimal, matrice aléatoire, etc.
- Séminaire (de calcul) stochastique

# L'équipe Probabilités

Plusieurs membres parmi lesquels M. :

- Vincent VIGON

Des activités diverses :

- Collaborations internationales (Allemagne, Australie, Chine, etc.) en actuariat, transport optimal, matrice aléatoire, etc.
- Séminaire (de calcul) stochastique

# L'équipe Probabilités

Plusieurs membres parmi lesquels M. :

- Vincent VIGON

Des activités diverses :

- Collaborations internationales (Allemagne, Australie, Chine, etc.) en actuariat, transport optimal, matrice aléatoire, etc.
- Séminaire (de calcul) stochastique

Introduction



Principe et méthode



Le problème en 1D



Le problème en 2D



Conclusion



Le sujet du stage

## Le stage et le projet

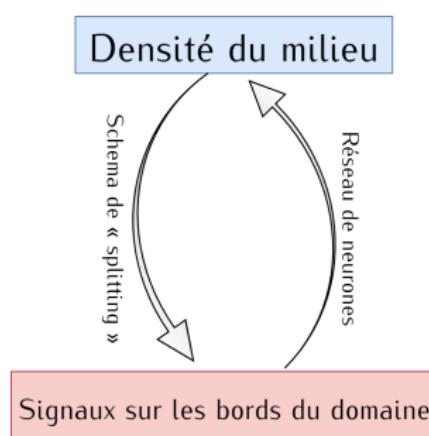
# Le(s) problème(s) à résoudre

## Problème direct

(Résolution de l'ETR par un schéma de  
"splitting")

## Problème inverse

(Reconstruction de la densité par un réseau  
de neurones)



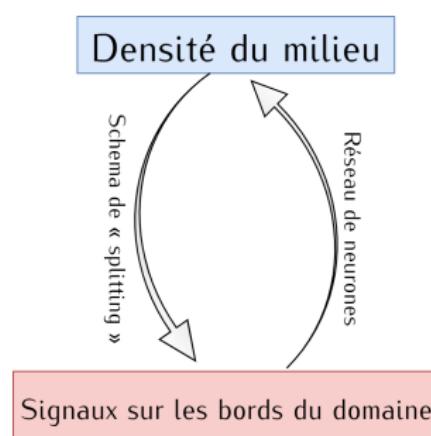
# Le(s) problème(s) à résoudre

## Problème direct

(Résolution de l'ETR par un schéma de  
"splitting")

## Problème inverse

(Reconstruction de la densité par un réseau  
de neurones)



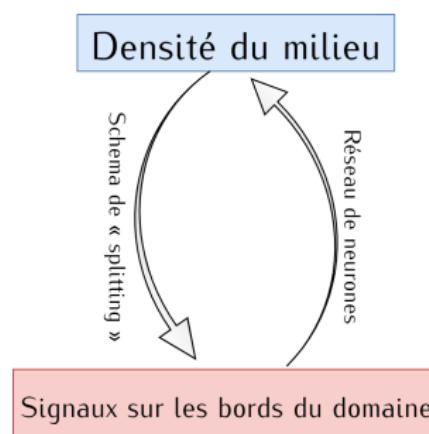
# Le(s) problème(s) à résoudre

## Problème direct

(Résolution de l'ETR par un schéma de "splitting")

## Problème inverse

(Reconstruction de la densité par un réseau de neurones)



# Les points pour situer le stage

- 1 Explosion du Deep Learning
- 2 Application du Machine Learning en imagerie médicale
- 3 Réévaluation des méthodes de résolution de problèmes inverse

# Les points pour situer le stage

- 1 Explosion du Deep Learning
- 2 Application du Machine Learning en imagerie médicale
- 3 Réévaluation des méthodes de résolution de problèmes inverse

# Les points pour situer le stage

- 1** Explosion du Deep Learning
- 2** Application du Machine Learning en imagerie médicale
- 3** Réévaluation des méthodes de résolution de problèmes inverse

# Sommaire

## 1 Introduction

- L'IRMA
- Le sujet du stage

## 2 Principe et méthode

- Simulation de l'ETR
- Réseaux de neurones

## 3 Le problème en 1D

- Simulation
- Apprentissage

## 4 Le problème en 2D

- Simulation
- Apprentissage

## 5 Conclusion

- Sur l'apprentissage
- Sur le stage

Introduction

○○○  
○○○○

Principe et méthode

●○○○○○○○  
○○○○○○○

Le problème en 1D

○○○  
○○○

Le problème en 2D

○○○  
○○○○

Conclusion

○○○  
○○○○

## Simulation de l'ETR

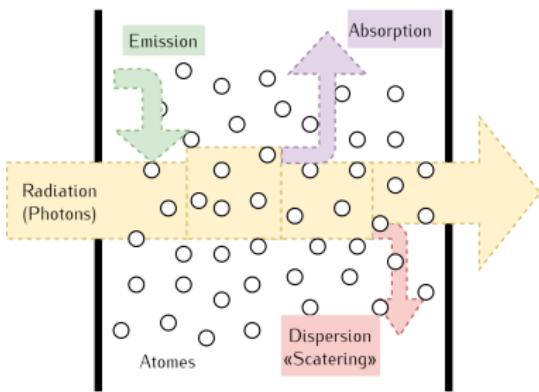
# Simulation de l'ETR

## Simulation de l'ETR

# Le transfert radiatif

Lorsque les photons se trouvent en présence de la matière, trois phénomènes majeurs (caractérisés par leurs opacités) se produisent :

- **Émission ( $\sigma_e$ )** : des photons sont émis en réponse aux électrons excités descendants à des niveaux d'énergie plus bas. Plus la température matière est élevée, plus l'émission est importante
- **Absorption ( $\sigma_a$ )** : à l'inverse, certains photons sont absorbés, les électrons deviennent plus excités (ou se libèrent complètement de leurs atomes), et la matière se réchauffe. À l'équilibre thermique,  $\sigma_a = \sigma_e$
- **Dispersion ( $\sigma_c$ )** : certains photons sont déviés de leur trajectoire originale par la matière. Il faut aussi tenir compte de la fonction de distribution angulaire de "scattering"  $p(\Omega' \rightarrow \Omega)$  (Turpault, 2004).



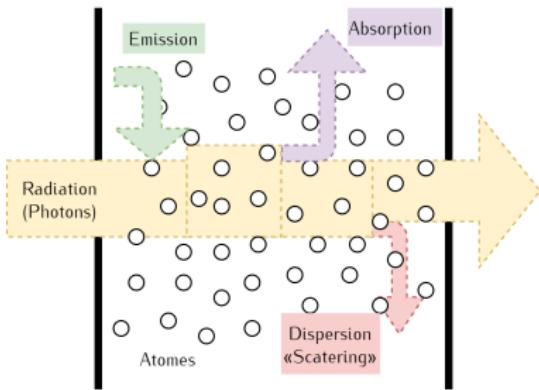
Interaction entre matière et radiation

## Simulation de l'ETR

# Le transfert radiatif

Lorsque les photons se trouvent en présence de la matière, trois phénomènes majeurs (caractérisés par leurs opacités) se produisent :

- **Émission ( $\sigma_e$ )** : des photons sont émis en réponse aux électrons excités descendants à des niveaux d'énergie plus bas. Plus la température matière est élevée, plus l'émission est importante
- **Absorption ( $\sigma_a$ )** : à l'inverse, certains photons sont absorbés, les électrons deviennent plus excités (ou se libèrent complètement de leurs atomes), et la matière se réchauffe. À l'équilibre thermique,  $\sigma_a = \sigma_e$
- **Dispersion ( $\sigma_c$ )** : certains photons sont déviés de leur trajectoire originale par la matière. Il faut aussi tenir compte de la fonction de distribution angulaire de "scattering"  $p(\Omega' \rightarrow \Omega)$  (Turpault, 2004).



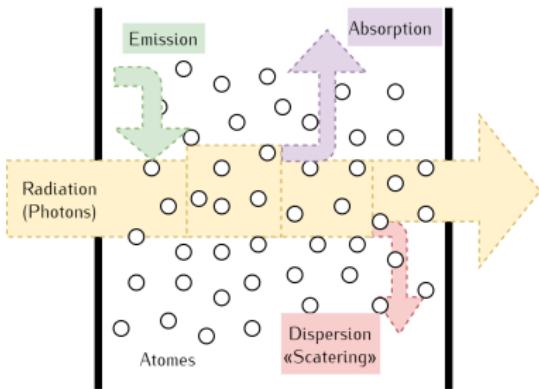
Interaction entre matière et radiation

## Simulation de l'ETR

# Le transfert radiatif

Lorsque les photons se trouvent en présence de la matière, trois phénomènes majeurs (caractérisés par leurs opacités) se produisent :

- **Émission ( $\sigma_e$ )** : des photons sont émis en réponse aux électrons excités descendants à des niveaux d'énergie plus bas. Plus la température matière est élevée, plus l'émission est importante
- **Absorption ( $\sigma_a$ )** : à l'inverse, certains photons sont absorbés, les électrons deviennent plus excités (ou se libèrent complètement de leurs atomes), et la matière se réchauffe. À l'équilibre thermique,  $\sigma_a = \sigma_e$
- **Dispersion ( $\sigma_c$ )** : certains photons sont déviés de leur trajectoire originale par la matière. Il faut aussi tenir compte de la fonction de distribution angulaire de "scattering"  $p(\Omega' \rightarrow \Omega)$  (Turpault, 2004).



Interaction entre matière et radiation

## Simulation de l'ETR

## L'ETR

L'équation du transfert radiatif (ETR) est un bilan d'énergie lié au rayonnement au niveau mésoscopique (dans la direction  $\Omega$ ).

$$\begin{aligned} \frac{1}{c} \frac{\partial}{\partial t} I(t, x, \Omega, \nu) + \Omega \cdot \nabla_x I(t, x, \Omega, \nu) \\ = \sigma_a(\rho, \Omega, \nu) (B(\nu, T) - I(t, x, \Omega, \nu)) \\ + \frac{1}{4\pi} \int_0^\infty \int_{S^2} \sigma_c(\rho, \Omega, \nu) p(\Omega' \rightarrow \Omega) (I(t, x, \Omega', \nu) - I(t, x, \Omega, \nu)) d\Omega' d\nu \end{aligned}$$

Où :

- $I(t, x, \Omega, \nu)$  désigne l'intensité radiative spécifique
- $B(\nu, T)$  la fonction de Planck
- $\oint p(\Omega' \rightarrow \Omega) d\Omega' = 1$

## Simulation de l'ETR

## L'ETR

L'équation du transfert radiatif (ETR) est un bilan d'énergie lié au rayonnement au niveau mésoscopique (dans la direction  $\Omega$ ).

$$\begin{aligned} \frac{1}{c} \frac{\partial}{\partial t} I(t, x, \Omega, \nu) + \Omega \cdot \nabla_x I(t, x, \Omega, \nu) \\ = \sigma_a(\rho, \Omega, \nu) (B(\nu, T) - I(t, x, \Omega, \nu)) \\ + \frac{1}{4\pi} \int_0^\infty \int_{S^2} \sigma_c(\rho, \Omega, \nu) p(\Omega' \rightarrow \Omega) (I(t, x, \Omega', \nu) - I(t, x, \Omega, \nu)) d\Omega' d\nu \end{aligned}$$

Où :

- $I(t, x, \Omega, \nu)$  désigne l'intensité radiative spécifique
- $B(\nu, T)$  la fonction de Planck
- $\oint p(\Omega' \rightarrow \Omega) d\Omega' = 1$

# L'ETR

L'équation du transfert radiatif (ETR) est un bilan d'énergie lié au rayonnement au niveau mésoscopique (dans la direction  $\Omega$ ).

$$\begin{aligned} \frac{1}{c} \frac{\partial}{\partial t} I(t, x, \Omega, \nu) + \Omega \cdot \nabla_x I(t, x, \Omega, \nu) \\ = \sigma_a(\rho, \Omega, \nu) (B(\nu, T) - I(t, x, \Omega, \nu)) \\ + \frac{1}{4\pi} \int_0^\infty \int_{S^2} \sigma_c(\rho, \Omega, \nu) p(\Omega' \rightarrow \Omega) (I(t, x, \Omega', \nu) - I(t, x, \Omega, \nu)) d\Omega' d\nu \end{aligned}$$

Où :

- $I(t, x, \Omega, \nu)$  désigne l'intensité radiative spécifique
- $B(\nu, T)$  la fonction de Planck
- $\oint p(\Omega' \rightarrow \Omega) d\Omega' = 1$

# Le modèle P1

D'après (Franck, 2012) :

$$\begin{cases} \partial_t E + c \operatorname{div} F = c\sigma_a (aT^4 - E) \\ \partial_t F + c \nabla E = -c\sigma_c F \\ \rho C_v \partial_t T = c\sigma_a (E - aT^4) \end{cases}$$

Où :

- Energie des photons :  $E(t, x) = \frac{4\pi}{c} \int_0^\infty \int_{S^2} I(t, x, \Omega, \nu) d\Omega d\nu$
- Flux des photons :  $F(t, x) = \frac{4\pi}{c} \int_0^\infty \int_{S^2} \Omega I(t, x, \Omega, \nu) d\Omega d\nu$

Ce modèle est :

- Linéaire et hyperbolique
- Macroscopique aux moments (d'ordre 2), dit "gris"
- Moins précis qu'un modèle résolut par la méthode de Monte-Carlo
- Peu coûteux et rapide à implémenter

# Le modèle P1

D'après (Franck, 2012) :

$$\begin{cases} \partial_t E + c \operatorname{div} F = c\sigma_a (aT^4 - E) \\ \partial_t F + c \nabla E = -c\sigma_c F \\ \rho C_v \partial_t T = c\sigma_a (E - aT^4) \end{cases}$$

Où :

- Energie des photons :  $E(t, x) = \frac{4\pi}{c} \int_0^\infty \int_{S^2} I(t, x, \Omega, \nu) d\Omega d\nu$
- Flux des photons :  $F(t, x) = \frac{4\pi}{c} \int_0^\infty \int_{S^2} \Omega I(t, x, \Omega, \nu) d\Omega d\nu$

Ce modèle est :

- Linéaire et hyperbolique
- Macroscopique aux moments (d'ordre 2), dit "gris"
- Moins précis qu'un modèle résolut par la méthode de Monte-Carlo
- Peu coûteux et rapide à implémenter

# Le modèle P1

D'après (Franck, 2012) :

$$\begin{cases} \partial_t E + c \operatorname{div} F = c\sigma_a (aT^4 - E) \\ \partial_t F + c \nabla E = -c\sigma_c F \\ \rho C_v \partial_t T = c\sigma_a (E - aT^4) \end{cases}$$

Où :

- Energie des photons :  $E(t, x) = \frac{4\pi}{c} \int_0^\infty \int_{S^2} I(t, x, \Omega, \nu) d\Omega d\nu$
- Flux des photons :  $F(t, x) = \frac{4\pi}{c} \int_0^\infty \int_{S^2} \Omega I(t, x, \Omega, \nu) d\Omega d\nu$

Ce modèle est :

- Linéaire et hyperbolique
- Macroscopique aux moments (d'ordre 2), dit "gris"
- Moins précis qu'un modèle résolut par la méthode de Monte-Carlo
- Peu coûteux et rapide à implémenter

# Le modèle P1

D'après (Franck, 2012) :

$$\begin{cases} \partial_t E + c \operatorname{div} F = c\sigma_a (aT^4 - E) \\ \partial_t F + c \nabla E = -c\sigma_c F \\ \rho C_v \partial_t T = c\sigma_a (E - aT^4) \end{cases}$$

Où :

- Energie des photons :  $E(t, x) = \frac{4\pi}{c} \int_0^\infty \int_{S^2} I(t, x, \Omega, \nu) d\Omega d\nu$
- Flux des photons :  $F(t, x) = \frac{4\pi}{c} \int_0^\infty \int_{S^2} \Omega I(t, x, \Omega, \nu) d\Omega d\nu$

Ce modèle est :

- Linéaire et hyperbolique
- Macroscopique aux moments (d'ordre 2), dit "gris"
- Moins précis qu'un modèle résolut par la méthode de Monte-Carlo
- Peu coûteux et rapide à implémenter

# Le modèle P1

D'après (Franck, 2012) :

$$\begin{cases} \partial_t E + c \operatorname{div} F = c\sigma_a (aT^4 - E) \\ \partial_t F + c \nabla E = -c\sigma_c F \\ \rho C_v \partial_t T = c\sigma_a (E - aT^4) \end{cases}$$

Où :

- Energie des photons :  $E(t, x) = \frac{4\pi}{c} \int_0^\infty \int_{S^2} I(t, x, \Omega, \nu) d\Omega d\nu$
- Flux des photons :  $F(t, x) = \frac{4\pi}{c} \int_0^\infty \int_{S^2} \Omega I(t, x, \Omega, \nu) d\Omega d\nu$

Ce modèle est :

- Linéaire et hyperbolique
- Macroscopique aux moments (d'ordre 2), dit "gris"
- Moins précis qu'un modèle résolut par la méthode de Monte-Carlo
- Peu coûteux et rapide à implémenter

# Le schéma de "splitting"

Le principe :

- 1 On sépare le problème en temps en deux

# Le schéma de "splitting"

Le principe :

- 1 On sépare le problème en temps en deux
- 2 On résout l'étape 1 (réglage de la température) : sur une maille, Euler implicite + Point fixe

$$\begin{cases} \partial_t E + \mathbf{c} \operatorname{div} \mathbf{F} = c\sigma_a (aT^4 - E) \\ \rho C_v \partial_t T = c\sigma_a (E - aT^4) \end{cases}$$

# Le schéma de "splitting"

Le principe :

- 1 On sépare le problème en temps en deux
- 2 On résout l'étape 1 (réglage de la température) : sur une maille, Euler implicite + Point fixe

$$\begin{cases} \partial_t E + c \operatorname{div} \mathbf{F} = c\sigma_a (aT^4 - E) \\ \rho C_v \partial_t T = c\sigma_a (E - aT^4) \end{cases}$$

- 3 On résout l'étape 2 (partie hyperbolique) : volume finis 2D, mais aussi 1D

$$\begin{cases} \partial_t E + c \operatorname{div} \mathbf{F} = c\sigma_a (aT^4 - E) \\ \partial_t \mathbf{F} + c \nabla E = -c\sigma_c \mathbf{F} \end{cases}$$

## Simulation de l'ETR

# Le schéma de "splitting"

Le principe :

- 1** On sépare le problème en temps en deux
- 2** On résout l'étape 1 (réglage de la température) : sur une maille, Euler implicite + Point fixe

$$\begin{cases} \partial_t E + c \operatorname{div} \mathbf{F} = c\sigma_a (aT^4 - E) \\ \rho C_v \partial_t T = c\sigma_a (E - aT^4) \end{cases}$$

- 3** On résout l'étape 2 (partie hyperbolique) : volume finis 2D, mais aussi 1D

$$\begin{cases} \partial_t E + c \operatorname{div} \mathbf{F} = c\sigma_a (aT^4 - E) \\ \partial_t \mathbf{F} + c \nabla E = -c\sigma_c \mathbf{F} \end{cases}$$

- 4** Tout ceci se fait sur le même pas de temps

# Le schéma de "splitting" : Étape 1

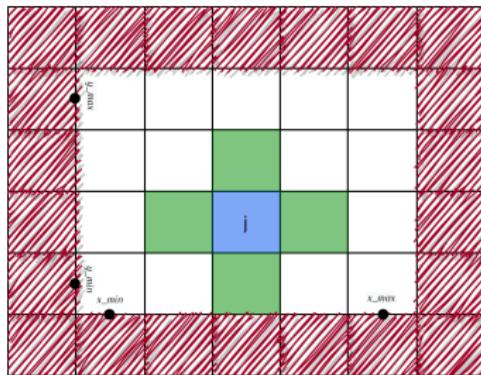
À l'itération  $n$ , on pose  $\Theta = aT^4$

$$\begin{cases} E_j^{q+1} = \frac{\alpha E_j^n + \beta \gamma \Theta_j^n}{1 - \beta \delta} \\ \Theta_j^{q+1} = \frac{\gamma \Theta_j^n + \alpha \delta E_j^n}{1 - \beta \delta} \end{cases}$$

Où  $\mu_q = \frac{1}{T^{3,n} + T^n T^{2,q} + T^q T^{2,n} + T^{3,q}}$

$$\alpha = \frac{1}{\Delta t \left( \frac{1}{\Delta t} + c\sigma_a \right)}, \quad \beta = \frac{c\sigma_a}{\Delta t + c\sigma_a}, \quad \gamma = \frac{\rho_j C_v \mu_q}{\Delta t \left( \frac{\rho_j C_v \mu_q}{\Delta t} + c\sigma_a \right)}, \quad \delta = \frac{c\sigma_a}{\frac{\rho_j C_v \mu_q}{\Delta t} + c\sigma_a}.$$

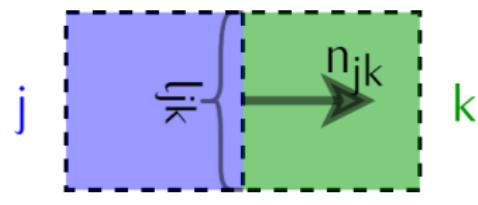
Itération sur  $q$  ; convergence vers  $E_j^*$  et  $\Theta_j^*$  ;  $F_j = F_j^*$  constant.



## Simulation de l'ETR

## Le schéma de "splitting" : Étape 2

$$\begin{cases} E_j^{n+1} = E_j^* + \alpha \sum_k (\mathbf{F}_{jk}, \mathbf{n}_{jk}) \\ \mathbf{F}_j^{n+1} = \beta \mathbf{F}_j^* + \gamma E_j^n + \delta \sum_k E_{jk} \mathbf{n}_{jk} \end{cases}$$



Avec :

$$\alpha = -\frac{c\Delta t}{|\Omega_j|},$$

$$\beta = \frac{1}{\Delta t} \left( \frac{1}{\Delta t} + c \sum_k M_{jk} \sigma_{jk} \right)^{-1},$$

$$\gamma = \frac{c}{|\Omega_j|} \left( \frac{1}{\Delta t} + c \sum_k M_{jk} \sigma_{jk} \right)^{-1} \left( \sum_k l_{jk} M_{jk} \mathbf{n}_{jk} \right)$$

$$\delta = -\frac{c}{|\Omega_j|} \left( \frac{1}{\Delta t} + c \sum_k M_{jk} \sigma_{jk} \right)^{-1}$$

$$(\mathbf{F}_{jk}, \mathbf{n}_{jk}) = l_{jk} M_{jk} \left( \frac{\mathbf{F}_j^n \cdot \mathbf{n}_{jk} + \mathbf{F}_k^n \cdot \mathbf{n}_{jk}}{2} - \frac{\mathbf{E}_k^n - \mathbf{E}_j^n}{2} \right)$$

$$E_{jk} \mathbf{n}_{jk} = l_{jk} M_{jk} \left( \frac{\mathbf{E}_j^n + \mathbf{E}_k^n}{2} - \frac{\mathbf{F}_k^n \cdot \mathbf{n}_{jk} - \mathbf{F}_j^n \cdot \mathbf{n}_{jk}}{2} \right) \mathbf{n}_{jk}$$

$$M_{jk} = \frac{2}{2 + \Delta x \sigma_{jk}}$$

$$\sigma_{jk} = \frac{1}{2} \left( \sigma_c(\rho_j, T_j^n) + \sigma_c(\rho_k, T_k^n) \right)$$

## Simulation de l'ETR

# Implémentation C++

- Temps final = 0.01 sh
- $c = 299 \text{ [cm/sh]}$
- $a = 0.01372 \text{ [g/cm/sh}^2\text{/keV]}$
- $C_v = 0.14361 \text{ [Jerk/g/keV]}$
- La densité  $\rho$  est un signal créneau  $[\text{g cm}^{-3}]$
- $\sigma_a = \rho T \text{ [cm}^{-1}\text{]}$
- $\sigma_c = \rho T \text{ [cm}^{-1}\text{]}$
- $T_0, T_{gauche} = 5 \text{ [keV]}$
- $E_0 = a T_0^4 \text{ [g/cm/sh}^2\text{]}$
- $E_{gauche*} = a T_0^4 + 5 \sin(2k\pi t) \text{ [g/cm/sh}^2\text{]}$
- $F_0, F_{gauche} = 0 \text{ [g/sh}^2\text{]}$
- Sorties libres sur les autres bords

# Implémentation C++

- Temps final = 0.01 sh
- $c = 299$  [cm/sh]
- $a = 0.01372$  [g/cm/sh<sup>2</sup>/keV]
- $C_v = 0.14361$  [Jerk/g/keV]
- La densité  $\rho$  est un signal créneau [g cm<sup>-3</sup>]
- $\sigma_a = \rho T$  [cm<sup>-1</sup>]
- $\sigma_c = \rho T$  [cm<sup>-1</sup>]
- $T_0, T_{gauche} = 5$  [keV]
- $E_0 = aT_0^4$  [g/cm/sh<sup>2</sup>]
- $E_{gauche*} = aT_0^4 + 5 \sin(2k\pi t)$  [g/cm/sh<sup>2</sup>]
- $F_0, F_{gauche} = 0$  [g/sh<sup>2</sup>]
- Sorties libres sur les autres bords

```
x_min 0
x_max 1
y_min 0
y_max 1
N 90

c 299
a 0.01372
C_v 0.14361

CFL 0.5
precision 1e-6
t_0 0
t_f 0.01

rho_crenau(0.5,0.5,0.1,10)
sigma_a rho*T
sigma_c rho*T

E_0 0.01372*(5^4)
F_0.x 0
F_0.y 0
T_0 5
```

```
E_u neumann
F_u.x neumann
F_u.y neumann
T_u neumann

E_d neumann
F_d.x neumann
F_d.y neumann
T_d neumann

E_l ponctuel(0.4,0.6)
F_l.x 0
F_l.y 0
T_l 5

E_r neumann
F_r.x neumann
F_r.y neumann
T_r neumann

export_file data/df_simu.csv
export_mode dataframe
write_mode truncate
```

Exemple de configuration

Introduction

○○○  
○○○○

Principe et méthode

○○○○○○○○  
●○○○○○○

Le problème en 1D

○○○  
○○○

Le problème en 2D

○○○  
○○○○

Conclusion

○○○  
○○○○

Réseaux de neurones

# Réseaux de neurones

# Le principe des réseaux de neurones

On désigne par  $X$  le signal sur les bords du domaine et  $y$  la densité du milieu.

- $f$  définit le problème direct :  $X = f(y)$
- $f^{-1}$  définit le problème inverse :  $y = f^{-1}(X)$
- On va se contenter de  $\hat{f}^{-1}$  telle que  $\hat{y} = \hat{f}^{-1}(X, \theta)$ , où  $\theta$  représente les paramètres du réseau de neurones

L'apprentissage :  $IA \rightarrow ML \rightarrow ANN \rightarrow MLP \rightarrow CNN$

- IA : Intelligence Artificielle
- ML : Machine Learning
- ANN : Artificial Neural Network
- MLP : Multi-Layer Perceptron
- CNN : Convolutional Neural Network

# Le principe des réseaux de neurones

On désigne par  $X$  le signal sur les bords du domaine et  $y$  la densité du milieu.

- $f$  définit le problème direct :  $X = f(y)$
- $f^{-1}$  définit le problème inverse :  $y = f^{-1}(X)$
- On va se contenter de  $\hat{f}^{-1}$  telle que  $\hat{y} = \hat{f}^{-1}(X, \theta)$ , où  $\theta$  représente les paramètres du réseau de neurones

L'apprentissage :  $IA \rightarrow ML \rightarrow ANN \rightarrow MLP \rightarrow CNN$

- IA : Intelligence Artificielle
- ML : Machine Learning
- ANN : Artificial Neural Network
- MLP : Multi-Layer Perceptron
- CNN : Convolved Neural Network

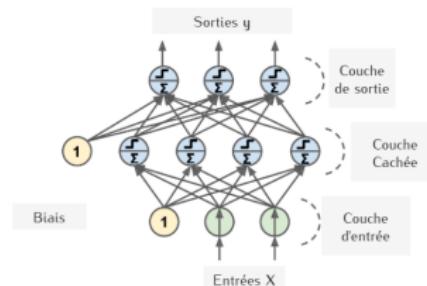
# Le principe des réseaux de neurones

On désigne par  $X$  le signal sur les bords du domaine et  $y$  la densité du milieu.

- $f$  définit le problème direct :  $X = f(y)$
- $f^{-1}$  définit le problème inverse :  $y = f^{-1}(X)$
- On va se contenter de  $\hat{f}^{-1}$  telle que  $\hat{y} = \hat{f}^{-1}(X, \theta)$ , où  $\theta$  représente les paramètres du réseau de neurones

L'apprentissage :  $IA \rightarrow ML \rightarrow ANN \rightarrow MLP \rightarrow CNN$

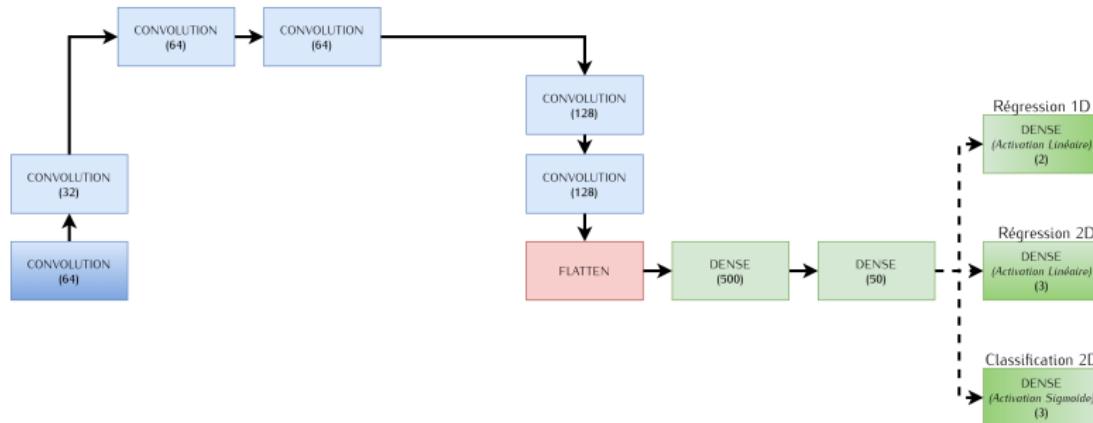
- IA : Intelligence Artificielle
- ML : Machine Learning
- ANN : Artificial Neural Network
- MLP : Multi-Layer Perceptron
- CNN : Convolved Neural Network



Exemple de MLP (Geron, 2017)

## Réseaux de neurones

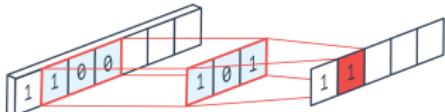
## L'architecture sous Keras



Architecture générale utilisée (CNN)

# Les couches utilisées : Convolution (cross-corrélation)

$$S(i) = (I * K)(i) = \sum_m I(i+m)K(m)$$

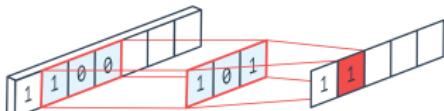


En 1D (Ganesh, 2019)

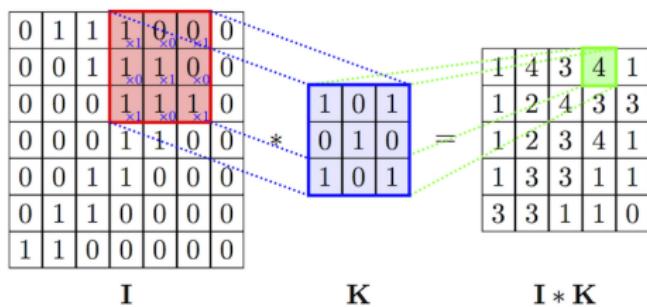
# Les couches utilisées : Convolution (cross-corrélation)

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(i+m, j+n)K(m, n)$$

$$S(i) = (I * K)(i) = \sum_m I(i+m)K(m)$$

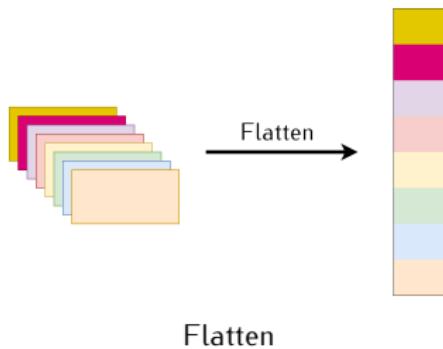


En 1D (Ganesh, 2019)

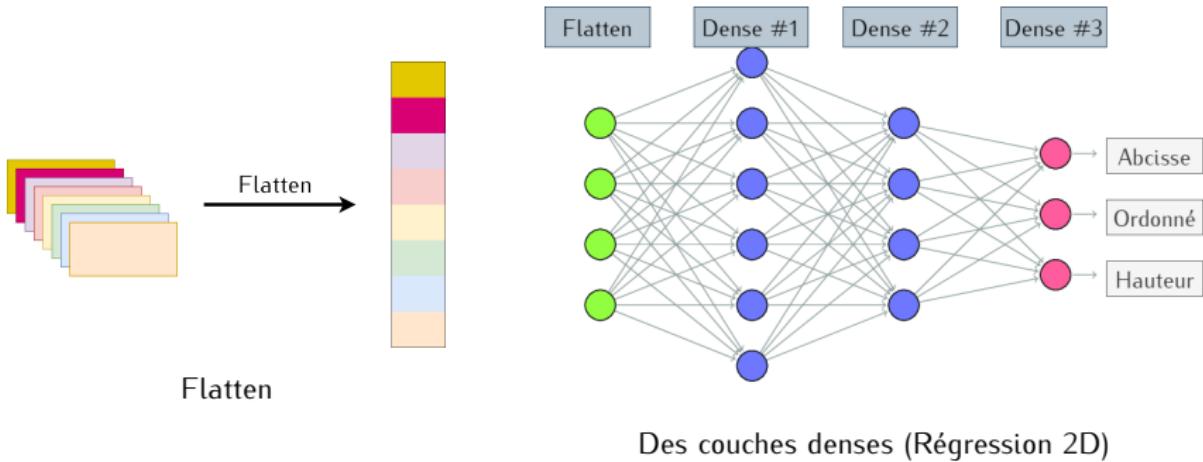


En 2D (Packt, s.d)

# Les couches utilisées : Flatten et Dense



## Les couches utilisées : Flatten et Dense



# Les métriques

## Coefficient de détermination R2

$$\text{Score } R2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

Avec

$$SS_{res} = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$SS_{tot} = \sum_{i=1}^n (y_i - \bar{y})^2$$

Où  $\bar{y} = \sum_{i=1}^n y_i$  représente la moyenne des valeurs observées.

# Les métriques

## Coefficient de détermination R2

### Score personnalisé

$$\text{Score } R2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

Avec

$$SS_{res} = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$SS_{tot} = \sum_{i=1}^n (y_i - \bar{y})^2$$

Pourcentage des prédictions correctes :

Où  $\bar{y} = \sum_{i=1}^n y_i$  représente la moyenne des valeurs observées.

# Les métriques

## Coefficient de détermination R2

### Score personnalisé

$$\text{Score } R2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

Avec

$$SS_{res} = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$SS_{tot} = \sum_{i=1}^n (y_i - \bar{y})^2$$

Pourcentage des prédictions correctes :

- au dixième près pour la position (suivant x ou y)

Où  $\bar{y} = \sum_{i=1}^n y_i$  représente la moyenne des valeurs observées.

# Les métriques

## Coefficient de détermination R2

### Score personnalisé

$$\text{Score } R2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

Avec

$$SS_{res} = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$SS_{tot} = \sum_{i=1}^n (y_i - \bar{y})^2$$

Pourcentage des prédictions correctes :

- au dixième près pour la position (suivant x ou y)
- à l'unité près pour la hauteur

Où  $\bar{y} = \sum_{i=1}^n y_i$  représente la moyenne des valeurs observées.

# Les hyper-paramètres

Liste des paramètres les plus influents pour l'entraînement

Hyper-paramètre	Définition	Valeur 1D / 2D
optimizer	algorithme d'optimisation	Adam
learning rate	taux d'apprentissage	1e-4 / 1e-5
batch size	taille d'un batch à chaque époque	32
epochs	nombre d'époques	100
patience	patience pour l'early stopping	10
kernel size	taille du noyau de convolution	3 / (6,2)
activation	type de fonction d'activation	relu, linear, sigmoid

# Les hyper-paramètres

Liste des paramètres les plus influents pour l'entraînement

Hyper-paramètre	Définition	Valeur 1D / 2D
optimizer	algorithme d'optimisation	Adam
learning rate	taux d'apprentissage	1e-4 / 1e-5
batch size	taille d'un batch à chaque époque	32
epochs	nombre d'époques	100
patience	patience pour l'early stopping	10
kernel size	taille du noyau de convolution	3 / (6,2)
activation	type de fonction d'activation	relu, linear, sigmoid

# Les hyper-paramètres

Liste des paramètres les plus influents pour l'entraînement

Hyper-paramètre	Définition	Valeur 1D / 2D
optimizer	algorithme d'optimisation	Adam
learning rate	taux d'apprentissage	1e-4 / 1e-5
batch size	taille d'un batch à chaque époque	32
epochs	nombre d'époques	100
patience	patience pour l'early stopping	10
kernel size	taille du noyau de convolution	3 / (6,2)
activation	type de fonction d'activation	relu, linear, sigmoid

# Les hyper-paramètres

Liste des paramètres les plus influents pour l'entraînement

Hyper-paramètre	Définition	Valeur 1D / 2D
optimizer	algorithme d'optimisation	Adam
learning rate	taux d'apprentissage	1e-4 / 1e-5
batch size	taille d'un batch à chaque époque	32
epochs	nombre d'époques	100
patience	patience pour l'early stopping	10
kernel size	taille du noyau de convolution	3 / (6,2)
activation	type de fonction d'activation	relu, linear, sigmoid

# Les hyper-paramètres

Liste des paramètres les plus influents pour l'entraînement

Hyper-paramètre	Définition	Valeur 1D / 2D
optimizer	algorithme d'optimisation	Adam
learning rate	taux d'apprentissage	1e-4 / 1e-5
batch size	taille d'un batch à chaque époque	32
epochs	nombre d'époques	100
patience	patience pour l'early stopping	10
kernel size	taille du noyau de convolution	3 / (6,2)
activation	type de fonction d'activation	relu, linear, sigmoid

# Les hyper-paramètres

Liste des paramètres les plus influents pour l'entraînement

Hyper-paramètre	Définition	Valeur 1D / 2D
optimizer	algorithme d'optimisation	Adam
learning rate	taux d'apprentissage	1e-4 / 1e-5
batch size	taille d'un batch à chaque époque	32
epochs	nombre d'époques	100
patience	patience pour l'early stopping	10
kernel size	taille du noyau de convolution	3 / (6,2)
activation	type de fonction d'activation	relu, linear, sigmoid

# Les hyper-paramètres

Liste des paramètres les plus influents pour l'entraînement

Hyper-paramètre	Définition	Valeur 1D / 2D
optimizer	algorithme d'optimisation	Adam
learning rate	taux d'apprentissage	1e-4 / 1e-5
batch size	taille d'un batch à chaque époque	32
epochs	nombre d'époques	100
patience	patience pour l'early stopping	10
kernel size	taille du noyau de convolution	3 / (6,2)
activation	type de fonction d'activation	relu, linear, sigmoid

Introduction

○○○  
○○○○

Principe et méthode

○○○○○○○○  
○○○○○○○○

Le problème en 1D

●○○  
○○○

Le problème en 2D

○○○  
○○○○

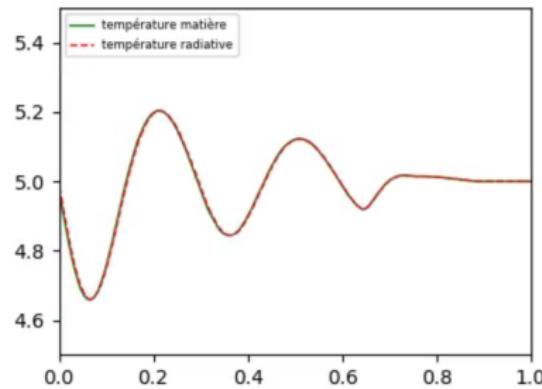
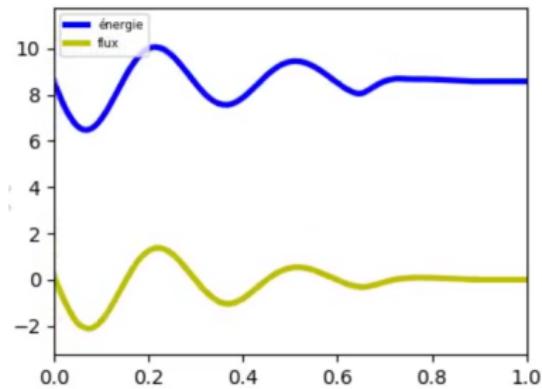
Conclusion

○○○  
○○○○

Simulation

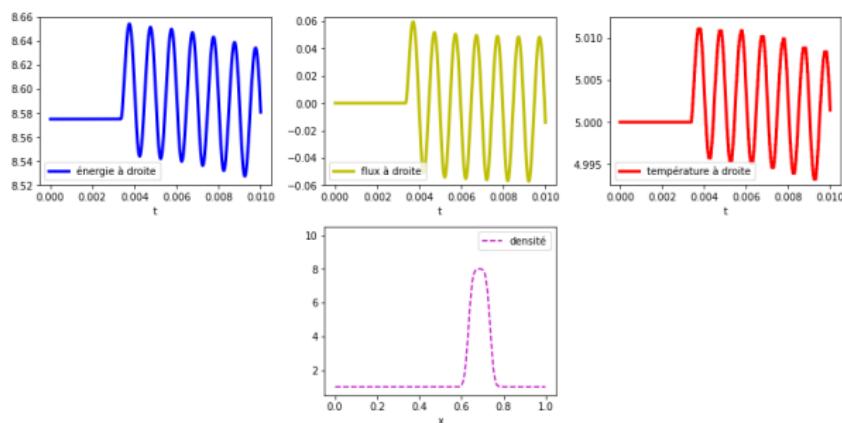
## Résultats 1D

# Exemple de simulation 1D



## Simulation

## Entrées/sorties 1D



(50, 3)

Dimension temporelle

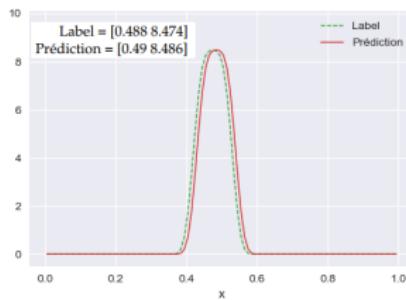
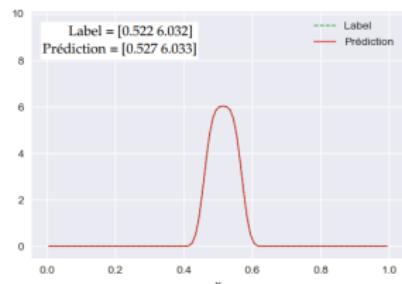
Nombre de canaux

Taille d'une entrée

Une entrée du réseau de neurones et la sortie attendue

## Apprentissage

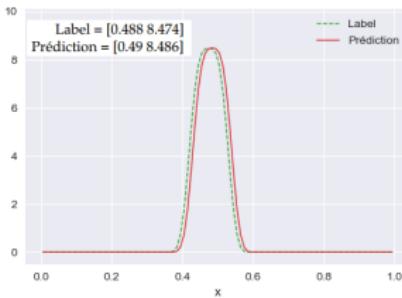
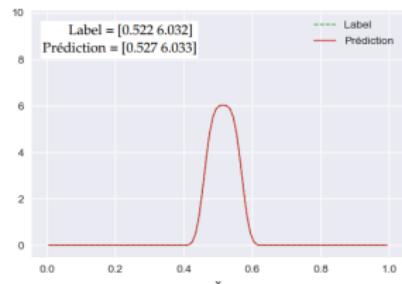
# Meilleures/piires prédictions du CNN



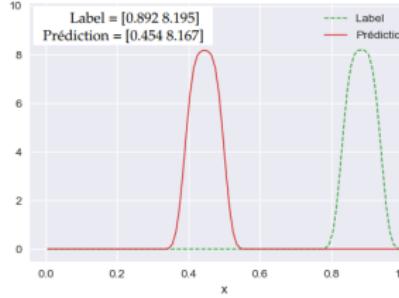
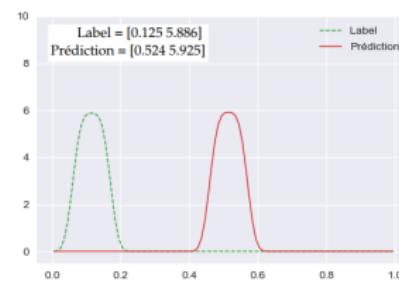
Les meilleures prédictions

## Apprentissage

## Meilleures/piques prédictions du CNN



Les meilleures prédictions



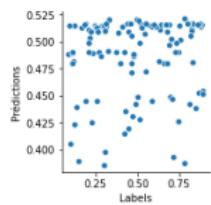
Les pires prédictions

# Les scores obtenus en 1D

Nom du score	Valeur
R2	99.50 %
Personnalisé	28.21 %

# Les scores obtenus en 1D

Nom du score	Valeur
R2	99.50 %
Personnalisé	28.21 %



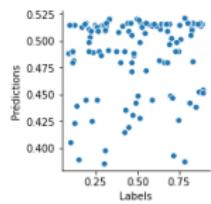
Correlation des positions

# Les scores obtenus en 1D

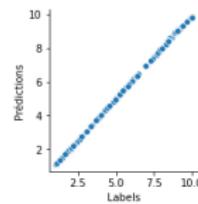
---

Nom du score	Valeur
R2	99.50 %
Personnalisé	28.21 %

---



Correlation des positions



Correlation des hauteurs

# Conclusion sur la régression 1D

## Les causes de l'échec :

- Le problème inverse est probablement mal posé
- Le score R2 est mal calculé

```
from keras import backend as K

def r2_score(y_true, y_pred):
    SS_res = K.sum(K.square(y_true - y_pred), axis=-1)
    SS_tot = K.sum(K.square(y_true - K.mean(y_true)), axis=-1)
    return 1.0 - SS_res/(SS_tot + K.epsilon())
```

# Conclusion sur la régression 1D

Les causes de l'échec :

- Le problème inverse est probablement mal posé
- Le score R2 est mal calculé

```
from keras import backend as K

def r2_score(y_true, y_pred):
    SS_res = K.sum(K.square(y_true - y_pred), axis=-1)
    SS_tot = K.sum(K.square(y_true - K.mean(y_true)), axis=-1)
    return 1.0 - SS_res/(SS_tot + K.epsilon())
```

Introduction

○○○  
○○○○

Principe et méthode

○○○○○○○○  
○○○○○○○○

Le problème en 1D

○○○  
○○○

Le problème en 2D

●○○  
○○○○

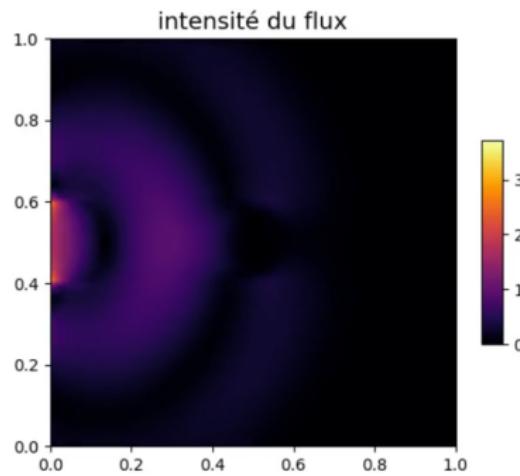
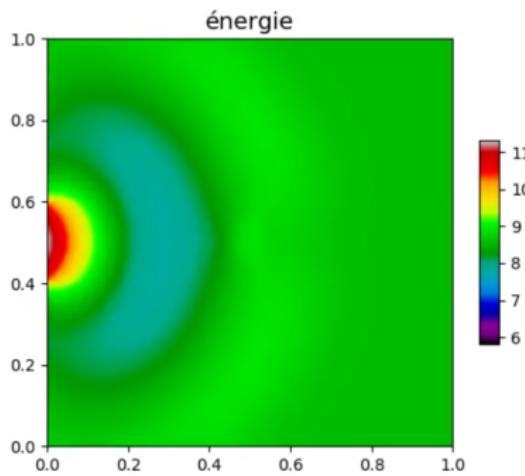
Conclusion

○○○  
○○○○

Simulation

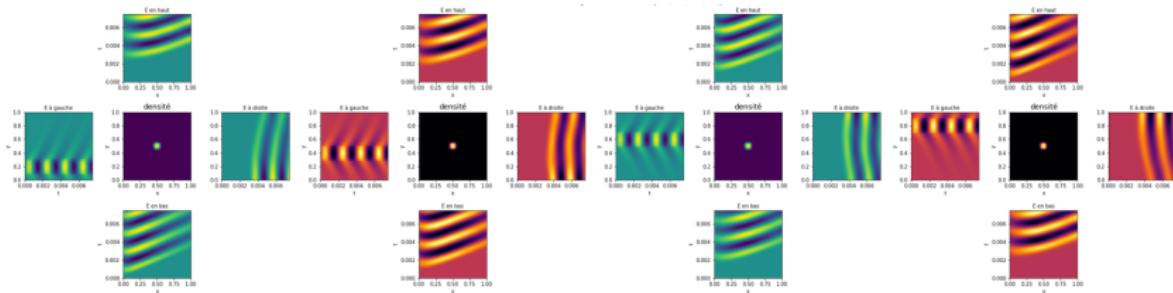
## Résultats 2D

# Exemple de simulation 2D



## Simulation

## Entrées/sorties 2D



Une entrée du réseau de neurones et la sortie attendue

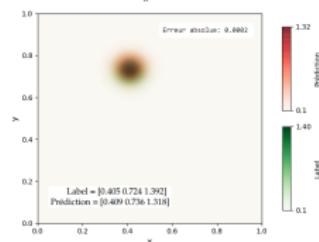
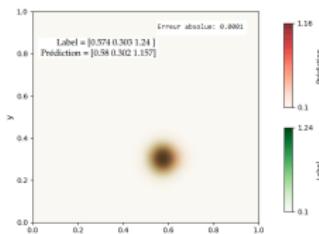
(168, 28, 48)

Dimension temporelle Dimension spatiale Nombre de canaux

Taille d'une entrée

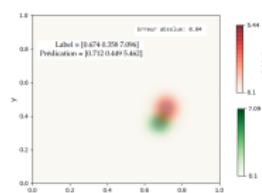
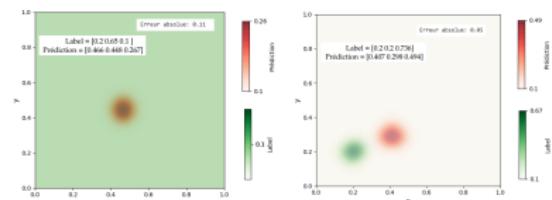
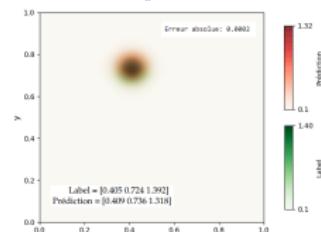
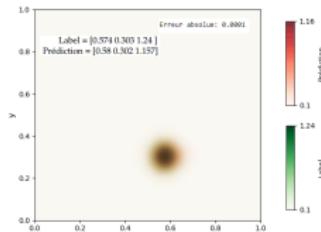
## Apprentissage

# Meilleures/piques prédictions du CNN



Les meilleures prédictions

# Meilleures/piques prédictions du CNN



Les pires prédictions

Les meilleures prédictions

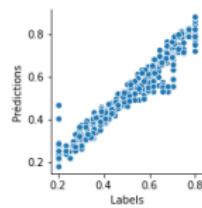
# Les scores obtenus en 2D

Nom du score	Valeur
R2	98.81 %
Personnalisé	93.50 %

## Apprentissage

## Les scores obtenus en 2D

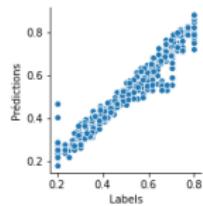
Nom du score	Valeur
R2	98.81 %
Personnalisé	93.50 %



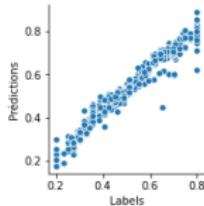
## Corrélation des abscisses

# Les scores obtenus en 2D

Nom du score	Valeur
R2	98.81 %
Personnalisé	93.50 %



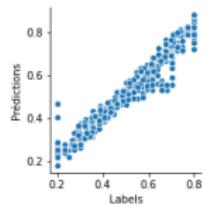
Corrélation des abscisses



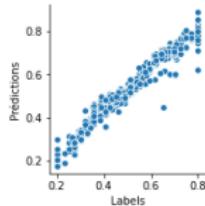
Corrélation des ordonnées

# Les scores obtenus en 2D

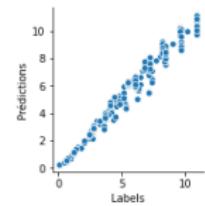
Nom du score	Valeur
R2	98.81 %
Personnalisé	93.50 %



Corrélation des abscisses



Corrélation des ordonnées



Corrélation des hauteurs

# Conclusion sur la régression 2D

Détection de toutes les variables :

- L'abscisse, l'ordonnée, et la hauteur sont relativement bien prédits
- La valeur de la densité en dehors du crâneau est connue : reconstruction complète de la densité du milieu

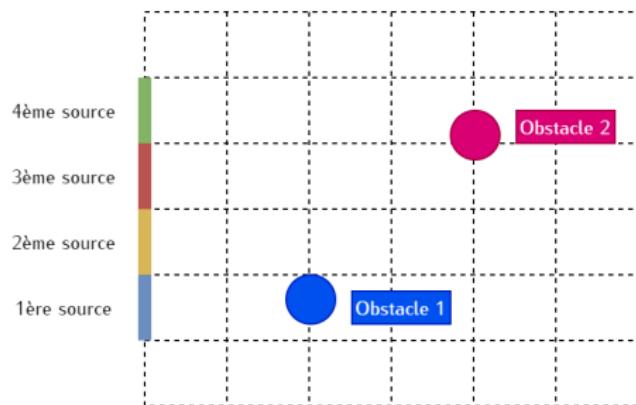
# Conclusion sur la régression 2D

Détection de toutes les variables :

- L'abscisse, l'ordonnée, et la hauteur sont relativement bien prédits
- La valeur de la densité en dehors du crâneau est connue : **reconstruction complète de la densité du milieu**

## Apprentissage

## Classification 2D

LabelsObstacle 1: 

1	0	0	0
---	---	---	---

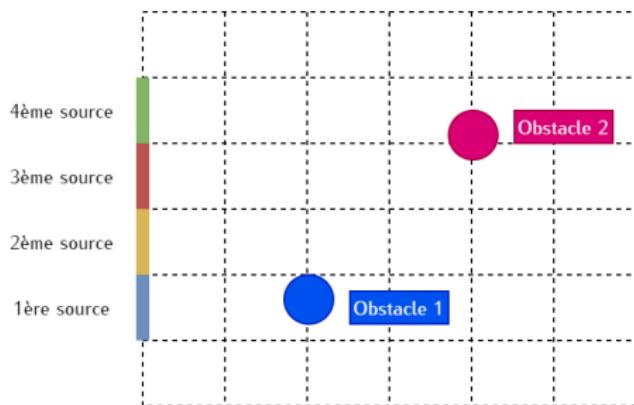
Obstacle 2: 

0	0	1	1
---	---	---	---

Labels pour la classification

## Apprentissage

## Classification 2D

LabelsObstacle 1: 

1	0	0	0
---	---	---	---

Obstacle 2: 

0	0	1	1
---	---	---	---

Labels pour la classification

Les scores obtenus

---

Nom	Valeur
R2	98.86 %
Pers.	95.45 %

---

Introduction

○○○  
○○○○

Principe et méthode

○○○○○○○○  
○○○○○○○○

Le problème en 1D

○○○  
○○○

Le problème en 2D

○○○  
○○○○

Conclusion

●○○  
○○○○○

Sur l'apprentissage

## Conclusion sur l'apprentissage

Introduction

○○○  
○○○○

Principe et méthode

○○○○○○○○  
○○○○○○○○

Le problème en 1D

○○○  
○○○

Le problème en 2D

○○○  
○○○○

Conclusion

○●○  
○○○○

Sur l'apprentissage

# Bilan de l'apprentissage

Par ordre croissant de réussite :

- Régression 1D : Permet de détecter la hauteur du crâneau
- Classification 2D : Permet de localiser l'ordonnée du crâneau
- Régression 2D : Permet de prédire tous les attributs essentiels du crâneau (abscisse, ordonnée, et hauteur)

Introduction  
○○○  
○○○○

Principe et méthode  
○○○○○○○○  
○○○○○○○○

Le problème en 1D  
○○○  
○○○

Le problème en 2D  
○○○  
○○○○

Conclusion  
○●○  
○○○○

Sur l'apprentissage

# Bilan de l'apprentissage

Par ordre croissant de réussite :

- 1 Régression 1D** : Permet de détecter la hauteur du crâneau
- 2 Classification 2D** : Permet de localiser l'ordonnée du crâneau
- 3 Régression 2D** : Permet de prédire tous les attributs essentiels du crâneau (abscisse, ordonnée, et hauteur)

Introduction  
○○○  
○○○○

Principe et méthode  
○○○○○○○○  
○○○○○○○○

Le problème en 1D  
○○○  
○○○

Le problème en 2D  
○○○  
○○○○

Conclusion  
○●○  
○○○○

Sur l'apprentissage

# Bilan de l'apprentissage

Par ordre croissant de réussite :

- 1 Régression 1D** : Permet de détecter la hauteur du crâneau
- 2 Classification 2D** : Permet de localiser l'ordonnée du crâneau
- 3 Régression 2D** : Permet de prédire tous les attributs essentiels du crâneau (abscisse, ordonnée, et hauteur)

Introduction  
○○○  
○○○○

Principe et méthode  
○○○○○○○○  
○○○○○○○○

Le problème en 1D  
○○○  
○○○

Le problème en 2D  
○○○  
○○○○

Conclusion  
○●○  
○○○○

Sur l'apprentissage

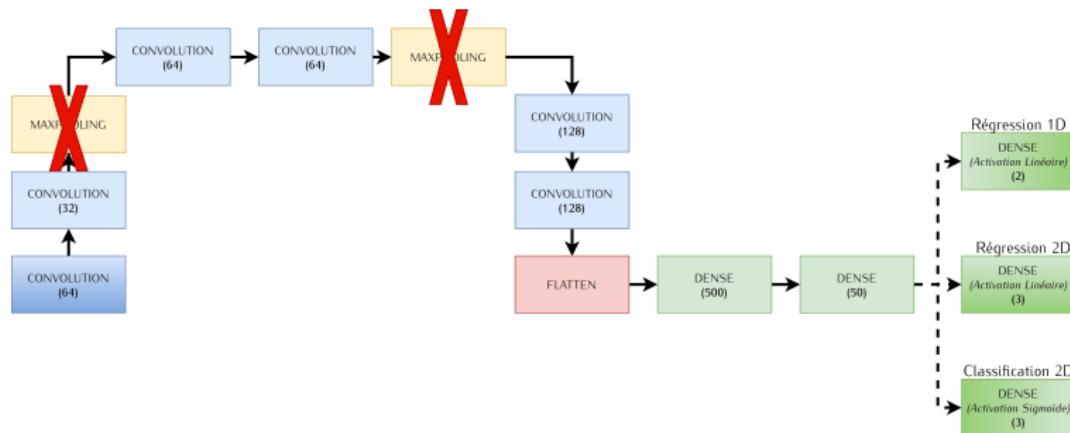
# Bilan de l'apprentissage

Par ordre croissant de réussite :

- 1 **Régression 1D** : Permet de détecter la hauteur du crâneau
- 2 **Classification 2D** : Permet de localiser l'ordonnée du crâneau
- 3 **Régression 2D** : Permet de prédire tous les attributs essentiels du crâneau (abscisse, ordonnée, et hauteur)

## Sur l'apprentissage

# Absence de régularisation



Architecture originale proposée par M.Vigon

Introduction

○○○  
○○○○

Principe et méthode

○○○○○○○○  
○○○○○○○○

Le problème en 1D

○○○  
○○○

Le problème en 2D

○○○  
○○○○

Conclusion

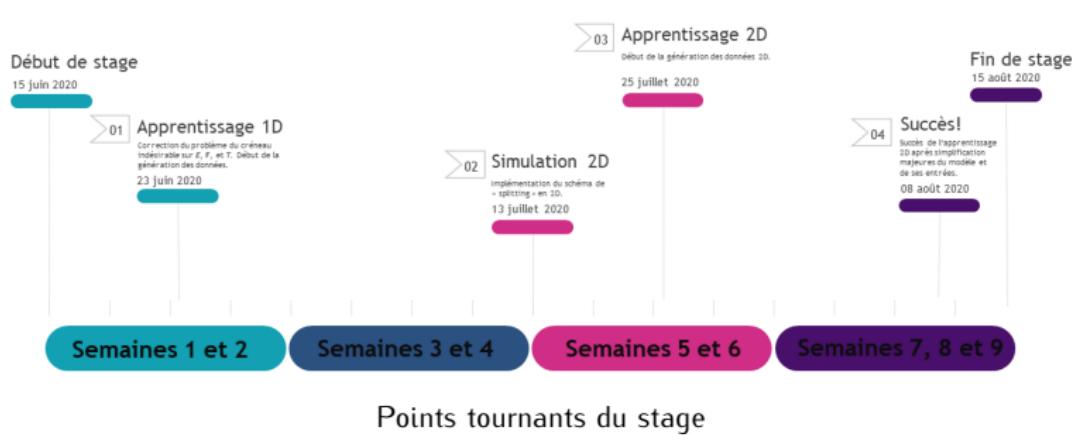
○○○  
●○○○○

Sur le stage

## Conclusion sur le stage

Sur le stage

# Déroulement du stage



Introduction

○○○  
○○○

Principe et méthode

○○○○○○○○  
○○○○○○○○

Le problème en 1D

○○○  
○○○

Le problème en 2D

○○○  
○○○○

Conclusion

○○○  
○○●○○

Sur le stage

# Apports et enseignements

- Développement C++ et Python
- Équations aux dérivées partielles
- Réseaux de neurones
- Expérience dans un milieu de recherche

Sur le stage

# Apports et enseignements

- Développement C++ et Python
- Équations aux dérivées partielles
- Réseaux de neurones
- Expérience dans un milieu de recherche

Sur le stage

# Apports et enseignements

- Développement C++ et Python
- Équations aux dérivées partielles
- Réseaux de neurones
- Expérience dans un milieu de recherche



Sur le stage

# Apports et enseignements

- Développement C++ et Python
- Équations aux dérivées partielles
- Réseaux de neurones
- Expérience dans un milieu de recherche



Sur le stage

# Perspectives

- Amélioration des résultats (différentes formes d'obstacles, d'opacités)
- Déploiement d'un tomographe en milieu médical
- Stage, thèse ultérieures ...

Sur le stage

# Perspectives

- Amélioration des résultats (différentes formes d'obstacles, d'opacités)
- Déploiement d'un tomographe en milieu médical
- Stage, thèse ultérieures ...

Sur le stage

# Perspectives

- Amélioration des résultats (différentes formes d'obstacles, d'opacités)
- Déploiement d'un tomographe en milieu médical
- Stage, thèse ultérieures ...

Introduction

○○○  
○○○○

Principe et méthode

○○○○○○○○  
○○○○○○○○

Le problème en 1D

○○○  
○○○

Le problème en 2D

○○○  
○○○○

Conclusion

○○○  
○○○●

Sur le stage

Merci pour votre attention

-  Franck, E. (Oct. 2012). "Construction et analyse numérique de schéma asymptotic preserving sur maillages non structurés. Application au transport linéaire et aux systèmes de Friedrichs". In: *Archive ouverte HAL*, pp. 12, 160–161. URL: <https://tel.archives-ouvertes.fr/file/index/docid/744371/filename/theseFranckv3.pdf>.
-  Ganesh, Prakhar (Oct. 2019). "Types of Convolution Kernels : Simplified". In: URL: <https://towardsdatascience.com/types-of-convolution-kernels-simplified-f040cb307c37>.
-  Geron, Aurelien (2017). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, Inc, pp. 227–325. URL: [https://books.google.fr/books/about/Hands\\_On\\_Machine\\_Learning\\_with\\_Scikit\\_Le.html?id=HnetDwAAQBAJ&printsec=frontcover&source=kp\\_read\\_button&redir\\_esc=y](https://books.google.fr/books/about/Hands_On_Machine_Learning_with_Scikit_Le.html?id=HnetDwAAQBAJ&printsec=frontcover&source=kp_read_button&redir_esc=y).
-  Packt (s.d.). "Convolutional neural networks". In: URL: <https://subscription.packtpub.com/book/game-development/9781789138139/4/ch04lvl1sec31/convolutional-neural-networks>.
-  Turpault, R. (Feb. 2004). "Modelisation, approximation numerique et applications du transfert radiatif en desequilibre spectral couple avec l'hydrodynamique". In: *Archive ouverte HAL*, pp. 13–22. URL: <http://docplayer.fr/177010094-Modelisation-approximation-numerique-et-applications-du-transfert-radiatif-en-desequilibre-spectral-couple-avec-l-hydrodynamique.html>.