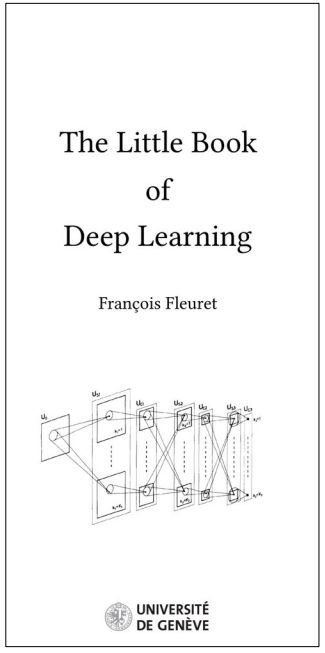
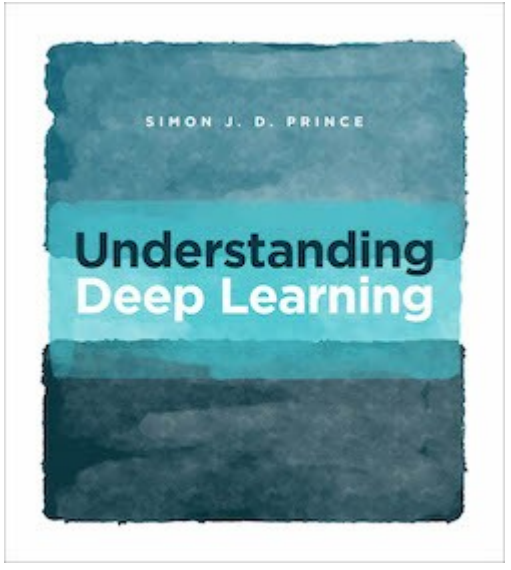
Introduction aux réseaux de neurones pour l'apprentissage supervisé

Guillaume Bourmaud

Livres



https://fleuret.org/francois/lbdl.html



https://udlbook.github.io/udlbook/

PLAN

I. Introduction

II. Apprentissage supervisé

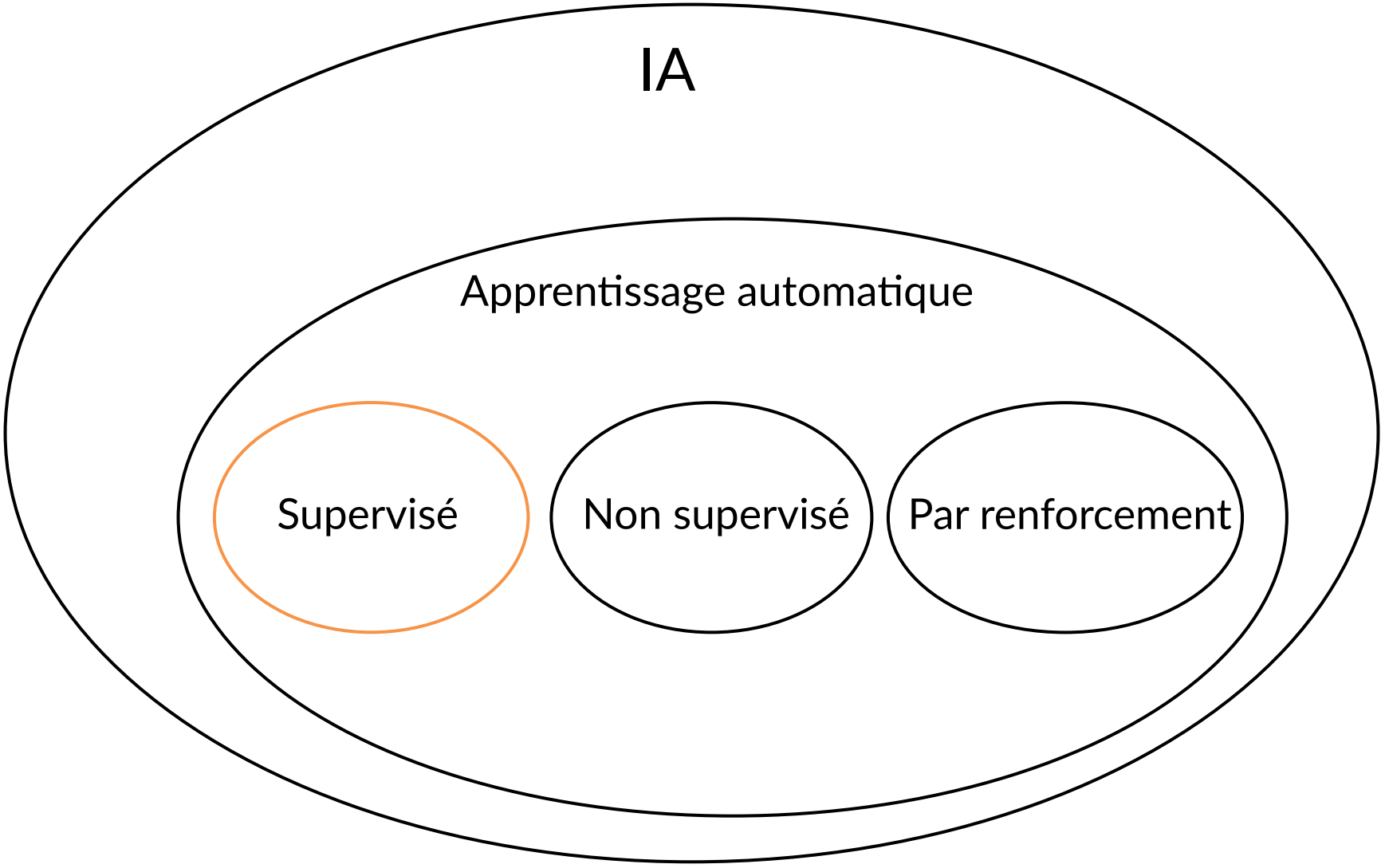
III. Approches paramétriques

IV. Réseaux de neurones

V. Risques

I) Introduction

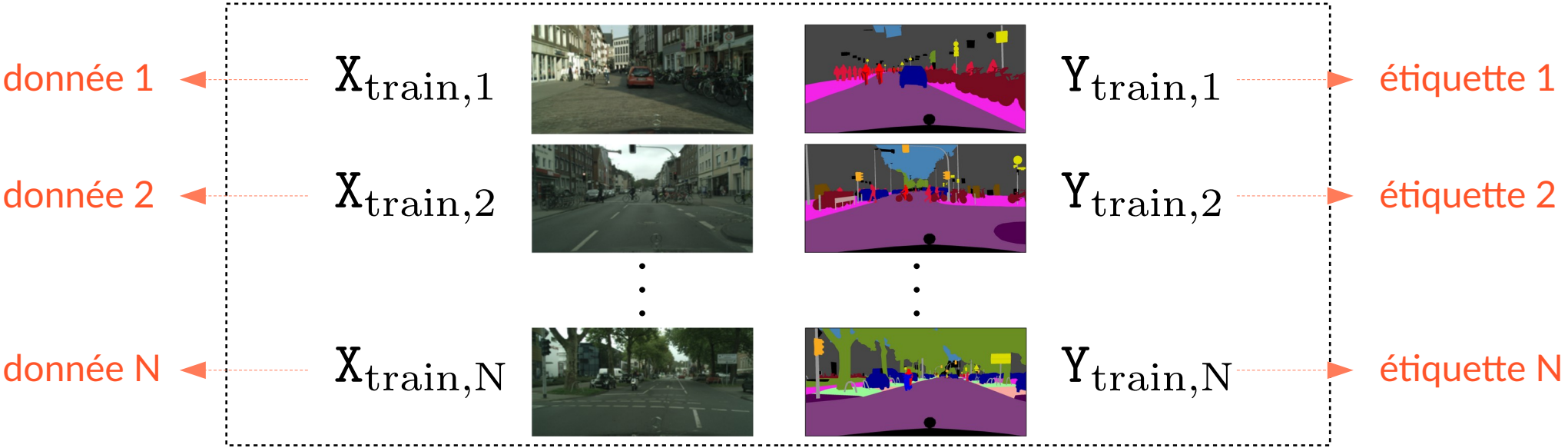
I)


IA
Apprentissage automatique
Supervisé
Non supervisé
Par renforcement

II) Apprentissage supervisé

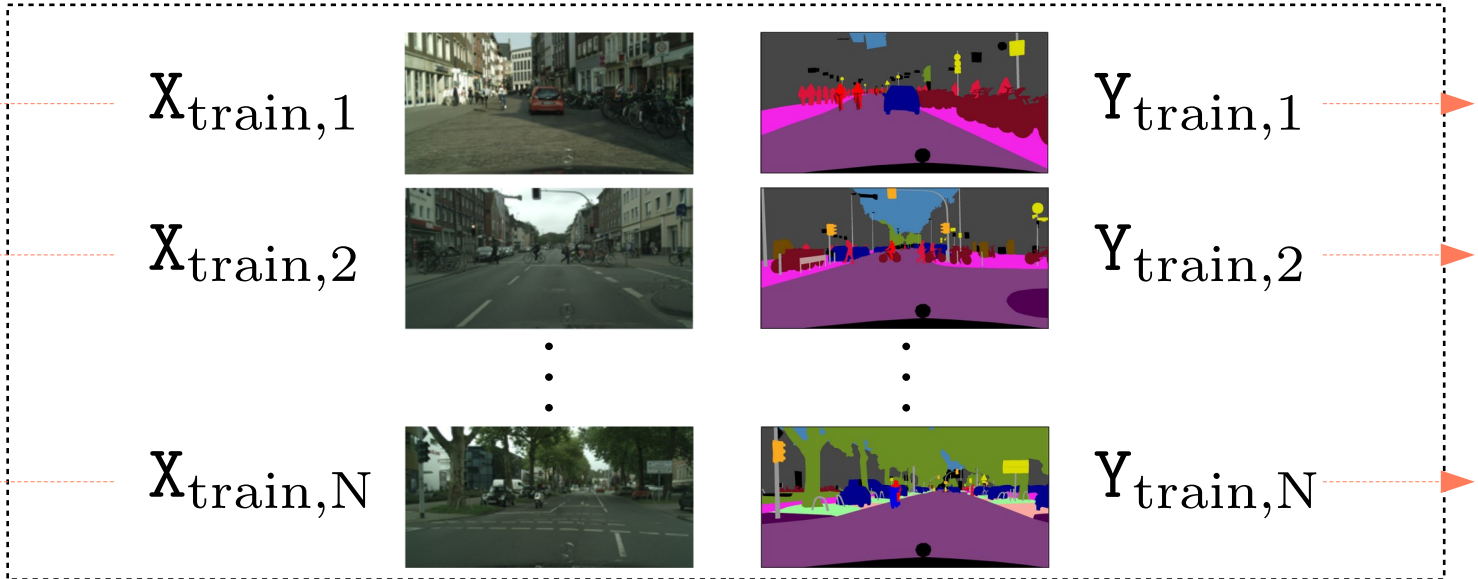
II)

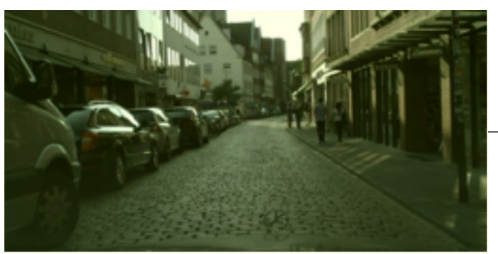
Apprentissage supervisé



II)

Apprentissage supervisé





Nouvelle donnée

$\mathbf{x}_{test}$

f

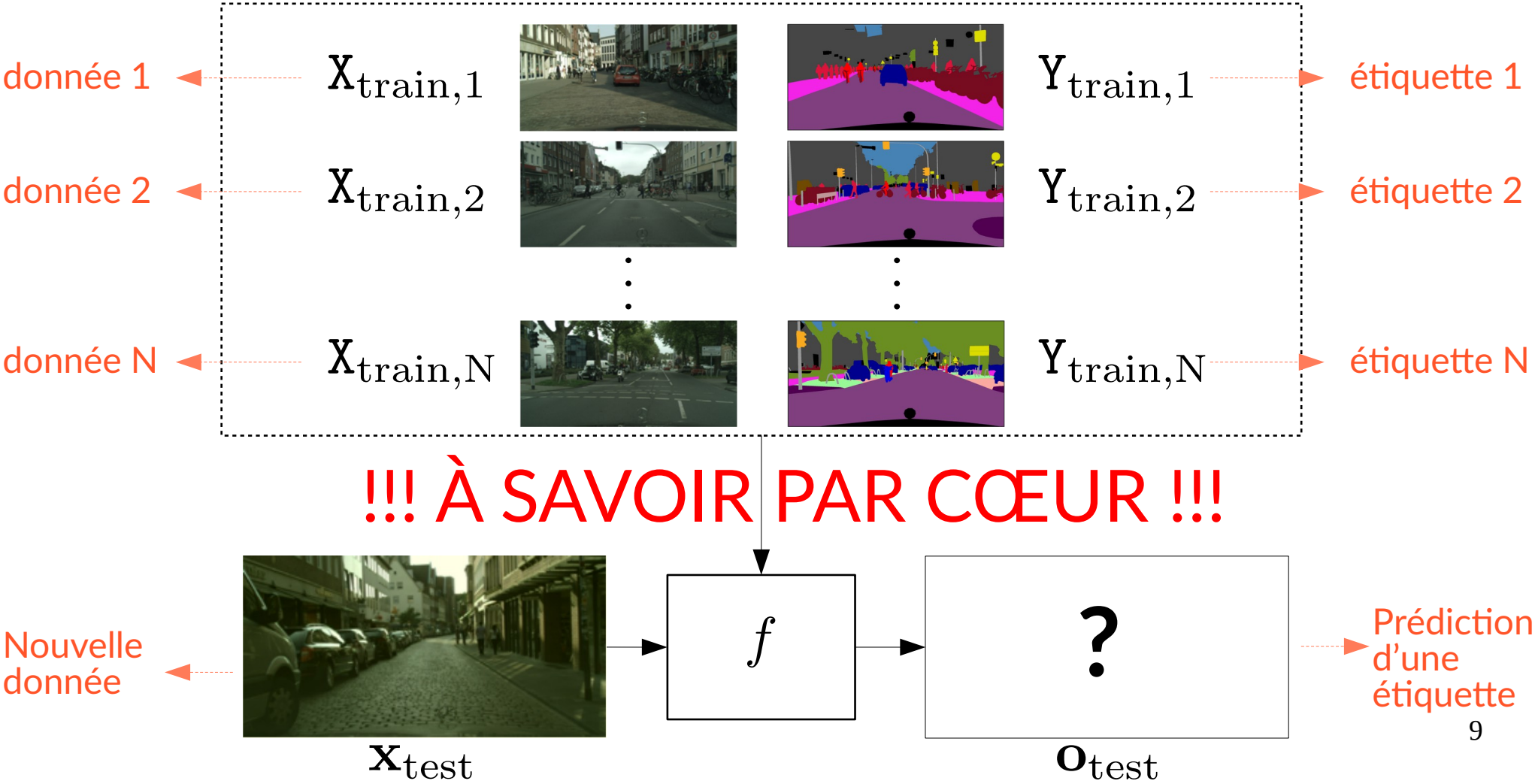
?

$\mathbf{o}_{test}$

Prédiction d'une étiquette

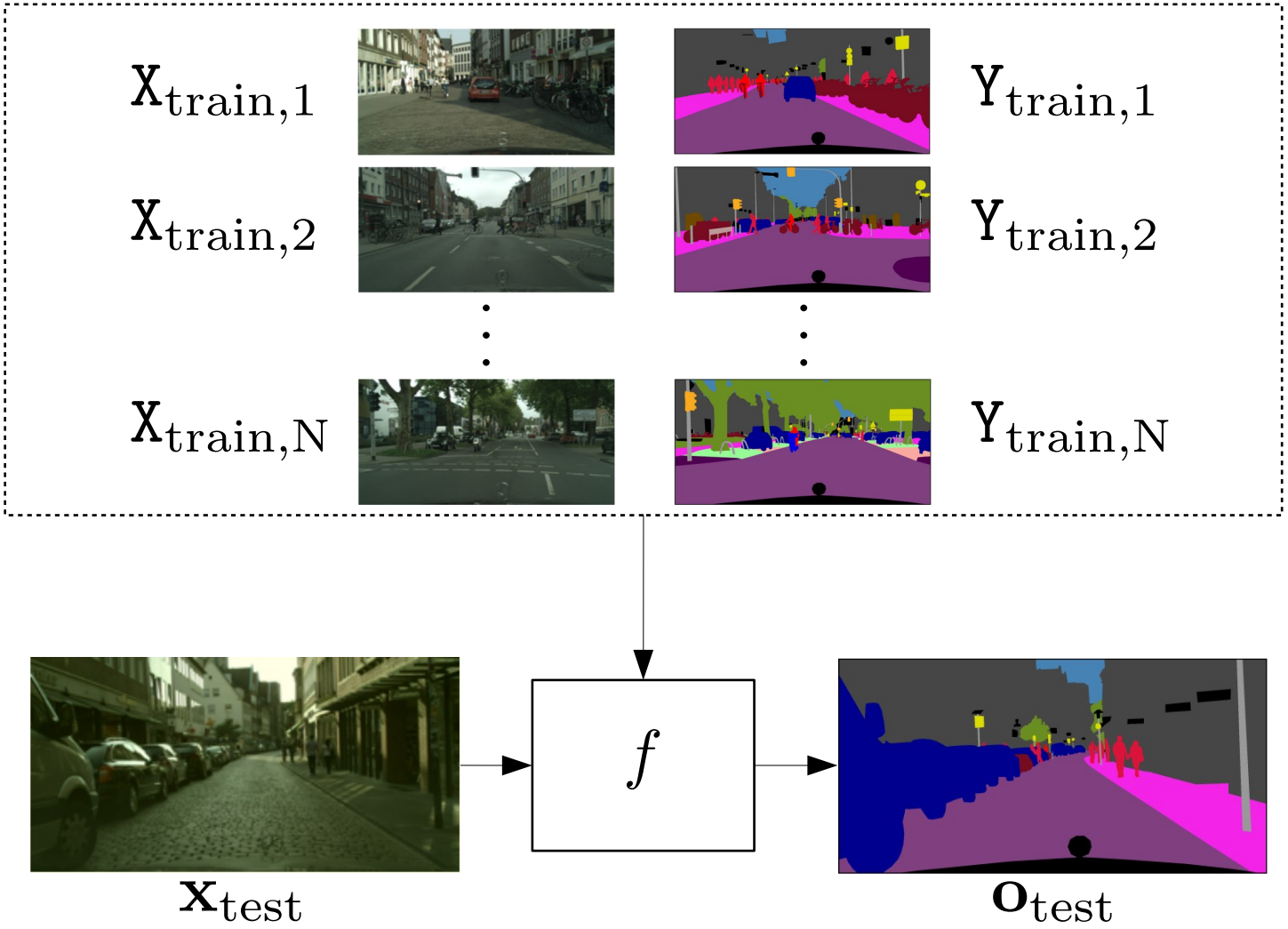
II)

Apprentissage supervisé



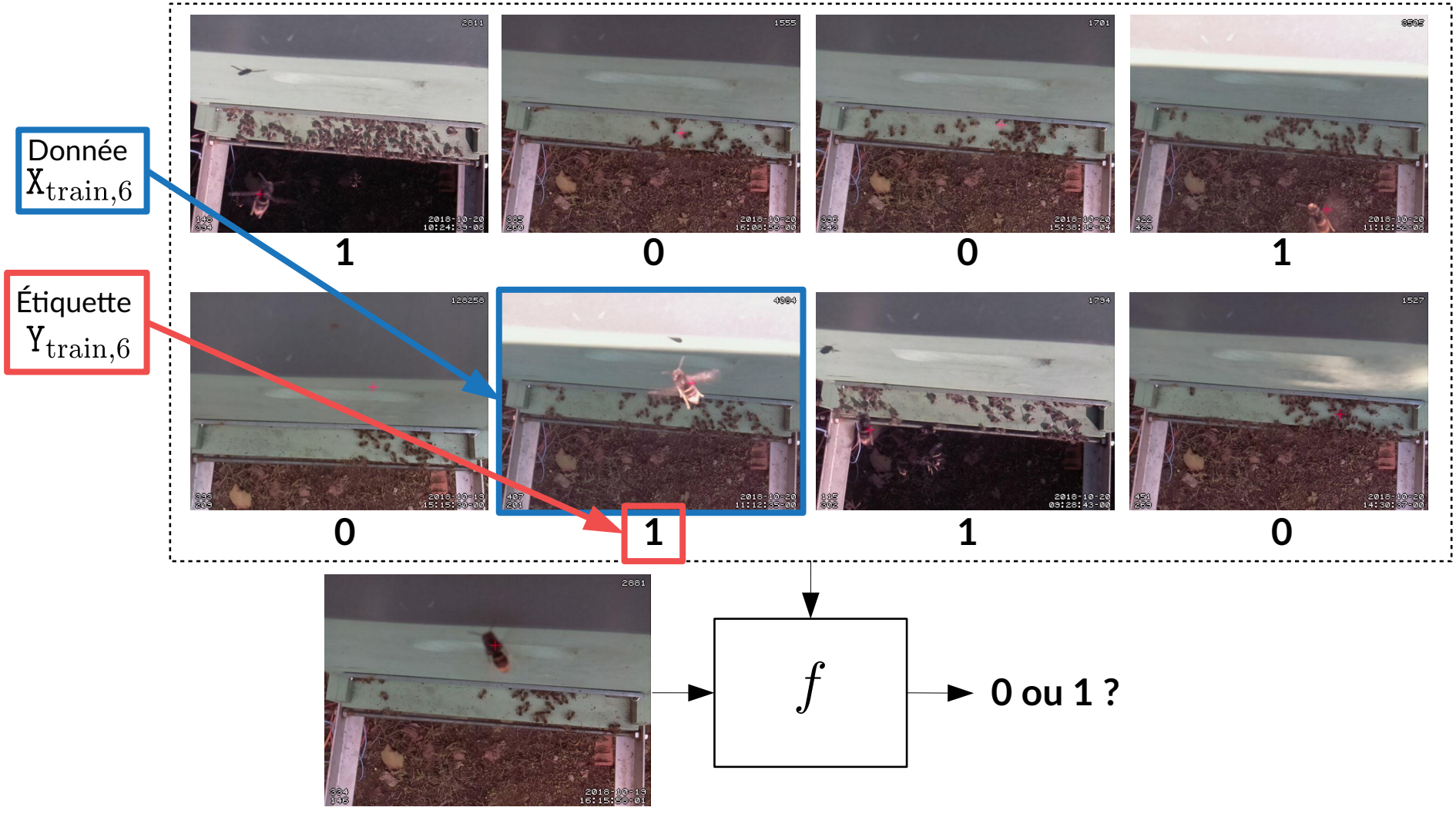
II)

Apprentissage supervisé



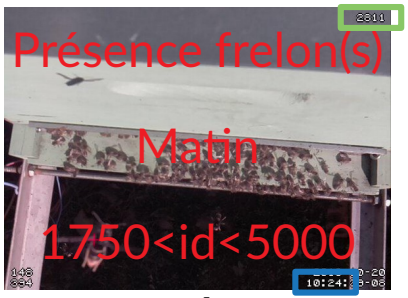
II)

Êtes-vous capable de résoudre ce problème d'apprentissage supervisé ?



II)

Mais de quel problème parle-t-on au juste ?



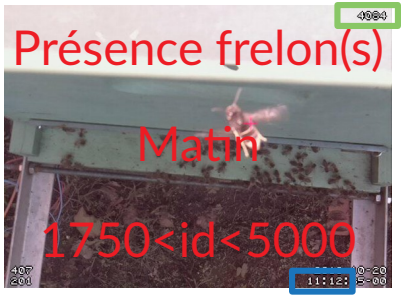
1

0

0

1

0



1

1

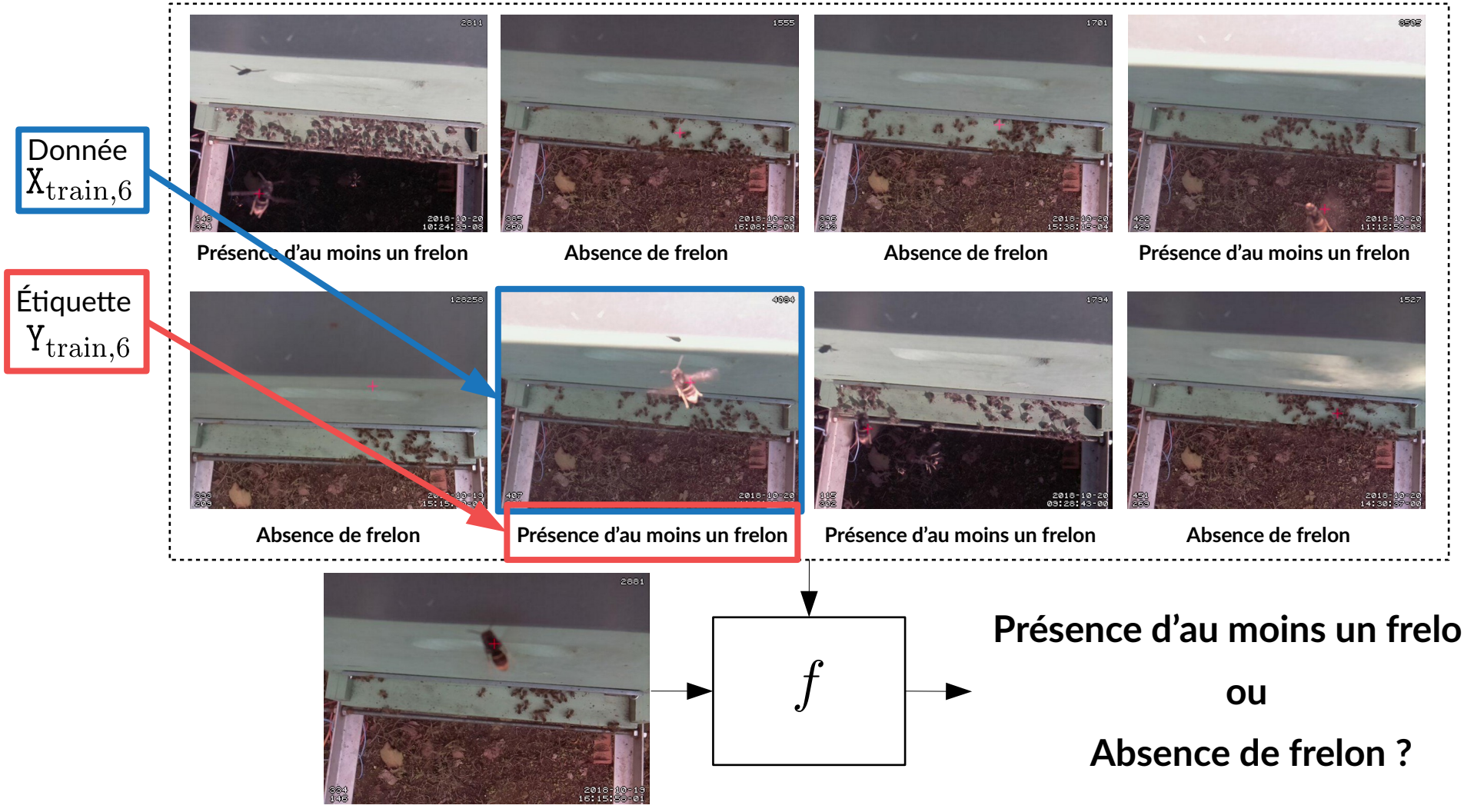
0

f

0 ou 1 ?

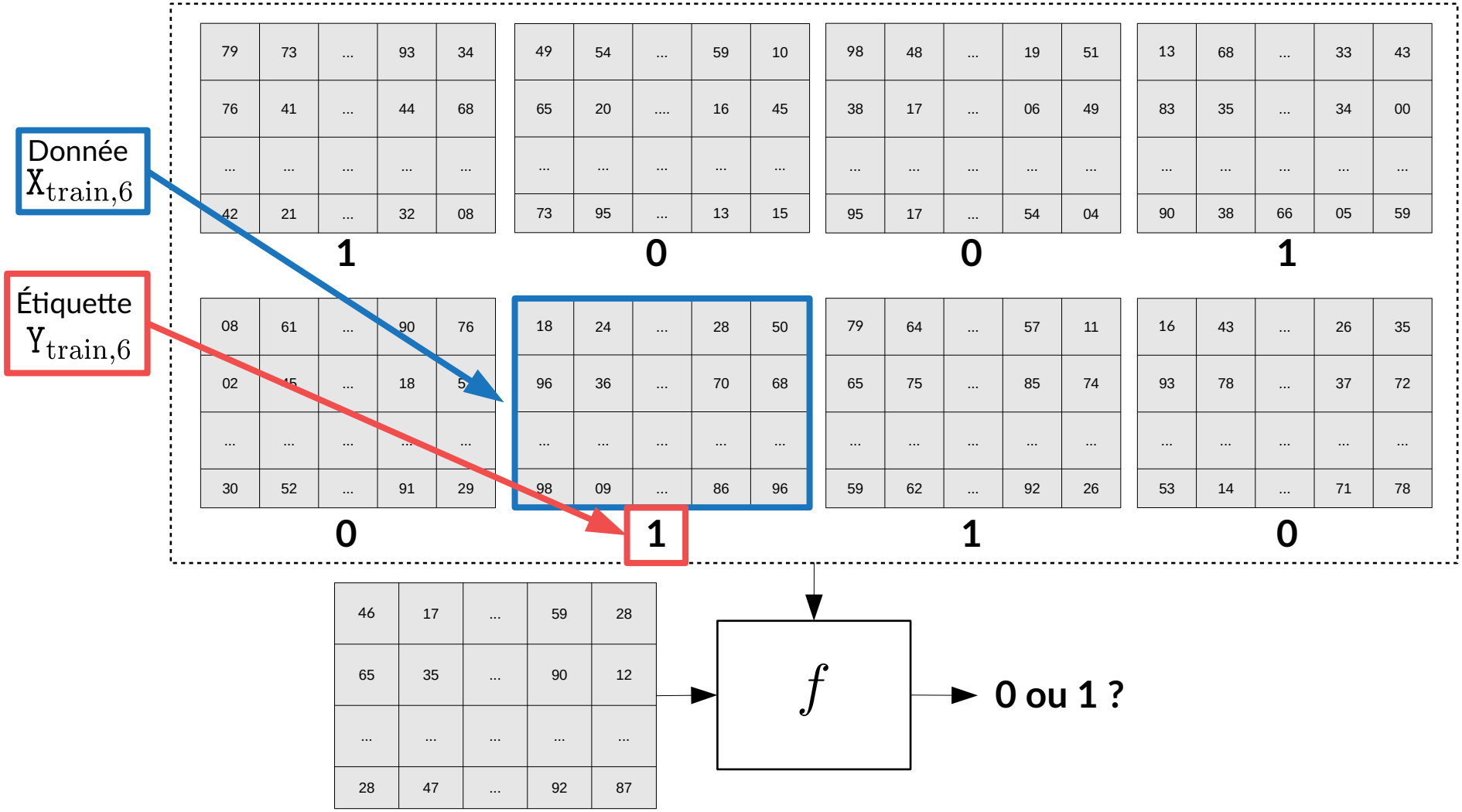
II)

Nous souhaitons que l'ordinateur apprenne à résoudre le problème suivant



II)

Mais une fois mis en forme pour l'ordinateur le problème devient



II)

Résumé

- L'ordinateur apprend une fonction qui effectue des calculs sur des tableaux de valeurs numériques et produit une valeur numérique en sortie.

- Nous interprétons cette valeur numérique en sortie comme une réponse à la question : « Y a-t-il (au moins) un frelon présent dans l'image ? »

- Il ne s'agit que d'une interprétation...

Tesla said autopilot was activated during a fatal Model X crash last week in California.

III) Approches paramétriques

III)

Fonction paramétrique

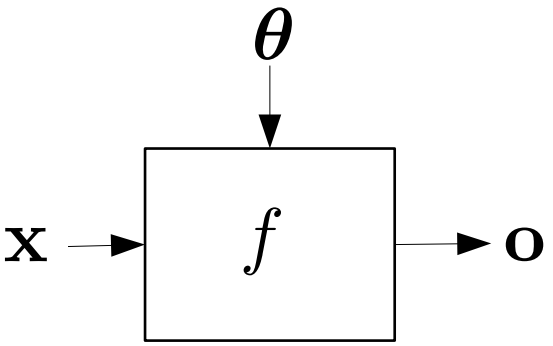
Mathématique

$$\mathbf{o} = f\left(\mathbf{x}; \boldsymbol{\theta}\right)$$

Informatique
(Python)

```
def f(x,theta):
    ...
    ...
    o = ...
    return o
```

Graphique
(Graphe de calcul)



III) Exemple de fonction paramétrique : transformation affine

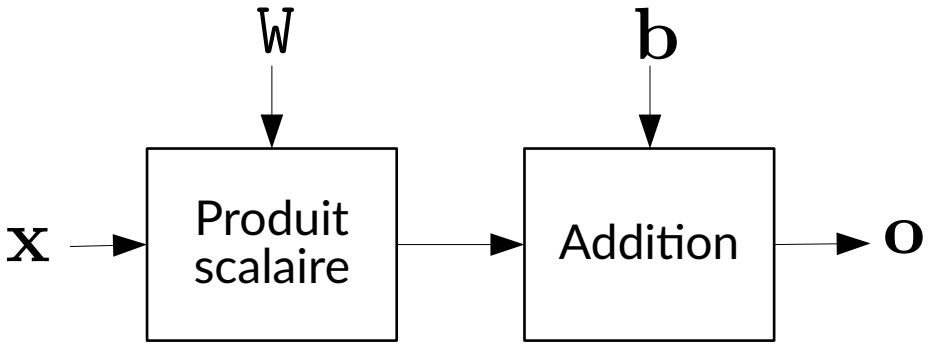
Mathématique

$$\mathbf{o} = f\left(\mathbf{x}; \boldsymbol{\theta} = \{\mathtt{W}, \mathbf{b}\}\right) = \mathtt{W}\mathbf{x} + \mathbf{b}$$

Informatique
(Python)

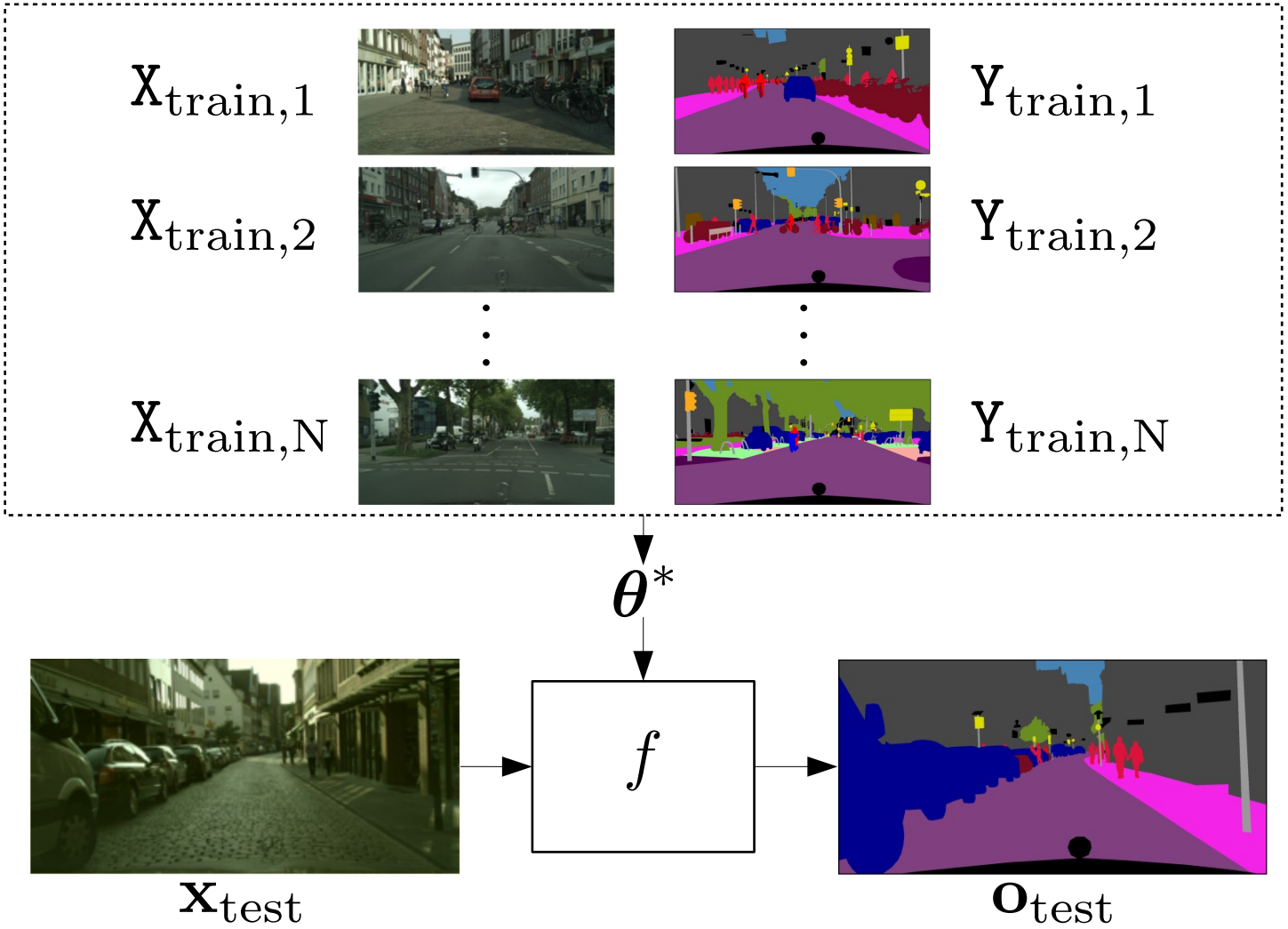
```
def f(x,W,b):
    o = x.dot(W) + b
    return o
```

Graphique
(Graphe de calcul)



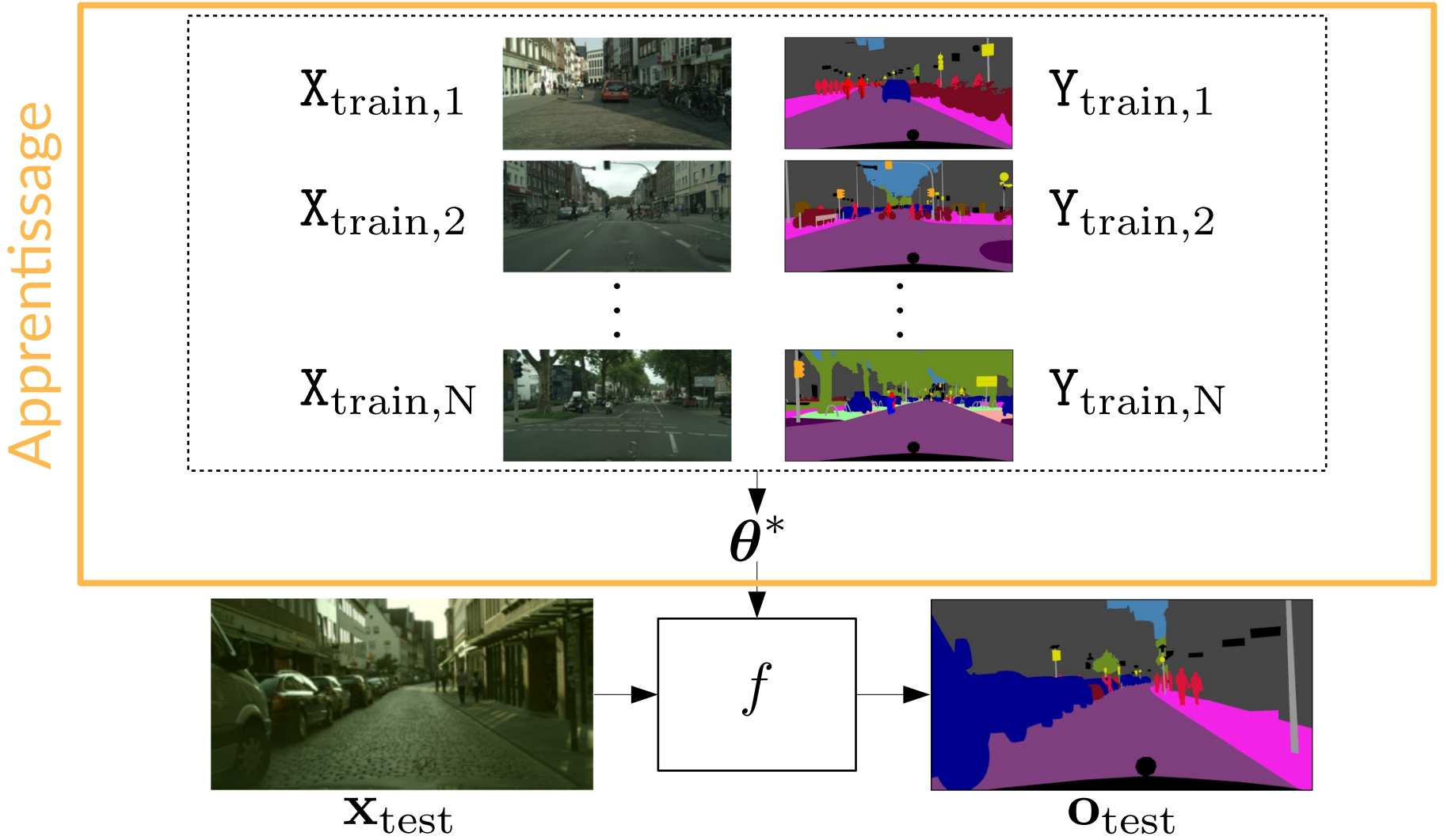
III)

Approches paramétriques



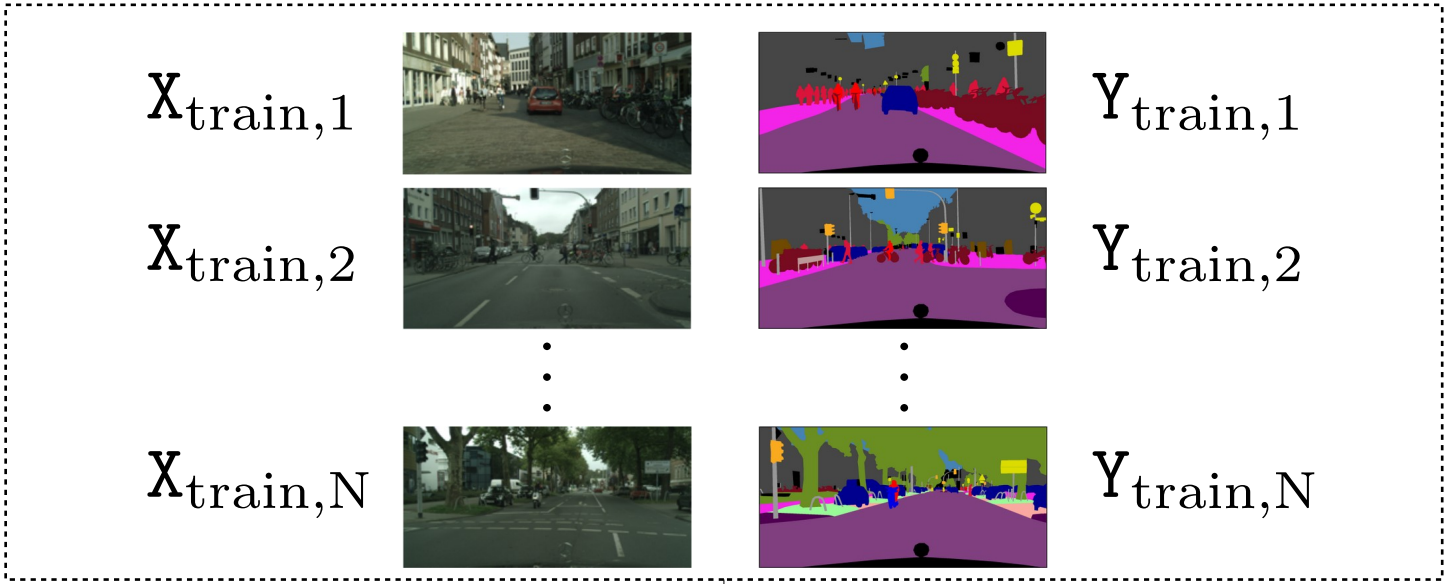
III)

Approches paramétriques

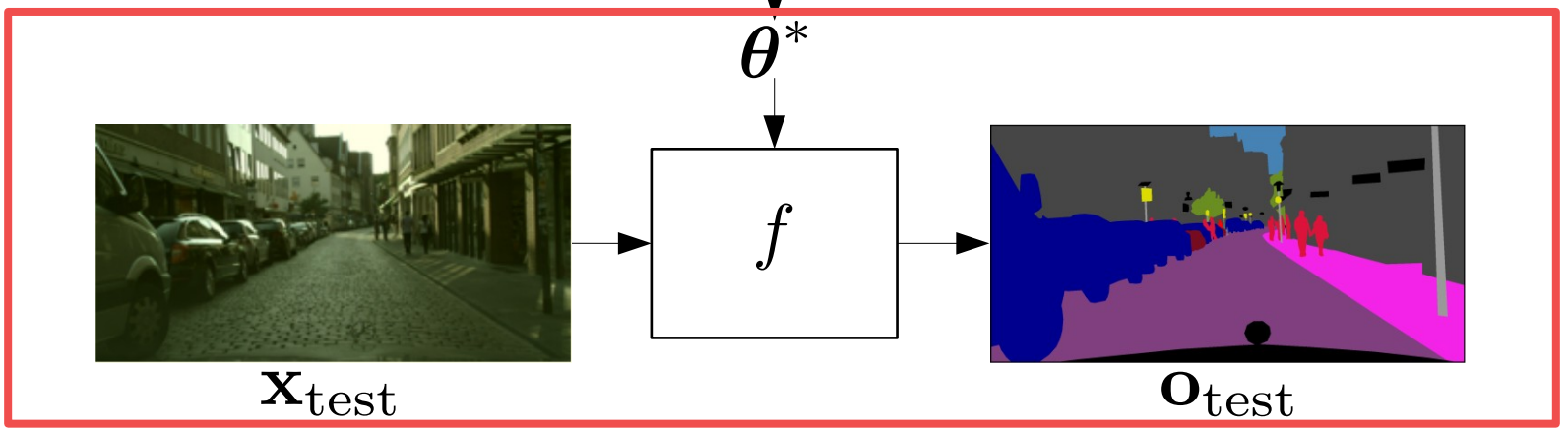


III)

Approches paramétriques

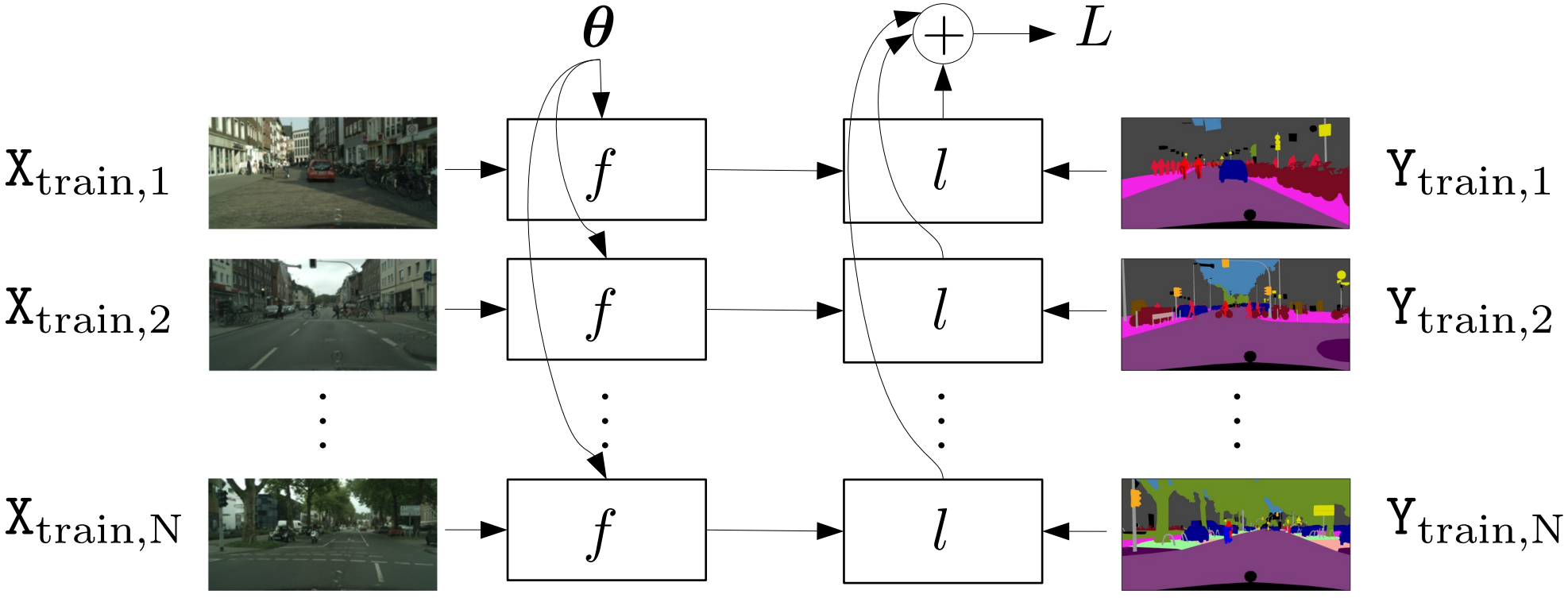


Inférence



III)

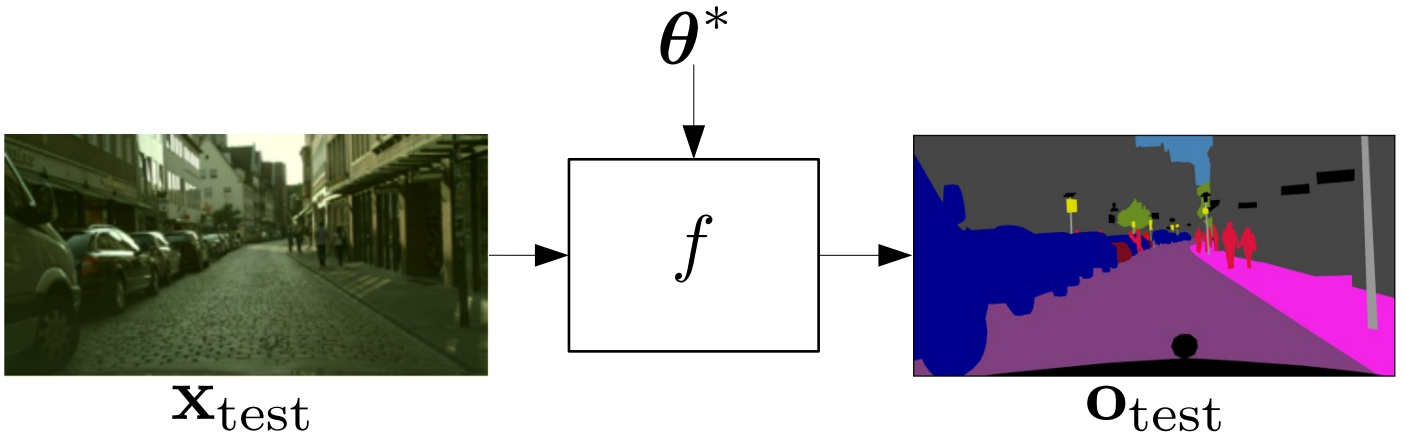
Étape d'apprentissage ("Training time")



$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{N} l\left(\mathrm{Y}_{\mathrm{train},i}, f\left(\mathrm{X}_{\mathrm{train},i}; \boldsymbol{\theta}\right)\right)$$

III)

Étape d'inférence ("Test time")



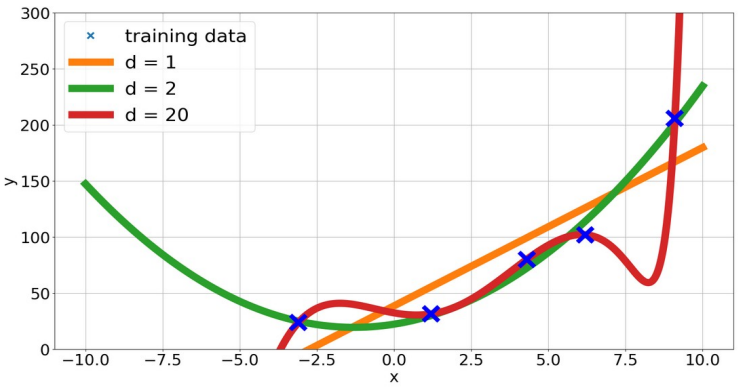
$$\mathbf{o}_{\text{test}} = f\left(\mathbf{x}_{\text{test}}; \boldsymbol{\theta}^*\right)$$

III)

Exemple de régression : 5 données, $\mathtt{X} \in \mathbb{R}$ et $\mathtt{Y} \in \mathbb{R}$

$\mathtt{X}_{\text{train},1} = -3.1$ $\mathtt{X}_{\text{train},2} = 1.2$ $\mathtt{X}_{\text{train},3} = 4.3$ $\mathtt{X}_{\text{train},4} = 6.2$ $\mathtt{X}_{\text{train},5} = 9.1$

$\mathtt{Y}_{\text{train},1} = 23.7$ $\mathtt{Y}_{\text{train},2} = 31.3$ $\mathtt{Y}_{\text{train},3} = 79.9$ $\mathtt{Y}_{\text{train},4} = 101.9$ $\mathtt{Y}_{\text{train},5} = 205.5$



III)

Exemple de régression : 5 données, $\mathtt{X} \in \mathbb{R}$ et $\mathtt{Y} \in \mathbb{R}$

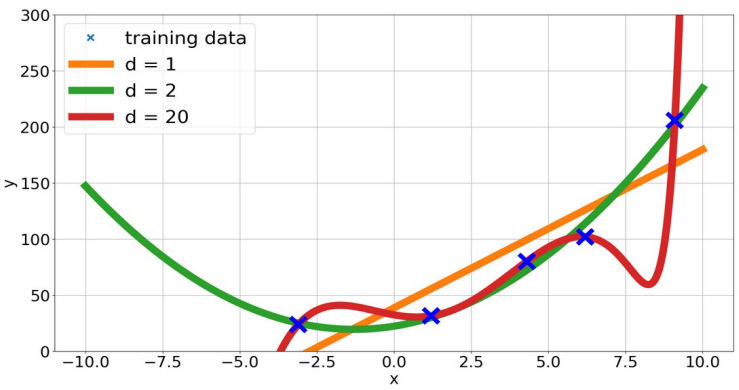
$\mathtt{X}_{\text{train},1} = -3.1 \quad \mathtt{X}_{\text{train},2} = 1.2 \quad \mathtt{X}_{\text{train},3} = 4.3 \quad \mathtt{X}_{\text{train},4} = 6.2 \quad \mathtt{X}_{\text{train},5} = 9.1$

$\mathtt{Y}_{\text{train},1} = 23.7 \quad \mathtt{Y}_{\text{train},2} = 31.3 \quad \mathtt{Y}_{\text{train},3} = 79.9 \quad \mathtt{Y}_{\text{train},4} = 101.9 \quad \mathtt{Y}_{\text{train},5} = 205.5$

Choix de la fonction $f(x;\boldsymbol{\theta}) = \boldsymbol{\theta}^\top \phi(x)$

où $\phi(x) = \begin{bmatrix} 1 & x & \dots & x^d \end{bmatrix}^\top$

hyper-paramètre : d



III)

Exemple de régression : 5 données, $\mathtt{X} \in \mathbb{R}$ et $\mathtt{Y} \in \mathbb{R}$

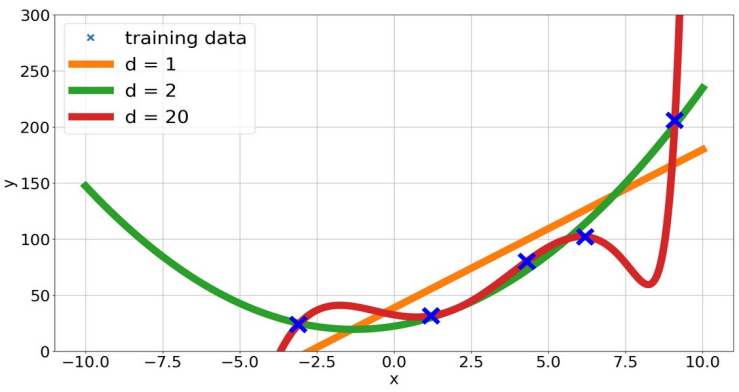
$$\mathtt{X}_{\text{train},1} = -3.1 \quad \mathtt{X}_{\text{train},2} = 1.2 \quad \mathtt{X}_{\text{train},3} = 4.3 \quad \mathtt{X}_{\text{train},4} = 6.2 \quad \mathtt{X}_{\text{train},5} = 9.1$$

$$\mathtt{Y}_{\text{train},1} = 23.7 \quad \mathtt{Y}_{\text{train},2} = 31.3 \quad \mathtt{Y}_{\text{train},3} = 79.9 \quad \mathtt{Y}_{\text{train},4} = 101.9 \quad \mathtt{Y}_{\text{train},5} = 205.5$$

Choix de la fonction $f(x;\boldsymbol{\theta}) = \boldsymbol{\theta}^\top \phi(x)$

où $\phi(x) = \begin{bmatrix} 1 & x & \dots & x^d \end{bmatrix}^\top$

hyper-paramètre : d



Apprentissage

Choix du coût $l(y,o) = (y-o)^2$

Optimisation $\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{5} \left(\mathtt{Y}_{\text{train},i} - \boldsymbol{\theta}^\top \phi(\mathtt{X}_{\text{train},i}) \right)^2$

III)

Exemple de régression : 5 données, $\mathtt{X} \in \mathbb{R}$ et $\mathtt{Y} \in \mathbb{R}$

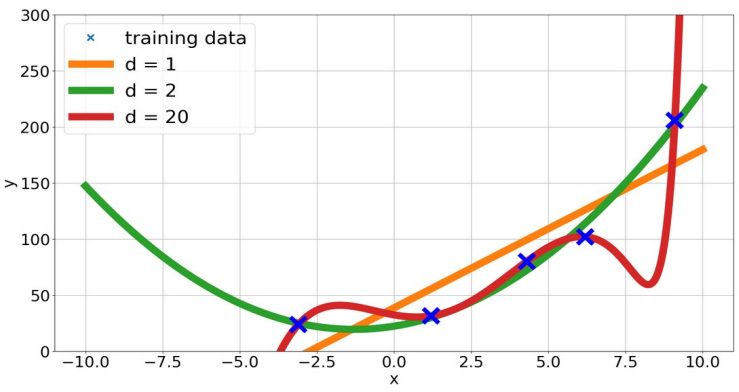
$$\mathtt{X}_{\text{train},1} = -3.1 \quad \mathtt{X}_{\text{train},2} = 1.2 \quad \mathtt{X}_{\text{train},3} = 4.3 \quad \mathtt{X}_{\text{train},4} = 6.2 \quad \mathtt{X}_{\text{train},5} = 9.1$$

$$\mathtt{Y}_{\text{train},1} = 23.7 \quad \mathtt{Y}_{\text{train},2} = 31.3 \quad \mathtt{Y}_{\text{train},3} = 79.9 \quad \mathtt{Y}_{\text{train},4} = 101.9 \quad \mathtt{Y}_{\text{train},5} = 205.5$$

Choix de la fonction $f(x;\boldsymbol{\theta}) = \boldsymbol{\theta}^\top \phi(x)$

où $\phi(x) = \begin{bmatrix} 1 & x & \dots & x^d \end{bmatrix}^\top$

hyper-paramètre : d



Apprentissage

Choix du coût $l(y,o) = (y-o)^2$

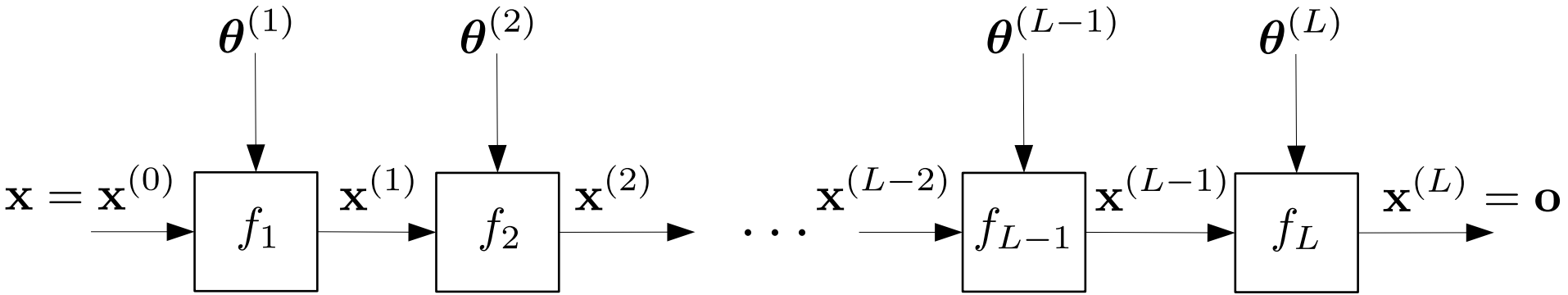
Optimisation $\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{5} \left(\mathtt{Y}_{\text{train},i} - \boldsymbol{\theta}^\top \phi(\mathtt{X}_{\text{train},i}) \right)^2$

Inférence $o_{\text{test}} = \boldsymbol{\theta}^{*\top} \phi(x_{\text{test}}) = \boldsymbol{\theta}^*_0 + \boldsymbol{\theta}^*_1 x_{\text{test}} + \dots + \boldsymbol{\theta}^*_d x^d_{\text{test}}$

IV) Réseaux de neurones

IV)

Choix de la fonction : Réseau de neurones

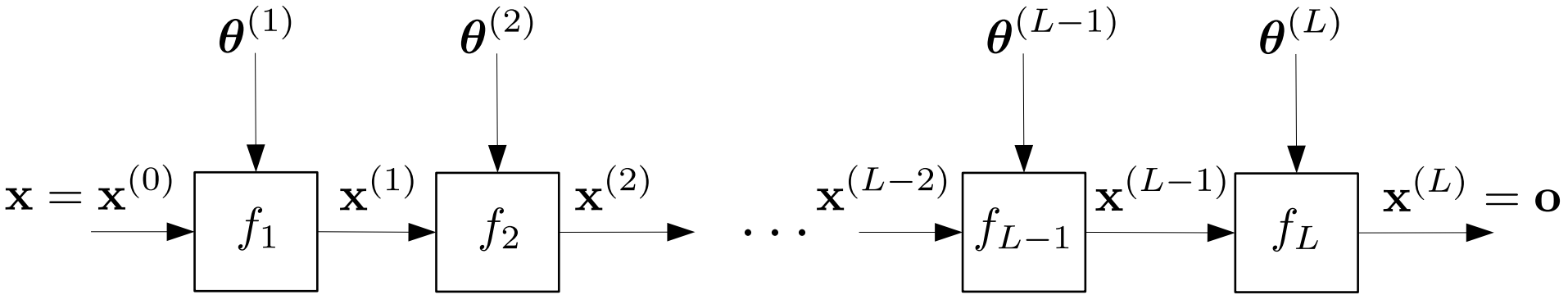


$$f(\mathbf{x};\boldsymbol{\theta}) = f_L\left(f_{L-1}\left(\ldots f_2\left(f_1\left(\mathbf{x};\boldsymbol{\theta}_1\right);\boldsymbol{\theta}_2\right)\ldots;\boldsymbol{\theta}_{L-1}\right);\boldsymbol{\theta}_L\right)$$

Réseau de neurones = Composition de fonctions paramétriques

IV)

Choix de la fonction : Réseau de neurones

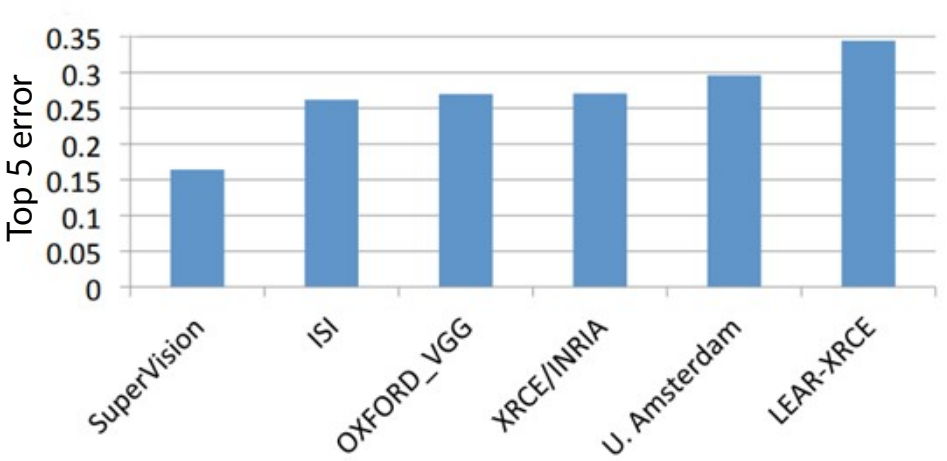


$$f(\mathbf{x};\boldsymbol{\theta}) = f_L\left(f_{L-1}\left(\ldots f_2\left(f_1\left(\mathbf{x};\boldsymbol{\theta}_1\right);\boldsymbol{\theta}_2\right)\ldots;\boldsymbol{\theta}_{L-1}\right);\boldsymbol{\theta}_L\right)$$

Réseau de neurones = Composition de fonctions paramétriques

!!! À SAVOIR PAR CŒUR !!!

ImageNet Large Scale Visual Recognition Challenge 2012



- ImageNet : 1.2 millions d'images annotées, 1000 classes
- SuperVision (*A. Krizhevsky*, *I. Sutskever*, *G. Hinton*, University of Toronto)
  - Réseau de neurones à convolution (AlexNet)
  - 62,3 millions de paramètres (pour information : GPT-3 → 175 000 millions de paramètres)
  - 6 jours d'apprentissage sur 2 GPUs (GTX 580 3GB)

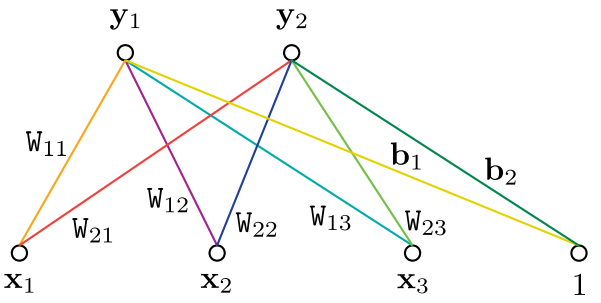
IV)

“Perceptron” multicouche (MLP)

Transformation affine = FC (“Fully Connected”)

$$\mathrm{FC}\left(\mathbf{x}; \boldsymbol{\theta} = \{\mathtt{W}, \mathbf{b}\}\right) = \mathtt{W}\mathbf{x} + \mathbf{b}$$

Pourquoi « Fully Connected » ?



IV)

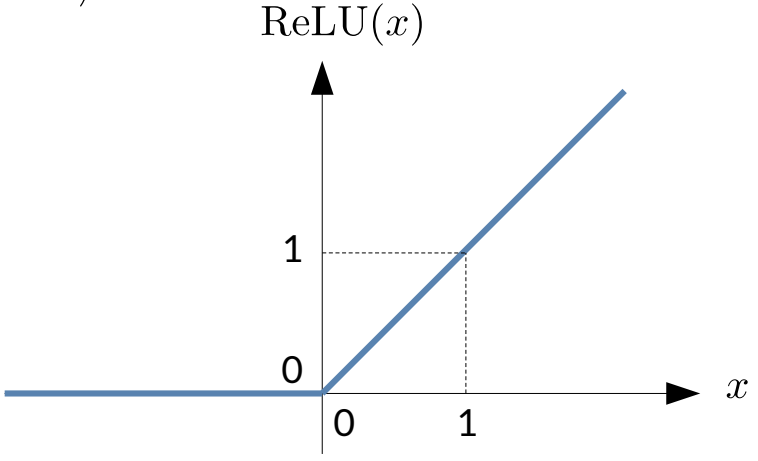
“Perceptron” multicouche (MLP)

Transformation affine = FC (“Fully Connected”)

$$\mathrm{FC}\left(\mathbf{x}; \boldsymbol{\theta} = \{\mathtt{W}, \mathbf{b}\}\right) = \mathtt{W}\mathbf{x} + \mathbf{b}$$

Non-linéarité = ReLU (“Rectified Linear Unit“)

$$\mathrm{ReLU}\left(x\right) = \max\left(0, x\right)$$



IV)

“Perceptron” multicouche (MLP)

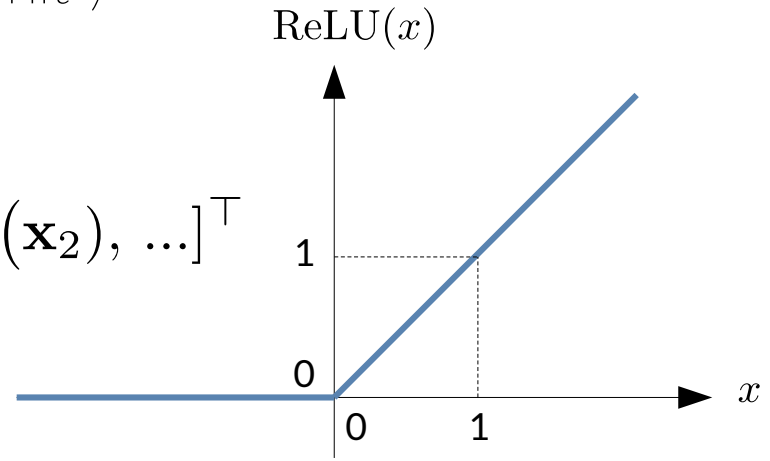
Transformation affine = FC (“Fully Connected”)

$$\mathrm{FC}\left(\mathbf{x}; \boldsymbol{\theta} = \{\mathtt{W}, \mathbf{b}\}\right) = \mathtt{W}\mathbf{x} + \mathbf{b}$$

Non-linéarité = ReLU (“Rectified Linear Unit“)

$$\mathrm{ReLU}\left(x\right) = \max\left(0, x\right)$$

Abus de notation : $\mathrm{ReLU}\left(\mathbf{x}\right) = \left[\mathrm{ReLU}\left(\mathbf{x}_1\right), \mathrm{ReLU}\left(\mathbf{x}_2\right), \ldots\right]^\top$



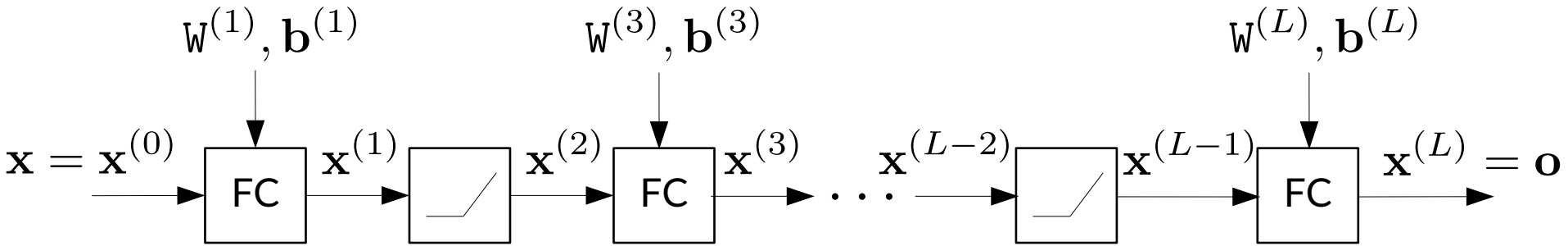
“Perceptron” multicouche (MLP)

Transformation affine = FC (“Fully Connected”)

$$\mathrm{FC}\left(\mathbf{x}; \boldsymbol{\theta} = \{\mathtt{W}, \mathbf{b}\}\right) = \mathtt{W}\mathbf{x} + \mathbf{b}$$

Non-linéarité = ReLU (“Rectified Linear Unit“)

$$\mathrm{ReLU}\left(x\right) = \max\left(0, x\right)$$



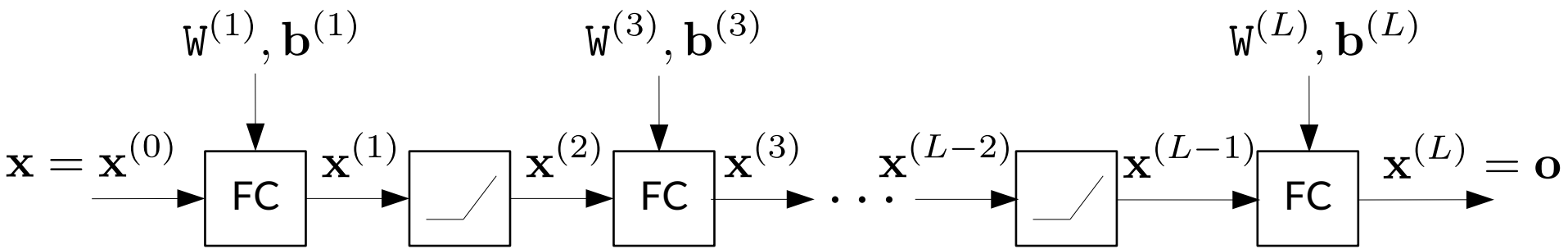
“Perceptron” multicouche (MLP)

Transformation affine = FC (“Fully Connected”)

$$\mathrm{FC}\left(\mathbf{x}; \boldsymbol{\theta} = \{\mathtt{W}, \mathbf{b}\}\right) = \mathtt{W}\mathbf{x} + \mathbf{b}$$

Non-linéarité = ReLU (“Rectified Linear Unit“)

$$\mathrm{ReLU}\left(x\right) = \max\left(0, x\right)$$



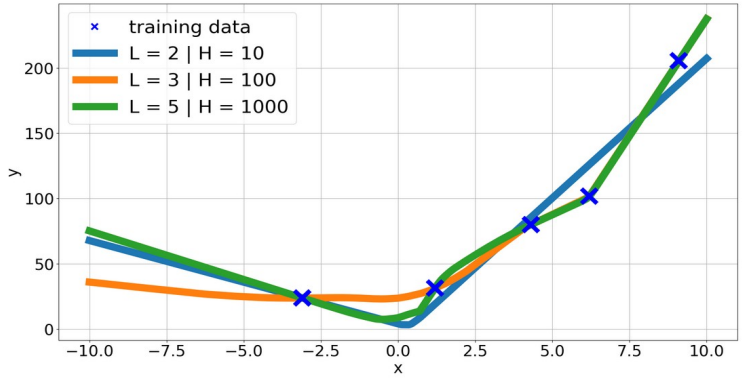
!!! À SAVOIR PAR CŒUR !!!

IV)

Exemple de régression : 5 données, $\mathtt{X} \in \mathbb{R}$ et $\mathtt{Y} \in \mathbb{R}$

$\mathtt{X}_{\text{train},1} = -3.1 \quad \mathtt{X}_{\text{train},2} = 1.2 \quad \mathtt{X}_{\text{train},3} = 4.3 \quad \mathtt{X}_{\text{train},4} = 6.2 \quad \mathtt{X}_{\text{train},5} = 9.1$

$\mathtt{Y}_{\text{train},1} = 23.7 \quad \mathtt{Y}_{\text{train},2} = 31.3 \quad \mathtt{Y}_{\text{train},3} = 79.9 \quad \mathtt{Y}_{\text{train},4} = 101.9 \quad \mathtt{Y}_{\text{train},5} = 205.5$



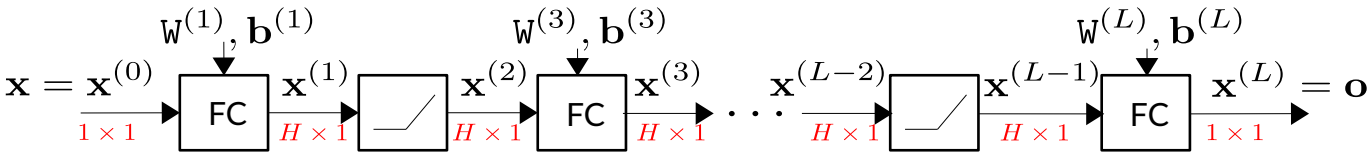
IV)

Exemple de régression : 5 données, $\mathtt{X} \in \mathbb{R}$ et $\mathtt{Y} \in \mathbb{R}$

$$\mathtt{X}_{\text{train},1} = -3.1 \quad \mathtt{X}_{\text{train},2} = 1.2 \quad \mathtt{X}_{\text{train},3} = 4.3 \quad \mathtt{X}_{\text{train},4} = 6.2 \quad \mathtt{X}_{\text{train},5} = 9.1$$

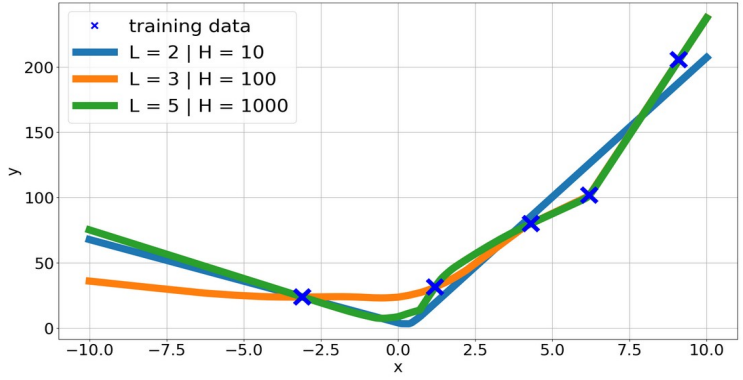
$$\mathtt{Y}_{\text{train},1} = 23.7 \quad \mathtt{Y}_{\text{train},2} = 31.3 \quad \mathtt{Y}_{\text{train},3} = 79.9 \quad \mathtt{Y}_{\text{train},4} = 101.9 \quad \mathtt{Y}_{\text{train},5} = 205.5$$

Choix de la fonction



$$f\,(x; \boldsymbol{\theta}) = \mathrm{MLP}\left(x; \boldsymbol{\theta} = \{\mathtt{W}^{(l)}, \mathbf{b}^{(l)}\}_l\right)$$

hyper-paramètres : L, H

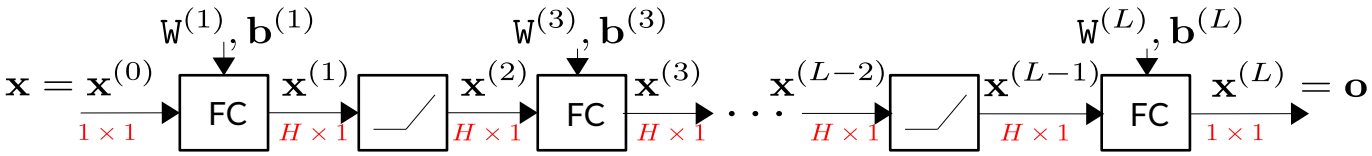


IV)

Exemple de régression : 5 données, $\mathtt{X} \in \mathbb{R}$ et $\mathtt{Y} \in \mathbb{R}$

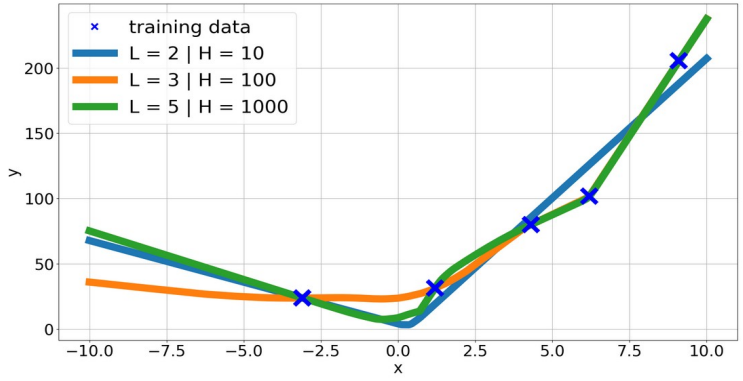
$$\mathtt{X}_{\text{train},1} = -3.1 \quad \mathtt{X}_{\text{train},2} = 1.2 \quad \mathtt{X}_{\text{train},3} = 4.3 \quad \mathtt{X}_{\text{train},4} = 6.2 \quad \mathtt{X}_{\text{train},5} = 9.1$$
$$\mathtt{Y}_{\text{train},1} = 23.7 \quad \mathtt{Y}_{\text{train},2} = 31.3 \quad \mathtt{Y}_{\text{train},3} = 79.9 \quad \mathtt{Y}_{\text{train},4} = 101.9 \quad \mathtt{Y}_{\text{train},5} = 205.5$$

Choix de la fonction



$$f(x; \boldsymbol{\theta}) = \text{MLP}\left(x; \boldsymbol{\theta} = \{\mathtt{W}^{(l)}, \mathbf{b}^{(l)}\}_l\right)$$

hyper-paramètres : L, H



Apprentissage

Choix du coût $l(y, o) = (y - o)^2$

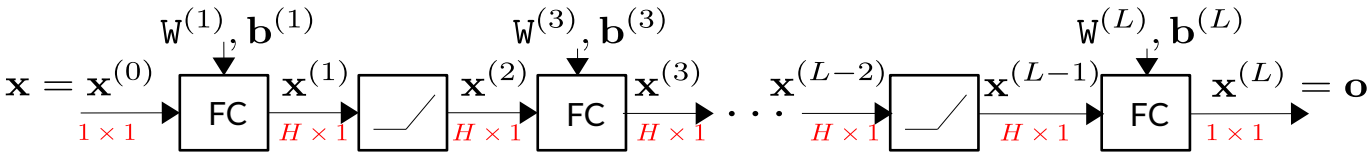
Optimisation $$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{5} \left(\mathtt{Y}_{\text{train},i} - \text{MLP}\left(\mathtt{X}_{\text{train},i}; \boldsymbol{\theta}\right)\right)^2$$

IV)

Exemple de régression : 5 données, $\mathtt{X} \in \mathbb{R}$ et $\mathtt{Y} \in \mathbb{R}$

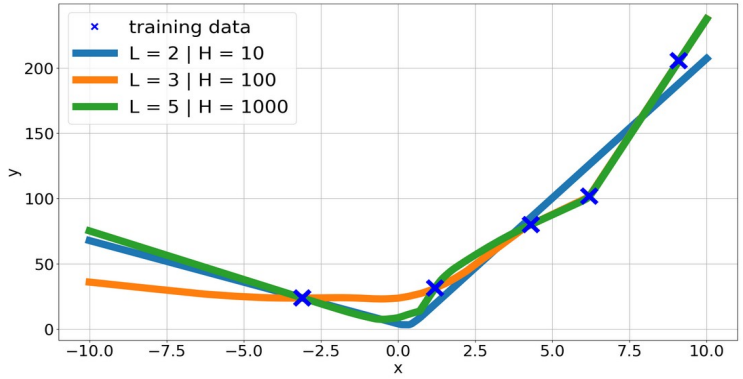
$$\mathtt{X}_{\text{train},1} = -3.1 \quad \mathtt{X}_{\text{train},2} = 1.2 \quad \mathtt{X}_{\text{train},3} = 4.3 \quad \mathtt{X}_{\text{train},4} = 6.2 \quad \mathtt{X}_{\text{train},5} = 9.1$$
$$\mathtt{Y}_{\text{train},1} = 23.7 \quad \mathtt{Y}_{\text{train},2} = 31.3 \quad \mathtt{Y}_{\text{train},3} = 79.9 \quad \mathtt{Y}_{\text{train},4} = 101.9 \quad \mathtt{Y}_{\text{train},5} = 205.5$$

Choix de la fonction



$$f\left(x;\boldsymbol{\theta}\right) = \text{MLP}\left(x;\boldsymbol{\theta} = \{\mathtt{W}^{(l)}, \mathbf{b}^{(l)}\}_l\right)$$

hyper-paramètres : L, H



Apprentissage

Choix du coût $l\left(y,o\right) = \left(y-o\right)^2$

Optimisation $$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{5} \left(\mathtt{Y}_{\text{train},i} - \text{MLP}\left(\mathtt{X}_{\text{train},i};\boldsymbol{\theta}\right)\right)^2$$

Inférence $$o_{\text{test}} = \text{MLP}\left(x_{\text{test}};\boldsymbol{\theta}^*\right)$$

IV)

Exemple de classification : 5 données, $\mathtt{X} \in \mathbb{R}^2$ et $\mathtt{Y} \in \{0, 1, 2\}$

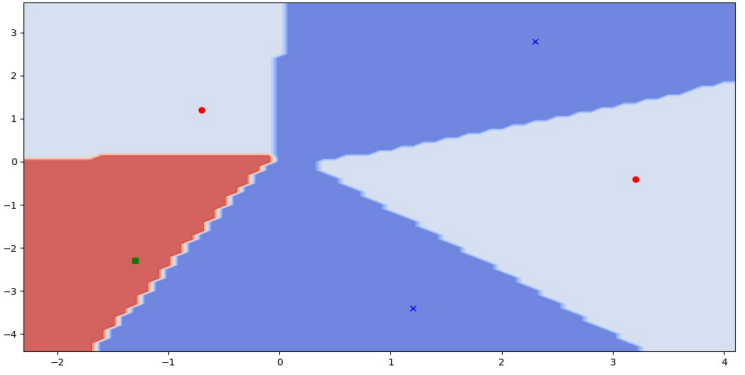
$\mathtt{X}_{\text{train},1} = [1.2, -3.4]$ $\mathtt{Y}_{\text{train},1} = 0$

$\mathtt{X}_{\text{train},2} = [2.3, 2.8]$ $\mathtt{Y}_{\text{train},2} = 0$

$\mathtt{X}_{\text{train},3} = [-0.7, 1.2]$ $\mathtt{Y}_{\text{train},3} = 1$

$\mathtt{X}_{\text{train},4} = [3.2, -0.4]$ $\mathtt{Y}_{\text{train},4} = 1$

$\mathtt{X}_{\text{train},5} = [-1.3, 2.3]$ $\mathtt{Y}_{\text{train},5} = 2$



IV)

Exemple de classification : 5 données, $\mathtt{X} \in \mathbb{R}^2$ et $\mathtt{Y} \in \{0, 1, 2\}$

$\mathtt{X}_{\text{train},1} = [1.2, -3.4]$ $\mathtt{Y}_{\text{train},1} = 0$

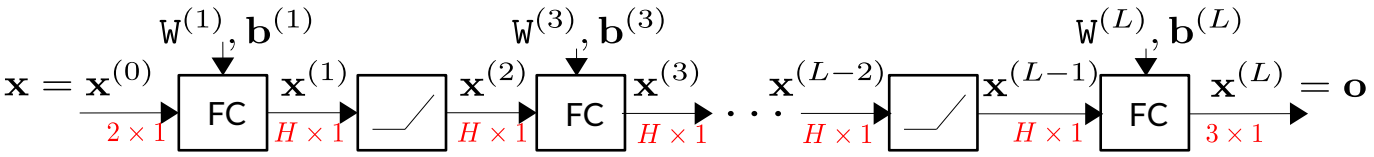
$\mathtt{X}_{\text{train},2} = [2.3, 2.8]$ $\mathtt{Y}_{\text{train},2} = 0$

$\mathtt{X}_{\text{train},3} = [-0.7, 1.2]$ $\mathtt{Y}_{\text{train},3} = 1$

$\mathtt{X}_{\text{train},4} = [3.2, -0.4]$ $\mathtt{Y}_{\text{train},4} = 1$

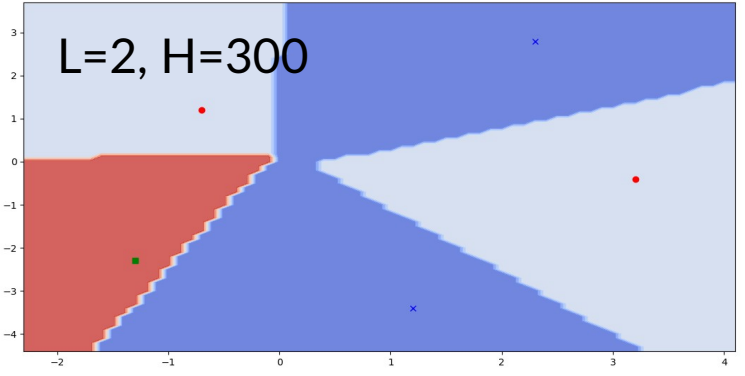
$\mathtt{X}_{\text{train},5} = [-1.3, 2.3]$ $\mathtt{Y}_{\text{train},5} = 2$

Choix de la fonction



$$f(x; \boldsymbol{\theta}) = \text{MLP}\left(x; \boldsymbol{\theta} = \{\mathtt{W}^{(l)}, \mathbf{b}^{(l)}\}_l\right)$$

hyper-paramètres : L, H



IV)

Exemple de classification : 5 données, $\mathtt{X} \in \mathbb{R}^2$ et $\mathtt{Y} \in \{0, 1, 2\}$

$\mathtt{X}_{\text{train},1} = [1.2, -3.4]$ $\mathtt{Y}_{\text{train},1} = 0$

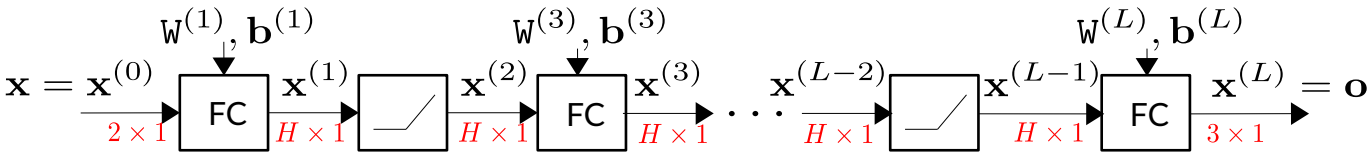
$\mathtt{X}_{\text{train},2} = [2.3, 2.8]$ $\mathtt{Y}_{\text{train},2} = 0$

$\mathtt{X}_{\text{train},3} = [-0.7, 1.2]$ $\mathtt{Y}_{\text{train},3} = 1$

$\mathtt{X}_{\text{train},4} = [3.2, -0.4]$ $\mathtt{Y}_{\text{train},4} = 1$

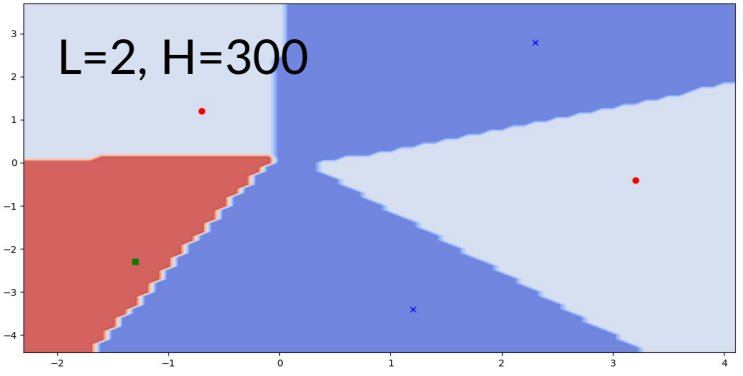
$\mathtt{X}_{\text{train},5} = [-1.3, 2.3]$ $\mathtt{Y}_{\text{train},5} = 2$

Choix de la fonction



$$f(x; \boldsymbol{\theta}) = \text{MLP}\left(x; \boldsymbol{\theta} = \{\mathtt{W}^{(l)}, \mathbf{b}^{(l)}\}_l\right)$$

hyper-paramètres : L, H



Apprentissage

Choix du coût $l(y, \mathbf{o}) = -\ln(\text{softmax}(\mathbf{o})[y])$

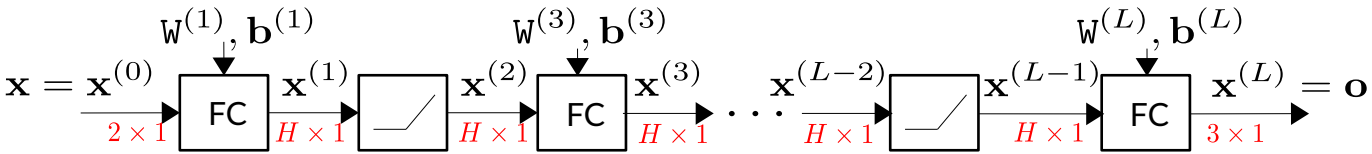
Optimisation $$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{5} -\ln(\text{softmax}(\text{MLP}(\mathtt{X}_{\text{train},i}; \boldsymbol{\theta}))[\mathtt{Y}_{\text{train},i}])$$

IV)

Exemple de classification : 5 données, $\mathtt{X} \in \mathbb{R}^2$ et $\mathtt{Y} \in \{0, 1, 2\}$

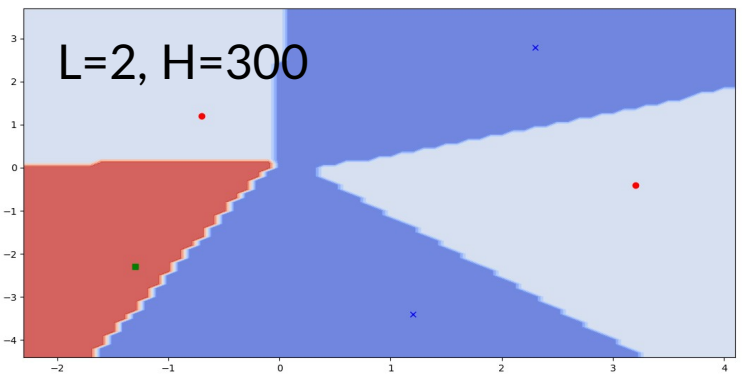
$\mathtt{X}_{\text{train},1} = [1.2, -3.4]$ $\mathtt{Y}_{\text{train},1} = 0$
$\mathtt{X}_{\text{train},2} = [2.3, 2.8]$ $\mathtt{Y}_{\text{train},2} = 0$
$\mathtt{X}_{\text{train},3} = [-0.7, 1.2]$ $\mathtt{Y}_{\text{train},3} = 1$
$\mathtt{X}_{\text{train},4} = [3.2, -0.4]$ $\mathtt{Y}_{\text{train},4} = 1$
$\mathtt{X}_{\text{train},5} = [-1.3, 2.3]$ $\mathtt{Y}_{\text{train},5} = 2$

Choix de la fonction



$$f(x; \boldsymbol{\theta}) = \text{MLP}\left(x; \boldsymbol{\theta} = \{\mathtt{W}^{(l)}, \mathbf{b}^{(l)}\}_l\right)$$

hyper-paramètres : L, H



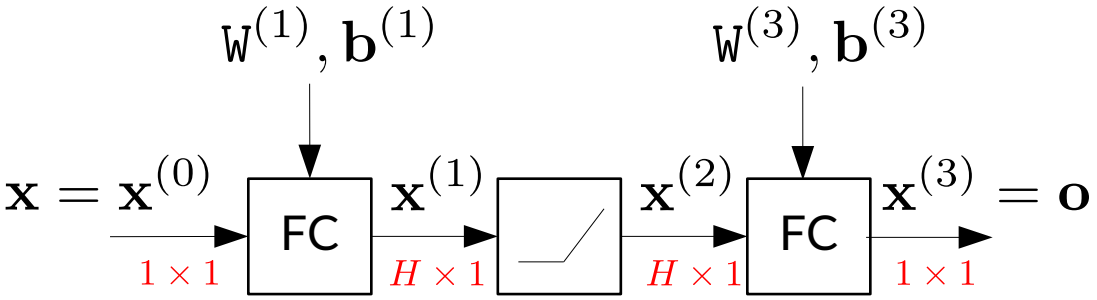
Apprentissage

Choix du coût $l(y, \mathbf{o}) = -\ln(\text{softmax}(\mathbf{o})[y])$

Optimisation $$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{5} -\ln(\text{softmax}(\text{MLP}(\mathtt{X}_{\text{train},i}; \boldsymbol{\theta}))[\mathtt{Y}_{\text{train},i}])$$

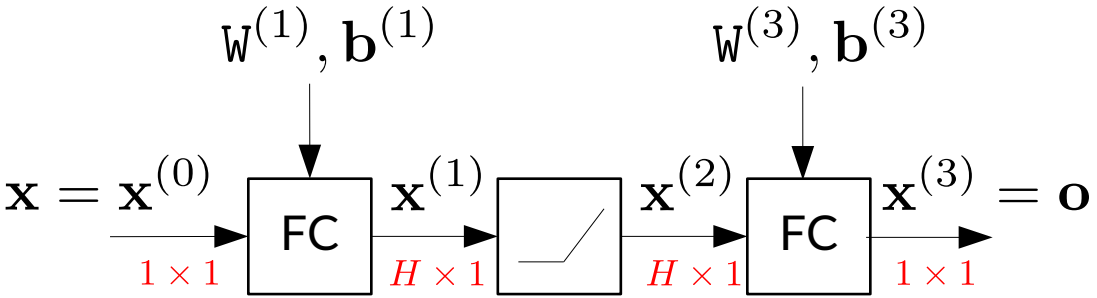
Inférence $\mathbf{o}_{\text{test}} = \text{MLP}(x_{\text{test}}; \boldsymbol{\theta}^*)$ Classe prédite : $\arg\max \mathbf{o}_{\text{test}}$

Étude du MLP à une couche cachée en 1D



IV)

Étude du MLP à une couche cachée en 1D



$\mathbf{W}^{(1)} \doteq \mathbf{w}_1$ Vecteur colonne Hx1

$\mathbf{b}^{(1)} \doteq \mathbf{b}_1$ Vecteur colonne Hx1

$\mathbf{W}^{(3)} \doteq \mathbf{w}_3^\top$ Vecteur ligne 1xH

$\mathbf{b}^{(3)} \doteq b_3$ Scalaire 1x1

$$f(x) = \mathbf{w}_3^\top (\mathrm{ReLU}(\mathbf{w}_1 x + \mathbf{b}_1)) + b_3$$

Étude du MLP à une couche cachée en 1D (suite)

$$f(x) = \mathbf{w}_3^\top \mathrm{ReLU}(\mathbf{w}_1 x + \mathbf{b}_1) + b_3$$

$$= \sum_{j=1}^{H} \mathbf{w}_{3,j} \mathrm{ReLU}(\mathbf{w}_{1,j} x + \mathbf{b}_{1,j}) + b_3$$

Somme pondérée de fonctions ReLU !

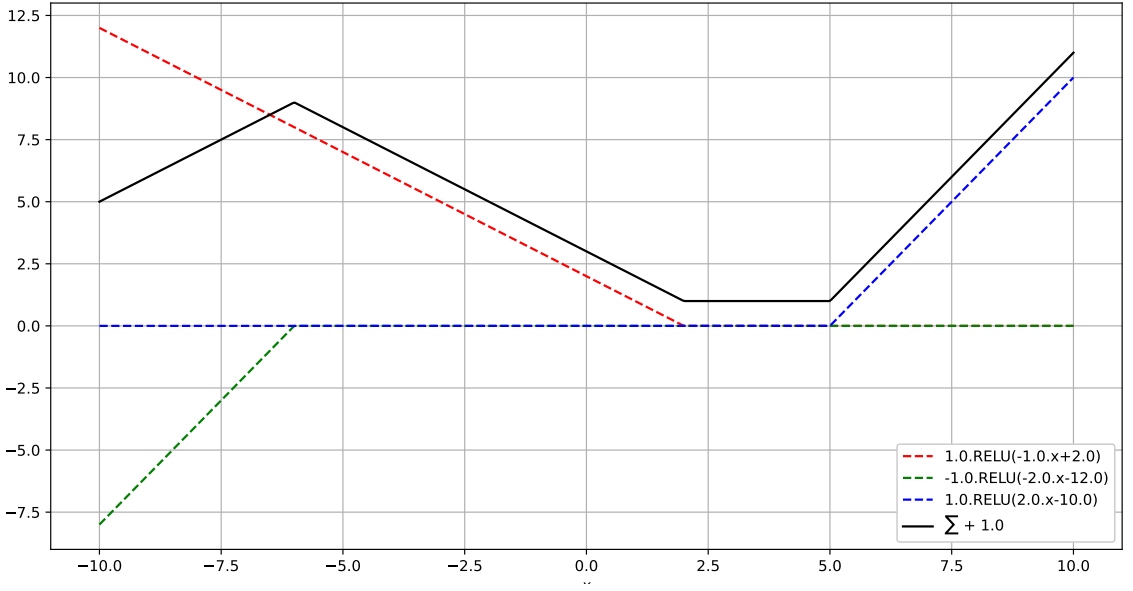
IV)

Étude du MLP à une couche cachée en 1D (suite)

$$f(x) = \mathbf{w}_3^\top \mathrm{ReLU}(\mathbf{w}_1 x + \mathbf{b}_1) + b_3$$

$$= \sum_{j=1}^{H} \mathbf{w}_{3,j} \mathrm{ReLU}(\mathbf{w}_{1,j} x + \mathbf{b}_{1,j}) + b_3$$

Exemple numérique avec H=3



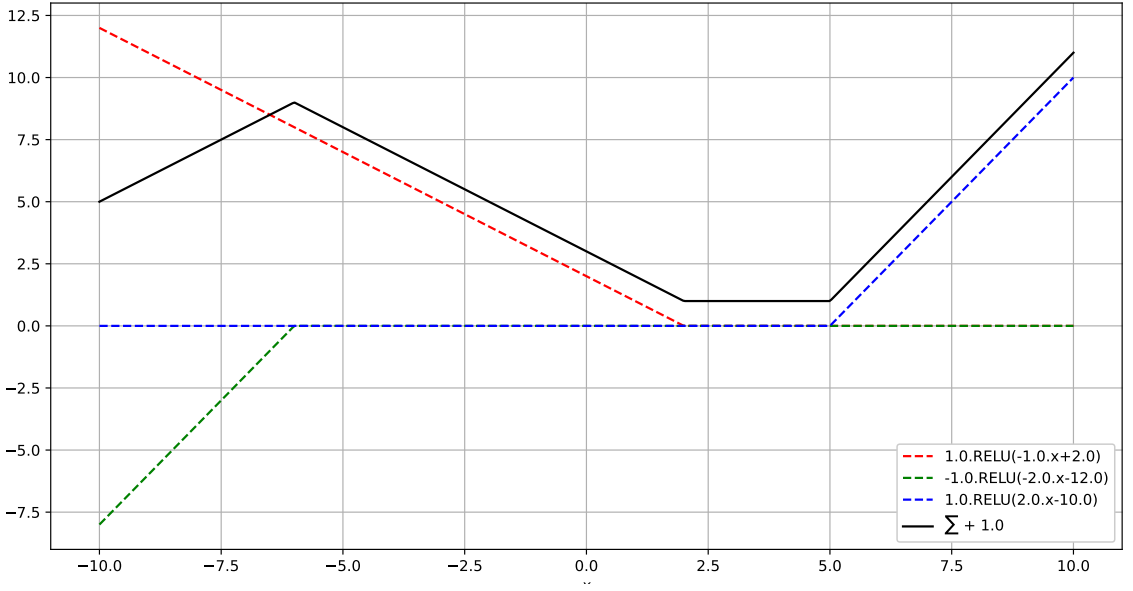
IV)

Étude du MLP à une couche cachée en 1D (suite)

$$f(x) = \mathbf{w}_3^\top \mathrm{ReLU}(\mathbf{w}_1 x + \mathbf{b}_1) + b_3$$

$$= \sum_{j=1}^{H} \mathbf{w}_{3,j} \mathrm{ReLU}(\mathbf{w}_{1,j} x + \mathbf{b}_{1,j}) + b_3$$

Exemple numérique avec H=3



Fonction continue et affine par morceaux

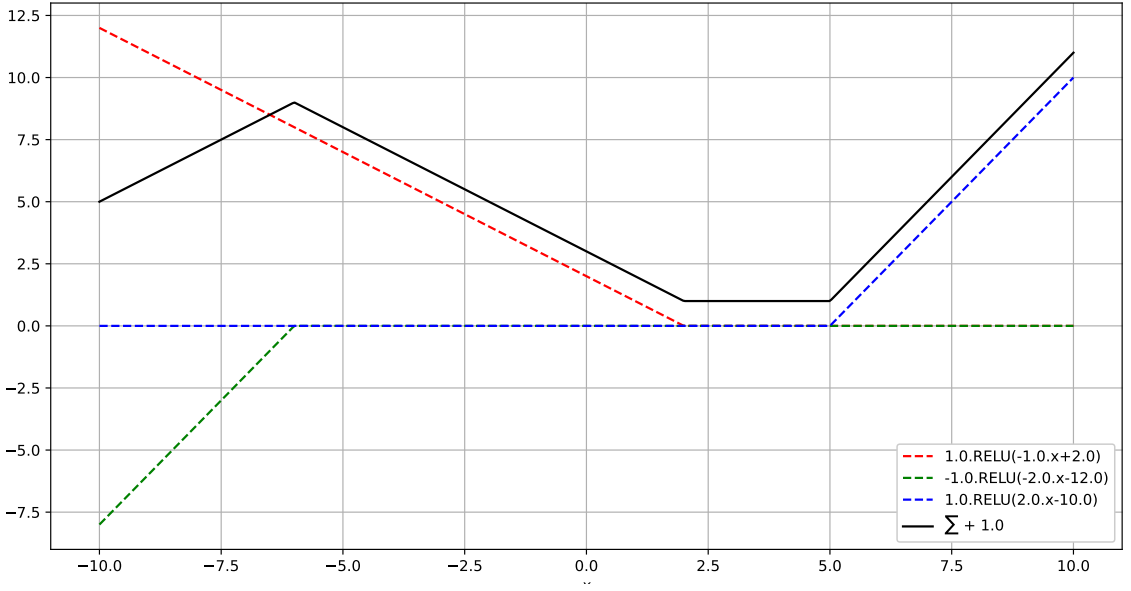
IV)

Étude du MLP à une couche cachée en 1D (suite)

$$f(x) = \mathbf{w}_3^\top \mathrm{ReLU}(\mathbf{w}_1 x + \mathbf{b}_1) + b_3$$

$$= \sum_{j=1}^{H} \mathbf{w}_{3,j} \mathrm{ReLU}(\mathbf{w}_{1,j} x + \mathbf{b}_{1,j}) + b_3$$

Exemple numérique avec H=3



Augmenter H accroît la **capacité** du réseau.

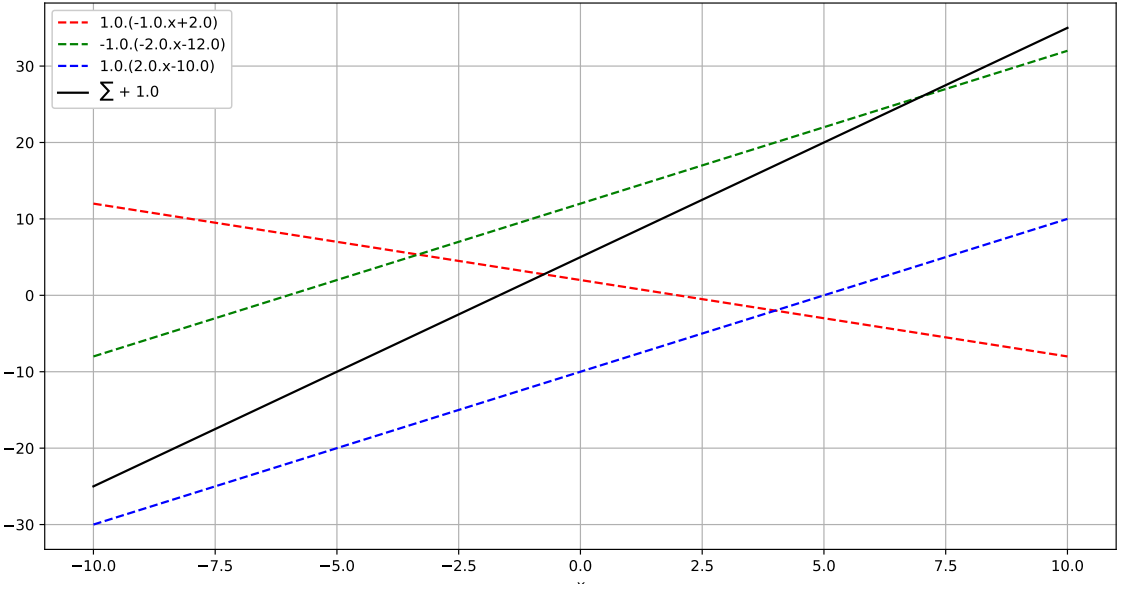
IV)

Étude du MLP à une couche cachée en 1D (suite)

$$f(x) = \mathbf{w}_3^\top \mathrm{ReLU}(\mathbf{w}_1 x + \mathbf{b}_1) + b_3$$

$$= \sum_{j=1}^{H} \mathbf{w}_{3,j} \mathrm{ReLU}(\mathbf{w}_{1,j} x + \mathbf{b}_{1,j}) + b_3$$

Et sans ReLU ?



Une somme de fonctions linéaires est une fonction linéaire...

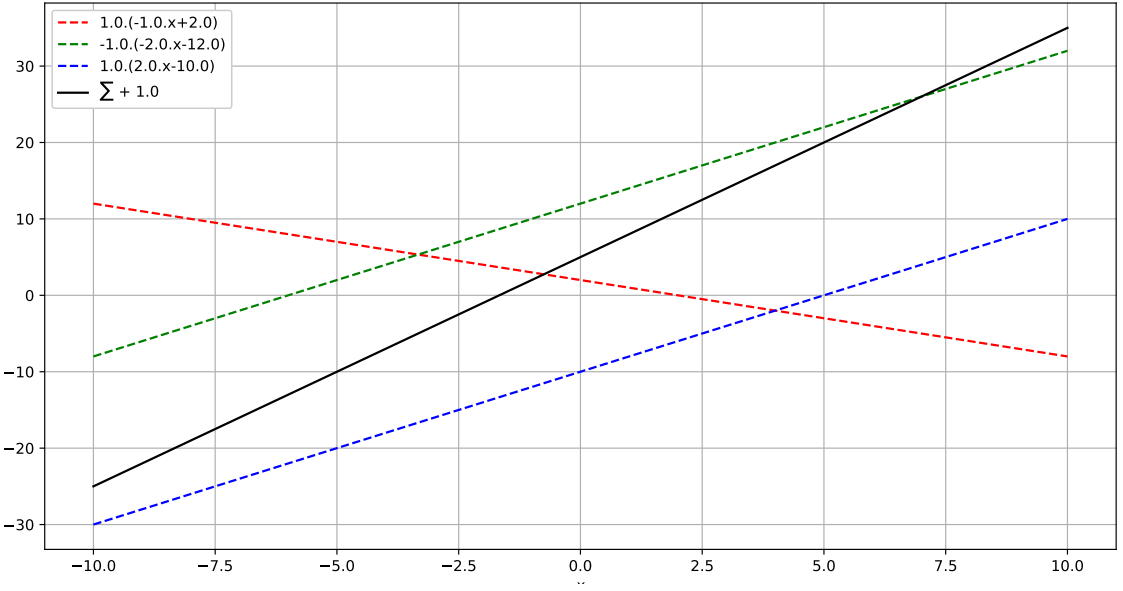
IV)

Étude du MLP à une couche cachée en 1D (suite)

$$f(x) = \mathbf{w}_3^\top \mathrm{ReLU}(\mathbf{w}_1 x + \mathbf{b}_1) + b_3$$

$$= \sum_{j=1}^{H} \mathbf{w}_{3,j} \mathrm{ReLU}(\mathbf{w}_{1,j} x + \mathbf{b}_{1,j}) + b_3$$

Et sans ReLU ?

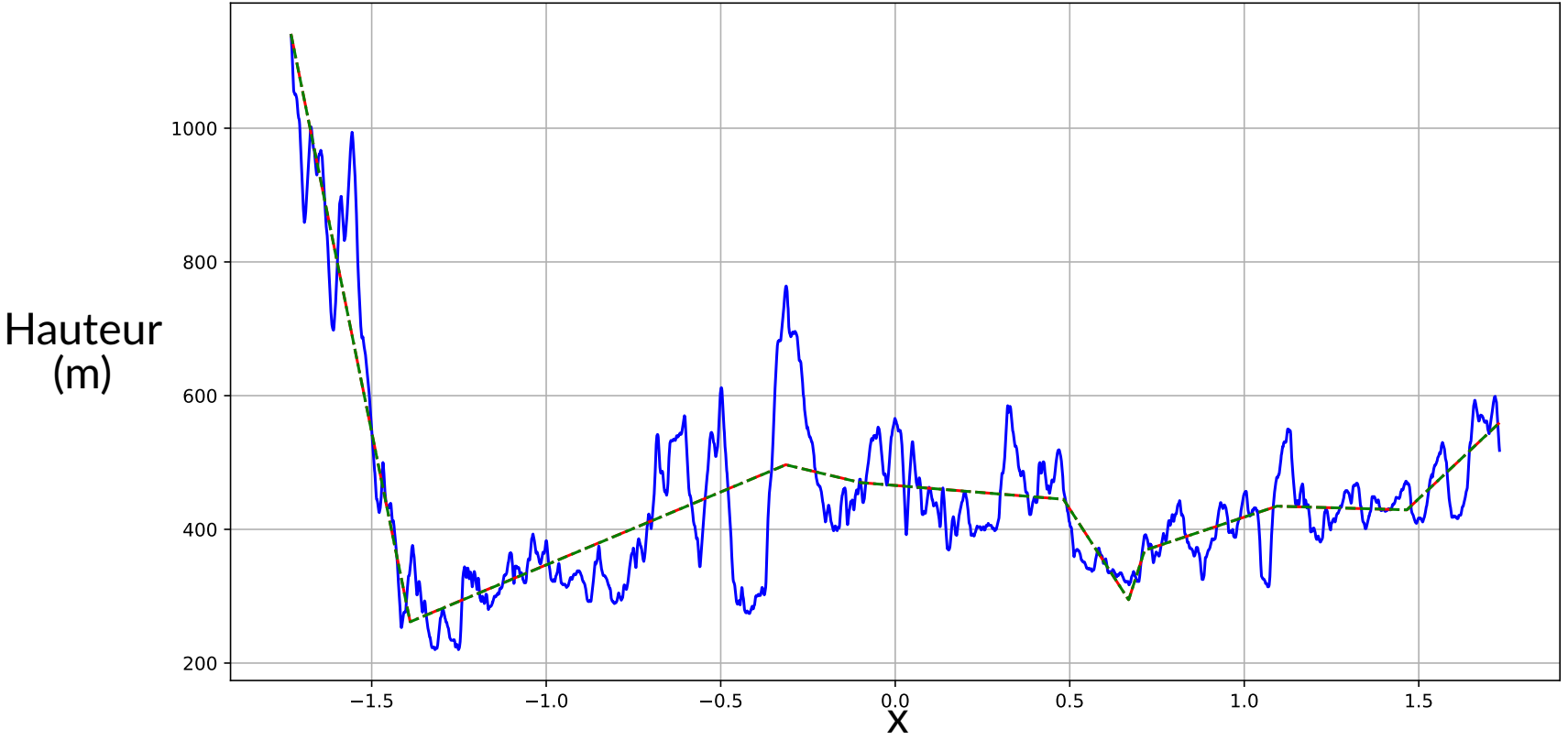


Augmenter H n'accroît pas la **capacité** du réseau.

IV)

Étude du MLP à une couche cachée en 1D (suite bis)

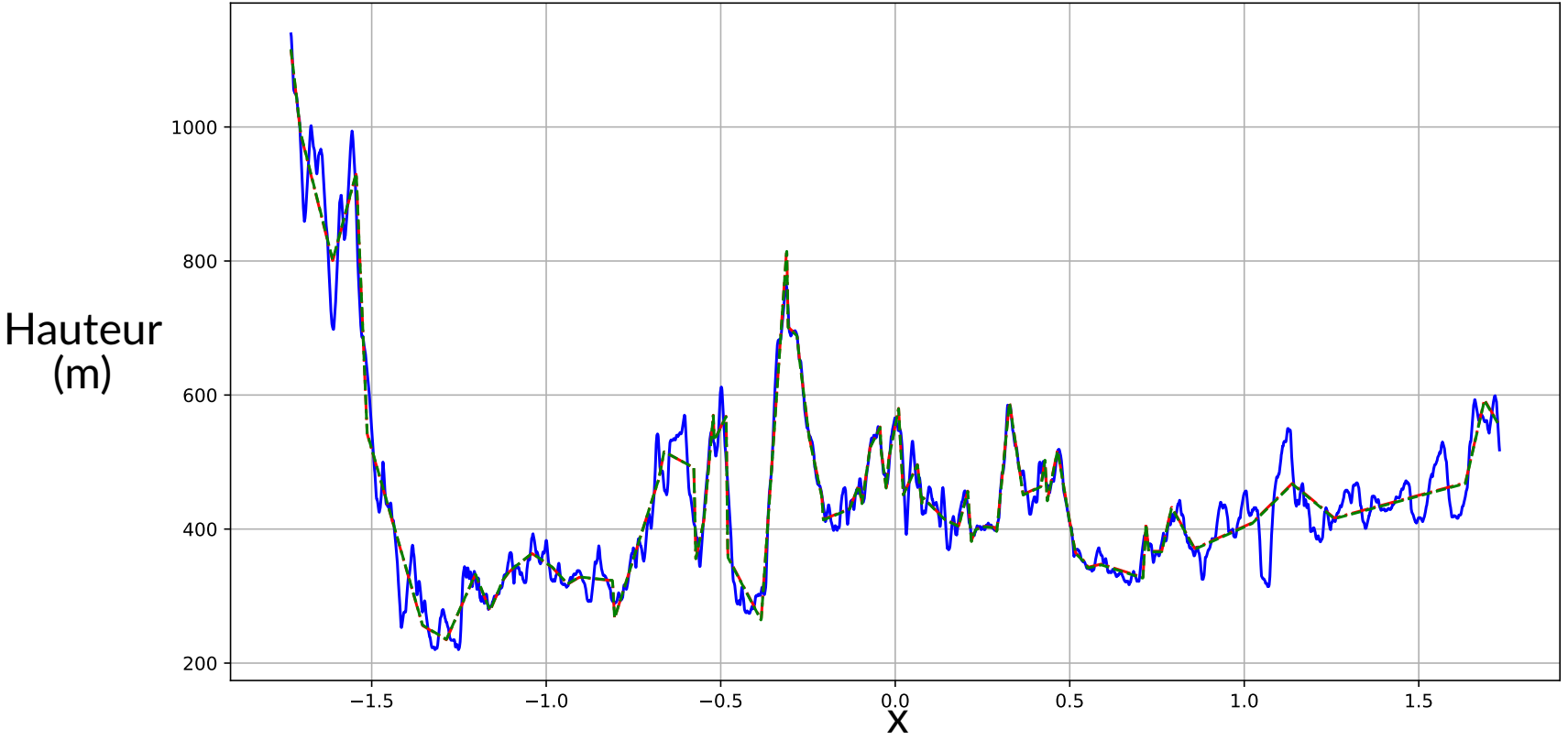
Optimisation d'un MLP sur un profil de terrain : H=10



IV)

Étude du MLP à une couche cachée en 1D (suite bis)

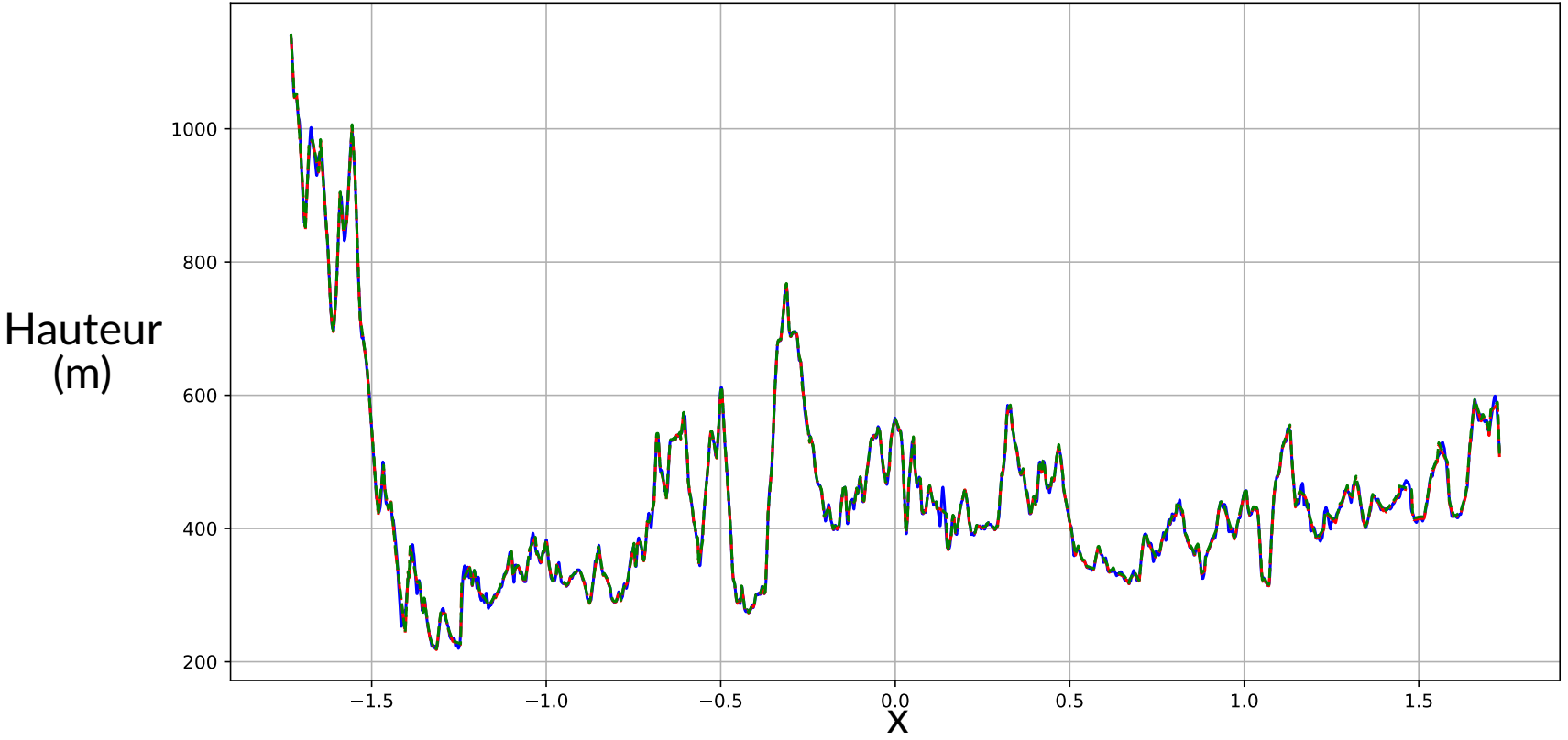
Optimisation d'un MLP sur un profil de terrain : H=100



IV)

Étude du MLP à une couche cachée en 1D (suite bis)

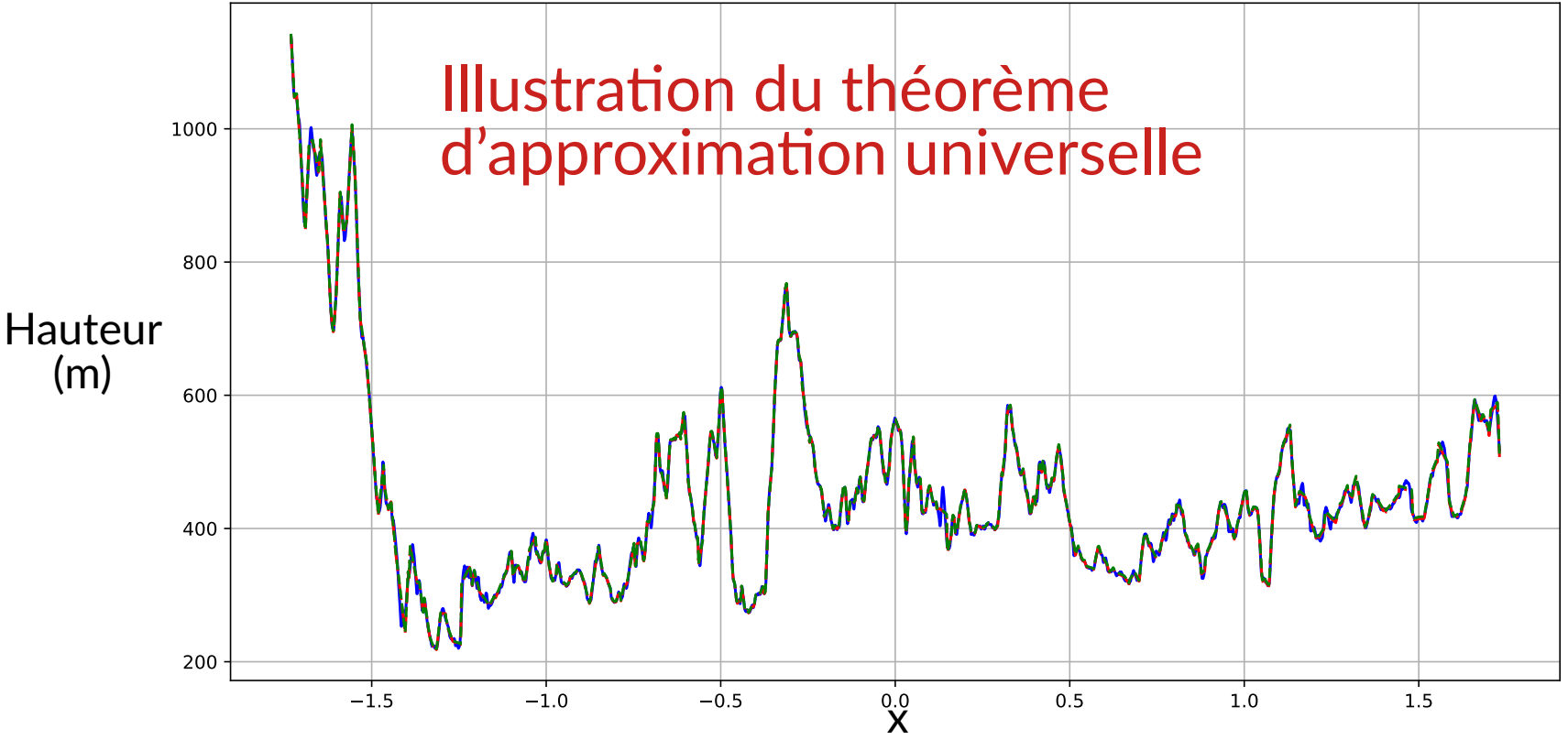
Optimisation d'un MLP sur un profil de terrain : H=1000



IV)

Étude du MLP à une couche cachée en 1D (suite bis)

Optimisation d'un MLP sur un profil de terrain : H=1000



V) Risques

V)

IA et vie privée

Are you ready? Here is all the data Facebook and Google have on you

Dylan Curran

- Google knows where you've been
- Google knows everything you've ever searched – and deleted
- Google has an advertisement profile of you
- Google knows all the apps you use
- Google has all of your YouTube history
- The data Google has on you can fill millions of Word documents

https://www.theguardian.com/commentisfree/2018/mar/28/all-the-data-facebook-google-has-on-you-privacy

V)

IA et vie privée (suite)

The Chinese state wants to control its citizens via a system of social scoring that punishes behavior it doesn't approve of. Image Credit: Telecoms

---

Le projet de smart city d'Alphabet à Toronto suscite l'inquiétude des experts

VU AILLEURS Le projet de smart city lancé à Toronto en 2017 par Sidewalk Labs, la branche d'innovation urbaine d'Alphabet, est vivement critiqué par la population locale et certains experts consultants du projet. La protection de la vie privée ne serait pas respectée.

FLORIANE LECLERC | PUBLIÉ LE 25 OCTOBRE 2018 À 14H10

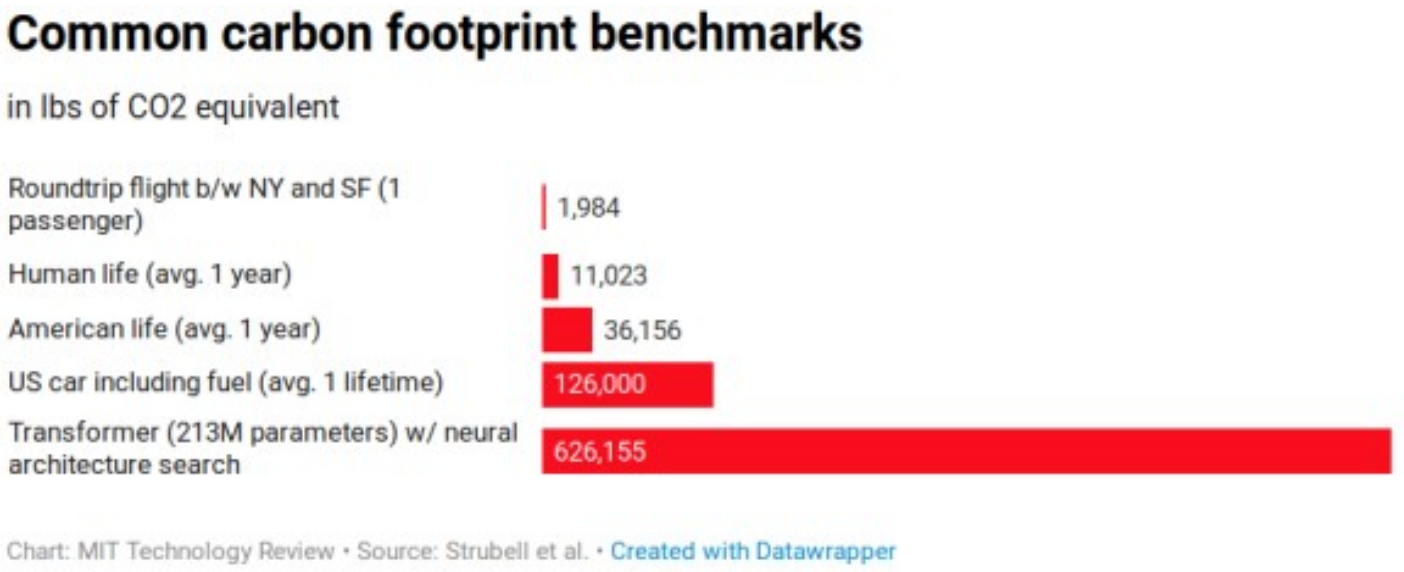
V)

La machine au service de l'homme ?

Mme Tang Yu, PDG du chinois NetDragon Websoft et de ses 6000 employés, est le premier robot à être nommé à la tête d'une société. Disponible H24, elle ne touche aucun salaire. *NetDragon Websoft*

V)

IA et consommation énergétique



<u>Conseils de lecture :</u>

L'Enfer Numérique, Guillaume Pitron, 2021

Training a single AI model can emit as much carbon as five cars in their lifetimes, MIT Press

Energy and Policy Considerations for Deep Learning in NLP, Strubell et al., 2019

V)

IA et transhumanisme

NEURALINK

Calico

<u>Conseil de lecture :</u> Leurre et malheur du transhumanisme, Olivier Rey