Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage

Deep Learning

Introduction

Thomas Ranvier

Université Claude Bernard, Lyon 1

Thomas Ranvier Deep Learning - Introduction 7 novembre 2022

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Sommaire 

1 Contexte

2 Perceptron

3 Réseaux de neurones

4 Apprentissage en théorie

5 Apprentissage en pratique

6 Types d’apprentissage

Thomas Ranvier Deep Learning - Introduction 1

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage

Contexte

Introduction au Deep Learning

Thomas Ranvier Deep Learning - Introduction 2

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Qu’est ce que le Deep Learning ? **Artificial**

**Intelligence Machine**

Méthodes capable d’imiter un

**Learning Deep**

comportement humain Méthodes capable d’apprendre automatiquement à partir de

**Learning**

données Utilisation de réseaux de neurones

profonds

Thomas Ranvier Deep Learning - Introduction 3

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage L’intérêt du Deep Learning Problématique

Coder manuellement un algorithme à reconnaître des features est très laborieux, long et peu efficace. Thomas Ranvier Deep Learning - Introduction 4

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage L’intérêt du Deep Learning Problématique

Coder manuellement un algorithme à reconnaître des features est très laborieux, long et peu efficace.

Solution

Le Deep Learning rend possible l’apprentissage automatique de ces features.

Thomas Ranvier Deep Learning - Introduction 4

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Domaines d’application du Deep Learning 

Domaines d’application

*•* Reconnaissance visuelle

Thomas Ranvier Deep Learning - Introduction 5

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Domaines d’application du Deep Learning 

Domaines d’application

*•* Reconnaissance visuelle

*•* Reconnaissance faciale

Thomas Ranvier Deep Learning - Introduction 5

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Domaines d’application du Deep Learning 

Domaines d’application

*•* Reconnaissance visuelle

*•* Reconnaissance faciale

*•* Reconnaissance de formes

Thomas Ranvier Deep Learning - Introduction 5

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Domaines d’application du Deep Learning 

Domaines d’application

*•* Reconnaissance visuelle

*•* Reconnaissance faciale

*•* Reconnaissance de formes

*•* Reconnaissance de texte

Thomas Ranvier Deep Learning - Introduction 5

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Domaines d’application du Deep Learning 

Domaines d’application

*•* Reconnaissance visuelle

*•* Reconnaissance faciale

*•* Reconnaissance de formes

*•* Reconnaissance de texte

*•* Etc.

Thomas Ranvier Deep Learning - Introduction 5

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Domaines d’application du Deep Learning 

Domaines d’application

*•* Reconnaissance visuelle

*•* Reconnaissance faciale

*•* Reconnaissance de formes

*•* Reconnaissance de texte

*•* Etc.

*•* Robotique

Thomas Ranvier Deep Learning - Introduction 5

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Domaines d’application du Deep Learning 

Domaines d’application

*•* Reconnaissance visuelle

*•* Reconnaissance faciale

*•* Reconnaissance de formes

*•* Reconnaissance de texte

*•* Etc.

*•* Robotique

*•* Sécurité

Thomas Ranvier Deep Learning - Introduction 5

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Domaines d’application du Deep Learning 

Domaines d’application

*•* Reconnaissance visuelle

*•* Reconnaissance faciale

*•* Reconnaissance de formes

*•* Reconnaissance de texte

*•* Etc.

*•* Robotique

*•* Sécurité

*•* Santé

Thomas Ranvier Deep Learning - Introduction 5

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Domaines d’application du Deep Learning 

Domaines d’application

*•* Reconnaissance visuelle

*•* Reconnaissance faciale

*•* Reconnaissance de formes

*•* Reconnaissance de texte

*•* Etc.

*•* Robotique

*•* Sécurité

*•* Santé

*•* Traduction

Thomas Ranvier Deep Learning - Introduction 5

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Domaines d’application du Deep Learning 

Domaines d’application

*•* Reconnaissance visuelle

*•* Reconnaissance faciale

*•* Reconnaissance de formes

*•* Reconnaissance de texte

*•* Etc.

*•* Robotique

*•* Sécurité

*•* Santé

*•* Traduction

*•* Etc.

Thomas Ranvier Deep Learning - Introduction 5

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Exemples d’applications du Deep Learning Thomas Ranvier Deep Learning - Introduction 6

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Exemples d’applications du Deep Learning Thomas Ranvier Deep Learning - Introduction 6

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Exemples d’applications du Deep Learning Thomas Ranvier Deep Learning - Introduction 6

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Exemples d’applications du Deep Learning 

Thomas Ranvier Deep Learning - Introduction 6

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Pourquoi le Deep Learning se développe-t-il maintenant ? 

**A.I. TIMELINE**

**1950**

**TURING TEST**

Computer scientist Alan Turing proposes a test for machine

intelligence. If a

machine can trick

humans into thinking it is human, then it has intelligence

**1999**

**AIBO**

Sony launches first consumer robot pet dog AiBO (AI robot) with skills and personality that develop over time

**1955**

**A.I. BORN**

Term 'artificial

intelligence' is coined by computer scientist, John McCarthy to

describe “the science and engineering of making intelligent

machines”

**2002**

**ROOMBA**

First mass produced autonomous robotic vacuum cleaner from

iRobot learns to navigate and clean homes

**1961**

**UNIMATE**

First industrial robot, Unimate, goes to work at GM replacing

humans on the

assembly line

**2011**

**SIRI**

Apple integrates Siri, an intelligent virtual assistant with a voice interface, into the iPhone 4S

**1964**

**ELIZA**

Pioneering chatbot developed by Joseph Weizenbaum at MIT holds conversations with humans

**2011**

**WATSON**

IBM’s question

answering computer Watson wins first place on popular $1M prize television quiz show *Jeopardy*

**1966**

**SHAKEY**

The ‘first electronic person’ from Stanford, Shakey is a general purpose mobile robot that reasons about its own actions

**2014**

**EUGENE**

Eugene Goostman, a chatbot passes the Turing Test with a third of judges believing Eugene is human

**A.I.**

**WINTER**

Many false starts and dead-ends leave A.I. out in the cold

**2014**

**ALEXA**

Amazon launches Alexa, an intelligent virtual assistant with a voice interface that completes shopping tasks

**1997**

**DEEP BLUE**

Deep Blue, a chess playing computer from IBM defeats world chess champion Garry

Kasparov

**2016**

**TAY**

Microsoft’s chatbot Tay goes rogue on social media making

inflammatory and

offensive racist

comments

**1998**

**KISMET**

Cynthia Breazeal at MIT introduces KISmet, an emotionally intelligent robot insofar as it

detects and responds to people’s feelings

**2017**

**ALPHAGO**

Google’s A.I. AlphaGo beats world champion Ke Jie in the complex

board game of Go, notable for its vast number (2170) of

possible positions

Thomas Ranvier Deep Learning - Introduction 7

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage

Le Perceptron

L’élément de base pour construire des réseaux de neurones

Thomas Ranvier Deep Learning - Introduction 8

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Le perceptron - forward propagation 

**x1**

**b**

Formalisation mathématique

**x2**

**… xn**

**w1**

**w2**

**… wn**

**∑ ŷ**

*y*ˆ = *g*(*b* +X*m i*=1

*xiwi*)

Thomas Ranvier Deep Learning - Introduction 9

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Le perceptron - forward propagation 

**x1**

**b**

Formalisation mathématique

**x2**

**w1 w2**

**∑ ŷ**

*y*ˆ = *g*(*b* +X*m i*=1

*xiwi*)

**… xn**

**… wn**

*⇔ y*ˆ = *g*(*b* + ***X****T****W***)





Où ***X****T* = [*x*1 *. . . xm*] et ***W*** =

*w*1 ...

*wm*

 

Thomas Ranvier Deep Learning - Introduction 9

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Le perceptron - fonction d’activation 

Les fonctions d’activation

*•* Dans *y*ˆ = *g*(*b* + ***X****T****W***), *g* est une fonction d’activation non linéaire (activation function / nonlinearity)

Thomas Ranvier Deep Learning - Introduction 10

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Le perceptron - fonction d’activation 

Les fonctions d’activation

*•* Dans *y*ˆ = *g*(*b* + ***X****T****W***), *g* est une fonction d’activation non linéaire (activation function / nonlinearity)

*•* L’intérêt des fonctions d’activation est d’introduire une non-linéarité dans le modèle Thomas Ranvier Deep Learning - Introduction 10

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Le perceptron - fonction d’activation 

Les fonctions d’activation

*•* Dans *y*ˆ = *g*(*b* + ***X****T****W***), *g* est une fonction d’activation non linéaire (activation function / nonlinearity)

*•* L’intérêt des fonctions d’activation est d’introduire une non-linéarité dans le modèle *•* Exemples de fonctions d’activations courantes :

Thomas Ranvier Deep Learning - Introduction 10

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Le perceptron - fonction d’activation 

Les fonctions d’activation

*•* Dans *y*ˆ = *g*(*b* + ***X****T****W***), *g* est une fonction d’activation non linéaire (activation function / nonlinearity)

*•* L’intérêt des fonctions d’activation est d’introduire une non-linéarité dans le modèle *•* Exemples de fonctions d’activations courantes :

*•* Sigmoid

Thomas Ranvier Deep Learning - Introduction 10

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Le perceptron - fonction d’activation 

Les fonctions d’activation

*•* Dans *y*ˆ = *g*(*b* + ***X****T****W***), *g* est une fonction d’activation non linéaire (activation function / nonlinearity)

*•* L’intérêt des fonctions d’activation est d’introduire une non-linéarité dans le modèle *•* Exemples de fonctions d’activations courantes :

*•* Sigmoid

*•* Tanh

Thomas Ranvier Deep Learning - Introduction 10

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Le perceptron - fonction d’activation 

Les fonctions d’activation

*•* Dans *y*ˆ = *g*(*b* + ***X****T****W***), *g* est une fonction d’activation non linéaire (activation function / nonlinearity)

*•* L’intérêt des fonctions d’activation est d’introduire une non-linéarité dans le modèle *•* Exemples de fonctions d’activations courantes :

*•* Sigmoid

*•* Tanh

*•* ReLU

Thomas Ranvier Deep Learning - Introduction 10

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Visualisation de l’intérêt des fonctions d’activation 

Fonctionnement d’un réseau de neurones

*•* Sans fonction d’activation un réseau de neurones ne peut apprendre qu’une séparation linéaire des données

Thomas Ranvier Deep Learning - Introduction 11

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Visualisation de l’intérêt des fonctions d’activation 

Fonctionnement d’un réseau de neurones

*•* Sans fonction d’activation un réseau de neurones ne peut apprendre qu’une séparation linéaire des données

*•* Pour apprendre à séparer des données non linéairement séparable on doit donc utiliser des fonctions d’activation non linéaires

Thomas Ranvier Deep Learning - Introduction 11

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Visualisation de l’intérêt des fonctions d’activation 

Fonctionnement d’un réseau de neurones

*•* Sans fonction d’activation un réseau de neurones ne peut apprendre qu’une séparation linéaire des données

*•* Pour apprendre à séparer des données non linéairement séparable on doit donc utiliser des fonctions d’activation non linéaires

Outil de visualisation

*•* On va utiliser un outil permettant de visualiser le fonctionnement d’un réseau de neurones couche par couche : https://cs.stanford.edu/people/karpathy/convnetjs/demo/classify2d.html

Thomas Ranvier Deep Learning - Introduction 11

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Visualisation de l’intérêt des fonctions d’activation 

Fonctionnement d’un réseau de neurones

*•* Sans fonction d’activation un réseau de neurones ne peut apprendre qu’une séparation linéaire des données

*•* Pour apprendre à séparer des données non linéairement séparable on doit donc utiliser des fonctions d’activation non linéaires

Outil de visualisation

*•* On va utiliser un outil permettant de visualiser le fonctionnement d’un réseau de neurones couche par couche : https://cs.stanford.edu/people/karpathy/convnetjs/demo/classify2d.html *•* Cet outil permet de visualiser la frontière apprise par le modèle, ainsi que les opérations effectuées à chaque étape du modèle

Thomas Ranvier Deep Learning - Introduction 11

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Visualisation de l’intérêt des fonctions d’activation 

Visualisation

Sans fonction d’activation, première et seconde couches : données non linéairement séparables (a) Données d’entrée (b) Première couche (c) Seconde couche

Thomas Ranvier Deep Learning - Introduction 11

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Visualisation de l’intérêt des fonctions d’activation 

Visualisation

Avec fonction d’activation, première couche

(a) Données d’entrée (b) Première couche (c) Première fonction d’activation

Thomas Ranvier Deep Learning - Introduction 11

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Visualisation de l’intérêt des fonctions d’activation 

Visualisation

Avec fonction d’activation, seconde couche : données linéairement séparables

(a) Données d’entrée (b) Seconde couche (c) Seconde fonction d’activation

Thomas Ranvier Deep Learning - Introduction 11

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Exemples de fonctions d’activation 

Sigmoid

*g*(*z*) = 1

1 + *e−z*

*g′*(*z*) = *g*(*z*) *·* (1 *− g*(*z*))

Thomas Ranvier Deep Learning - Introduction 12

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Exemples de fonctions d’activation 

Sigmoid

*g*(*z*) = 1

1 + *e−z*

*g′*(*z*) = *g*(*z*) *·* (1 *− g*(*z*))

Hyperbolic Tangent (tanh)

*g*(*z*) = *ez − e−z*

*ez* + *e−z*

*g′*(*z*) = 1 *− g*(*z*)2

Thomas Ranvier Deep Learning - Introduction 12

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Exemples de fonctions d’activation 

Sigmoid

*g*(*z*) = 1 1 + *e−z*

Hyperbolic Tangent (tanh)

*g*(*z*) = *ez − e−z*

*ez* + *e−z*

Rectified Linear Unit (ReLU)

*g*(*z*) = *max*(0*, z*)

(

*g′*(*z*) = *g*(*z*) *·* (1 *− g*(*z*))

*g′*(*z*) = 1 *− g*(*z*)2

*g′*(*z*) =

1 if *z >* 0 0 otherwise

Thomas Ranvier Deep Learning - Introduction 12

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Le perceptron - exercice Paramètres

*W* =

4 *−*2

, *b* = 1, donc *y*ˆ = *g*(*XTW* + *b*) = *g*( **1**

*x*1 *x*2

*T* 4 *−*2

+ 1), avec *g*(*z*) =1 1 + *e−z*

**x2**

**x1 x2**

**4 -2**

**∑ ŷ**

**x1**

Thomas Ranvier Deep Learning - Introduction 13

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Le perceptron - exercice 

Points

Calculer *y*ˆ pour les points *a* =

**1**

**x1**

*−*1 2

et *b* =

0 *−*1

**a**

**x2**

**x1**

**4**

**-2**

**x2**

**∑ ŷ**

**b**

Thomas Ranvier Deep Learning - Introduction 13

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Le perceptron - exercice 

Points

Calculer *y*ˆ pour les points *a* =

**1**

**x1**

*−*1 2

et *b* =

0 *−*1

, réponse : *y*ˆ*a ≈* 0*.*00091 et *y*ˆ*b ≈* 0*.*95257

**x2**

**a**

**x1**

**4**

**-2**

**x2**

**∑ ŷ**

**b**

Thomas Ranvier Deep Learning - Introduction 13

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Le perceptron - exercice 

Visualisation de la frontière

On peut tracer la frontière de décision du perceptron se situant sur *y*ˆ = 0*.*5

**z < 0 ŷ < 0.5**

**x2**

**a**

**.5**

**0**

**) =**

**x2**

**x1 x2**

**4 -2**

**1**

**∑ ŷ**

**b**

**- 2**

**x1**

**4**

**(1 +**

**g**

**x1**

**ŷ > 0.5 z > 0**

Thomas Ranvier Deep Learning - Introduction 13

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage

Réseaux de neurones

Organisation de perceptrons en une architecture permettant l’apprentissage

Thomas Ranvier Deep Learning - Introduction 14

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Composition d’un réseau de neurones 

Réseau de neurones

*•* Il est composé de perceptrons, organisés par couches : Perceptron Multicouche (MLP)

**Input Hidden layers**

**Output**

Thomas Ranvier Deep Learning - Introduction 15

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Composition d’un réseau de neurones 

Réseau de neurones

*•* Il est composé de perceptrons, organisés par couches : Perceptron Multicouche (MLP) *•* La couche d’entrée correspond aux données en entrée, la couche de sortie correspond aux résultats donnés par le modèle

**Input Hidden layers**

**Output**

Thomas Ranvier Deep Learning - Introduction 15

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Composition d’un réseau de neurones 

Réseau de neurones

*•* Il est composé de perceptrons, organisés par couches : Perceptron Multicouche (MLP) *•* La couche d’entrée correspond aux données en entrée, la couche de sortie correspond aux résultats donnés par le modèle

*•* Les couches entre les deux sont des couches dites cachées, c’est là que se passe l’essentiel de l’apprentissage

**Input Hidden layers**

**Output**

Thomas Ranvier Deep Learning - Introduction 15

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Fonctionnement d’un réseau de neurones 

Réseau de neurones

*•* Il prend des données en entrées

**Input Hidden layers**

**Output**

Thomas Ranvier Deep Learning - Introduction 16

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Fonctionnement d’un réseau de neurones 

Réseau de neurones

*•* Il prend des données en entrées

*•* On l’entraîne pour qu’il découvre et apprenne des patterns au sein de données connues

**Input Hidden layers**

**Output**

Thomas Ranvier Deep Learning - Introduction 16

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Fonctionnement d’un réseau de neurones 

Réseau de neurones

*•* Il prend des données en entrées

*•* On l’entraîne pour qu’il découvre et apprenne des patterns au sein de données connues *•* On l’utilise ensuite pour générer des prédictions sur de nouvelles données inconnues

**Input Hidden layers**

**Output**

Thomas Ranvier Deep Learning - Introduction 16

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Fonctionnement d’un réseau de neurones 

Réseau de neurones

*•* Il prend des données en entrées

**Input Hidden layers**

**Output**

Thomas Ranvier Deep Learning - Introduction 17

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Fonctionnement d’un réseau de neurones 

Réseau de neurones

*•* Il prend des données en entrées

*•* On l’entraîne pour qu’il découvre et apprenne des patterns au sein de données connues

**Input Hidden layers**

**Output**

Thomas Ranvier Deep Learning - Introduction 17

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Fonctionnement d’un réseau de neurones 

Réseau de neurones

*•* Il prend des données en entrées

*•* On l’entraîne pour qu’il découvre et apprenne des patterns au sein de données connues *•* On l’utilise ensuite pour générer des prédictions sur de nouvelles données inconnues

**Input Hidden layers**

**Output**

Thomas Ranvier Deep Learning - Introduction 17

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Modéliser une couche de perceptrons de manière efficiente 

Forward propagation d’une donnée à *m* dimensions dans un perceptron seul

***X*** et ***W*** sont des vecteurs, *b* est un scalaire, donc *y*ˆ est un scalaire également

Thomas Ranvier Deep Learning - Introduction 18

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Modéliser une couche de perceptrons de manière efficiente 

Forward propagation d’une donnée à *m* dimensions dans un perceptron seul

***X*** et ***B*** sont des vecteurs, ***W*** est une matrice, donc ***Y***ˆ est un vecteur

Thomas Ranvier Deep Learning - Introduction 18

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Modéliser une couche de perceptrons de manière efficiente 

Forward propagation d’une donnée à *m* dimensions dans un perceptron seul

***B*** est un vecteur, ***X*** et ***W*** sont des matrices, donc ***Y***ˆ est une matrice

**w1,1 w2,**

**…**

**1**

**w1,2 w2,2 …**

**…**

**w1,n**

**…**

**w2,n**

**…**

**wm,1**

**x1,1 x1,2 … x1,m**

**wm,2**

**… wm,n**

**b1 b2 … bn**

**x2,1x2,2 … x2,m**

**\*+ = Z**

**…**

**…**

**…**

**xd,1xd,2 … xd,m**

**Avec g(Z) = Ŷ**

Thomas Ranvier Deep Learning - Introduction 18

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Programmer une couche de perceptrons sous TensorFlow Coder manuellement une couche “dense”

class CustomDense ( t f . keras . layers . Layer ) :

def \_ \_ i n i t \_ \_ ( s e l f , input\_dim , u n i t s ) :

super ( CustomDense , s e l f ) . \_ \_ i n i t \_ \_ ( )

# Create weights and biases

s e l f . weights = s e l f . add\_weight ( shape =( input\_dim , u n i t s ) )

s e l f . biases = s e l f . add\_weight ( shape =( units , 1 ) )

def c a l l ( s e l f , x ) :

# Forward propagation

z = t f . matmul ( x , s e l f . weights ) + s e l f . biases

# Add non− l i n e a r i t y using an a c t i v a t i o n f u n c t i o n

y = t f . keras . a c t i v a t i o n s . r e l u ( z )

r e t u r n y

Thomas Ranvier Deep Learning - Introduction 19

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Programmer une couche de perceptrons sous TensorFlow Coder manuellement une couche “dense”

class CustomDense ( t f . keras . layers . Layer ) :

def \_ \_ i n i t \_ \_ ( s e l f , input\_dim , u n i t s ) :

super ( CustomDense , s e l f ) . \_ \_ i n i t \_ \_ ( )

# Create weights and biases

s e l f . weights = s e l f . add\_weight ( shape =( input\_dim , u n i t s ) )

s e l f . biases = s e l f . add\_weight ( shape =( units , 1 ) )

def c a l l ( s e l f , x ) :

# Forward propagation

z = t f . matmul ( x , s e l f . weights ) + s e l f . biases

# Add non− l i n e a r i t y using an a c t i v a t i o n f u n c t i o n

y = t f . keras . a c t i v a t i o n s . r e l u ( z )

r e t u r n y

Créer une couche de 10 neurones en utilisant simplement le code déjà existant sous TensorFlow dense\_layer = t f . keras . layers . Dense ( u n i t s =10 , a c t i v a t i o n = ’ r e l u ’ )

Thomas Ranvier Deep Learning - Introduction 19

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Programmer un réseau de neurones multi-couches sous TensorFlow Créer un modèle de dimensions (5, 4, 2)

model = t f . keras . Sequential ( [

t f . keras . layers . Dense ( u n i t s =5 , a c t i v a t i o n = ’ r e l u ’ ) ,

t f . keras . layers . Dense ( u n i t s =4 , a c t i v a t i o n = ’ r e l u ’ ) ,

t f . keras . layers . Dense ( u n i t s =2 , a c t i v a t i o n = ’ softmax ’ ) ,

] )

**Input Hidden 1 Output**

**Hidden 2**

**x1**

**x2**

**…**

**xn**

Thomas Ranvier Deep Learning - Introduction 20

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage

Apprentissage en théorie

Apprendre une tâche à un réseau de neurones en théorie

Thomas Ranvier Deep Learning - Introduction 21

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Exemple de problème de classification 

Thomas Ranvier Deep Learning - Introduction 22

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Exemple de problème de classification 

Thomas Ranvier Deep Learning - Introduction 22

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Exemple de problème de classification 

Forward propagation

*•* On associe la valeur **0** pour la classe **Blue** et **1** pour la classe **Red**.

Thomas Ranvier Deep Learning - Introduction 22

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Exemple de problème de classification 

Forward propagation

*•* On associe la valeur **0** pour la classe **Blue** et **1** pour la classe **Red**.

*•* On passe le point

7 3

en entrée de notre modèle.

Thomas Ranvier Deep Learning - Introduction 22

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Exemple de problème de classification 

Forward propagation

*•* On associe la valeur **0** pour la classe **Blue** et **1** pour la classe **Red**.

*•* On passe le point

7 3

en entrée de notre modèle.

*•* Le modèle nous renvoi une valeur : **0.7**.

Thomas Ranvier Deep Learning - Introduction 22

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Exemple de problème de classification 

Thomas Ranvier Deep Learning - Introduction 22

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Loss functions 

Intérêt d’une loss function

*•* Elle permet de mesurer l’écart entre la prédiction faite par le modèle et la cible qu’on lui a donné. Thomas Ranvier Deep Learning - Introduction 23

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Loss functions 

Intérêt d’une loss function

*•* Elle permet de mesurer l’écart entre la prédiction faite par le modèle et la cible qu’on lui a donné. *•* Le modèle est entraîné à minimiser la valeur de cette fonction, c’est ce qui lui permet d’apprendre la tâche souhaitée.

Thomas Ranvier Deep Learning - Introduction 23

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Loss functions 

Intérêt d’une loss function

*•* Elle permet de mesurer l’écart entre la prédiction faite par le modèle et la cible qu’on lui a donné. *•* Le modèle est entraîné à minimiser la valeur de cette fonction, c’est ce qui lui permet d’apprendre la tâche souhaitée.

Exemples de fonctions adaptées selon le problème

*•* Pour permettre un bon apprentissage il faut choisir une loss function adaptée à la tâche attendue. Thomas Ranvier Deep Learning - Introduction 23

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Loss functions 

Intérêt d’une loss function

*•* Elle permet de mesurer l’écart entre la prédiction faite par le modèle et la cible qu’on lui a donné. *•* Le modèle est entraîné à minimiser la valeur de cette fonction, c’est ce qui lui permet d’apprendre la tâche souhaitée.

Exemples de fonctions adaptées selon le problème

*•* Pour permettre un bon apprentissage il faut choisir une loss function adaptée à la tâche attendue. *•* Problèmes de classification : Cross-entropy

Thomas Ranvier Deep Learning - Introduction 23

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Loss functions 

Intérêt d’une loss function

*•* Elle permet de mesurer l’écart entre la prédiction faite par le modèle et la cible qu’on lui a donné. *•* Le modèle est entraîné à minimiser la valeur de cette fonction, c’est ce qui lui permet d’apprendre la tâche souhaitée.

Exemples de fonctions adaptées selon le problème

*•* Pour permettre un bon apprentissage il faut choisir une loss function adaptée à la tâche attendue. *•* Problèmes de classification : Cross-entropy

*• L*(*y*ˆ*, y*) = *−*(*y · log*(*y*ˆ) + (1 *− y*) *· log*(1 *− y*ˆ)).

Thomas Ranvier Deep Learning - Introduction 23

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Loss functions 

Intérêt d’une loss function

*•* Elle permet de mesurer l’écart entre la prédiction faite par le modèle et la cible qu’on lui a donné. *•* Le modèle est entraîné à minimiser la valeur de cette fonction, c’est ce qui lui permet d’apprendre la tâche souhaitée.

Exemples de fonctions adaptées selon le problème

*•* Pour permettre un bon apprentissage il faut choisir une loss function adaptée à la tâche attendue. *•* Problèmes de classification : Cross-entropy

*• L*(*y*ˆ*, y*) = *−*(*y · log*(*y*ˆ) + (1 *− y*) *· log*(1 *− y*ˆ)).

*•* t f . keras . losses . BinaryCrossentropy ( ) ou t f . keras . losses . CategoricalCrossentropy ( ) . Thomas Ranvier Deep Learning - Introduction 23

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Loss functions 

Intérêt d’une loss function

*•* Elle permet de mesurer l’écart entre la prédiction faite par le modèle et la cible qu’on lui a donné. *•* Le modèle est entraîné à minimiser la valeur de cette fonction, c’est ce qui lui permet d’apprendre la tâche souhaitée.

Exemples de fonctions adaptées selon le problème

*•* Pour permettre un bon apprentissage il faut choisir une loss function adaptée à la tâche attendue. *•* Problèmes de classification : Cross-entropy

*• L*(*y*ˆ*, y*) = *−*(*y · log*(*y*ˆ) + (1 *− y*) *· log*(1 *− y*ˆ)).

*•* t f . keras . losses . BinaryCrossentropy ( ) ou t f . keras . losses . CategoricalCrossentropy ( ) . *•* Problèmes de régression : Mean Squared Error (MSE)

Thomas Ranvier Deep Learning - Introduction 23

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Loss functions 

Intérêt d’une loss function

*•* Elle permet de mesurer l’écart entre la prédiction faite par le modèle et la cible qu’on lui a donné. *•* Le modèle est entraîné à minimiser la valeur de cette fonction, c’est ce qui lui permet d’apprendre la tâche souhaitée.

Exemples de fonctions adaptées selon le problème

*•* Pour permettre un bon apprentissage il faut choisir une loss function adaptée à la tâche attendue. *•* Problèmes de classification : Cross-entropy

*• L*(*y*ˆ*, y*) = *−*(*y · log*(*y*ˆ) + (1 *− y*) *· log*(1 *− y*ˆ)).

*•* t f . keras . losses . BinaryCrossentropy ( ) ou t f . keras . losses . CategoricalCrossentropy ( ) . *•* Problèmes de régression : Mean Squared Error (MSE)

*• L*(*y*ˆ*, y*) = *||y*ˆ *− y||*22.

Thomas Ranvier Deep Learning - Introduction 23

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Loss functions 

Intérêt d’une loss function

*•* Elle permet de mesurer l’écart entre la prédiction faite par le modèle et la cible qu’on lui a donné. *•* Le modèle est entraîné à minimiser la valeur de cette fonction, c’est ce qui lui permet d’apprendre la tâche souhaitée.

Exemples de fonctions adaptées selon le problème

*•* Pour permettre un bon apprentissage il faut choisir une loss function adaptée à la tâche attendue. *•* Problèmes de classification : Cross-entropy

*• L*(*y*ˆ*, y*) = *−*(*y · log*(*y*ˆ) + (1 *− y*) *· log*(1 *− y*ˆ)).

*•* t f . keras . losses . BinaryCrossentropy ( ) ou t f . keras . losses . CategoricalCrossentropy ( ) . *•* Problèmes de régression : Mean Squared Error (MSE)

*• L*(*y*ˆ*, y*) = *||y*ˆ *− y||*22.

*•* t f . keras . losses . MeanSquaredError ( ) .

Thomas Ranvier Deep Learning - Introduction 23

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Loss functions 

Intérêt d’une loss function

*•* Elle permet de mesurer l’écart entre la prédiction faite par le modèle et la cible qu’on lui a donné. *•* Le modèle est entraîné à minimiser la valeur de cette fonction, c’est ce qui lui permet d’apprendre la tâche souhaitée.

Exemples de fonctions adaptées selon le problème

*•* Pour permettre un bon apprentissage il faut choisir une loss function adaptée à la tâche attendue. *•* Problèmes de classification : Cross-entropy

*• L*(*y*ˆ*, y*) = *−*(*y · log*(*y*ˆ) + (1 *− y*) *· log*(1 *− y*ˆ)).

*•* t f . keras . losses . BinaryCrossentropy ( ) ou t f . keras . losses . CategoricalCrossentropy ( ) . *•* Problèmes de régression : Mean Squared Error (MSE)

*• L*(*y*ˆ*, y*) = *||y*ˆ *− y||*22.

*•* t f . keras . losses . MeanSquaredError ( ) .

*•* On notera *L*(*f*(*x,* ***W***)*, y*) les fonctions de perte pour le reste du cours, avec *f* le réseau de neurones, *x* un vecteur de données, ***W*** les paramètres (poids et biais) du modèle et *y* la cible.

Thomas Ranvier Deep Learning - Introduction 23

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Fonction objectif **x ŷ y**

1,2 7,3 4,1

**…**

**x1 x2**

**ŷ**

0.2 0.7 0.8

**…**

1

0

1

**…**

Intérêt d’une fonction objectif

*•* La fonction objectif mesure la moyenne de la loss sur l’ensemble du jeu de données. Thomas Ranvier Deep Learning - Introduction 24

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Fonction objectif **x ŷ y**

1,2 7,3 4,1

**…**

**x1 x2**

**ŷ**

0.2 0.7 0.8

**…**

1

0

1

**…**

Intérêt d’une fonction objectif

*•* La fonction objectif mesure la moyenne de la loss sur l’ensemble du jeu de données. *•* Elle est notée *J*(***W***) = 1*n*P*ni*=1 *L*(*f*(*xi,* ***W***)*, yi*)

Thomas Ranvier Deep Learning - Introduction 24

Sommaire Contexte Perceptron Réseaux de neurones Apprentissage en théorie Apprentissage en pratique Types d’apprentissage Fonction objectif **x ŷ y**

1,2 7,3 4,1

**…**

**x1 x2**

**ŷ**

0.2 0.7 0.8

**…**

1

0

1

**…**

Intérêt d’une fonction objectif

*•* La fonction objectif mesure la moyenne de la loss sur l’ensemble du jeu de données. *•* Elle est notée *J*(***W***) = 1*n*P*ni*=1 *L*(*f*(*xi,* ***W***)*, yi*)

*•* On cherche les poids et biais qui permettent d’atteindre la loss la plus basse possible :

*W∗* = *argmin* ***W***

1

*J*(***W***) = *argmin n*

***W***

P*n*

*i*=1 *L*(*f*(*xi,* ***W***)*, yi*)

Thomas Ranvier Deep Learning - Introduction 24