Université Nouveaux Horizons

Machine Learning: Étude de la minimisation d'erreur dans l'apprentissage supervisé, avec une application de la technologie ANPR

Auteur :
TSHELEKA KAJILA Hassan

Directeur :
Prof. MASAKUNA Jordan

Mémoire présenté à la Faculté des Sciences Informatiques en vue de l'obtention du grade de Licencié en informatique.

en

Calcul Scientifique

18 avril 2022

RÉSUMÉ

Au cours de la dernière décennie, la taille des données a augmenté plus rapidement que la vitesse des processeurs. Dans ce contexte, faire un traitement de reconnaissance des formes dans des images et vidéos, les ensembles de données d'entraînement pour les problèmes de détection d'objets sont généralement très volumineux et les capacités des méthodes d'apprentissage automatique statistique sont limitées par le temps de calcul plutôt que par la taille de l'échantillon.

Le cas des problèmes d'apprentissage à grande échelle implique la complexité de calcul de l'algorithme d'optimisation sous-jacent de manière non triviale. Des algorithmes d'optimisation improbables tels que la **descente de gradient stochastique** (en anglais : **Stochastic Gradient Descent** ou SGD) montre des performances étonnantes pour les problèmes à grande échelle, lorsque l'ensemble d'apprentissage est volumineux.
En particulier, les variants du SGD n'utilisent qu'un seul nouvel échantillon d'apprentissage à chaque itération, sont asymptotiquement efficaces après un seul passage sur l'ensemble d'apprentissage.

Ce travail vise à proposer une méthode intelligente, basée sur l'intelligence artificielle, qui permet aux ordinateurs et aux systèmes informatiques de dériver des informations significatives à partir d'images numériques, de vidéos et d'autres entrées visuelles, avec un coût plus bas que possible. Dans notre contexte la reconnaissance des plaques d'immatriculation des véhicules à l'aide d'un classificateur de la famille de descente de gradient stochastique. Pour minimiser la **fonction coût** du classificateur, la SGD adopte un modèle d'optimisation convexe. De plus, pour augmenter la vitesse de convergence du classificateur, la descente de gradient stochastique, à chaque étape, elle tire un échantillon aléatoire de l'ensemble des fonctions (f_i), de la fonction objectif, constituant la somme.

Mots clés : Apprentissage supervisé, vision par ordinateur, Descente de gradient stochastique, Adaline, ANPR, ALPR.

ABSTRACT

Over the past decade, data size has grown faster than processor speeds. In this context, doing pattern recognition processing in real-time videos, training datasets for object detection problems are usually very large, and the capabilities of statistical machine learning methods are limited by computation time rather than sample size.

The case of large scale learning problems involves the computational complexity of the underlying optimization algorithm in a nontrivial way.
Improbable optimization algorithms such as **Stochastic Gradient Descent** (SGD) show amazing performance for large scale problems, when the training set is bulky.
In particular, SGD variants use only one new training sample at each iteration, are asymptotically efficient after a single pass over the training set.

This work aims to provide an intelligent method, based on artificial intelligence, that allows computers and computer systems to derive meaningful information from digital images, videos and other visual inputs, with a lower cost. as possible. In our context the recognition of vehicle license plates using a classifier of the family of stochastic gradient descent. To minimize the **cost function** of the classifier, the SGD adopts a convex optimization model. Moreover, to increase the speed of convergence of the classifier, the stochastic gradient descent, at each step, it draws a random sample from the set of functions (f_i), of the objective function, constituting the sum.

Key words : Supervised learning, computer vision, Stochastic gradient descent, Adaline, ANPR, ALPR.

0 INTRODUCTION

0.1 PRÉSENTATION (GÉNÉRALITÉS)

L'intelligence désigne communément le potentiel des capacités mentales et cognitives d'un individu, animal ou humain, lui permettant de résoudre un problème ou de s'adapter à son environnement. L'intelligence nous fait ressentir ce besoin d'apprendre pour arriver à nos fins, extrinsèquement l'intelligence c'est l'apprentissage. Pour que nous puissions dire qu'une machine est intelligente, premièrement elle doit passer par une phase d'apprentissage. Apprendre à résoudre des problèmes ou à réaliser des tâches par lui-même d'une façon autonome. Dans le IA nous parlons de l'apprentissage automatique (en anglais : Machine Learning, ML), nous utilisons plusieurs paradigmes d'apprentissage automatique, selon un contexte purement informatique : apprentissage supervisé, apprentissage non supervisé, apprentissage par renforcement, apprentissage profond.

L'apprentissage supervisé représente une grande partie de l'activité de recherche en apprentissage automatique (ML) et de nombreuses techniques d'apprentissage supervisé ont trouvé une application dans le traitement de contenu multimédia. La caractéristique qui définit l'apprentissage supervisé est la disponibilité de données d'apprentissage annotées[7]. Le nom évoque l'idée d'un **superviseur** qui instruit le système d'apprentissage sur les étiquettes à associer à des modèles [1] d'entraînement.

L'application de cette étude est orientée vers la reconnaissance automatique d'objet dans les vidéos et images, une des applications intéressantes, parmi tant d'autres, dans l'intelligence artificielle.
La reconnaissance automatique d'objet est un problème important dans la vision par ordinateur (Computer Vision [2]) et en traitement d'images. Cette tâche est très utile vue l'accroissement du nombre de vidéos générées par des smartphones, des systèmes de sécurité, des caméras de circulation et autres dispositifs dotés d'instruments visuels. La reconnaissance automatique des objets en vidéo peut ainsi renforcer la sécurité, faciliter la gestion des vidéos ainsi que permettre de nouvelles applications en interaction homme/machine.

1 Un modèle de machine learning est le résultat généré lorsque vous entraînez votre algorithme d'apprentissage automatique avec des données.

2 La vision par ordinateur est un domaine de l'intelligence artificielle (IA) qui permet aux ordinateurs et aux systèmes de dériver des informations significatives à partir d'images numériques, de vidéos et d'autres entrées visuelles, et de prendre des mesures ou de faire des recommandations sur la base de ces informations.

Par ailleurs, les images numériques et la vidéo sont devenues indispensables pour divers domaines d'application, tels que la détection d'intrusions pour la sécurité, la surveillance du trafic routier, la médecine pour l'imagerie médicale, ou encore lors des événements sportifs (ex., renforcement de l'arbitrage, création automatique de résumés). Des contraintes d'exploitation découlent des observations citées ci-dessus, parmi lesquelles nous citerons celles qui sont liées à la reconnaissance des objets en mouvement dans les vidéos. Par exemple, de nos jours, un très grand nombre de caméras est déployé exclusivement pour la surveillance vidéo [1] . Souvent, le contenu de ces vidéos est interprété par des opérateurs humains qui engendrent des coûts exorbitants pour le suivi et l'analyse du contenu, sans mentionner les erreurs qui peuvent être induites par la fatigue et l'inattention humaine. Une des interrogations importantes abordés lors l'apprentissage supervisé appliqué dans la surveillance vidéo est la reconnaissance des types d'objets en mouvement et leurs actions. Afin de détecter, par exemple, des menaces potentielles (ex., vols, attentats, accidents), ou tout simplement pour des fins de statistiques (ex., compter le nombre d'individus, de voitures dans une entrée de parc). Les applications du monde réel démontrent l'importance de la vision par ordinateur pour les entreprises, les secteurs du divertissement, des transports, des soins de santé et dans la vie quotidienne. L'un des principaux moteurs de la croissance de ces applications est le flot d'informations visuelles provenant des médias numériques (ex., internet, la télévision, les vidéos personnelles, la surveillance vidéo).

0.2 CONTEXTE ET PROBLÉMATIQUE DE NOTRE RECHERCHE

Ce travail présente les résultats d'une étude approfondie sur les algorithmes de minimisation d'erreur, la fonction coût[3] (en anglias : loss function).

Dans ce contexte, faire une une application dans le traitement de reconnaissance des formes dans des vidéos, les ensembles de données d'entraînement pour les problèmes de détection d'objets sont généralement très volumineux et les capacités des méthodes d'apprentissage automatique statistique sont limitées par le temps de calcul plutôt que par la taille de l'échantillon [3].
Par exemple, pour entraîner une machine à reconnaître des plaques d'immatriculation de voiture, elle doit recevoir de grandes quantités d'images de plaques d'immatriculation et d'éléments liés aux plaques pour apprendre les différences et reconnaître une plaque, en particulier la voiture qui porte une plaque sans défaut. Plus nous avons des données, plus nous gagnons en précision et plus la complexité en temps augmente.

Une analyse plus précise révèle des compromis qualitativement différents pour le cas des problèmes d'apprentissage à petite et à grande échelle [3]. La complexité de calcul de l'algorithme d'apprentissage devient le facteur limitant critique

3 Dans l'optimisation mathématique et en statistique, une fonction de perte ou une fonction de coût est généralement utilisée pour l'estimation des paramètres , et l'événement en question est une fonction de la différence entre les valeurs estimées et vraies pour une instance de données.

lorsque l'on envisage de très grands ensembles de données. C'est à ce point critique qu'entre en jeu cette étude, la minimisation des erreurs sans alourdir la complexité en temps et espace de l'algorithme d'apprentissage. Minimiser les erreurs dans les modèles d'apprentissage a toujours été une tâche très importante pour renforcer la fiabilité de notre Machine Learning Model [10]. Établir un algorithme d'apprentissage qui s'adapte au mieux à notre modèle, selon la nature du problème métier traité, il existe différentes approches qui varient selon le type et le volume des données. Dans cette section, nous discutons des algorithmes de descente de gradient stochastique parce qu'ils montrent des performances d'optimisation incroyables pour les problèmes à grande échelle [3].

Le travail de Léon bottou et al (e. g., [3] [14] [4]), présente *la descente de gradient stochastique comme un algorithme d'apprentissage fondamental*. L'un des piliers de l'apprentissage automatique est l'optimisation mathématique [Jorge Nocedal dans 5, page : 3], qui, dans ce contexte, implique le calcul numérique de minimisation des paramètres d'un système conçu pour prendre des décisions basées sur des données actuellement disponibles, ces paramètres sont choisis pour être optimaux par rapport à un problème d'apprentissage donné.

Dans l'ensemble, ce document tente d'apporter des réponses aux questions suivantes.

1. Comment les problèmes de minimisation surviennent-ils dans les applications d'apprentissage automatique et qu'est-ce qui les rend difficiles ?
2. Quelles ont été les méthodes minimisation les plus efficaces pour l'apprentissage supervisé à grande échelle et pourquoi ?
3. Comment des algorithmes d'apprentissage supervisé arrivent-t-ils résoudre le problème de la reconnaissance automatique d'objet ?
4. Quelles avancées récentes ont été réalisées dans la conception d'algorithmes d'apprentissage et quelles sont les questions ouvertes dans ce domaine de recherche ?

0.3 OBJECTIFS DE NOTRE ÉTUDE

Le but de cette étude est de fournir une revue et un commentaire sur le passé, le présent et le futur de l'utilisation des algorithmes d'optimisation numérique, précisément de minimisation, dans le contexte des applications d'apprentissage automatique qui permet aux ordinateurs et aux systèmes informatiques de dériver des informations significatives à partir d'images numériques, de vidéos et d'autres entrées visuelles, avec un coût plus bas que possible.

Expressément, nous faisons la reconnaissance des plaques d'immatriculation des véhicules à l'aide d'un classificateur de la famille de descente de gradient stochastique (SGD) [?]. Pour minimiser la fonction de coût du classificateur, le SGD adopte un modèle d'optimisation convexe. De plus, pour augmenter la vitesse de convergence du classificateur, la descente de gradient stochastique, à

chaque étape, elle tire un échantillon aléatoire de l'ensemble des fonctions (f_i), de la fonction objectif, constituant la somme.

Pour chaque algorithme, nous examinons l'efficacité et comparons le score pour différents cas. Les objectifs de notre travail regroupent les points suivants :

1. État des connaissances, c'est sont les éléments sur lequel je me base pour constituer ce travail, nous parlons des base mathématique essentiel pour le Machine Learning : les éléments différentiel, statistique, l'optimisation de modèle linéaire convexe, etc.

2. La méthodologie utilisée parmi tant d'autres, pour entraîner les modèles d'apprentissage automatique (Machine learning Model) de façon optimale. Pour la minimisation de la fonction coût nous utilisons des algorithmes comme SGD, ADAM, ADAGRAD, ADADELTA, ASGD, NAG. Puis faire une étude comparative de leurs performances. Et aussi des méthodes intelligentes de classification des images pour application reconnaissance automatique d'objet.

3. Nous construirons des modèles à partir d'une base de données annotée pour l'apprentissage et pour les tests de reconnaissance d'objets. Les résultats concluants de cette étude pourront conduire à un déploiement de notre système dans les domaines comme celui de la surveillance vidéo de voitures dans une entrée de parking. Des métriques connues pour mesurer les erreurs et en déduire le score du classificateur seront utilisées pour évaluer la qualité de la reconnaissance automatique des plaques d'immatriculation (en anglais : Automatic Number Plate Recognition ou ANPR) par notre approche.

Première partie

ÉTAT DES CONNAISSANCES (BACKGROUND MATERIAL)

État des connaissances, c'est sont les éléments sur lequel je me base pour constituer ce travail, nous parlons des base mathématique essentiel pour le Machine Learning : les éléments différentiel, statistique, l'optimisation numérique de modèle linéaire convexe, etc.

1

LES BASES MATHÉMATIQUES POUR LE MACHINE LEARNING

1.1 ELÉMENTS DE CALCUL DIFFÉRENTIEL

Cette section est inspirée des notes écrites par le Professeur TSHIMANGA [voir 9, page :45-82] et d'autres consignes données par Nocedal et al dans [5] [6][??].

1.1.1 *Convexité*

DÉFINITION : (ENSEMBLE CONVEXE) Une partie $\mathcal{C} \subset \mathbb{R}^n$ est dite convexe si et seulement si pour tout $(x, y) \in \mathcal{C}^2$, et pour tout $\alpha \in [0, 1]$, $\alpha x + (1 - \alpha)y \in \mathcal{C}$ combinaison convexe [9].

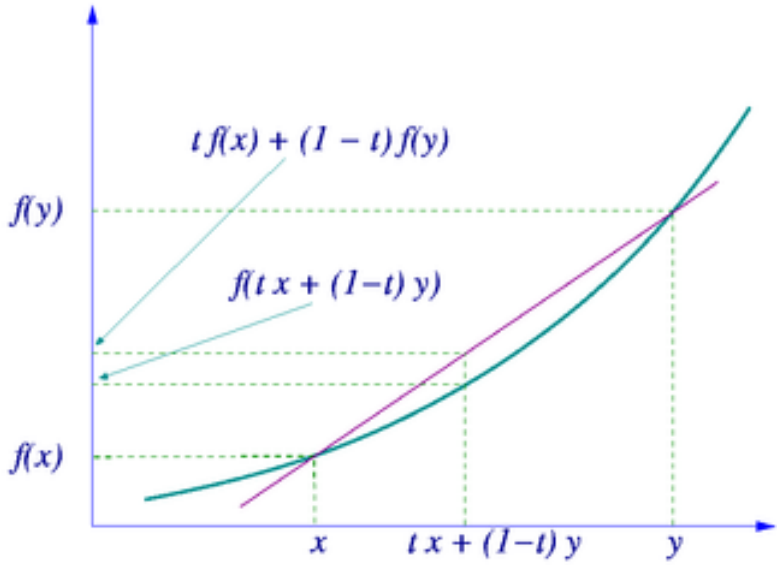

FIGURE 1 : Illustration fonction convexe [image de Wikipédia]

DÉFINITION : (FONCTION CONVEXE) Une fonction f d'un intervalle réel $I \in \mathcal{C}$ est dite fonction convexe lorsque, $\forall (x, y)$ de I tel que $(x, y) \in \mathcal{C}^2$ et tout $\alpha \in [0, 1]$ on a :

$$f(\alpha x + (1 - \alpha)y) \leqslant \alpha f(x) + (1 - \alpha)f(y) \tag{1}$$

et si

$$f(\alpha x + (1-\alpha)y) < \alpha f(x) + (1-\alpha)f(y) \tag{2}$$

on dit que la fonction est strictement convexe dans $\mathcal{C}$, [voir dans 9, page :45]

Exemple [9] :

- La fonction $f(x) = x^2$ est convexe.
- La fonction $f(x) = x^T x$ est convexe.
- La fonction $f(x) = x^T Ax$ est convexe, ssi A est symétrique semi-définie positive.

PROPRIÉTÉ D'UNE FONCTION DÉRIVABLE : (Extremum local) Parmi les propriétés de dérivabilité il existe une qui est mise en relation avec l'effect qu'une fonction doit être convexe. énoncé ci-dessous [voir dans 6, page :212].
Soit $I \to \mathbb{R}$ une fonction et a un point de I.

+ On dit que m est un **minimum local** de f s'il existe $a > 0$ tel que m soit le minimum de f restreinte à $I \cap]a-\alpha, a+\alpha[$.
+ On dit que M est un **maximum local** de f s'il existe $a > 0$ tel que M soit le maximum de f restreinte à $I \cap]a-\alpha, a+\alpha[$.

Donc nous pouvons dire qu'une fonction convexe à un unique point minimum.

Développement limité

En physique et en mathématiques, un développement limité (noté DL) d'une fonction en un point est une approximation polynomiale de cette fonction au voisinage de ce point, c'est-à-dire l'écriture de cette fonction sous la forme de la somme d'une fonction polynomiale et d'un reste négligeable au voisinage du point considéré [6].

Soit f une fonction à valeurs réelles définie sur un intervalle I, et $x_0 \in I$. On dit que f admet un développement limité d'ordre n[2] (abrégé par DL_n) en x_0, s'il existe $n+1$ réels $a_0, a_1, \ldots, a_n$ tels que la fonction $R : I \to \mathbb{R}$ définie par :

$$f(x) = a_0 + a_1(x-x_0) + a_2(x-x_0)^2 + \cdots + a_n(x-x_0)^n + R(x) = \sum_{i=0}^{n} a_i(x-x_0)^i + R(x)$$

vérifie : $R(x)$ tend vers 0 lorsque x tend vers x_0, et ce plus rapidement que le dernier terme de la somme, c'est-à-dire que :

$$\lim_{x \to x_0} \frac{R(x)}{(x-x_0)^n} = 0.$$

La fonction reste $R(x)$ vérifiant ceci est notée $o((x–x0)^n)$ (selon la notation de Landau). On écrit donc :

$$f(x) = \sum_{i=0}^{n} a_i(x-x_0)^i + R(x) = \sum_{i=0}^{n} a_i(x-x_0)^i + o((x-x_0)^n)$$

Il est fréquent d'écrire un développement limité en posant $x = x0 + h$ on aura :

$$f(x_0+h) = \sum_{i=0}^{n} a_i h^i + o(h^n)$$

CONSÉQUENCES IMMÉDIATES

- Si f admet un DL_0 en x_0, alors $a_0 = f(x_0)$. [6]
- Si f admet un DL_n en x_0, alors elle admet un DL_k en x_0 pour tout entier $k < n$ [6].
- Une condition nécessaire et suffisante pour que f admette un DL_n en x_0 est l'existence d'un polynôme P tel que $f(x) = P(x) + o((x–x_0)^n)$ [6]. S'il existe un tel polynôme P, alors il en existe une infinité d'autres, mais un seul d'entre eux est de degré inférieur ou égal à n : le reste de la division euclidienne de $P(X)$ par $(X–x_0)^{n+1}$. On l'appelle la partie régulière, ou partie principale, du DL_n de f en x_0.

Le théorème de Taylor-Young assure [voir dans 6, page :241] qu'une fonction f dérivable n fois au point x_0 (avec $n \geqslant 1n \geqslant 1$) admet un DLn en ce point :

$$f(x) = f(x_0) + f'(x_0)(x-x_0) + \frac{f''(x_0)}{2!}(x-x_0)^2 + \cdots + \frac{f^{(n)}(x_0)}{n!}(x-x_0)^n + o((x-x_0)^n)$$

soit en écriture abrégée

$$f(x) = \sum_{i=0}^{n} \frac{f^{(i)}(x_0)}{i!}(x-x_0)^i + o((x-x_0)^n)$$

Le développement d'ordre 0 en x_0 revient à écrire que f est continue en x_0 :

$$f(x) = f(x_0) + o((x-x_0)^0) = f(x_0) + o(1)$$

Le développement limité d'ordre 1 en x_0 revient à approcher une courbe par sa tangente en x_0 on parle aussi d'approximation affine :

$$f(x) = f(x_0) + f'(x_0)\cdot(x-x_0) + o(x-x_0)$$

Différentiabilité au sens de Fréchet

Soient E un espace vectoriel normé, F un espace vectoriel topologique séparé, f une application de E dans F et a un point de E. On abandonne la notation des vecteurs par des flèches dans ce paragraphe.

On dit que f est différentiable en a (au sens de Fréchet) s'il existe une application linéaire continue $L : E \to f$ telle que :

$$\forall h \in E \quad f(a+h) = f(a) + L(h) + o(\|h\|)$$

ou, de manière équivalente :

$$\lim_{h \to 0} \frac{f(a+h) - f(a) - L(h)}{\|h\|} = 0.$$

Une telle application linéaire L est alors unique. L'opérateur L est appelé différentielle de Fréchet (ou F-différentielle, ou Fréchet-différentielle) de f au point a, et f est dite Fréchet-différentiable (ou différentiable, ou différentiable au sens de Fréchet) au point a. La différentielle de f au point a est souvent notée $Df(a)$, la notation $f'(a)$ est aussi utilisée.

1.1.2 *Fonctions dérivables*

Gradient

DÉFINITION : Le gradient d'une fonction de plusieurs variables en un certain point est un vecteur qui caractérise la variabilité de cette fonction au voisinage de ce point. Défini en tout point où la fonction est différentiable, il définit un champ de vecteurs, également dénommé gradient. Le gradient est la généralisation à plusieurs variables de la dérivée d'une fonction d'une seule variable.

DÉFINITION MATHÉMATIQUE : Dans un système de coordonnées cartésiennes, le gradient d'une fonction $f(x_1, x_2, \ldots, x_n)$ est le vecteur de composantes $\partial f / \partial x_i$ ($i = 1, 2, \ldots, n$), c'est-à-dire les dérivées partielles de f par rapport aux coordonnées [9].

$$\nabla f(x) = \begin{bmatrix} \frac{\partial f(x)}{\partial x_1} \\ \vdots \\ \frac{\partial f(x)}{\partial x_n} \end{bmatrix} \in \mathbb{R}^n$$

GRADIENT SOUS FORME DE DÉVELOPPEMENT LIMITÉ : *Si une application admet un gradient en un point, alors on peut écrire ce développement limité du premier ordre (voir le point 1.1.1).*

$$f(x+h) = f(x) + \langle \nabla f(x) \mid h \rangle + o(h)$$

ou

$$f(x-h) = f(x) - \langle \nabla f(x) \mid h \rangle + o(h)$$

Numériquement, il est très intéressant de faire ensuite la demi-différence des deux développements pour obtenir la valeur du gradient et on note que celui-ci ne dépend pas en fait de la valeur de la fonction au point $x : f(x)$. *Cette formule a l'avantage de tenir compte des gradients du 2e ordre et est donc beaucoup plus précise et numériquement robuste. L'hypothèse est, en pratique, de connaitre les valeurs "passé" et "futur" de la fonction autour d'un petit voisinage du point* x.

DÉFINITION NUMÉRIQUE : Une fonction multivariée (à variable vectorielle) $f(x) : \mathbb{R}^n \to \mathbb{R} : x \to f(x)$ définie sur un ouvert $O \in \mathbb{R}^n$ est dite dérivable (au sens de Fréchet, voir le point 1.1.1) en x ssi il existe un vecteur noté $\nabla f(x) \in \mathbb{R}^n$ tel que

$$f(x+h) = f(x) + \nabla f(x)^T h + o(||h||) \tag{3}$$

$\nabla f(x) \in \mathbb{R}^n$ et où l'on a posé que le reste $o(||h||) = ||h||\epsilon(h) \in \mathbb{R}^n$, avec $h \in \mathbb{R}^n$

$$\epsilon(h) : \mathbb{R}^n \to \mathbb{R}, \qquad \lim_{||h|| \to 0} \epsilon(h) = 0.$$

Le vecteur $\nabla f(x)$ est unique et s'appelle **gradient** de $f(x)$ en x. Le gradient s'adresse aux fonctions scalaires à variables vectorielles.

A PROPOS DE LA NOTATION **o(| | h | |)** : La notation de Landau $o(||h||)$ traduit le comportement d'une fonction de h qui [est ? ?] tend vers 0 d'un ordre de grandeur plus vite que $||h||$.

Elle est infiniment plus petit que h dans le voisinage de 0

Hessienne

DÉFINITION MATHÉMATIQUE : Étant donnée une fonction f à valeurs réelles

$$f : \mathbb{R}^n \to \mathbb{R}; (x_1, ..., x_n) \mapsto f(x_1, ..., x_n)$$

dont toutes les dérivées partielles secondes existent, le coefficient d'indice i, j de la **matrice hessienne**[1] $H(f)$ vaut $H_{ij}(f) = \frac{\partial^2 f}{\partial x_i \partial x_j}$.
Autrement dit,

$$H(f) = \begin{bmatrix} \frac{\partial^2 f}{\partial {x_1}^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial {x_2}^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial {x_n}^2} \end{bmatrix}.$$

DÉFINITION NUMÉRIQUE : Supposons que $f : \mathbb{R}^n \to \mathbb{R}$ définie sur un ouvert $\mathcal{O} \in \mathbb{R}^n$. La fonction $f(x)$ est dite 2 fois continûment dérivable (au sens de Fréchet ??) si en tout $x \in \mathcal{O}$ on a

$$f(x+h) = f(x) + \nabla f(x)^T h + \frac{1}{2} h^T \nabla^2 f(x) h + o(||h||^2) \tag{4}$$

avec$\nabla f(x) \in \mathbb{R}^{n \times n}$ et où on a posé que le reste $o(||h||^2) = ||h|| \epsilon(h) \in \mathbb{R}$ avec $\lim_{||h|| \to 0} \epsilon(h) = 0$ La matrice carrée symétrique $\nabla^2 f(x)$ appelée **Hessien** de $f(x)$ en x.
Remarque :

$$\lim_{||h|| \to h} \frac{o(||h||^2)}{||h||} = 0 \in \mathbb{R}$$

La Hessienne s'adresse aux fonctions scalaires à variables vectorielles.

Jacobienne

DÉFINITION MATHÉMATIQUE : Soit F une fonction d'un ouvert de $\mathbb{R}^n$ à valeurs dans $\mathbb{R}^m$ ($F : \mathbb{R}^n \to \mathbb{R}^m$). Une telle fonction est définie par ses m fonctions composantes à valeurs réelles :

$$F : \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \longmapsto \begin{pmatrix} f_1(x_1, \ldots, x_n) \\ \vdots \\ f_m(x_1, \ldots, x_n) \end{pmatrix}.$$

1 En mathématiques, la matrice hessienne (ou simplement la hessienne) d'une fonction numérique f est la matrice carrée, notée $H(f)$, de ses dérivées partielles secondes.

Les dérivées partielles de ces fonctions en un point M, si elles existent, peuvent être rangées dans une matrice à m lignes et n colonnes, appelée **matrice jacobienne**[2] de F :

$$J_F(M) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{pmatrix}.$$

La case sur la ligne i et la colonne j contient $\frac{\partial f_i}{\partial x_j}$ qui est la dérivée partielle de fi selon la variable xj. Cette matrice est notée :

$$J_F(M), \qquad \frac{\partial(f_1,\ldots,f_m)}{\partial(x_1,\ldots,x_n)} \qquad \text{ou} \qquad \frac{D(f_1,\ldots,f_m)}{D(x_1,\ldots,x_n)}$$

Pour $i = 1,\ldots,m$, la i-ème ligne de cette matrice est la transposée du vecteur **gradient** (voir le point 1.1.2) au point M de la fonction f_i, lorsque celui-ci existe. La matrice jacobienne est également la matrice de la différentielle de la fonction, lorsque celle-ci existe.

DÉFINITION NUMÉRIQUE : Soit $f(x) : \mathbb{R}^n \to \mathbb{R}^m$ définie sur un ouvert $\mathcal{O} \subset \mathbb{R}$. On dit que $f(x)$ est dérivable (au sens de Fréchet) en x, si chacune des composantes $f_i(x)$ est dérivable en x. On a alors

$$f(x+h) = f(x) + D_f(x)h + o(||h||) \tag{5}$$

avec $D_f(x) \in \mathbb{R}^{n\times m}$ et/où $o(||h||) = ||h||\epsilon(h) \in \mathbb{R}^m$ avec $\lim_{||h||\to 0} \epsilon(h) = 0$. Remarque :

$$\lim_{||h||\to h} \frac{o(||h||^2)}{||h||} = 0 \in \mathbb{R}$$

Soient $x = \begin{bmatrix} x_1 & x_2 & \cdots & x_n \end{bmatrix}^T \in \mathbb{R}^n$ et $f(x) = \begin{bmatrix} f_1(x) & f_2(x) & \cdots & f_n(x) \end{bmatrix}^T \in \mathbb{R}^m$

$$D_f(x) = \begin{bmatrix} \frac{\partial f_1(x)}{\partial x_1} & \cdots & \frac{\partial f_1(x)}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m(x)}{\partial x_1} & \cdots & \frac{\partial f_m(x)}{\partial x_n} \end{bmatrix} = \begin{bmatrix} \nabla f_1(x)^T \\ \nabla f_2(x)^T \\ \vdots \\ \nabla f_m(x)^T \end{bmatrix} \in \mathbb{R}^{n\times m},$$

La matrice $D_f(x) \in \mathbb{R}^{n\times m}$ est appelée **Jacobienne** de f(x) en x. La Jacobienne s'adresse aux fonctions vectorielles à variables vectorielles.

NOTE : Lorsque $m = 1$ (m : nombre des lignes), la Jacobienne est la même que le gradient car il s'agit d'une généralisation du gradient.

2 En analyse vectorielle, la matrice jacobienne est la matrice des dérivées partielles du premier ordre d'une fonction vectorielle en un point donné.

1.2 STATISTIQUE & PROBABILITÉ

1.2.1 *Échantillonnage (Statistique)*

En statistiques, l'échantillonnage est la sélection d'un sous-ensemble (un échantillon statistique) d'individus au sein d'une population statistique pour estimer les caractéristiques de l'ensemble de la population.

Sur un échantillon, on peut calculer différents paramètres statistiques de position (moyenne, etc.) ou de dispersion (écart type, etc.) issus de la statistique descriptive, de la même manière que l'on peut déterminer des paramètres statistiques d'une population par son recensement exhaustif.

On peut également déduire des propriétés de la population à partir de celles de l'échantillon par inférence statistique. D'après la loi des grands nombres, plus la taille de l'échantillon augmente, plus ses propriétés seront proches de celle de la population. En particulier, on peut estimer une probabilité sur les individus d'une population par la fréquence observée sur un échantillon si sa taille est suffisamment grande.

Cette méthode présente plusieurs avantages : une étude restreinte sur une partie de la population, un moindre coût, une collecte des données plus rapide que si l'étude avait été réalisé sur l'ensemble de la population, la réalisation de contrôles destructifs, etc.

On peut procéder de différentes manières pour collecter les données de l'échantillon, il existe en effet plusieurs méthodes d'échantillonnage [12] :

- **Échantillonnage aléatoire et simple** : le tirage des individus de l'échantillon est aléatoire, c'est-à-dire que chaque individu a la même probabilité d'être choisi, et simple, c'est-à-dire que les choix des différents individus sont réalisés indépendamment les uns des autres.
- **Échantillonnage systématique** : le premier individu est choisi de manière aléatoire, puis les suivants sont déterminés à intervalle régulier. Par exemple, dans un verger, on choisit au hasard le 7e pommier, puis les 27e, 47e, 67e, etc.
- **Échantillonnage stratifié** : on subdivise la population en plusieurs parties avant de prendre l'échantillon1.
- **Échantillonnage par quotas** : la composition de l'échantillon doit être représentative de celle de la population selon certains critères jugés particulièrement importants. On utilise cette méthode pour réaliser les sondages d'opinions.

La collecte de données

La collecte de données est le processus de collecte et de mesure des informations sur des variables ciblées dans un système établi, qui permet ensuite de répondre aux questions pertinentes et d'évaluer les résultats.

Une bonne collecte de données implique :

- Suivre le processus d'échantillonnage défini
- Garder les données dans l'ordre du temps
- Noter les commentaires et autres événements contextuels
- Enregistrement des non-réponses

ERREUR D'ÉCHANTILLONNAGE : Dans les statistiques, les erreurs d'échantillonnage se produisent lorsque les caractéristiques statistiques d'une population sont estimées à partir d'un sous-ensemble, ou échantillon, de cette population. Étant donné que l'échantillon n'inclut pas tous les membres de la population, les statistiques de l'échantillon (souvent appelées estimateurs), telles que les moyennes et les quartiles, diffèrent généralement des statistiques de l'ensemble de la population (appelées paramètres). La différence entre la statistique d'échantillon et le paramètre de population est considérée comme l'erreur d'échantillonnage [12].

1.2.2 *Analyse bayésienne*

La statistique bayésienne est une théorie dans le domaine des statistiques basée sur l' interprétation bayésienne de la probabilité où la probabilité exprime un degré de croyance en un événement. Le degré de croyance peut être basé sur des connaissances antérieures sur l'événement, telles que les résultats d'expériences précédentes, ou sur des croyances personnelles sur l'événement. Cela diffère d'un certain nombre d'autres interprétations de la probabilité , telles que l' interprétation fréquentiste qui considère la probabilité comme la limite de la fréquence relative d'un événement après de nombreux essais [??].

Les statistiques bayésiennes portent le nom de Thomas Bayes[3], qui a formulé un cas spécifique du théorème de Bayes dans un article publié en 1763.

Theorem 1 (Théorème de Bayes) *Le théorème de Bayes est utilisé dans les méthodes bayésiennes pour mettre à jour les probabilités, qui sont des degrés de croyance, après avoir obtenu de nouvelles données. Compte tenu de deux événements* A *et* B*, la probabilité conditionnelle de* A *étant donné que* B *est vrai s'exprime comme suit :*

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(B|A)\mathbb{P}(A)}{\mathbb{P}(B)} \tag{6}$$

3 Thomas Bayes était un Anglais statisticien , philosophe et ministre presbytérien qui est connu pour la formulation d' un cas spécifique du théorème qui porte son nom : théorème de Bayes.

où $\mathbb{P}(B) \neq 0$ Bien que le théorème de Bayes soit un résultat fondamental de la théorie des probabilités , il a une interprétation spécifique dans les statistiques bayésiennes [2].

2

APPRENTISSAGE AUTOMATIQUE : MODÉLISATION ET CLASSIFICATION

2.1 GÉNÉRALITÉ

2.1.1 *Les ingrédients d'apprentissage*

Résoudre un problème d'apprentissage, c'est d'abord le comprendre, c'est-à-dire discuter longuement avec les experts du domaine concerné pour identifier quelles sont les "entrées", les "sorties" ou résultats désirés, les connaissances disponibles, les particularités des données, par exemple : valeurs manquantes, taux de bruit dans les mesures des attributs de description, proportions des classes, stationnarité ou pas de l'environnement. C'est aussi réaliser un gros travail de *préparation des données* : nettoyage, ré-organisation, enrichissement, intégration avec d'autres sources de données, etc. Ces étapes de compréhension du problème, de préparation des données, de mise au point du protocole d'apprentissage et des mesures d'évaluation des résultats, prennent, et de loin, la plus grande partie du temps pour (tenter de) résoudre un problème d'apprentissage [2]. Nous avons toujours tendance à largement sous-estimer ces étapes et à vouloir se concentrer uniquement sur la phase excitante de l'essai de méthodes d'apprentissage sur des données supposées bonnes à la consommation.

Algorithme qui apprennent

2.1.2 *Modélisation*

La modélisation est la conception et l'utilisation d'un *modèle*. Selon son objectif et les moyens utilisés, la modélisation est dite mathématique, géométrique, 3D, empirique, etc. En informatique, la modélisation permet de concevoir l'architecture globale d'un système d'information, ainsi que l'organisation des informations à l'aide de la modélisation des données ;

MODÈLE (INFORMATIQUE) : En informatique, un modèle a pour objectif de structurer les informations et activités d'une organisation : données, traitements, et flux d'informations entre entités.

MODÈLE (MATHÉMATIQUE) : Un modèle mathématique est une description d'un système utilisant des concepts et un langage mathématiques.

Un modèle peut aider à expliquer un système et à étudier les effets de différents composants, et à faire des prédictions sur le comportement.

E. g. : Prenons l'exemple de données décrites dans l'espace d'entrée $\mathcal{X} = \mathbb{R}^n$ avec n variables réelles et supposons-les étiquetées par $\times$ ou par $\bullet$. On cherche donc une fonction de décision h, appelée hypothèse ou modèle, telle qu'elle soit capable d'étiqueter toute entrée $x \in \mathcal{X}, h : x \to \{\times, \bullet\}$. Reste à définir l'espace des hypothèses ou modèles $\mathcal{H}$ que l'on est prêt à considérer.

Toujours en considérant le problème de prédiction basique (présenté ci-dessus), on pourrait définir une hypothèse par une procédure qui examine les trois plus proches voisins du point à étiqueter x et qui choisit l'étiquette majoritaire parmi ces trois points pour étiqueter x. Il n'y a évidemment plus de paramètres pour définir les modèles possibles [2].

Un **modèle non paramétrique** est construit selon les informations provenant des données. Dans [2] il est expliqué que : La régression non paramétrique exige des tailles d'échantillons plus importantes que celles de la régression basée sur des modèles paramétriques parce que les données doivent fournir la structure du modèle ainsi que les estimations du modèle

Un **modèle paramétrique** est, s'il est approximativement valide, plus puissant qu'un modèle non paramétrique, produisant des estimations d'une fonction de régression qui ont tendance à être plus précises que ce que nous donne l'approche non paramétrique [11]. Cela devrait également se traduire par une prédiction plus précise.

Entraînement du modèle

Tout modèle, où toutes les informations nécessaires ne sont pas disponibles, contient certains paramètres qui peuvent être utilisés pour adapter le modèle au système qu'il est censé décrire. Si la modélisation est effectuée par un réseau de neurones artificiels ou un autre apprentissage automatique, l'optimisation des paramètres est appelée **entraînement** (en anglais : **training**), tandis que l'optimisation des hyperparamètres du modèle est appelée **réglage** (en anglais : **tuning**) et utilise souvent la validation croisée [??]. Dans une modélisation plus conventionnelle à travers des fonctions mathématiques explicitement données, les paramètres sont souvent déterminés par ajustement de courbe.

Une partie cruciale du processus de modélisation consiste à évaluer si oui ou non un modèle mathématique donné décrit un système avec précision. Il peut être difficile de répondre à cette question car elle implique plusieurs types d'évaluation différents.

???? Un modèle de régression linéaire ajusté peut être utilisé pour identifier la relation entre une seule variable prédictive x_j et la variable de réponse y lorsque toutes les autres variables prédictives du modèle sont "maintenues fixes". Plus précisément, l'interprétation de β_j est la variation attendue de y pour une variation d'une unité de x_j lorsque les autres covariables sont maintenues fixes,

c'est-à-dire la valeur attendue de la dérivée partielle de y par rapport à x_j. Ceci est parfois appelé l'effet unique de x_j sur y. En revanche, l'effet marginal de x_j sur y peut être évalué à l'aide d'un coefficient de corrélation ou d'un simple modèle de régression linéaire reliant uniquement x_j à y ; cet effet est la dérivée totale de y par rapport à x_j.

2.2 RÉGRESSION LINÉAIRE

Dans la modélisation statistique, l'analyse de régression est un ensemble de processus statistiques permettant d'estimer les relations entre une variable dépendante et une ou plusieurs variables indépendantes [??].
En statistique, la régression linéaire est une approche linéaire pour modéliser (voir le point 2.1.2) la relation entre une réponse scalaire et une ou plusieurs variables explicatives (également appelées variables dépendantes et indépendantes). Le cas d'une variable explicative est appelé régression linéaire simple ; pour plus d'un, le processus est appelé régression linéaire multiple.

Dans la régression linéaire, les relations sont modélisées à l'aide de *fonctions prédictives*[1] linéaires dont les paramètres de modèle inconnus sont estimés à partir des données [11]. De tels modèles sont appelés modèles linéaires.

La régression linéaire a de nombreuses utilisations pratiques. Si l'objectif est la prédiction, la prévision ou la réduction des erreurs, la régression linéaire peut être utilisée pour ajuster un modèle prédictif à un ensemble de données observées de valeurs de la réponse et de variables explicatives [8]. Après avoir développé un tel modèle, si des valeurs supplémentaires des variables explicatives sont collectées sans valeur de réponse d'accompagnement, le modèle ajusté peut être utilisé pour faire une prédiction de la réponse.

2.2.1 *Le problème de la régression linéaire*

On appelle problèmes de régression de tels problèmes, dans lesquels la sortie est numérique, généralement un vecteur de réels, supposé dépendre de la valeur d'un certain nombre de facteurs en entrée.

Le vecteur d'entrée $x = (1, 2, ..., x)^T$ est souvent appelé variable indépendante, tandis que le vecteur de sortie y est appelé variable dépendante. On formalise le problème en supposant que la sortie résulte de la somme d'une fonction déterministe f de l'entrée et d'un bruit aléatoire :

$$y = f(x) + \epsilon \tag{7}$$

1 En statistique et en apprentissage automatique, une fonction de prédicteur linéaire est une fonction linéaire d'un ensemble de coefficients et de variables explicatives, dont la valeur est utilisée pour prédire le résultat d'une variable dépendante.

où $f(x)$ est la fonction inconnue que nous souhaitons approcher par un estimateur $h(x|w)$, où h est défini à l'aide d'un vecteur w de paramètres.
Si l'on suppose que le bruit ϵ est un phénomène gaussien de moyenne nulle et de variance constante σ^2, c'est-à-dire $\epsilon = \mathcal{N}(0, \sigma^2)$, alors, en plaçant notre estimateur $h(\cdot)$ à la place de la fonction inconnue, on devrait avoir la densité conditionnelle réelle $p(y|x)$ vérifiant :

$$p(y|x) = \mathcal{N}(h(x|w), \sigma^2) \tag{8}$$

On peut estimer le vecteur de paramètres w grâce au principe de maximisation de la vraisemblance. On suppose que les couples (y_t, y_t) de l'échantillon d'apprentissage sont tirés par tirages indépendants d'une distribution de probabilités jointes inconnue $p(x, y)$, qui peut s'écrire :

$$p(y|x) = p(y|x)p(x)$$

où $p(y|x)$ est la probabilité de la sortie étant donnée l'entrée et $p(x)$ est la densité de probabilité sur les entrées.

[11]

2.2.2 *Le cas de la régression générale*

La plupart des modèles de régression proposent que Y_i est une fonction de X_i et β, avec ϵ_i représentant un terme d'erreur additif qui peut remplacer des déterminants non modélisés de Y_i ou bruit statistique aléatoire :

$$Y_i = f(X_i, \beta) + \epsilon_i \tag{9}$$

L'objectif est d'estimer la fonction $f(X_i, \beta)$ qui correspond le mieux aux données.

Pour effectuer une analyse de régression, la forme de la fonction f doit être spécifié. Parfois, la forme de cette fonction est basée sur la connaissance de la relation entre Y_i et X_i. Si ces connaissances ne sont pas disponibles, un formulaire souple ou pratique pour f est choisi. Par exemple, une simple régression univariée peut proposer

$$f(X_i, \beta) = \beta_0 + \beta_1 X_i$$

ou

$$Y_i = \beta_0 + \beta_1 X_i + e_i$$

être une approximation raisonnable du processus statistique générant les données.

Différentes formes d'analyse de régression fournissent des outils pour estimer les paramètres. β. Par exemple, les moindres carrés trouvent la valeur de β qui minimise la somme des carrés des erreurs

$$\sum_i (Y_i - f(X_i, \beta))^2$$

Étant donné un ensemble de données $\{y_i, x_{i1}, \ldots, x_{ip}\}_{i=1}^{n}$ de n unités statistiques, un modèle de régression linéaire suppose que la relation entre la variable dépendante y et le vecteur p des régresseurs x est linéaire. Cette relation est modélisée par un terme de perturbation ou une variable d'erreur ϵ : une variable aléatoire non observée qui ajoute du "bruit" à la relation linéaire entre la variable dépendante et les régresseurs. Ainsi le modèle prend la forme

$$y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i = \mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon_i, \qquad \text{avec} \quad i = 1, \ldots, n,$$

Souvent, ces n équations sont empilées et écrites en notation matricielle comme

$$\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon},$$

où

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}, \quad X = \begin{pmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \vdots \\ \mathbf{x}_n^T \end{pmatrix} = \begin{pmatrix} 1 & x_{11} & \cdots & x_{1p} \\ 1 & x_{21} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \cdots & x_{np} \end{pmatrix}, \quad \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{pmatrix}, \quad \boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{pmatrix}.$$

$\mathbf{y}$ est un vecteur de valeurs observées y_i $(i = 1, \ldots, n)$ de la variable appelée variable mesurée ou variable dépendante.

X peut être vu comme une matrice de vecteurs-lignes $\mathbf{x}_i$ ou de vecteurs-colonnes à n dimensions X_j, appelées régresseurs, variables explicatives, variables d'entrée, variables prédictives ou variables indépendantes. La matrice X est parfois appelée la matrice de conception.

$\boldsymbol{\beta}$ est un vecteur de paramètre de dimension $(p+1)$, où β_0 est le terme d'interception, s'il n'est inclus dans le modèle $\boldsymbol{\beta}$ est de dimension p. Ses éléments sont appelés effets ou coefficients de régression. En régression linéaire simple , $p = 1$, et le coefficient est appelé **pente** de régression.
L'estimation statistique et l'inférence dans la régression linéaire se concentrent sur β. Les éléments de ce vecteur de paramètres sont interprétés comme les dérivées partielles de la variable dépendante par rapport aux différentes variables indépendantes.

2.3 RÉGRESSION LOGISTIQUE

2.3.1 *Le problème de classification*

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

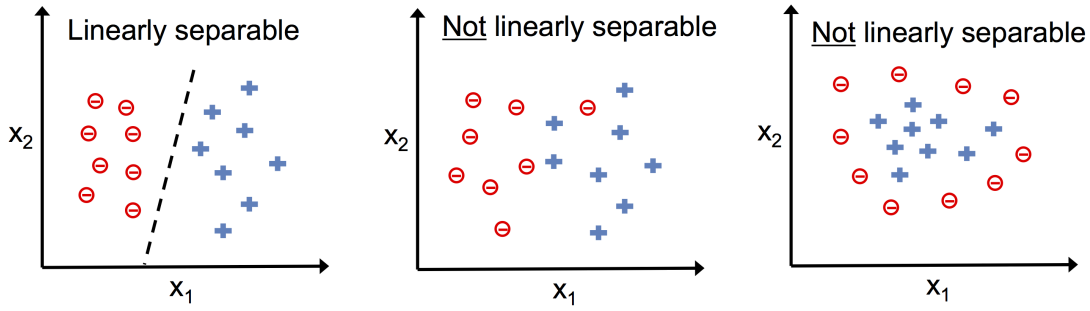

FIGURE 2 : Classes linéairement séparables [image de 13, page–48]

2.3.2 *Le cas séparable*

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

2.3.3 *Le cas non séparable*

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

2.3.4 *Le modèle de la régression logistique*

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

2.4 CLASSIFICATIONS

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

2.5 APPRENTISSAGE PROFOND (DEEP LEARNING)

2.5.1 *Perceptron*

2.5.2 *Neurones*

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Réseau neuronal convolutif (CNN)

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec

varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Réseau neuronal récurrent (RNN)

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Deuxième partie

ANNEXES ET BIBLIOGRAPHIES

BIBLIOGRAPHIE

[1] Yaovi Ahadjitse. "Reconnaissance d'objets en mouvement dans la vidéo par description géométrique et apprentissage supervisé". Thèse de doct. Université du Québec en Outaouais, 2013.

[2] Vincent Barra Antoine Cornuéjols Laurent Michet. *Apprentissage automatique : Deep leaning, concepts et algorithmes*. 3rd. Eyrolles, 2018, p. 239-263.

[3] Léon Bottou. "Large-scale machine learning with stochastic gradient descent". In : *Proceedings of COMPSTAT'2010*. Springer, 2010, p. 177-186.

[4] Léon Bottou. "Stochastic gradient descent tricks". In : *Neural networks: Tricks of the trade*. Springer, 2012, p. 421-436.

[5] Léon Bottou, Frank E Curtis et Jorge Nocedal. "Optimization methods for large-scale machine learning". In : *Siam Review* 60.2 (2018), p. 223-311.

[6] F. Coulombeau, G. Debeaumarché, B. David, F. Dorra, S. Dupont et M. Hochart. *Mathématiques MPSI-PCSI: Programme 2013 avec algorithmique en Scilab*. Cap Prépa. Pearson, 2013. isbn : 9782744076527. url : https://books.google.cd/books?id=e4vfnQEACAAJ.

[7] Pádraig Cunningham, Matthieu Cord et Sarah Jane Delany. "Supervised learning". In : *Machine learning techniques for multimedia*. Springer, 2008, p. 21-49.

[8] R.B. Darlington et A.F. Hayes. *Regression Analysis and Linear Models: Concepts, Applications, and Implementation*. Methodology in the Social Sciences. Guilford Publications, 2016. isbn : 9781462521135. url : https://books.google.cd/books?id=YDgoDAAAQBAJ.

[9] Jean Tshimanga Ilunga. "Optimisation Numerique". In : *Jorge Nocedal and Steve Wright, (2000), Numerical Optimization, Springer Verlag*. T. 72. UNH. 2021, p. 21-23.

[10] Daniel Kirsch Judith Hurwitz. *Machine Learning For Dummies*. IBM Limited Edition. John Wiley et Sons, Inc., 2018.

[11] N. Matloff. *Statistical Regression and Classification: From Linear Models to Machine Learning*. Chapman & Hall/CRC Texts in Statistical Science. CRC Press, 2017. isbn : 9781351645898. url : https://books.google.cd/books?id=IHs2DwAAQBAJ.

[12] Carl-Erik Särndal, Bengt Swensson et Jan Wretman. *Model assisted survey sampling*. Springer Science & Business Media, 2003.

[13] Vahid Mirjalili Sebastien Raschka. *Python Machine Learning and Deep Learning, with sckit-learn and Tensorflow*. 2nd. Packt, 2017, p. 17-139.

[14] Rob GJ WIJNHOVEN et PHN de WITH. "Fast training of object detection using stochastic gradient descent". In : *2010 20th International Conference on Pattern Recognition*. IEEE. 2010, p. 424-427.