Apprentissage-Automatisé

Release 1.0.0-beta0

IVIA-AF Team

Jul 30, 2023

Contents

1 | Introduction Générale à l'Apprentissage Automatique

Nous parlerons de:

- Apprentissage Supervisé
- Apprentissage Non-Supervisé
- Les méthodes à noyaux (Kernel methods)
- Apprentissage par Renforcement

Motivations et les applications pour chaque type d'apprentissage.

1.1 C'est quoi Apprentissage Automatique?

Fig. [1.1].

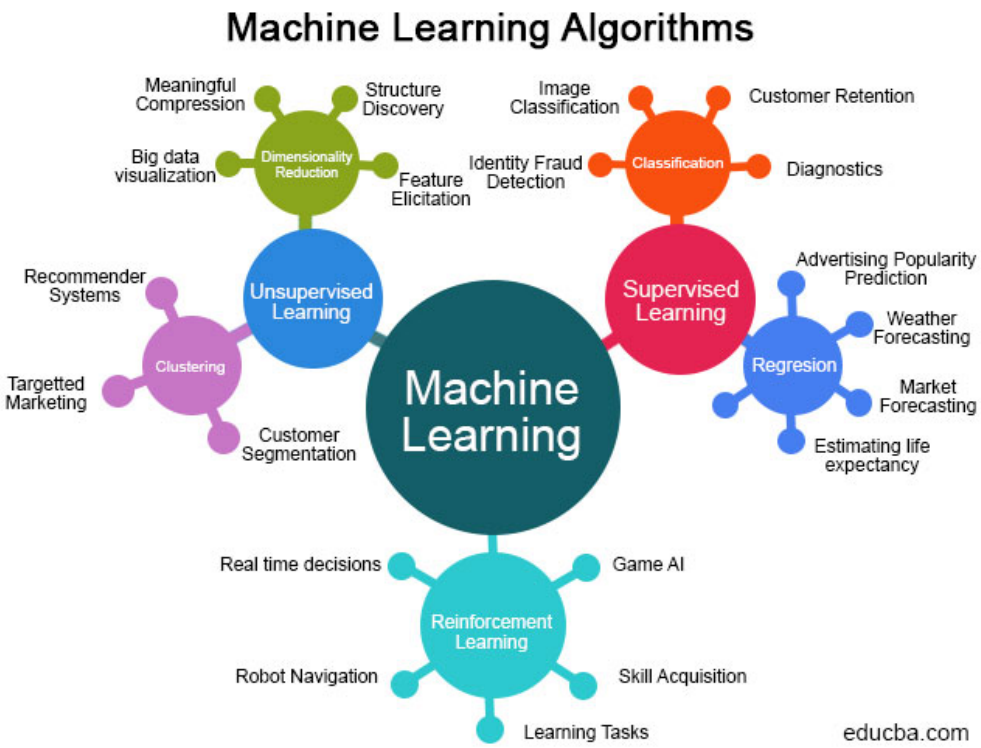

Fig. 1.1.1: Les types d'I.A

1.2 Convention Mathématiques pour le document

- Les matrices seront notées en lettre **majuscule** et seront mises en **gras**. Par exemple,

  $\mathbf{X}$

- Les vecteurs seront notés en lettre **miniscule** et mises en **gras**. Par exemple,

  $\mathbf{x}$.

- Lécriture mathématique de probabilités, espérance seront respectivement:

  $\mathbb{P}$, $\mathbb{E}$

- Il sera aussi important de ponctuer les équations.
- Numéroter les équations principales.
- Tous les ensemble seront notés en utilisant

  $\mathbb{R}$

  par example.

- les expressions mathématiques qui sont écrites à travers les textes seront écrites dans

  Prob

- Si c'est un symbole qui est un vecteur, on écrit (par exemple, si c'est alpha)

  $\boldsymbol{\alpha}$

  par exemple.

2 | Pré-requis

Python est le langage de programmation préféré des Data Scientistes. Ils ont besoin d'un langage facile à utiliser, avec une disponibilité décente des bibliothèques et une grande communauté. Les projets ayant des communautés inactives sont généralement moins susceptibles de mettre à jour leurs plates-formes. Mais alors, pourquoi Python est populaire en Data Science ?

Python est connu depuis longtemps comme un langage de programmation simple à maîtriser, du point de vue de la syntaxe. Python possède également une communauté active et un vaste choix de bibliothèques et de ressources. Comme résultat, vous disposez d'une plate-forme de programmation qui est logique d'utiliser avec les technologies émergentes telles que l'apprentissage automatique (Machine Learning) et la Data Science.

2.1 Langage Python et ses Librairies

Python est un langage de programmation puissant et facile à apprendre. Il dispose de structures de données de haut niveau et permet une approche simple mais efficace de la programmation orientée objet. Parce que sa syntaxe est élégante, que son typage est dynamique et qu'il est interprété, Python est un langage idéal pour l'écriture de scripts quand on fait de l'apprentissage automatique et le développement rapide d'applications dans de nombreux domaines et sur la plupart des plate-formes.

2.1.1 Installation de Python et Anaconda

L'installation de Python peut-être un vrai challenge. Déjà il faut se décider entre les versions 2.X et 3.X du langage, par la suite, choisir les librairies nécessaires (ainsi que les versions compatibles) pour faire de l'apprentissage automatique (Machine Learning); sans oublier les subtilités liées aux différents Systèmes d'exploitation (Windows, Linux, Mac…) qui peuvent rendre l'installation encore plus *"douloureuse"*.

Dans cette partie nous allons installer pas à pas un environnement de développement Python en utilisant Anaconda[1]. A l'issue de cette partie, nous aurons un environnement de développement fonctionnel avec les librairies (packages) nécessaires pour faire de l'apprentissage automatique (Machine Learning).

Qu'est ce que Anaconda ?

L'installation d'un environnement Python complet peut-être assez complexe. Déjà, il faut télécharger Python et l'installer, puis télécharger une à une les librairies (packages) dont on a besoin. Parfois, le nombre de ces librairies peut-être grand. Par ailleurs, il faut s'assurer de la compatibilité entre les versions des différents packages qu'on a à télécharger. *Bref, ce n'est pas amusant*!

Alors Anaconda est une distribution Python. À son installation, Anaconda installera Python ainsi qu'une multitude de packages dont vous pouvez consulter la liste[2]. Cela nous évite de nous ruer dans les problèmes

[1] http://docs.anaconda.com/anaconda/navigator/

[2] https://docs.anaconda.com/anaconda/packages/pkg-docs/#Python-3-7

d'incompatibilités entre les différents packages. Finalement, Anaconda propose un outil de gestion de packages appelé Conda. Ce dernier permettra de mettre à jour et installer facilement les librairies dont on aura besoin pour nos développements.

Téléchargement et Installation de Anaconda

Note: Les instructions qui suivent ont été testées sur Linux/Debian. Le même processus d'installation pourra s'appliquer pour les autres systèmes d'exploitation.

Pour installer Anaconda sur votre ordinateur, vous allez vous rendre sur le site officiel[3] depuis lequel l'on va télécharger directement la dernière version d'Anaconda. Prenez la version du binaire qu'il vous faut :

- Choisissez le système d'exploitation cible (Linux, Windows, Mac, etc…)
- Sélectionnez la version 3.X (à l'heure de l'écriture de ce document, c'est la version 3.8 qui est proposée, surtout pensez à toujours installer la version la plus récente de Python), compatible (64 bits ou 32 bits) avec l'architecture de votre ordinateur.

Après le téléchargement, si vous êtes sur Windows, alors rien de bien compliqué double cliquez sur le fichier exécutable et suivez les instructions classique d'installation d'un logiciel sur Windows.

Si par contre vous êtes sur Linux, alors suivez les instructions qui suivent:

- Ouvrez votre terminal et rassurez vous que votre chemin accès est celui dans lequel se trouve votre fichier d'installation.
- Exécutez la commande: $ bash Anaconda3-2020.02-Linux-x86_64.sh, rassurez vous du nom du fichier d'installation, il peut changer selon la version que vous choisissez.

Après que l'installation se soit déroulée normalement, éditez le fichier caché **.bashrc** pour ajouter le chemin d'accès à Anaconda. Pour cela exécutez les commandes suivantes:

- $ cd ~
- $ gedit .bashrc
- Ajoutez cette commande à la dernière ligne du fichier que vous venez d'ouvrir
- export PATH= ~/anaconda3/bin:$PATH

Maintenant que c'est fait, enregistrez le fichier et fermez-le. Puis exécutez les commandes suivantes

- $ conda init
- $ Python

Pour ce qui est de l'installation sur Mac, veuillez suivre la procédure d'installation dans la documentation d'Anaconda[4].

Il existe une distribution appelée Miniconda[5] qui est un programme d'installation minimal gratuit pour conda. Il s'agit d'une petite version bootstrap d'Anaconda qui inclut uniquement conda, Python, les packages dont ils dépendent, et un petit nombre d'autres packages utiles.

Terminons cette partie en nous familiarisant avec quelques notions de la programmation Python.

Première utilisation de Anaconda

La distribution Anaconda propose deux moyens d'accéder à ses fonctions: soit de manière graphique avec Anaconda-Navigator, soit en ligne de commande (depuis Anaconda Prompt sur Windows, ou un terminal pour

[3] http://docs.anaconda.com/anaconda/navigator/
[4] https://docs.anaconda.com/anaconda/install/mac-os/
[5] https://docs.conda.io/en/latest/miniconda.html

Linux ou MacOS). Sous Windows ou MacOs, démarrez Anaconda-Navigator dans le menu des programmes. Sous Linux, dans un terminal, tapez la commande : $ anaconda-navigator (cette commande est aussi disponible dans le prompt de Windows). Anaconda-Navigator propose différents services (déjà installés, ou à installer). Son onglet Home permet de lancer le service désiré. Les principaux services à utiliser pour développer des programmes Python sont :

- Spyder
- IDE Python
- Jupyter notebook et jupyter lab : permet de panacher des cellules de commandes Python (code) et des cellules de texte (Markdown).

Pour la prise en main de Python nous allons utiliser jupyter lab.

2.1.2 Prise en main de Python

Nous avons préparé un notebook qui nous permettra d'aller de zèro à demi Héros en Python. Le notebook se trouve ici[6].

2.2 Les Bases Mathématiques pour l'Apprentissage Automatique

Dans cette section, nous allons présenter les notions mathématiques essentielles à l'apprentissage automatique (machine learning). Nous n'aborderons pas les théories complexes des mathématiques afin de permettre aux débutants (en mathématiques) ou mêmes les personnes hors du domaine mais intéressées à l'apprentissage automatique de pouvoir en profiter.

2.2.1 Algèbre linéaire et Analyse

Définition d'espaces vectoriels. Un espace vectoriel est un triplet $(V, +, *)$ formé d'un ensemble V muni de deux lois,

$$\begin{aligned} + : \quad & V \times V \longrightarrow V \\ & (u, v) \mapsto u + v \\ \text{et} & \\ * : \quad & \mathbb{K} \times V \longrightarrow V, \text{ avec } \mathbb{K} \text{ un corps commutatif} \\ & (\lambda, v) \mapsto \lambda * v = \lambda v \end{aligned} \tag{2.2.1}$$

qui vérifient:

1. associativité de $+ : \forall\, u, v, w \in V, \quad (u + v) + w = u + (v + w)$
2. commutativité de $+ : \forall\, u, v \in V, \quad u + v = v + u$
3. existence d'élément neutre pour $+ : \exists\, e \in V : \forall\, u \in V, \quad u + e = e + u = u$
4. existence d'élément opposé pour $+ : \forall\, u \in V, \exists\, v \in V : u + v = v + u = 0$. On note $v = -u$ et v est appelé l'opposé de u
5. existence de l'unité pour $* : \exists\, e \in \mathbb{K}$ tel que $\forall\, u \in V, \quad e * u = u$

[6] https://colab.research.google.com/drive/1zILtNrCmPDFyQQ1Ev1H4jeHx7FuyEZ27?usp=sharing

6. associativité de $* : \forall\, (\lambda_1, \lambda_2, u) \in \mathbb{K} \times \mathbb{K} \times V, \quad (\lambda_1 \lambda_2) * u = \lambda_1 * (\lambda_2 * u)$
7. somme de vecteurs (distributivité de * sur +) : $\forall\, (\lambda, u, v) \in \mathbb{K} \times V \times V, \quad \lambda * (u + v) = \lambda * u + \lambda * v$
8. : $\forall\, (\lambda_1, \lambda_2, u) \in \mathbb{K} \times \mathbb{K} \times V, \quad (\lambda_1 + \lambda_2) * u = \lambda_1 * u + \lambda_2 * u.$

Remarque 1: Les éléments de V sont appelés des **vecteurs**, ceux de $\mathbb{K}$ sont appelés des **scalaires** et l'élément neutre pour $+$ est appelé **vecteur nul**. Finalement, V est appelé $\mathbb{K}$-espace vectoriel ou espace vectoriel sur $\mathbb{K}$.
Base d'un espace vectoriel. Soit V un $\mathbb{K}$-espace vectoriel. Une famille de vecteurs $\mathcal{B} = \{b_1, b_2, \ldots, b_n\}$ est appelée base de V si les deux propriétés suivantes sont satisfaites:

- $\forall\, u \in V, \exists\, c_1, \ldots, c_n \in \mathbb{K}$ tels que $u = \sum_{i=1}^{n} c_i b_i$ (On dit que $\mathcal{B}$ est **une famille génératrice** de V).
- $\forall\, \lambda_1, \ldots, \lambda_n \in \mathbb{K}, \quad \sum_{i=1}^{n} \lambda_i b_i = 0 \Longrightarrow \lambda_i = 0 \quad \forall\, i.$ (On dit que les éléments de $\mathcal{B}$ sont *linéairement indépendants*).

Lorsque $u = \sum_{i=1}^{n} c_i b_i$, on dit que $c_1, \ldots, c_n$ sont les coordonnées de u dans la base $\mathcal{B}$. Si de plus aucune confusion n'est à craindre, on peut écrire:

$$\mathbf{u} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix}. \tag{2.2.2}$$

Définition. Le nombre d'éléments dans une base d'un espace vectoriel est appelé **dimension** de l'espace vectoriel.

NB: Un espace vectoriel ne peut être vide (il contient toujours le vecteur nul). L'**espace vectoriel nul** $\{0\}$ n'a pas de base et est **de dimension nulle**. Tout **espace vectoriel non nul** de dimension finie admet une infinité de bases mais sa **dimension est unique**.
Exemples d'espaces vectoriels: Pour tous $n, m \geq 1$, l'ensemble des matrices $\mathcal{M}_{nm}$ à coefficients réels et l'ensemble $\mathbb{R}^n$ sont des $\mathbb{R}$-espace vectoriels. En effet, il est très facile de vérifier que nos exemples satisfont les huit propriétés énoncées plus haut. Dans le cas particulier $V = \mathbb{R}^n$, toute famille d'exactement n vecteurs linéairement indépendants en est une base. En revanche, toute famille de moins de n vecteurs ou qui contient plus que n vecteurs ne peut être une base de $\mathbb{R}^n$.

Matrices: Soit $\mathbb{K}$ un corps commutatif. Une matrice en mathématiques à valeurs dans $\mathbb{K}$ est un tableau de nombres, où chaque nombre est un élément de $\mathbb{K}$. Chaque ligne d'une telle matrice est un vecteur (élément

d'un $\mathbb{K}$-espace vectoriel). Une matrice est de la forme:

$$\mathbf{M} = \begin{bmatrix} a_{11} & a_{12} \dots a_{1n} \\ a_{21} & a_{22} \dots a_{2n} \\ \vdots & \\ a_{m1} & a_{m2} \dots a_{mn} \end{bmatrix}. \tag{2.2.3}$$

On note aussi $\mathbf{M} = \big(a_{ij}\big)_{1 \leq i \leq m, 1 \leq j \leq n}$.

La matrice ci-dessus est carrée si $m = n$. Dans ce cas, la suite $[a_{11}, a_{22}, \dots, a_{mm}]$ est appelée **diagonale** de M. Si tous les coefficients hors de la diagonale sont zéro, on dit que la matrice est diagonale. Une matrice avec tous ses coefficients nuls est dite matrice **nulle**.

Produit de matrices. Soient $\mathbf{A} = \big(a_{ij}\big)_{1 \leq i \leq m, 1 \leq j \leq n}, \mathbf{B} = \big(b_{ij}\big)_{1 \leq i \leq n, 1 \leq j \leq q}$ deux matrices.
On définit le produit de **A** par **B** et on note $\mathbf{A} \times \mathbf{B}$ ou simplement **AB**, la matrice M définie par:

$$\mathbf{M}_{ij} = \sum_{\ell=1}^{n} a_{i\ell} b_{\ell j}, \text{ pour tout } i \text{ et } j. \tag{2.2.4}$$

Important.

- Le produit **AB** est possible si et seulement si le nombre de colonnes de **A** est égal au nombre de lignes de **B**.
- Dans ce cas, **AB** a le même nombre de lignes que **A** et le même nombre de colonnes que **B**.
- Un autre point important à noter est que le produit matriciel n'est pas commutatif (**AB** n'est pas toujours égal à **BA**).

Exemple. Soient les matrices **A** et **B** définies par:

$$\mathbf{A} = \begin{bmatrix} 2 & -3 & 0 \\ 5 & 11 & 5 \\ 1 & 2 & 3 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1 & 3 \\ -5 & 1 \\ 1 & 2 \end{bmatrix}, \mathbf{A} + \mathbf{B} = \begin{bmatrix} 1 & 3 \\ -5 & 1 \\ 1 & 2 \end{bmatrix} \tag{2.2.5}$$

Le nombre de colonnes de la matrice **A** est égal au nombre de lignes de la matrice **B**.

$$\mathbf{AB} = \begin{bmatrix} 2 \times 1 + (-3) \times (-5) + 0 \times 1 & 2 \times 3 + (-3) \times 1 + 0 \times 2 \\ 5 \times 1 + 11 \times (-5) + 5 \times 1 & 5 \times 3 + 11 \times 1 + 5 \times 2 \\ 1 \times 1 + 2 \times (-5) + 3 \times 1 & 1 \times 3 + 2 \times 1 + 3 \times 2 \end{bmatrix} = \begin{bmatrix} 17 & 3 \\ -45 & 33 \\ -6 & 11 \end{bmatrix}. \tag{2.2.6}$$

Le produit **BA** n'est cependant pas possible.

Somme de matrices et multiplication d'une matrice par un scalaire.
La somme de matrices et multiplication d'une matrice par un scalaire se font coefficients par coefficients.
Avec les matrice $\mathbf{A}, \mathbf{B}$ de l'exemple précédent, et $\mathbf{C} = \begin{bmatrix} -2 & -7 & 3 \\ 5 & 10 & 5 \\ 12 & 9 & 3 \end{bmatrix}$, on a:

$$\mathbf{A}+\mathbf{C}=\begin{bmatrix}2+(-2) & -3+(-7) & 0+3\\ 5+5 & 11+10 & 5+5\\ 1+12 & 2+9 & 3+3\end{bmatrix}=\begin{bmatrix}0 & -10 & 3\\ 10 & 21 & 10\\ 13 & 11 & 6\end{bmatrix},\ \text{et pour tout } \lambda\in\mathbb{R},\quad \lambda\mathbf{B}=\begin{bmatrix}\lambda & 3\lambda\\ -5\lambda & \lambda\\ \lambda & 2\lambda\end{bmatrix}. \tag{2.2.7}$$

NB: La somme de matrice n'est définie que pour des matrices de même taille.

Déterminant d'une matrice.

Soit $\mathbf{A}=(a_{ij})_{1\leq i\leq n,1\leq j\leq n}$ une matrice carrée d'ordre n. Soit $\mathbf{A}_{i,j}$ la sous-matrice de $\mathbf{A}$ obtenue en supprimant la ligne i et la colonne j de $\mathbf{A}$. On appelle **déterminant** (au développement suivant la ligne i) de $\mathbf{A}$ et on note $\det(\mathbf{A})$, le nombre

$$\det(\mathbf{A})=\sum_{j=1}^{n}a_{ij}(-1)^{i+j}\det(\mathbf{A}_{i,j}), \tag{2.2.8}$$

avec le déterminant d'une matrice carrée de taille 2×2 donné par:

$$\det\left(\begin{bmatrix}a & b\\ c & d\end{bmatrix}\right)=ad-bc. \tag{2.2.9}$$

NB: Le développement suivant toutes les lignes donne le même résultat.

Le déterminant d'une matrice a une deuxième formulation dite de Leibniz[7] que nous n'introduisons pas dans ce document.

Inverse d'une matrice. Soit $\mathbf{A}$ une matrice carrée d'ordre n. $\mathbf{A}$ est **inversible** s'il existe une autre matrice notée $\mathbf{A}^{-1}$ telle que $\mathbf{A}\mathbf{A}^{-1}=\mathbf{A}^{-1}\mathbf{A}=\mathbf{I}_n$, où $\mathbf{I}_n$ est la matrice identité de taille $n\times n$.

Les matrices, leurs inverses et les opérations sur les matrices sont d'une importance capitale dans l'apprentissage automatique.

Vecteurs propres, valeurs propres d'une matrice.

Soient E un espace vectoriel et $\mathbf{A}$ une matrice. Un vecteur $\mathbf{v}\in E$ est dit **vecteur propre** de $\mathbf{A}$ si $\mathbf{v}\neq 0$ et il existe un scalaire λ tel que $\mathbf{A}\mathbf{v}=\lambda\mathbf{v}$. Dans ce cas, on dit que λ est la **valeur propre** associée au vecteur propre $\mathbf{v}$.

Applications linéaires et changement de base d'espaces vectoriels.

Soient $(E,\mathcal{B})$, $(F,\mathcal{G})$ deux $\mathbb{K}$-espace vectoriels, chacun muni d'une base et $f:\ E\to F$ une application.

On dit que f est **linéaire** si les propriétés suivantes sont satisfaites:

1. Pour tous $\mathbf{u},\mathbf{v}\in E$, $f(\mathbf{u}+\mathbf{v})=f(\mathbf{u})+f(\mathbf{v})$.
2. Pour tout $(\lambda,\mathbf{u})\in\mathbb{K}\times E$, $f(\lambda\mathbf{u})=\lambda f(\mathbf{u})$.

On suppose que $\mathcal{B}=\{e_1,e_2,\ldots,e_n\}$ et $\mathcal{G}=\{e'_1,e'_2,\ldots,e'_m\}$.

De manière équivalente, f est linéaire s'il existe une matrice $\mathbf{A}$ telle que pour tout $\mathbf{x}\in E,\quad f(x)=\mathbf{A}\mathbf{x}$.

[7] https://fr.wikipedia.org/wiki/Formule_de_Leibniz#D

Dans ce cas, la matrice **A** que l'on note $Mat_{\mathcal{B},\mathcal{G}}(f)$ est appelée matrice (représentative) de l'application linéaire f dans le couple de coordonnées $(\mathcal{B}, \mathcal{G})$.
La matrice **A** est unique et de taille $m \times n$ (notez la permutation *dimension de l'espace d'arrivée puis dimension de l'espace de départ dans la taille de la matrice*). De plus, la colonne j de la matrice **A** est constituée des coordonnées de $f(e_j)$ dans la base $\mathcal{G}$ de F. Lorsque $E = F$, l'application linéaire f est appelée **endomorphisme** de E et on écrit simplement $Mat_{\mathcal{B}}(f)$ au lieu de $Mat_{\mathcal{B},\mathcal{G}}(f)$.

Définition. Soient E un espace vectoriel de dimension finie et, $\mathcal{B}$ et $\mathcal{C}$, deux bases de E. On appelle **matrice de passage** de la base $\mathcal{B}$ à la base $\mathcal{C}$ la matrice de l'application identité

$$\begin{aligned} id_E : \quad & (E, \mathcal{C}) \to (E, \mathcal{B}) : \\ & x \mapsto x \quad . \end{aligned} \tag{2.2.10}$$

Cette matrice est notée $P_{\mathcal{B}}^{\mathcal{C}}$ et on a $P_{\mathcal{B}}^{\mathcal{C}} := Mat_{\mathcal{C},\mathcal{B}}(id_E)$.

Note: Si $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$ est un vecteur de E exprimé dans la base $\mathcal{B}$, alors l'expression de **x** dans la base $\mathcal{C}$ est donnée par $\begin{bmatrix} x'_1 \\ x'_2 \\ \vdots \\ x'_n \end{bmatrix} = (P_{\mathcal{B}}^{\mathcal{C}})^{-1}\mathbf{x} = P_{\mathcal{C}}^{\mathcal{B}}\mathbf{x}$.

Exemple. Si $E = \mathbb{R}^3$ avec ses deux bases

$$\mathcal{B} = \left(\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right) \text{ et } \mathcal{C} = \left(\begin{bmatrix} -1 \\ 2 \\ 3 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 5 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right), \tag{2.2.11}$$

on a $P_{\mathcal{B}}^{\mathcal{C}} = \begin{bmatrix} -1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & 5 & 1 \end{bmatrix}$ (c'est-à-dire qu'on exprime les vecteurs de $\mathcal{C}$ dans $\mathcal{B}$ pour former $P_{\mathcal{B}}^{\mathcal{C}}$).

Formule du changement de base pour une application linéaire.
Soient E une application linéaire et, $\mathcal{B}$ et $\mathcal{C}$, deux bases de E. Alors

$$Mat_{\mathcal{C}}(f) = P_{\mathcal{C}}^{\mathcal{B}} Mat_{\mathcal{B}}(f) P_{\mathcal{B}}^{\mathcal{C}}, \tag{2.2.12}$$

ou encore

$$Mat_{\mathcal{C}}(f) = (P_{\mathcal{B}}^{\mathcal{C}})^{-1} Mat_{\mathcal{B}}(f) P_{\mathcal{B}}^{\mathcal{C}}. \tag{2.2.13}$$

Diagonalisation et décomposition en valeurs singulières.

Diagonalisation. Soit **A** une matrice carrée à coefficients dans $\mathbb{K} = \mathbb{R}$ ou $\mathbb{C}$. On dit que **A** est **diagonalisable** s'il existe une matrice inversible **P** et une matrice diagonale **D** telles que $\mathbf{A} = \mathbf{PDP}^{-1}$. On dit aussi que **A** est similaire à **D**.

Important. Soient E un espace vectoriel de dimension finie et f un endomorphisme de E de matrice représentative (dans une base $\mathcal{B}$ de E) diagonalisable $\mathbf{A} = \mathbf{PDP}^{-1}$. On rappelle que les colonnes de **P** sont les vecteurs propres de **A**. Alors ces colonnes (dans leur ordre) constituent une base de E, et dans cette base, la matrice **A** est représentée par la matrice diagonale **D**. En d'autres termes, si $\mathcal{C}$ est la base des vecteurs propres de **A**, alors $Mat_{\mathcal{C}}(f) = \mathbf{D}$. Enfin, la matrice **D** est constituée des valeurs propres de **A** et le processus de calcul de **P** et **D** est appelé **diagonalisation**.

Décomposition en valeurs singulières.

Soit **M** une matrice de taille $m \times n$ et à coefficients dans $\mathbb{K} = \mathbb{R}$ ou $\mathbb{C}$. Alors **M** admet une factorisation de la forme $\mathbf{M} = \mathbf{U}$?$\mathbf{V}^*$, où $$

- **U** est une matrice unitaire (sur $\mathbb{K}$) de taille $m \times m$.
- $\mathbf{V}^*$ est l'adjoint (conjugué de la transposée) de **V**, matrice unitaire (sur $\mathbb{K}$) de taille $n \times n$
- ? est une matrice de taille $m \times n$ dont les coefficients diagonaux sont les valeurs singulières de **M**, i.e, les racines carrées des valeurs propres de $\mathbf{M}^*\mathbf{M}$ et tous les autres coefficients sont nuls.

Cette factorisation est appelée **la décomposition en valeurs singulières** de **M**. **Important.** Si la matrice **M** est de rang r, alors

- les r premières colonnes de **U** sont les vecteurs singuliers à gauche de **M**
- les r premières colonnes de **V** sont les vecteurs singuliers à droite de **M**
- les r premiers coefficients strictement positifs de la diagonale de ? sont les valeurs singulières de **M** et tous les autres coefficients sont nuls.

Produit scalaire et normes vectorielles. Soit V un espace vectoriel sur $\mathbb{R}$.

On appelle produit scalaire sur V toute application

$$\begin{aligned} \langle ., . \rangle : \quad & V \times V \to \mathbb{R} \\ & (\mathbf{u}, \mathbf{v}) \mapsto \langle \mathbf{u}, \mathbf{v} \rangle, \end{aligned} \tag{2.2.14}$$

telle que, $\forall (\lambda_1, \lambda_2, \mathbf{u}, \mathbf{v}, \mathbf{w}) \in \mathbb{R} \times \mathbb{R} \times V \times V \times V$,

- $\langle \mathbf{u}, \mathbf{v} \rangle = \langle \mathbf{v}, \mathbf{u} \rangle$(symétrie)
- 1. $\langle \lambda_1 \mathbf{u} + \lambda_2 \mathbf{v}, \mathbf{w} \rangle = \lambda_1 \langle \mathbf{u}, \mathbf{w} \rangle + \lambda_2 \langle \mathbf{v}, \mathbf{w} \rangle$ (linéarité à gauche)
  2. $\langle \mathbf{u}, \lambda_1 \mathbf{v} + \lambda_2 \mathbf{w} \rangle = \lambda_1 \langle \mathbf{u}, \mathbf{v} \rangle + \lambda_2 \langle \mathbf{u}, \mathbf{w} \rangle$ (linéarité à droite)
- $\langle \mathbf{u}, \mathbf{u} \rangle \geq 0$ (positive)
- $\langle \mathbf{u}, \mathbf{u} \rangle = 0 \implies \mathbf{u} = 0$ (définie)

$$\begin{aligned} \|.\| : \quad & V \to \mathbb{R}_+ \\ & \mathbf{v} \mapsto \|\mathbf{v}\| \end{aligned} \tag{2.2.15}$$

$V \; \forall \; (\lambda, \mathbf{u}, \mathbf{v}) \in \mathbb{R} \times V \times V$

- $\|\lambda \mathbf{u}\| = |\lambda| \times \|\mathbf{u}\|$
- $\|\mathbf{u} + \mathbf{v}\| \leq \|\mathbf{u}\| + \|\mathbf{u}\| + \|v\|$(inégalité triangulaire)

Remarque 2 Si $\langle ., . \rangle$ est un produit scalaire sur V, alors $\langle ., . \rangle$ induit une norme sur V. En effet,

$$\begin{aligned} \|.\|_{\langle \ldots \rangle} : \quad & V \to \mathbb{R}_+ \\ & \mathbf{u} \mapsto \|\mathbf{u}\| = \sqrt{\langle \mathbf{u}, \mathbf{u} \rangle} \end{aligned} \tag{2.2.16}$$

Exemples de normes et produits scalaires.
Prenons $V = \mathbb{R}^n$.
• Les applications

$$\begin{aligned} \rho : \quad & V \times V \to \mathbb{R} \\ & (\mathbf{u}, \mathbf{v}) \mapsto \sum_{i=1}^{n} u_i v_i, \end{aligned} \tag{2.2.17}$$

et

$$\begin{aligned} \mu : \quad & V \to \mathbb{R}_+ \\ & \mathbf{u} \mapsto \sqrt{\sum_{i=1}^{n} u_i^2}, \end{aligned} \tag{2.2.18}$$

sont respectivement un produit scalaire et une norme sur V. Il faut remarquer que $\forall \; \mathbf{u} \in V, \quad \mu(\mathbf{u}) = \sqrt{\rho(\mathbf{u}, \mathbf{u})}$.

• Pour tout $p \in \mathbb{N}^*$, l'application

$$\begin{aligned} \mu_p : \quad & V \to \mathbb{R}_+ \\ & \mathbf{u} \mapsto \left(\sum_{i=1}^{n} |u_i|^p \right)^{\frac{1}{p}}, \end{aligned} \tag{2.2.19}$$

est une norme sur V appelée norme p.

Dans le cas $p = 2$, on retrouve la norme μ ci-dessus appelée norme euclidienne.

Remarque 3. Un espace vectoriel muni d'une norme est appelé **espace vectoriel normé**.

Notion de distance.

Soit E un ensemble non vide. Toute application $d : E \times E \to \mathbb{R}_+$ qui satisfait pour tout $x, y, z \in E$:

- $d(x, y) = d(y, x)$ (symétrie)
- $d(x, y) = 0 \Longrightarrow x = y$ (séparation) (2.2.20)
- $d(x, y) \leq d(x, z) + d(z, y)$ (inégalité triangulaire)

Exemples de distances.

- $$\begin{aligned} d : \quad & \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}_+ \\ & (\mathbf{u}, \mathbf{v}) \mapsto \left(\sum_{i=1}^{n} |u_i - v_i| \right). \end{aligned}$$

- $$\begin{aligned} d : \quad & \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}_+ \\ & (\mathbf{u}, \mathbf{v}) \mapsto \left(\sum_{i=1}^{n} |u_i - v_i|^2 \right)^{\frac{1}{2}}. \end{aligned}$$

• C'est la généralisation de la distance euclidienne et de la distance de Manhattan

$$\begin{aligned} d_{Minkowski} : \quad & \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}_+ \\ & (\mathbf{u}, \mathbf{v}) \mapsto \left(\sum_{i=1}^{n} |u_i - v_i|^p \right)^{\frac{1}{p}}, p \geq 1. \end{aligned}$$

Espaces métriques.

Définition. Un **espace métrique** est un ensemble E muni d'une distance d; on écrit (E, d).

Remarque 4. Tout espace vectoriel normé est un espace métrique.

Suites dans un espace métrique.

Soit (E, d) un espace métrique. On appelle **suite** (d'éléments de E) et on note $(u_n)_{n \in I}$ ou $(u)_n$ une application:

$$\begin{aligned} u: \quad & I \to E \\ & n \mapsto u(n) := u_n \end{aligned} \tag{2.2.21}$$

où I est une partie infinie de $\mathbb{N}$. On dit que la suite $(u)_n$ converge vers $u^* \in E$ si pour tout $\epsilon > 0$ il existe $N \in \mathbb{N}$ tels que:

$$\forall\, n \in \mathbb{N}, \quad n > N \Longrightarrow d(u_n, u^*) < \epsilon \tag{2.2.22}$$

En d'autres termes, la suite $(u)_n$ converge vers $u^* \in E$ si pour tout $\epsilon > 0$, il existe un entier $N \in \mathbb{N}$ tel que pour tout $n > N$, u_n est contenu dans la boule $\mathcal{B}_\epsilon$ centrée en u^* et de rayon ϵ.

NB: La suite $(u)_n$ à valeurs dans E peut converger dans un ensemble autre que E.
Définition. La suite $(u)_n$ d'éléments de E est dite de Cauchy si pour tout $\epsilon > 0$, il existe $N \in \mathbb{N}$ tel que:

$$\forall\, n > m \in \mathbb{N}, \quad m > N \Longrightarrow d(u_n, u_m) < \epsilon. \tag{2.2.23}$$

Autrement dit, tous les termes u_n, u_m d'une suite de Cauchy se rapprochent de plus en plus lorsque n et m sont suffisamment grands.
Espaces métriques complets.
Définition. Un espace métrique (E, d) est dit **complet** si toute suite de Cauchy de E converge dans E.
Un espace métrique complet est appelé **espace de Banach**.

2.2.2 Calcul du gradient (dérivation).

Fonction réelle.
Définition.
Soit $f : J \to \mathbb{R}$ une fonction, avec J un intervalle ouvert de $\mathbb{R}$.
On dit que f est **dérivable** en $a \in J$ si la limite:

$$\lim_{h \to 0} \frac{f(a+h) - f(a)}{h} \text{ est finie.} \tag{2.2.24}$$

Si f est dérivable en a, la dérivée de f en a est notée $f'(a)$. La fonction dérivée de f est notée f' ou $\frac{\mathrm{d}f}{\mathrm{d}x}$ ou $\mathrm{d}f$.
Exemple de dérivées.

- **Fonctions polynomiales.**
  La dérivée de la fonction $f(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_1 x + a_0$, avec les a_i des constantes, est $f'(x) = n a_n x^{n-1} + (n-1)a_{n-1}x^{n-2} + \cdots + a_1$.
- **Fonction exponentielle de base :math:`e`.**
  La dérivée de la fonction $f(x) = \exp(x)$ est la fonction f elle-même, i.e, $\frac{\mathrm{d}\exp}{\mathrm{d}x}(x) = \exp(x)$.
- **Fonctions trigonométriques.**
  $\frac{\mathrm{d}\cos}{\mathrm{d}x}(x) = -\sin x$ et $\frac{\mathrm{d}\sin}{dx}(x) = \cos x$.
- **Fonction logarithme népérien**.
  $\frac{\mathrm{d}\ln}{\mathrm{d}x}(x) = \frac{1}{x}$.

Propriétés.
Soient $J \subseteq \mathbb{R}$ un intervalle ouvert, $u,\ v\ : J \to \mathbb{R}$ deux fonctions et $\lambda \in \mathbb{R}$. Alors on a les propriétés suivantes de la dérivée:

1. $(u+v)' = u' + v'$
2. $(uv)' = uv' + u'v$
3. $(\lambda u)' = \lambda u'$

Ces propriétés s'étendent aux fonctions vectorielles en dimension supérieure.

Fonctions vectorielles.
Soit $f : \mathcal{O} \to \mathbb{R}^p$ une fonction, avec $\mathcal{O}$ une partie ouverte de $\mathbb{R}^n, n, p \geq 1$.
On dit que f est **différentiable** (au sens de Fréchet) en $\mathbf{a} \in \mathcal{O}$, s'il existe une application linéaire continue $L : \mathbb{R}^n \to \mathbb{R}^p$ telle que pour tout $h \in \mathbb{R}^n$, on a

$$\lim_{h \to 0} \frac{f(\mathbf{a}+h) - f(\mathbf{a}) - L(h)}{\|h\|} = 0. \tag{2.2.25}$$

Si f est différentiable en tout point de $\mathcal{O}$, on dit que f est différentiable sur $\mathcal{O}$.

La différentielle de f est notée Df.

Dérivées partielles.
Soient $\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} \in \mathcal{O} \subseteq \mathbb{R}^n$ et $f : \mathcal{O} \to \mathbb{R}^p$ une fonction.
On dit que f admet une dérivée partielle par rapport à la $j - me$ variable x_j si la limite:

$$\lim_{h \to 0} \frac{f(a_1, a_2, \ldots, a_j + h, \ldots, a_n) - f(\mathbf{a})}{h} \text{ est finie.} \tag{2.2.26}$$

La dérivée partielle par rapport à la variable x_j de f en $\mathbf{a}$ est notée $\frac{\partial f}{\partial x_j}(\mathbf{a})$.

Note. Si f est différentiable, alors f admet des dérivées partielles par rapport à toutes les variables.

Gradient et Matrice Jacobienne. Soit $f : \mathcal{O} \subseteq \mathbb{R}^n \to \mathbb{R}^p$ une fonction différentiable.
On suppose que les fonctions composantes de f sont $f_1, f_2, \ldots, f_p$.
Alors la matrice des dérivées partielles

$$\begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & & \vdots \\ \frac{\partial f_p}{\partial x_1} & \frac{\partial f_p}{\partial x_2} & \cdots & \frac{\partial f_p}{\partial x_n} \end{bmatrix} \tag{2.2.27}$$

est appelée la **matrice jacobienne** de f, notée $\mathbf{J}_f$ ou $\mathbf{J}(f)$.

Dans le cas $p = 1$, le vecteur $\begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix}$ est appelé **gradient** de f et noté ∇f ou **grad**(f).

Exemples du calcul de dérivées et de gradients sur :math:`mathbb{R}^n`.

- $f(\mathbf{x}) = \langle \mathbf{x}, \mathbf{x} \rangle = \mathbf{x}^T \mathbf{x}$. Le gradient de f est $\nabla f(\mathbf{x}) = 2\mathbf{x}$
- $f(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b}$, avec $\mathbf{A}$ une matrice et $\mathbf{b}$ un vecteur. On a $Df(\mathbf{x}) = \mathbf{A}$.

Dérivées de fonctions composées.

Il existe souvent des fonctions dont le gradient ne peut facilement être calculé en utilisant les formules précédentes. Pour trouver le gradient d'une telle fonction, on va réécrire la fonction comme étant une composition de fonctions dont le gradient est facile à calculer en utilisant les techniques que nous allons introduire. Dans cette partie nous allons présenter trois formules de dérivation de fonctions composées.

Composition de fonctions à une seule variable.

Soit $f, g, h : \mathbb{R} \to \mathbb{R}$, trois fonctions réelles telles que $f(x) = g(h(x))$.

$$\frac{\mathrm{d}f}{\mathrm{d}x} = \frac{\mathrm{d}g}{\mathrm{d}h} \frac{\mathrm{d}h}{\mathrm{d}x} \tag{2.2.28}$$

Formule de dérivée totale.

Soit $f : \mathbb{R}^{n+1} \to \mathbb{R}$ telle que $f = f(x, u_1(x), \ldots, u_n(x))$ avec $u_i : \mathbb{R} \to \mathbb{R}$ alors

$$\frac{\mathrm{d}f\left(x, u_1, \ldots, u_n\right)}{\mathrm{d}x} = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial u_1} \frac{\mathrm{d}u_1}{\mathrm{d}x} + \frac{\partial f}{\partial u_2} \frac{\mathrm{d}u_2}{\mathrm{d}x} + \ldots + \frac{\partial f}{\partial u_n} \frac{\mathrm{d}u_n}{\mathrm{d}x} = \frac{\partial f}{\partial x} + \sum_{i=1}^{n} \frac{\partial f}{\partial u_i} \frac{\mathrm{d}u_i}{\mathrm{d}x}. \tag{2.2.29}$$

Formule générale de dérivées de fonctions composées.

Soit

$$\begin{array}{ll} f : \mathbb{R}^k \to \mathbb{R}^m & g : \mathbb{R}^n \to \mathbb{R}^k \\ \mathbf{x} \mapsto f(\mathbf{x}) & \mathbf{x} \mapsto g(\mathbf{x}) \end{array} \tag{2.2.30}$$

où $\mathbf{x} = (x_1, \ldots, x_n)$, $f(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_m(\mathbf{x}))$ et $g(\mathbf{x}) = (g_1(\mathbf{x}), \ldots, g_k(\mathbf{x}))$.

Le gradient de $f(g(\mathbf{x}))$ est défini comme suit:

$$\frac{\partial}{\partial \mathbf{x}} \mathbf{f}(\mathbf{g}(\mathbf{x})) = \begin{bmatrix} \frac{\partial f_1}{\partial g_1} & \frac{\partial f_1}{\partial g_2} & \cdots & \frac{\partial f_1}{\partial g_k} \\ \frac{\partial f_2}{\partial g_1} & \frac{\partial f_2}{\partial g_2} & \cdots & \frac{\partial f_2}{\partial g_k} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial g_1} & \frac{\partial f_m}{\partial g_2} & \cdots & \frac{\partial f_m}{\partial g_k} \end{bmatrix} \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \frac{\partial g_1}{\partial x_2} & \cdots & \frac{\partial g_1}{\partial x_n} \\ \frac{\partial g_2}{\partial x_1} & \frac{\partial g_2}{\partial x_2} & \cdots & \frac{\partial g_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial g_k}{\partial x_1} & \frac{\partial g_k}{\partial x_2} & \cdots & \frac{\partial g_k}{\partial x_n} \end{bmatrix} \tag{2.2.31}$$

2.2.3 Probabilités

La théorie des probabilités constitue un outil fondamental dans l'apprentissage automatique. Les probabilités vont nous servir à modéliser une expérience aléatoire, c'est-à-dire un phénomène dont on ne peut pas prédire l'issue avec certitude, et pour lequel on décide que le dénouement sera le fait du hasard.

Définition.

Une probabilité est une application sur $\mathcal{P}(\Omega)$, l'ensemble des parties de Ω telle que:

- $0 \leq \mathbb{P}(A) \leq 1$, pour tout événement $A \subseteq \Omega$;
- $\mathbb{P}(A) = \sum_{\{\omega\} \in A} \mathbb{P}(\omega)$, pour tout événement A;
- $\mathbb{P}(\Omega) = \sum_{A_i} \mathbb{P}(A_i) = 1$, avec les $A_i \subseteq \Omega$ une partition de Ω.

Proposition. Soient A et B deux événements,

1. Si A et B sont incompatibles, $\mathbb{P}(A \cup B) = \mathbb{P}(A) + \mathbb{P}(B)$.
2. $\mathbb{P}\left(A^c\right) = 1 - \mathbb{P}(A)$, avec A^c le complémentaire de A.
3. $\mathbb{P}(\emptyset) = 0$.
4. $\mathbb{P}(A \cup B) = \mathbb{P}(A) + \mathbb{P}(B) - \mathbb{P}(A \cap B)$.

Preuve voir [@epardoux]

Ci-dessous une définition plus générale de probabilité, valable pour des espaces des événements possibles non dénombrables.

Définition. Soit A une expérience alátoire et Ω l'espace des événements possibles associés. Une probabilité sur Ω est une application définie sur l'ensemble des événements, qui vérifie:

::: {.center}

- **Axiome 1:** $0 \leq \mathbb{P}(A) \leq 1$, pour tout événement A;
- **Axiome 2:** Pour toute suite d'événements $(A_i)_{i \in \mathbf{N}}$, deux à deux incompatibles,

$$\mathbb{P}\left(\bigcup_{i \in \mathbf{N}} A_i\right) = \sum_{i \in \mathbf{N}} \mathbb{P}\left(A_i\right); \tag{2.2.32}$$

- **Axiome 3:** $\mathbb{P}(\Omega) = 1$. ::

NB : Les événements $(A_i)_{i\in\mathbf{N}}$ sont deux à deux incompatibles, si pour tous $i \neq j$, $A_i \cap A_j = \emptyset$.

Indépendance et conditionnement.

Motivation.

Quelle est la probablité d'avoir un cancer du poumon?

Information supplémentaire: vous fumez une vingtaine de cigarettes par jour. Cette information va changer la probabilité. L'outil qui permet cette mise à jour est la probabilité conditionnelle.

Définition.

Étant donnés deux événements A et B, avec $\mathbb{P}(A) > 0$, on appelle probabilité de B conditionnellement à A, ou sachant A, la probabilité notée $\mathbb{P}(B \mid A)$ définie par:

$$\mathbb{P}(B \mid A) = \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(A)}. \tag{2.2.33}$$

L'équation *[prob_condit]* (page **??**) peut aussi s'écrire comme $\mathbb{P}(A \cap B) = \mathbb{P}(B \mid A)\mathbb{P}(A)$.

De plus, la probabilité conditionnelle sachant A, notée $\mathbb{P}(. \mid A)$ est une nouvelle probabilité et possède toutes les propriétés d'une probabilité.

Proposition. Formule des probabilités totales généralisée

Soit $(A_i)_{i\in I}$ (I un ensemble fini d'indices) une partition de Ω telle que $0 < \mathbb{P}(A_i) \leq 1 \quad \forall\, i \in I$. Pour tout événement B, on a

$$\mathbb{P}(B) = \sum_{i\in I} \mathbb{P}\left(B|A_i\right) \mathbb{P}\left(A_i\right). \tag{2.2.34}$$

La formule des probabilités totales permet de servir les étapes de l'expérience aléatoire dans l'ordre chronologique. **Proposition.** Formule de Bayes généralisée

Soit $(A_i)_{i\in I}$ une partition de Ω tel que $0 \leq \mathbb{P}(A_i) \leq 1, \forall\, i \in I$. Soit un événement B, tel que $\mathbb{P}(B) > 0$. Alors pour tout $i \in I$,

$$\mathbb{P}(A_i|B) = \frac{\mathbb{P}\left(B|A_i\right) \mathbb{P}\left(A_i\right)}{\sum_{i\in I} \mathbb{P}\left(B|A_i\right) \mathbb{P}\left(A_i\right)}. \tag{2.2.35}$$

Définition.

Deux événements A et B sont dits **indépendants** si

$$\mathbb{P}(A \cap B) = \mathbb{P}(A)\mathbb{P}(B). \tag{2.2.36}$$

S'ils sont de probabilité non nulle, alors

$$\mathbb{P}(B|A) = \mathbb{P}(B) \Leftrightarrow \mathbb{P}(A|B) = \mathbb{P}(A) \Leftrightarrow \mathbb{P}(A \cap B) = \mathbb{P}(A)\mathbb{P}(B). \tag{2.2.37}$$

Variables aléatoires.

Définition.
Une variable aléatoire (v.a) X est une fonction définie sur l'espace fondamental Ω, qui associe une valeur numérique à chaque résultat de l'expérience aléatoire étudiée. Ainsi, à chaque événement élémentaire ω, on associe un nombre $X(\omega)$.

Une variable qui ne prend qu'un nombre dénombrable de valeurs est dite **discrète** (par exemple le résultat d'une lancée d'une pièce de monnaie, …), sinon, elle est dite **continue** (par exemple le prix d'un produit sur le marché au fil du temps, distance de freinage d'une voiture roulant à 100 km/h).

Variable aléatoire discrète

Définition.
L'espérance mathématique ou moyenne d'une v.a discrète X est le réel

$$\mathbb{E}[X] = \sum_{k=0}^{\infty} k\mathbb{P}[X = k]. \tag{2.2.38}$$

Pour toute fonction g,

$$\mathbb{E}[g(X)] = \sum_{k=0}^{\infty} g(k)\mathbb{P}[X = k]. \tag{2.2.39}$$

Définition.
La variance d'une v.a discrète X est le réel positif

$$Var[X] = \mathbb{E}\left[(X - \mathbb{E}[X])^2\right] = \sum_{k=0}^{\infty} (k - \mathbb{E}[X])^2 \mathbb{P}[X = k] = \mathbb{E}[X^2] - \mathbb{E}[X]^2 \tag{2.2.40}$$

et l'écart-type de X est la racine carrée de sa variance. $$

Exemple: Loi de Bernoulli

La loi de Bernoulli est fondamentale pour la modélisation des problèmes de classification binaire en apprentissage automatique. On étudie que les expériences aléatoires qui n'ont que deux issues possibles (succès ou échec). Une expérience aléatoire de ce type est appelée une épreuve de Bernoulli. Elle se conclut par un succès si l'évènement auquel on s'intéresse est réalisé ou un échec sinon. On associe à cette épreuve une variable aléatoire Y qui prend la valeur 1 si l'évènement est réalisé et la valeur 0 sinon. Cette v.a. ne prend donc que deux valeurs (0 et 1) et sa loi est donnée par :

$$\mathbb{P}[Y = 1] = p, \quad \mathbb{P}[Y = 0] = q = 1 - p. \qquad \text{Avec } p \in [0, 1]. \tag{2.2.41}$$

On dit alors que Y suit une loi de Bernoulli de paramètre p, notée $\mathcal{B}(p)$. La v.a. Y a pour espérance p et pour variance $p(1-p)$. En effet,

$$\mathbb{E}[Y] = 0 \times (1 - p) + 1 \times p = p \tag{2.2.42}$$

et

$$\text{Var}(Y) = \mathbb{E}\left[Y^2\right] - \mathbb{E}[Y]^2 = \mathbb{E}[Y] - \mathbb{E}[Y]^2 = p(1 - p). \tag{2.2.43}$$

Schéma de Bernoulli :

- Chaque épreuve a deux issues : succès $[S]$ ou échec $[E]$.
- Pour chaque épreuve, la probabilité d'un succès est la même, notons $\mathbb{P}(S) = p$ et $\mathbb{P}(E) = q = 1 - p$.
- Les n épreuves sont **indépendantes** : la probabilité d'un succès ne varie pas, elle ne dépend pas des informations sur les résultats des autres épreuves.

Variable aléatoire continue

Contrairement aux v.a. discrètes, les v.a. continues sont utilisées pour mesurer des grandeurs "continues" (comme distance, masse, pression…). Une variable aléatoire continue est souvent définie par sa densité de probabilité ou simplement densité. Une densité f décrit la loi d'une v.a X en ce sens:

$$\forall\, a, b \in \mathbb{R}, \quad \mathbb{P}[a \leq X \leq b] = \int_a^b f(x)dx \tag{2.2.44}$$

et

$$\forall\, x \in \mathbb{R}, \quad F(x) = \mathbb{P}[X \leq x] = \int_{-\infty}^{x} f(t)dt \tag{2.2.45}$$

. On en déduit qu'une densité doit vérifier

$$\forall\, x \in \mathbb{R}, \quad f(x) \geq 0 \text{ et } \int_{\mathbb{R}} f(x)dx = 1 \tag{2.2.46}$$

.

Définition.
On appelle densité de probabilité toute fonction réelle positive, d'intégrale 1.

Définition.
L'espérance mathématiques de la v.a X est définie par

$$\mathbb{E}[X] = \int_{\mathbb{R}} x f(x) dx. \tag{2.2.47}$$

Exemple. La loi normale
C'est la loi de probabilité la plus importante. Son rôle est central dans de nombreux modèles probabilistes et en statistique. Elle possède des propriétés intéressantes qui la rendent agréable à utiliser. La densité d'une variable aléatoire suivant la loi normale de moyenne μ et d'écart-type σ ($\mathcal{N}(\mu, \sigma^2)$) est définie par

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right), \quad \forall\, x \in \mathbb{R}. \tag{2.2.48}$$

Quand $\mu = 0$ et $\sigma = 1$, on parle de loi normale centrée et réduite.

Loi des grands nombres

Considérons une suite $(X_n)_{n\geq 1}$ de v.a. indépendantes et de même loi. Supposons que ces v.a. ont une espérance, m et une variance, σ^2.

Théorème.

$$\mathbb{E}\left[\sum_{i=1}^{n} X_i\right] = nm \tag{2.2.49}$$

$$\text{Var}\left[\sum_{i=1}^{n} X_i\right] = n\sigma^2 \tag{2.2.50}$$

Définition.
La moyenne empirique des v.a. $X_1, \ldots, X_n$ est la v.a.

$$\bar{X_n} = \frac{X_1 + \cdots + X_n}{n}. \tag{2.2.51}$$

On sait d'ores et déjà que la moyenne empirique a pour espérance m et pour variance $\frac{\sigma^2}{n}$. Ainsi, plus n est grand, moins cette v.a. varie. A la limite, quand n tend vers l'infini, elle se concentre sur son espérance, m. C'est la loi des grands nombres.

Théorème. Convergence en Probabilité
Quand n est grand, $\bar{X_n}$ est proche de m avec une forte probabilité. Autrement dit,
$\forall\, \varepsilon \geq 0, \quad \lim\_n \to \infty \mathbb{P}(|\bar{X_n} - m| > \varepsilon) = 0.$

Théorème central limite Le Théorème central limite est très important en apprentissage automatique. Il est souvent utilisé pour la transformation des données surtout au traitement de données aberrantes.

Théorème.
Pour tous réels $a < b$, quand n tend vers $+\infty$,

$$\mathbb{P}\left(a \leq \frac{\bar{X}_n - m}{\sigma/\sqrt{n}} \leq b\right) \longrightarrow \int_a^b \frac{1}{\sqrt{2\pi}} e^{-x^2/2} \mathrm{d}x. \tag{2.2.52}$$

On dit que $\dfrac{\bar{X}_n - m}{\sigma/\sqrt{n}}$ converge en loi vers la loi normale $\mathcal{N}(0,1)$.

Intervalles de confiance

Soit X un caractère (ou variable) étudié sur une population, de moyenne m et de variance σ^2. On cherche ici à donner une estimation de la moyenne m de ce caractère, calculée à partir de valeurs observées sur un échantillon $(X_1, ..., X_n)$. La fonction de l'échantillon qui estimera un paramètre est appelée estimateur, son écart-type est appelé erreur standard et est noté SE. L'estimateur de la moyenne m est la moyenne empirique:

$$\frac{1}{n}\sum_{i=1}^{n} X_i \tag{2.2.53}$$

D'après les propriétés de la loi normale, avec un erreur $\alpha = 5\%$ quand n est grand on sait que

$$\mathbb{P}\left[m - 2\sigma/\sqrt{n} \leq \bar{X_n} \leq m + 2\sigma/\sqrt{n}\right] = 1 - \alpha = 0.954 \tag{2.2.54}$$

ou, de manière équivalente,

$$\mathbb{P}\left[\bar{X_n} - 2\sigma/\sqrt{n} \leq m \leq \bar{X_n} + 2\sigma/\sqrt{n}\right] = 1 - \alpha = 0.954 \tag{2.2.55}$$

Ce qui peut se traduire ainsi: quand on estime m par $\bar{X_n}$, l'erreur faite est inférieure à $2\sigma/\sqrt{n}$, pour $95,4\%$ des échantillons. Ou avec une probabilité de $95,4\%$, la moyenne inconnue m est dans l'intervalle $\left[\bar{X_n} - 2\sigma/\sqrt{n},\ \bar{X_n} + 2\sigma/\sqrt{n}\right]$.

Voir [@estatML] pour plus d'explication.

Définition.
On peut associer à chaque incertitude α, un intervalle appelé intervalle de confiance de niveau de confiance $1 - \alpha$, qui contient la vraie moyenne m avec une probabilité égale à $1 - \alpha$.

Définition.
Soit Z une v.a.. Le fractile supérieur d'ordre α de la loi de Z est le réel z qui vérifie

$$\mathbb{P}[Z \geq z] = \alpha \tag{2.2.56}$$

Le fractile inférieur d'ordre α de la loi Z est le réel z qui vérifie

$$\mathbb{P}[Z \leq z] = \alpha. \tag{2.2.57}$$

Quand l'écart-type théorique de la loi du caractère X étudié n'est pas connu, on l'éstime par l'écart-type empirique s_{n-1}. Comme on dispose d'un grand échantillon, l'erreur commise est petite. L'intervalle de confiance, de niveau de confiance $1-\alpha$ devient :

$$\left[\bar{x}_n - z_{\alpha/2}\frac{s_{n-1}}{\sqrt{n}},\ \bar{x}_n + z_{\alpha/2}\frac{s_{n-1}}{\sqrt{n}}\right] \tag{2.2.58}$$

où

$$s_{n-1}^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2. \tag{2.2.59}$$

2.2.4 Estimations paramétriques

Soit $(\Omega, \mathcal{A}, \mathbf{P})$ un espace probabilisé et **x** une $v.a.$ de $(\Omega, \mathcal{A})$ dans $(E, \mathcal{E})$. La donnée d'un modèle statistique c'est la donnée d'une famille de probabilités sur $(E, \mathcal{E})$, $\{\mathbb{P}_\theta, \theta \in \Theta\}$. Le modèle étant donné, on suppose alors que la loi de **x** appartient au modèle $\{\mathbf{P}_\theta, \theta \in \Theta\}$. Par exemple dans le modèle de Bernoulli, $\mathbf{x} = (x_1, \ldots, x_n)$ où les x_i sont $i.i.d.$ (indépendantes et identiquement distribuées) de loi de Bernoulli de paramètre $\theta \in]0,1]$. $E = \{0,1\}^n$, $\mathcal{E} = \mathcal{P}(E)$, $\Theta =]0,1]$ et $P_\theta = ((1-\theta)\delta_0 + \theta\delta_1)^{\otimes n}$.

Définition.
On dit que le modèle $\{\mathbb{P}_\theta, \theta \in \Theta\}$ est identifiable si l'application

$$\begin{array}{l}\Theta \to \{P_\theta, \theta \in \Theta\} \\ \theta \mapsto P_\theta\end{array} \tag{2.2.60}$$

est injective.

Définition.
Soit $g:\ \Theta \to \mathbb{R}^k$. On appelle estimateur de $g(\theta)$ au vu de l'observation x, toute application $T : \Omega \to \mathbb{R}^k$ de la forme $T = h(x)$ où $h : E \mapsto \mathbb{R}^k$ mesurable. Un estimateur ne doit pas dépendre de la quantité $g(\theta)$ que l'on cherche à estimer. On introduit les propriétés suivantes d'un estimateur.

Définition.
T est un estimateur sans biais de $g(\theta)$ si pour tout $\theta \in \Theta$, $\quad \mathbb{E}_\theta[T] = g(\theta)$.

Dans le cas contraire, on dit que l'estimateur T est biaisé et on appelle biais la quantité $\mathbb{E}_\theta[T] - g(\theta)$.

Généralement $\mathbf{x}$ est un vecteur $(x_1, \ldots, x_n)$ d'observations (n étant le nombre d'entre elles). Un exemple important est le cas où $x_1, \ldots, x_n$ forme un n-échantillon c'est à dire lorsque que $x_1, \ldots, x_n$ sont i.i.d. On peut alors regarder des propriétés asymptotiques de l'estimateur, c'est-à-dire en faisant tendre le nombre d'observations n vers $+\infty$. Dans ce cas, il est naturel de noter $T = T_n$ comme dépendant de n. On a alors la définition suivante :

Définition.

T_n est un estimateur consistant de $g(\theta)$ si pour tout $\theta \in \Theta$, T_n converge en probabilité vers $g(\theta)$ sous P_θ lorsque $n \to \infty$.

On définit le risque quadratique de l'estimateur dans le cas où $g(\theta) \in \mathbb{R}$.

Définition.

Soit T_n un estimateur de $g(\theta)$. Le risque quadratique de T_n est défini par

$$R(T_n, g(\theta)) = \mathbb{E}_\theta[(T_n - g(\theta))^2]. \tag{2.2.61}$$

Estimation par la méthode des moments

Considérons un échantillon $\mathbf{x} = (x_1, \ldots, x_n)$. Soit $f = (f_1, \ldots, f_k)$ une application de $\mathcal{X}$ dans $\mathbb{R}^k$ tel que le modèle $\{\mathbb{P}_\theta, \theta \in \Theta\}$ est identifiable si l'application Φ

$$\begin{aligned} \Phi : \ & \Theta \to \mathbb{R}^k \\ & \theta \mapsto \Phi(\theta) = \mathbb{E}_\theta[f(x)] \end{aligned} \tag{2.2.62}$$

est injective. On définit l'estimateur $\hat{\theta}_n$ comme la solution dans Θ (quand elle existe) de l'équation

$$\mathbb{E}_\theta[f(\mathbf{x})] \approx \frac{1}{n} \sum_{i=1}^{n} f(x_i). \tag{2.2.63}$$

Souvent, lorsque $\mathcal{X} \subset \mathbb{R}$, on prend $f_i(x) = x^i$ et Φ correspond donc au ième moment de la variable de X_i sous $\mathbb{P}_\theta$. Ce choix justifie le nom donné à la méthode. Voici quelques exemples d'estimateurs bâtis sur cette méthode.

Exemple. Loi uniforme

Ici $k = 1$, Q_θ est la loi uniforme sur $[0, \theta]$ avec $\theta > 0$. On a pour tout θ, $\mathbb{E}_\theta[X_1] = \dfrac{\theta}{2}$, on peut donc prendre par exemple $\Phi(\theta) = \dfrac{\theta}{2}$ et $f(x) = x$. L'estimateur obtenu par la méthode des moments est alors $\hat{\theta}_n = 2\bar{X}_n$. Cet estimateur est sans biais et constant.

Exemple. Loi normale

Ici $k = 2$, on prend $Q_\theta = \mathcal{N}(m, \sigma^2)$ avec $\theta = (m, \sigma^2) \in \mathbb{R} \times \mathbb{R}_+^*$. Pour tout θ, $\mathbb{E}_\theta[X_1] = m$ et $\mathbb{E}_\theta[X_1^2] = m^2 + \sigma^2$, on peut donc prendre par exemple, $f_1(x) = x$ et $f_2(x) = x^2$ ce qui donne $\Phi(m,\ \sigma^2) = (m, m^2 + \sigma^2)$. L'estimateur obtenu par la méthode des moments vérifie

$$\hat{m}_n = \bar{X}_n \text{ et } \hat{m}_n^2 + \hat{\sigma}_n^2 = \frac{1}{n} \sum_{i=1}^{n} X_i^2, \tag{2.2.64}$$

c'est-à-dire

$$\hat{\theta}_n = \left(\bar{X}_n,\ \frac{1}{n} \sum_{i=1}^{n} \left(X_i - \bar{X}_n \right)^2 \right). \tag{2.2.65}$$

L'estimateur est consistant mais l'estimateur de la variance est biaisé.

Estimation par maximum de vraisemblance

Soit $\{E,\ \mathcal{E},\ \{P_\theta,\ \theta \in \Theta\}\}$ un modèle statistique, où $\Theta \subset \mathbb{R}^k$. On suppose qu'il existe une mesure σ-finie μ qui domine le modèle, c'est à dire que $\forall\ \theta \in \Theta$, P_θ admet une densité par rapport à μ.

Définition.

Soit **x** une observation. On appelle vraisemblance de **x** l'application

$$\begin{array}{l} \Theta \to \mathbb{R}_+ \\ \theta \mapsto \mathbb{P}(\theta,\ \mathbf{x}) \end{array} \tag{2.2.66}$$

On appelle estimateur du maximum de vraisemblance de θ, tout élément $\hat{\theta}$ de Θ maximisant la vraisemblance, c'est à dire vérifiant

$$\hat{\theta} = \arg\max_{\theta \in \Theta} \mathbf{P}(\theta,\ \mathbf{x}) \tag{2.2.67}$$

Considérons le cas typique où $\mathbf{x} = (x_1, \ldots, x_n)$, les x_i formant un n-échantillon de loi Q_{θ_0} où Q_{θ_0} est une loi sur $\mathcal{X}$ de paramètre inconnu $\theta_0 \in \Theta \subset \mathbb{R}^k$. On suppose en outre que pour tout $\theta \in \Theta, Q_\theta$ est absolument continue par rapport à une mesure ν sur $\mathcal{X}$. Dans ce cas, en notant

$$q(\theta, x) = \frac{dQ_\theta}{d\nu}(x) \tag{2.2.68}$$

et en prenant $\mu = \nu^{\otimes n}$ on a la vraisemblance qui s'écrit sous la forme

$$\mathbb{P}(\theta, \mathbf{x}) = \prod_{i=1}^{n} q\left(\theta, x_i\right) \tag{2.2.69}$$

et donc

$$\hat{\theta}_n = \arg\max_{\theta \in \Theta} \frac{1}{n} \sum_{i=1}^{n} \log\left[q\left(\theta, x_i\right)\right] \tag{2.2.70}$$

avec la convention $\log(0) = -\infty$.

Exemple. Modèle de Bernoulli

Soit $Q_{\theta_0} = \mathcal{B}(\theta)$ avec $\theta \in [0,1] = \Theta$. Pour tout $\theta \in]0,1[$ et $x \in \{0,1\}$

$$q(\theta, x) = \theta^x (1-\theta)^{1-x} = (1-\theta) \exp\left[x \log\left(\frac{\theta}{1-\theta} \right) \right] \tag{2.2.71}$$

et donc l'estimateur du maximum de vraisemblance doit maximiser dans l'intervalle [0,1].

$$\log\left(\theta^{S_n} (1-\theta)^{n-S_n}\right) = S_n \log\left(\frac{\theta}{1-\theta} \right) + n \log(1-\theta) \tag{2.2.72}$$

avec $S_n = \sum_i x_i$ ce qui conduit à $\hat{\theta}_n = \bar{\mathbf{x}}$ en résolvant l'équation $\nabla \log(q(\theta, x)) = 0$.

3 | Bibliography

The breakthrough of deep learning origins from (Krizhevsky *et al.*, 2017) for computer vision, there is a rich of following up works, such as (He *et al.*, 2016). NLP is catching up as well, the recent work (Devlin *et al.*, 2018) shows significant improvements.

Two keys together (Devlin *et al.*, 2018, He *et al.*, 2016). Single author , two authors Newell and Rosenbloom (1980)

Bibliography

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6), 84–90.

Newell, A., & Rosenbloom, P. S. (1980). *Mechanisms of skill acquisition and the law of practice.* CARNEGIE-MELLON UNIV PITTSBURGH PA DEPT OF COMPUTER SCIENCE.