

## Matrices

Une matrice est un tableau de nombres disposés en  $m$  lignes et  $n$  colonnes. Soit  $A$  de taille  $(m, n)$  :

$$A = (a_{ij})_{\substack{i=1,\dots,m \\ j=1,\dots,n}} = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix}$$

Le terme  $a_{ij}$  est situé à la  $i^{\text{ème}}$  ligne de la  $j^{\text{ème}}$  colonne de la matrice  $A$ .

### Exemples

On considère :

$$A = \begin{pmatrix} -2 & 1 \\ 8 & -3 \\ 1 & 3 \end{pmatrix}; B = \begin{pmatrix} -2 \\ -3 \\ 3 \end{pmatrix}$$

$$C = \begin{pmatrix} 3 & -1 & 6 \end{pmatrix}; D = \begin{pmatrix} 3 & -1 \\ 4 & -3 \end{pmatrix}$$

$A$  est une matrice de taille  $(3, 2)$ .  
 $B$  est une matrice de taille  $(3, 1)$ .  
 $C$  est une matrice de taille  $(1, 3)$ .  
 $D$  est une matrice de taille  $(2, 2)$ .

Une matrice  $(m, 1)$  est dite matrice **colonne**.  
 Une matrice  $(1, n)$  est dite une matrice **ligne**.  
 Une matrice  $(n, n)$  est dite une matrice **carrée d'ordre  $n$** .

On appelle transposée de  $A$  (et on note  $A^t$  ou  $A'$ ), la matrice dont les lignes sont les colonnes de  $A$ , et dont les colonnes sont les lignes de  $A$ .

### Exemples

$$A = \begin{pmatrix} -2 & 1 \\ 8 & -3 \\ 1 & 3 \end{pmatrix} \Rightarrow A^t = \begin{pmatrix} -2 & 8 & 1 \\ 1 & -3 & 3 \end{pmatrix}$$

### Propriétés

$$(A + B)^t = A^t + B^t$$



$$(AB)^t = B^t A^t$$

## Matrices usuelles

Une matrice carrée d'ordre  $n$  est dite diagonale lorsque, pour tout  $i \neq j$ , on a  $a_{ij} = 0$

$$A = \begin{pmatrix} -2 & 0 & 0 \\ 0 & -3 & 0 \\ 0 & 0 & 5 \end{pmatrix}$$

La matrice unité  $I_n$  est la matrice diagonale telle que  $(\forall i = 1, \dots, n; a_{ii} = 1)$

$$I_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

## Opérations de Base

Soient  $A$  et  $B$  deux matrices de même taille  $(m, n)$  de termes généraux respectifs  $a_{ij}$  et  $b_{ij}$  et  $\lambda$  un nombre réel.

$A = B$  si et seulement si  $a_{ij} = b_{ij}$  pour tous  $i, j$

La somme de  $A$  et  $B$  est la matrice de terme général  $a_{ij} + b_{ij}$

Le produit de  $A$  par  $\lambda$  est la matrice de terme général  $\lambda a_{ij}$ .

### Exemples

$$A = \begin{pmatrix} 5 & 2 \\ 6 & 1 \end{pmatrix}; B = \begin{pmatrix} -4 & 1 \\ 1 & -3 \end{pmatrix}; \lambda = 3$$

$$A + B = \begin{pmatrix} 1 & 3 \\ 7 & -2 \end{pmatrix} \text{ et } \lambda A = \begin{pmatrix} 15 & 6 \\ 18 & 3 \end{pmatrix}$$

## Multiplication matricielle

Le produit de matrices  $A$  et  $B$  (noté  $AB$ ) n'est défini que si **le nombre de colonnes de  $A$  est égal au nombre de lignes de  $B$** .

$$A = \begin{pmatrix} 2 & -2 & 1 \\ -3 & 6 & 8 \\ 5 & 2 & 1 \end{pmatrix} \text{ et } B = \begin{pmatrix} -1 & 5 \\ 1 & 2 \\ 3 & 4 \end{pmatrix} \text{ alors}$$

C'est-à-dire  $A$  doit être de taille  $(m, p)$  et  $B$  de taille  $(p, n)$ . Alors  $AB$  est de taille  $(m, n)$ .

$$C = AB = \begin{pmatrix} -1 & 10 \\ 33 & 29 \\ 0 & 33 \end{pmatrix}$$

De plus, soient  $a_{ij}$  et  $b_{ij}$  les termes généraux respectifs de  $A$  et  $B$ , alors le terme général de  $C = AB$  est  $c_{ij}$  défini par :

En effet, l'élément qui se trouve au croisement de la  $i^{\text{ème}}$  ligne et la  $j^{\text{ème}}$  colonne est :

$$c_{ij} = \sum_{k=1}^p a_{ik} b_{kj}; \text{ par exemple}$$

$$c_{2,1} = (-3) \times (-1) + 6 \times 1 + 8 \times 3$$



**Le produit matriciel n'est pas commutatif**

## Propriétés

Le produit est distributif par rapport à l'addition. C'est-à-dire :

$$\begin{cases} A(B + C) = AB + AC \\ (B + C)A = BA + CA \end{cases}$$

Le produit est associatif, c'est-à-dire :  $ABC = A(BC) = (AB)C$ .

Si  $A$  est une matrice carrée d'ordre  $n$  alors  $AI_n = I_n A = A$ .

Le produit de deux matrices peut être nul sans qu'aucune des deux matrices ne soit la matrice nulle, exemple :  $\begin{pmatrix} 1 & 2 \\ 2 & 4 \end{pmatrix} \times \begin{pmatrix} -2 & 10 \\ 1 & -5 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$ .

## Systèmes linéaires

Grâce au produit matriciel, on peut représenter un système linéaire par une équation matricielle.

Soit le système linéaire de  $n$  équations et  $p$  inconnues

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1p}x_p = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2p}x_p = b_2 \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{np}x_p = b_n \end{cases}$$

On peut le représenter par  $AX = B$ .

$$A = \begin{pmatrix} a_{11} & \dots & a_{1p} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{np} \end{pmatrix}; X = \begin{pmatrix} x_1 \\ \vdots \\ x_p \end{pmatrix}; B = \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix}$$

## Inverse

Une matrice carrée  $A$  d'ordre  $n$  est dite inversible lorsqu'il existe  $B$  telle que :

$$AB = BA = I_n.$$

$B$  est alors notée  $A^{-1}$  : inverse de  $A$

## Déterminant

Soit  $A = (a_{ij})$  une matrice carrée d'ordre 2. Le déterminant de A est le réel noté **det(A)** tel que :

$$\det(A) = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11}a_{22} - a_{12}a_{21}$$

Soient  $A = (a_{ij})$  et  $B = (b_{ij})$  **deux matrices carrées** d'ordre n et  $\lambda \in \mathbf{R}$ . On a :

- $\det(AB) = \det(A) \det(B) = \det(BA)$
- $\det(A^t) = \det(A)$
- $\det(\lambda A) = \lambda^n \det(A)$
- Si A est diagonale, alors  $\det(A) = a_{11}a_{22} \dots a_{nn}$ .
- $\det(I_n) = 1$
- A est inversible si et seulement si  $\det(A) \neq 0$
- Si  $\det(A) \neq 0$ , alors  $\det(A^{-1}) = \frac{1}{\det(A)}$

## Diagonalisation

### VALEUR PROPRE / VECTEUR PROPRE

Une **valeur propre** de A est un **scalaire**  $\lambda$  tel qu'il existe un **vecteur colonne** non nul V vérifiant :  $AV = \lambda V$

V est alors appelé **vecteur propre** de A associé à  $\lambda$ .

### DIAGONALISATION

Une matrice carrée A d'ordre n est **diagonalisable** lorsqu'il existe une **matrice diagonale** D et une matrice **inversible** P telles que :  $A = PDP^{-1}$

D est constituée des **valeurs propres** de A.

P est obtenue par la concaténation des **vecteurs propres** de A.

Si les valeurs propres de A sont **distinctes**, alors A est **diagonalisable (réciproque fausse)**

## Matrice symétrique

Une matrice carrée A d'ordre n est symétrique lorsque  $A^t = A$ .

Si A est symétrique, alors :

- A a des valeurs propres réelles.
- A est diagonalisable.
- Il existe P telle que  $P^{-1} = P^t$  et  $A = PDP^{-1}$ .

## Produit scalaire

Soit x, y, z trois vecteurs de  $\mathbf{R}^n$  et  $\lambda$  un scalaire. On définit par produit scalaire (qu'on note  $\langle \cdot, \cdot \rangle$ ) toute application qui vérifie les propriétés suivantes :

- $\langle x + \lambda y, z \rangle = \langle x, z \rangle + \lambda \langle y, z \rangle$
- $\langle x, y + \lambda z \rangle = \langle x, y \rangle + \lambda \langle x, z \rangle$
- $\langle x, y \rangle = \langle y, x \rangle$
- $\langle x, x \rangle \geq 0$
- $\langle x, x \rangle = 0 \Rightarrow x = 0$

Le produit scalaire canonique et usuel est :

$$\langle x, y \rangle = \sum_{i=1}^n x_i * y_i$$

## Norme

On appelle **norme associée à un produit scalaire**, le réel  $\|x\| = \sqrt{\langle x, x \rangle}$ .

Elle vérifie les propriétés suivantes :

- $|\langle x, y \rangle| \leq \|x\| \|y\|$  (**Inégalité de Cauchy-Schwartz**)
- $\|x+y\| \leq \|x\| + \|y\|$  (**Inégalité triangulaire**)
- $\|x\| \geq 0$  avec égalité si  $x = 0$
- $\|\lambda x\| = |\lambda| \|x\|$
- $\|x+y\|^2 = \|x\|^2 + \|y\|^2 + 2\langle x, y \rangle$

Un vecteur est dit **unitaire** ou **normé** si  $\|x\| = 1$ .

## Orthogonalité

Deux vecteurs x et y sont dits orthogonaux si  $\langle x, y \rangle = 0$ . On note  $x \perp y$ .

Une famille de vecteurs  $x_i$  est dite orthogonale si tous ses vecteurs sont deux à deux orthogonaux. Toute famille orthogonale  $(x_i)_{i=1, \dots, p}$  vérifie le théorème de Pythagore :

$$\left\| \sum_{i=1}^n x_i \right\|^2 = \sum_{i=1}^n \|x_i\|^2$$

Soit E un espace muni d'un produit scalaire et X une partie de E. On appelle orthogonal de X et on note  $X^\perp$  l'ensemble :  $X^\perp = \{y \in E \mid (\forall x \in X), \langle x, y \rangle = 0\}$ .

On dit que  $(e_i)_{i=1, \dots, p}$  est une base orthonormée de E si et seulement si :

- si  $a_1 e_1 + a_2 e_2 + \dots + a_n e_n = 0$ , alors  $\forall i \in \{1, \dots, n\}, a_i = 0$
- $(\forall x \in E) (\exists a_1, a_2, \dots, a_n \in \mathbf{R}), x = a_1 e_1 + a_2 e_2 + \dots + a_n e_n$
- $e_i$  est orthogonal à  $e_j$  pour tout  $i \neq j$ .
- $\forall i \in \{1, \dots, n\}, \|e_i\| = 1$

## Projection orthogonale

Soit x un vecteur d'un espace muni d'un produit scalaire E.

Soit F un sous-espace vectoriel de E, x s'écrit de façon unique sous la forme :

$$x = f + f^\perp \text{ où } f \in F \text{ et } f^\perp \in F^\perp.$$

On dit que f est le projeté orthogonal de x sur F et on note  $f = P_F(x)$ .

Pour x, y de E, on a  $\langle P_F(x), y \rangle = \langle x, P_F(y) \rangle$ .

Si  $(e_1, e_2, \dots, e_n)$  est une base orthonormée, alors :

$$P_F(x) = \sum_{i=1}^p \langle x, e_i \rangle e_i$$

Notons que :  $\|x - P_F(x)\| = \inf_{f \in F} \|x - f\|$

## Matrice orthogonale

Soit M une matrice carrée d'ordre n.

On dit que M est une **matrice orthogonale** si elle vérifie :

$$M \times M^t = M^t \times M = I_n.$$

Le déterminant d'une matrice orthogonale est égal à  $\pm 1$ .

L'ensemble des valeurs propres de M est inclus dans l'ensemble  $\{0, 1\}$ .

# Fondements de probabilité : niveau élémentaire

## Quelques définitions

- On appelle épreuve E toute expérience probabiliste.
  - On appelle univers de E l'ensemble, généralement noté  $\Omega$ , de tous les résultats possibles de l'épreuve E (appelés "événements élémentaires")
- Lancer une paire de dés équilibrés et en retenir la somme est une épreuve.

$$\Omega = \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\}$$



## Événements

Un événement est un sous-ensemble de  $\Omega$ .

- L'**intersection** de A et B, notée  $A \cap B$ , est un événement. Il est réalisé uniquement si A et B se produisent.

- La **réunion** de A et B, notée  $A \cup B$ , est un événement. Il est réalisé si A ou B se produit. Deux événements remarquables sont à retenir:

— L'événement certain  $\Omega$  ;

— L'événement impossible  $\emptyset$  ;

- Tous les éléments qui n'appartiennent pas à A appartiennent à un événement que l'on appelle le **complémentaire de A**. On le note  $\bar{A}$  ou  $A^c$ .

- On dit que deux événements A et B sont **incompatibles** s'ils ne peuvent pas être réalisés en même temps.

Si A, B et C sont des événements de  $\Omega$ , les propriétés suivantes sont toujours vérifiées:

$$- A \cup \bar{A} = \Omega \text{ et } A \cap \bar{A} = \emptyset$$

$$- \overline{A \cap B} = \bar{A} \cup \bar{B} \text{ et } \overline{A \cup B} = \bar{A} \cap \bar{B} \quad (\text{lois de De Morgan})$$

$$- A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$$

$$- A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$$

## Partitions

La famille d'événements forme une partition de  $\Omega$  si :

$$\bigcup_{i \in I} A_i = \Omega \text{ et } A_i \cap A_j = \emptyset ; \forall i \neq j$$

Une partition remarquable est la famille qui contient l'événement A et son complémentaire.

## Tribus et boréliens

### COMMENT POUVONS-NOUS QUALIFIER L'ENSEMBLE DES ÉVÉNEMENTS?

Une tribu est une famille  $\mathcal{T}$  de parties de l'ensemble  $\Omega$  qui vérifie les propriétés suivantes :

—  $\Omega \in \mathcal{T}$

— Si  $(A_n)_n$  est une suite dénombrable d'éléments de  $\mathcal{T}$ , alors  $\bigcup A_n \in \mathcal{T}$

Si A est un élément de  $\mathcal{T}$ , alors son complémentaire l'est aussi

De plus, si  $\mathcal{T}$  est une tribu, alors:

—  $\emptyset \in \mathcal{T}$ .

— Si  $(A_n)_n$  est une suite d'éléments de  $\mathcal{T}$ , alors  $\bigcap A_n \in \mathcal{T}$ .

### EXEMPLE DE TRIBUS

Commençons par le cas discret.

On considère l'expérience "Lancer une pièce de monnaie équilibrée".

On notera : P "PILE apparaît" et F "FACE apparaît".

Dans ce cas, l'univers est l'ensemble  $\{P, F\}$  et

$\mathcal{T} = \{\Omega, \emptyset, P, F\}$  est une tribu.

En général, l'ensemble des parties est une tribu (classique).

Pour le cas continu, les intervalles du type

$[a, +\infty[ ; ]a, +\infty[ ; ]-\infty, a[ ; ]-\infty, a]$  sont des tribus.

Nous les appelons **DES BORÉLIENS**.

## Mesure

Soit E un ensemble muni d'une tribu  $\mathcal{T}$ . On appelle mesure toute application  $m : \mathcal{T} \rightarrow \mathbb{R}^+$  telle que :

$$- m(\emptyset) = 0$$

- Si  $(A_n)_n$  est une suite d'éléments de  $\mathcal{T}$  deux à deux disjoints alors :

$$m(\bigcup A_n) = \sum m(A_n).$$

## Probabilité

Soit E un ensemble muni d'une tribu  $\mathcal{T}$ . On appelle probabilité toute application

$m : \mathcal{T} \rightarrow \mathbb{R}^+$  telle que :

$$- P(\emptyset) = 0$$

- Si  $(A_n)_n$  est une suite d'éléments de  $\mathcal{T}$  deux à deux disjoints alors :

$$P(\bigcup A_n) = \sum P(A_n)$$

## Tribus et boréliens

Soient A et B deux événements. Les propriétés suivantes sont toujours vraies:

$$1- P(\bar{A}) = 1 - P(A)$$

$$2- P(B) = P(A \cap B) + P(\bar{A} \cap B)$$

$$3- \text{Si } A \subset B \text{ alors } P(A) \leq P(B)$$

$$4- 0 \leq P(A) \leq 1$$

$$5- P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

De plus, considérant une suite  $(A_n)_n$  d'événements. On a alors :

$$P\left(\bigcup_{k=1}^{+\infty} A_k\right) = \lim_{n \rightarrow +\infty} P\left(\bigcup_{k=1}^n A_k\right)$$

$$P\left(\bigcap_{k=1}^{+\infty} A_k\right) = \lim_{n \rightarrow +\infty} P\left(\bigcap_{k=1}^n A_k\right)$$

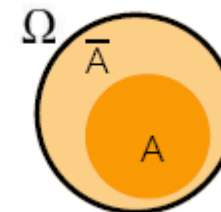
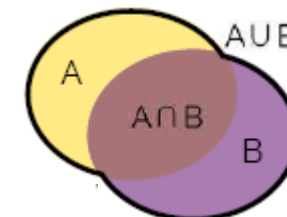
$$P\left(\bigcup_{k=1}^{+\infty} A_k\right) \leq \sum_{k=1}^{+\infty} P(A_k)$$

Et si :

$$\bigcup_{k=1}^n A_k = \Omega$$

Alors :

$$P(B) = \sum_{k=1}^n P(B \cap A_k)$$



# SEP guide Fondements de probabilités : niveau élémentaire

## Proba Conditionnelles

En théorie des probabilités, nous nous intéressons souvent au comportement d'un aléa, sachant qu'un autre événement est déjà passé.

C'est ce que nous appelons **Les Probabilités Conditionnelles**.

Considérant deux événements de proba non nulles A et B, la probabilité conditionnelle de A sachant que B est réalisé (couramment dit A sachant B) est :

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

Par commutativité de l'intersection nous avons :  $P(A \cap B) = P(B \cap A)$

Et donc en utilisant la formule ci-dessus :

$$P(B|A)P(A) = P(A|B)P(B)$$

D'où alors :

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

C'est ce que nous appelons :

**LA FORMULE DE BAYES**

## Indépendance

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

$$P(A \cap B) = P(A)P(B)$$

En termes courants, deux événements sont indépendants si le résultat de l'un n'influence aucunement l'aboutissement de l'autre.

Sous condition d'indépendance de A et B, la notion de la probabilité conditionnelle tombe à l'eau, car les événements évoluent l'un sans se soucier de l'autre.

Ceci se traduit par :

$$P(A|B) = P(A)$$

$$P(B|A) = P(B)$$

Notons que si A est indépendant de B, il le sera par rapport à son complémentaire également. Et vice versa.

En général, pour une suite  $(A_n)_n$  d'événements indépendants, on a :

$$P\left(\bigcap_{i=1}^n A_i\right) = \prod_{i=1}^n P(A_i) = P(A_1) \dots P(A_n)$$

Cette formule est largement utilisée en statistique.



**NE PAS CONFONDRE INDÉPENDANCE ET INCOMPATIBILITÉ DES ÉVÉNEMENTS.**

## Variable aléatoire

Une variable aléatoire est un nombre qui dépend du résultat d'une expérience aléatoire. Chaque exécution de l'expérience génère une réalisation de la variable aléatoire.

Mathématiquement, on définit une variable aléatoire X comme une fonction  $X : \mathcal{T} \rightarrow \mathbb{R}$  qui associe à chaque événement s, un réel X(s)

Par exemple, dans une queue pour la caisse d'un magasin, le nombre de clients est une variable aléatoire. La durée de traitement de chaque requête aussi.

Remarquons que la première est un nombre entier. On dit qu'elle est à support discret. Alors que la deuxième est une durée (un nombre réel). On dit qu'elle est à support continu

## Qu'est-ce qui caractérise une variable aléatoire ?

### Fonction de répartition

Une VA traduit le résultat d'une expérience aléatoire en nombre réel. La fonction de répartition transporte le calcul des probabilités concernant les réalisations de la VA.

C'est la fonction définie par :

$$F_X(x) = P(X \leq x)$$

Propriétés :

$$\forall x ; 0 \leq F_X(x) \leq 1$$

$F_X$  est une fonction croissante

$$\lim_{x \rightarrow -\infty} F_X(x) = 0 \text{ et } \lim_{x \rightarrow \infty} F_X(x) = 1$$



Cas discret



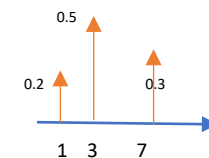
Cas continu

### Probabilité ponctuelle / Densité

#### CAS DISCRET : Probabilité ponctuelle

La probabilité ponctuelle est la fonction qui décrit les sauts de la fonction de répartition :

$$P(X = k) = P(X \leq k) - P(X \leq k-1) = p_k$$



$$\sum p_i = 1$$

#### CAS CONTINU : densité de probabilité

La densité est la fonction qui décrit les variations de la fonction de répartition :

$$f_X(x) = \frac{dF_X}{dx}(x)$$

$$\int f_X = 1$$

## Moments

### ESPÉRANCE

L'espérance d'une variable aléatoire est sa valeur attendue. C'est une mesure de localisation de la distribution.

Dans le cas discret :

$$E(X) = \sum_{k \in X(\Omega)} k \cdot P(X = k)$$

Alors que dans le cas continu :

$$E(X) = \int_{x \in X(\Omega)} x \cdot f_x(x) \cdot dx$$

THEOREME DE TRANSFERT

$$E(\varphi(X)) = \sum_{k \in X(\Omega)} \varphi(k) \cdot P(X = k)$$

$$E(\varphi(X)) = \int_{x \in X(\Omega)} \varphi(x) \cdot f_x(x) \cdot dx$$

### VARIANCE

La variance d'une variable aléatoire décrit la dispersion de la variable aléatoire autour de sa valeur moyenne (son espérance). Elle est définie par :

$$V(X) = E(X^2) - (E(X))^2 = E((X - E(X))^2)$$

Sa racine carrée est appelée écart-type et notée généralement :

$$\sigma(X) = \sqrt{V(X)}$$

### CENTRAGE ET REDUCTION

Le centrage consiste à localiser la distribution autour de l'origine et la réduction consiste à normaliser la dispersion. La technique est simple :

$$Y = \frac{X - E(X)}{\sigma(X)}$$

### MOMENTS D'ORDRE r :

Le moment d'ordre r est défini par :

$$\mu_r = E(X^r)$$

Le moment centré d'ordre r est défini ainsi :

$$\tilde{\mu}_r = E((X - E(X))^r)$$

## Couples aléatoires

La fonction  $F_{x,y}(x, y) = P(X \leq x \cap Y \leq y)$  est dite distribution conjointe de X et Y. Dans le cas continu, la fonction définie par :

$$f_{x,y}(x, y) = \frac{\partial^2}{\partial x \partial y} F_{x,y}(x, y)$$

Est une densité conjointe du couple (X, Y).

On a donc :

$$F_{x,y}(x, y) = \int_{-\infty}^x \int_{-\infty}^y f_{x,y}(t, u) dt du$$

Dans le **cas discret**, on définit la fonction de fréquences conjointes :

$$P(X = x_i, Y = y_j) = p_{ij}$$

Et on a donc :

$$F_{x,y} = \sum_{i: x_i \leq x} \sum_{j: y_j \leq y} p_{ij}$$

### LOI MARGINALE

On définit la loi marginale de X :

$$f_x(x) = \int_{-\infty}^{+\infty} f_{x,y}(x, y) dy$$

dans le cas continu, ou encore :

$$f_x(x_i) = \sum_j p_{ij}$$

dans le cas discret.

(De même on peut définir la densité marginale de Y)

Si X et Y sont indépendants, alors :

$$f_{x,y}(x, y) = f_x(x) f_y(y)$$

### COVARIANCE

La covariance mesure l'intensité de la relation linéaire entre deux variables aléatoires X et Y.

$$Cov(X, Y) = E(XY) - E(X)E(Y)$$

Si X et Y sont indépendants, alors :

$$Cov(X, Y) = 0$$

**Attention :** La réciproque n'est pas vraie.

## À mémoriser

Soient U, V, X et Y des variables aléatoires et a, b, c et d des constantes réelles.

### ESPÉRANCE

$$E(aX + bY) = aE(X) + bE(Y) \\ E(a) = a$$

### VARIANCE

$$V(AX) = a^2 V(X) \\ V(A) = 0 \\ V(X + Y) = V(X) + V(Y) + 2Cov(X, Y) \\ V(X - Y) = V(X) + V(Y) - 2Cov(X, Y)$$

### COVARIANCE

$$Cov(X, Y) = Cov(Y, X)$$

$$Cov(aX + b, cY + d) = ac Cov(X, Y)$$

$$Cov(aX + bY, U) = aCov(X, U) + bCov(Y, U)$$

$$Cov(X, cU + dV) = cCov(X, U) + dCov(X, V)$$

$$Cov(aX + bY, cU + dV) = acCov(X, U) + adCov(X, V) + bcCov(Y, U) + bdCov(Y, V)$$

## Vecteurs aléatoires

Un vecteur aléatoire est un n-uplet formé de variables aléatoires. On note  $(X_1, X_2, \dots, X_n)^t$

L'espérance est toujours linéaire. Pour une suite  $(a_i)_{i \in \{1, \dots, n\}}$  de réels, on a :  $E(a_1 X_1 + a_2 X_2 + \dots + a_n X_n) = a_1 E(X_1) + a_2 E(X_2) + \dots + a_n E(X_n)$

Pour les variables **indépendantes**, on a :  $V(X_1 + X_2 + \dots + X_n) = V(X_1) + \dots + V(X_n)$

## Lois usuelles

Ce tableau récapitule les lois usuelles que vous pourrez rencontrer dans différents cours du master.

Lois discrètes				
Loi X	Support	Densité	E(X)	V(X)
Bernoulli B(p)	{0,1}	$P(X = 1) = p = 1 - P(X = 0)$	p	p(1-p)
Binomiale B(n,p)	{0,...,n}	$P(X = k) = C_n^k p^k (1-p)^{n-k}$	np	np(1-p)
Poisson P(λ)	N	$P(X = k) = \frac{\lambda^k}{k!} e^{-\lambda}$	λ	λ

Lois continues				
Loi X	Support	Densité	E(X)	V(X)
Uniforme U[a,b]	[a,b]	$f(x) = \frac{1}{a-b} 1_{[a,b]}(x)$	$\frac{a+b}{2}$	$\frac{(b-a)^2}{12}$
Exponentielle E(λ)	R <sup>+</sup>	$f(x) = \lambda e^{-\lambda x} 1_{R^+}(x)$	$\frac{1}{\lambda}$	$\frac{1}{\lambda^2}$
Normale N(μ, σ <sup>2</sup> )	R	$f(x) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}$	μ	σ <sup>2</sup>
Chi-deux à n degrés de liberté χ <sub>n</sub> <sup>2</sup> (n>0)	R <sup>+</sup>	$f(x) = \frac{1}{2^{\frac{n}{2}} \Gamma(\frac{n}{2})} e^{-\frac{x}{2}} x^{\frac{n}{2}-1} 1_{R^+}(x)$		
Student à n degrés de liberté t(n) (n>0)	R	$f(x) = \frac{1}{\sqrt{n} B(\frac{1}{2}, \frac{n}{2})} \left(1 + \frac{x^2}{n}\right)^{-\frac{n+1}{2}}$		
Fisher à (p,q) degrés de liberté F(p,q) (p,q>0)	R <sup>+</sup>	$\frac{1}{B(\frac{p}{2}, \frac{q}{2})} p^{\frac{p}{2}} q^{\frac{q}{2}} \frac{x^{\frac{p}{2}-1}}{(q + px)^{\frac{p+q}{2}}} 1_{R^+}(x)$		

$\Gamma(x) = \int_0^{+\infty} t^{x-1} e^{-t} dt$  désigne la fonction Gamma d'Euler

$B(x, y) = \Gamma(x) \Gamma(y) = \int_0^1 t^{x-1} (1-t)^{y-1} dt = \frac{\Gamma(x) \Gamma(y)}{\Gamma(x+y)}$  désigne la fonction Béta

Nous allons souvent rencontrer les lois grisées dans les Tests statistiques. Là encore, connaître les densités ne servirait pas à grand-chose, mais ceci nous évitera de parler de lois dont nous ne connaissons pas la tête.

## Vecteurs Aléatoires

### INDEPENDANCE DEUX A DEUX

Les variables  $X_1, \dots, X_2$  sont deux à deux indépendantes si et seulement si :  $\forall i \neq j; X_i$  et  $X_j$  sont indépendantes

### Indépendance mutuelle

Les variables  $X_1, \dots, X_2$  sont mutuellement indépendantes si et seulement si :

$$P(X_1 = x_1, \dots, X_n = x_n) = P(X_1 = x_1) \times \dots \times P(X_n = x_n)$$

**L'INDÉPENDANCE MUTUELLE IMPLIQUE  
L'INDÉPENDANCE DEUX À DEUX.  
LA RÉCROQUE EST FAUSSE.**

Si  $X_1, \dots, X_n$  sont mutuellement indépendantes, alors pour toute famille de fonctions réelles  $(f_i)_i$ , on a :

$(f_1(X_1), \dots, f_n(X_n))$  sont indépendantes.

### ESPÉRANCE ET VARIANCE-COVARIANCE

Soit  $X = (X_1, \dots, X_2)^t$  un vecteur aléatoire. Dans le cas multidimensionnel, l'espérance scalaire est remplacée par un vecteur espérance.

$$E(X) = (E(X_1), \dots, E(X_2))^t$$

La variance unidimensionnelle est remplacée par la matrice symétrique de variance-covariance. Elle contient les variances en diagonale et les covariances ailleurs. On la note généralement  $\sum_x$ .

$$\sum_x = \begin{pmatrix} V(X_1) & \dots & Cov(X_1, X_n) \\ \vdots & \ddots & \vdots \\ Cov(X_n, X_1) & \dots & V(X_n) \end{pmatrix}$$

## Notions de Convergence

Si l'on pense à des données, vues comme réalisation de variables aléatoires  $X_1, \dots, X_n$ . Il serait intéressant de se poser la question de savoir comment évolue cette suite lorsque  $n$  tend vers l'infini.

### Convergence presque sûre

On dit que  $(X_n)$  converge presque sûrement vers  $X$  et on note  $X_n \xrightarrow{p.s.} X$  si et seulement si :

$$P(\{\lim_{n \rightarrow +\infty} X_n = X\}) = 1$$

### Convergence en probabilité

On dit que  $(X_n)$  converge en probabilité vers  $X$  et on note  $X_n \xrightarrow{p} X$  si et seulement si :

$$(\forall \varepsilon > 0); P(|X_n - X| > \varepsilon) \rightarrow 0$$

### Convergence en Loi

On dit que  $(X_n)$  converge en loi vers  $X$  et on note  $X_n \xrightarrow{Loi} X$  si et seulement si :

$$F_{X_n} \xrightarrow{n \rightarrow +\infty} F_X$$

Où  $F_X$  dénote la fonction de répartition de  $X$

### Convergence en M.Q

On dit que  $(X_n)$  converge en moyenne quadratique vers  $X$  et on note  $X_n \xrightarrow{m.q} X$  si et seulement si :

$$E((X_n - X)^2) \rightarrow 0$$

Cette définition peut se généraliser jusqu'à l'ordre  $n$ , mais nous n'en aurons pas besoin.

### Loi faible des grands nombres

Soit  $X_1, \dots, X_n$  une suite de variables aléatoires indépendantes et de même loi telle que :

$$E(X_i) = \mu \text{ et } V(X_i) = \sigma^2$$

Alors :

$$\frac{1}{n} \sum_{i=1}^n X_i \xrightarrow{p} \mu$$

### Loi forte des grands nombres

Soit  $X_1, \dots, X_n$  une suite de variables aléatoires indépendantes et de même loi telle que :

$$E(X_i) = \mu \text{ et } V(X_i) = \sigma^2$$

Alors :

$$\frac{1}{n} \sum_{i=1}^n X_i \xrightarrow{p.s.} \mu$$

### Théorème Central Limite

Soit  $X_1, \dots, X_n$  une suite de variables aléatoires indépendantes et de même loi telle que :

$$E(X_i) = \mu \text{ et } V(X_i) = \sigma^2$$

Alors :

$$\sqrt{n} \frac{\bar{X}_n - \mu}{\sigma} \xrightarrow{Loi} N(0, 1)$$



## Echantillon / Estimateur

Le point de départ est un vecteur (ou un tableau dans le cas multidimensionnel) de données.

Ces données peuvent être vues comme les **réalisations**  $(x_1, x_2, \dots, x_n)$  d'une variable aléatoire  $X$  qui dépend d'un certain paramètre  $\theta$  que nous allons chercher à estimer.

Pour ce faire, nous allons construire un échantillon de cette variable. Un **échantillon**  $(X_1, X_2, \dots, X_n)$  est un n-uplet de variables aléatoires **indépendantes** et **qui suivent tous la même loi** (celle de  $X$ ).

Un **estimateur** de  $\theta$  est une fonction  $\hat{\theta} = f(X_1, X_2, \dots, X_n)$  de notre échantillon, qui possède une loi de probabilité.

Lorsque l'aléa est réalisé,  $\hat{\theta}(\omega) = f(x_1, x_2, \dots, x_n)$  est une **estimation** de  $\theta$ . Le but de ce cours est de construire le **meilleur** estimateur possible de  $\theta$ .

## Estimateur sans biais

Pour que l'estimation soit bonne, il faut que  $\hat{\theta}$  **soit proche** de  $\theta$ . Comme  $\hat{\theta} = f(X_1, X_2, \dots, X_n)$  est une variable aléatoire, on ne peut imposer de condition qu'à sa **valeur moyenne**.

On définit ainsi le biais :

$$b_n(\hat{\theta}, \theta) = E(\hat{\theta}_n) - \theta$$

Un estimateur est dit **sans biais** si  $b_n(\hat{\theta}, \theta) = 0$ .

C'est-à-dire :

$$E(\hat{\theta}_n) = \theta$$



## Estimateur convergent

Un estimateur est dit **convergent** s'il converge en probabilité vers le paramètre à estimer :

$$\hat{\theta}_n \rightarrow \theta$$

En **pratique**, tout estimateur **sans biais** et dont la **variance tend vers 0** est convergent.

## Estimateur optimal

### Qualité d'un estimateur

La **qualité d'un estimateur** est mesurée à travers son erreur quadratique moyenne définie par :

$$EQM(\hat{\theta}_n) = (b_n(\hat{\theta}, \theta))^2 + V(\hat{\theta}_n)$$

Comme nous cherchons tout le temps (presque) des estimateurs sans biais, il reste à **comparer les variances**.

Un estimateur  $\hat{\theta}_1$  est **meilleur** que  $\hat{\theta}_2$  si :

$$V(\hat{\theta}_1) < V(\hat{\theta}_2)$$

### Inégalité de Rao-Cramer/ Efficacité

On définit la **quantité d'information** apportée par l'estimateur par :

$$I(\hat{\theta}_n) = - \left( E \left( \frac{\partial L}{\partial \theta} \right) \right)^2$$

Où  $L(x, \theta) = \prod_{i=1}^n f(x_i)$  est la **vraisemblance**. (nous reviendrons sur sa définition)

L'**inégalité de Rao-Cramer** postule que la variance d'un estimateur ne peut pas aller en delà d'un certain seuil :

$$V(\hat{\theta}_n) \geq \frac{1}{I(\hat{\theta}_n)}$$

Un estimateur est **optimal** (ou **efficace**) si sa variance vérifie le cas d'égalité.

## Construction d'un estimateur

### Méthode du maximum de vraisemblance

La méthode de maximum de vraisemblance consiste à affecter  $\theta$  la valeur qui maximise la probabilité d'observer  $(x_1, x_2, \dots, x_n)$  lorsque l'aléa du vecteur  $(X_1, X_2, \dots, X_n)$  tombe. Sans trop rentrer dans la théorie de la vraisemblance, nous allons présenter un algorithme en cinq étapes pour calculer cet estimateur (qui présente des propriétés assez séduisantes) :

**Etape 1 : Calculer la fonction de vraisemblance :**

$$L(x, \theta) = \prod_{i=1}^n f(x_i) \text{ dans le cas continu, ou } L(x, \theta) = \prod_{i=1}^n P(X_i = x_i) \text{ dans le cas discret.}$$

**Etape 2 : Calculer le log-vraisemblance :**

Il s'agit de calculer un maximum, ce qui revient à dériver. Il s'agit ici d'un produit de  $n$  facteurs ce qui rend la dérivation assez coriace. La fonction logarithmique présente des propriétés assez sympas pour faciliter cette tâche.

**Etape 3 : Calculer la dérivée de la log-vraisemblance.**

**Etape 4 : Résoudre l'équation d'inconnue  $\theta$  :**

$$\frac{\partial(\ln(L))}{\partial \theta} = 0 \Rightarrow \theta = \theta_0$$

**Etape 5 : Vérifier qu'il s'agit d'un maximum :** effectivement en s'assurant que :

$$\frac{\partial^2(\ln(L))}{\partial \theta^2} < 0$$

### Méthode des moments

Comme le paramètre à estimer intervient dans la densité de probabilité, les moments théoriques sont souvent en fonction de ce paramètre. Ainsi la méthode des moments consiste à égaliser les moments théoriques (espérance, variance) à leurs équivalents empiriques et à en dégager une estimation ponctuelle.

En pratique, il faut résoudre l'(les) équation(s) :  $E(X) = \overline{X_n}$  et  $V(X) = S_n^2$  avec :

$$\overline{X_n} = \frac{1}{n} \sum_{i=1}^n X_i; S_n^2 = \frac{1}{n} \sum_{i=1}^n (X_i - \overline{X_n})^2$$

### Méthode des moindres carrés ordinaires

Lorsqu'il s'agit de prendre une mesure  $\theta$  avec un appareil doté d'une imprécision  $\varepsilon$ , alors le problème d'estimation peut s'écrire :  $X = \theta + \varepsilon$

La méthode des moindres-carrés ordinaires consiste à trouver le paramètre  $\theta$  qui minimise la somme des carrés des erreurs :

$$\hat{\theta}_{MCO} = \operatorname{argmin} \left( \sum_{i=1}^n \varepsilon_i^2 \right) = \operatorname{argmin} \left( \sum_{i=1}^n (X_i - \theta)^2 \right)$$




## Intervalles de confiance

Un intervalle de confiance  $[A, B]$  de niveau  $1 - \alpha$  est un intervalle aléatoire qui a la probabilité  $1 - \alpha$  de contenir le paramètre à estimer  $\theta$ . Formellement, on écrit :  $P(t_1(\theta) \leq f(X_1, \dots, X_n) \leq t_2(\theta)) = P(A \leq \theta \leq B) = 1 - \alpha$

## Test d'hypothèses

Dans le cadre d'un test d'hypothèse, nous cherchons à faire valoir une hypothèse en dépit d'une autre, qui lui est contradictoire.

On appellera la première (celle dont le rejet à tort sera le plus préjudiciable) « Hypothèse nulle » et la deuxième « Hypothèse alternative ».

		Réalité		
		$H_0$ vraie	$H_0$ fausse	
Décision	Garder $H_0$	$1 - \alpha$	$\beta$	 Risque de deuxième espèce
	Rejeter $H_0$	$\alpha$	$1 - \beta$	
		 Risque de première espèce	 Puissance du test	

Les calculs qui se cachent derrière le choix de l'hypothèse à garder sont compliqués. Mais **BONNE NOUVELLE**, la machine fera tour à notre place. Il suffit juste de suivre correctement la méthode :

**Etape 1 :** Choisir judicieusement les hypothèses à évaluer et fixer le risque  $\alpha$ .

**Etape 2 :** Choisir le test adapté à la procédure.

**Etape 3 :** Rentrer la commande correspondante sur R et exécuter.

**Etape 4 :** Lire dans les sorties la p-value. si elle est supérieure à  $\alpha$  on accepte  $H_0$ . Si elle lui est inférieure, on rejette  $H_0$

**Le tableau qui va suivre est à lire avec la plus grande des attentions !!**



# Constructions d'Intervalle de confiance

Construction d'intervalles de confiance usuels				
Pour les paramètres d'une loi $N(\mu, \sigma^2)$				Pour une proportion p
Pour $\mu$		Pour $\sigma^2$		
Avec $\sigma$ connu	Avec $\sigma$ inconnu	Avec $\mu$ connu	Avec $\mu$ inconnu	
<p>L'EMV de <math>\mu</math> est :</p> $\overline{X}_n = \frac{1}{n} \sum_{i=0}^n X_i, \overline{X}_n \sim N\left(\mu, \frac{\sigma^2}{n}\right)$ <p>On a : <math>\sqrt{n} \frac{\overline{X}_n - \mu}{\sigma} \sim N(0, 1)</math></p> <p>Nous allons donc utiliser les fractiles d'une loi normale centrée réduite.</p> $P\left(-u_{1-\frac{\alpha}{2}} \leq \sqrt{n} \frac{\overline{X}_n - \mu}{\sigma} \leq u_{1-\frac{\alpha}{2}}\right) = 1 - \alpha$ <p>Ceci donne :</p> $P\left(\overline{X}_n - u_{1-\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}} \leq \mu \leq \overline{X}_n + u_{1-\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}}\right) = 1 - \alpha$ <p>Et donc un intervalle de confiance pour <math>\mu</math> de niveau de confiance <math>1 - \alpha</math> est :</p> $\left[\overline{X}_n - u_{1-\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}}, \overline{X}_n + u_{1-\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}}\right]$ <p>Où <math>u_{1-\frac{\alpha}{2}}</math> est le fractile d'ordre <math>1 - \frac{\alpha}{2}</math> d'une loi normale centrée réduite.</p>	<p>L'EMV de <math>\mu</math> est :</p> $\overline{X}_n = \frac{1}{n} \sum_{i=0}^n X_i, \overline{X}_n \sim N\left(\mu, \frac{\sigma^2}{n}\right)$ <p>Comme <math>\sigma</math> inconnu, on l'estime avec son équivalent empirique.</p> $s_n'^2 = \frac{1}{n-1} \sum_{i=0}^n (X_i - \overline{X}_n)^2$ <p>On a : <math>\sqrt{n} \frac{\overline{X}_n - \mu}{s_n'} \sim St(n-1)</math></p> <p>Nous allons donc utiliser les fractiles d'une loi de Student à n-1 degrés de liberté.</p> $P\left(-t_{1-\frac{\alpha}{2}} \leq \sqrt{n} \frac{\overline{X}_n - \mu}{s_n'} \leq t_{1-\frac{\alpha}{2}}\right) = 1 - \alpha$ <p>Ceci donne :</p> $P\left(\overline{X}_n - t_{1-\frac{\alpha}{2}} \frac{s_n'}{\sqrt{n}} \leq \mu \leq \overline{X}_n + t_{1-\frac{\alpha}{2}} \frac{s_n'}{\sqrt{n}}\right) = 1 - \alpha$ <p>Et donc un intervalle de confiance pour <math>\mu</math> de niveau de confiance <math>1 - \alpha</math> est :</p> $\left[\overline{X}_n - t_{1-\frac{\alpha}{2}} \frac{s_n'}{\sqrt{n}}, \overline{X}_n + t_{1-\frac{\alpha}{2}} \frac{s_n'}{\sqrt{n}}\right]$ <p>Où <math>t_{1-\frac{\alpha}{2}}</math> est le fractile d'ordre <math>1 - \frac{\alpha}{2}</math> d'une loi de Student à n-1 degrés de liberté.</p>	<p>L'EMV de <math>\sigma^2</math> est :</p> $s_n^2 = \frac{1}{n} \sum_{i=0}^n (X_i - \mu)^2$ <p>On a : <math>\frac{ns_n^2}{\sigma^2} \sim \chi_n^2</math></p> <p>Nous allons donc utiliser les fractiles d'une loi de Khi-deux à n degrés de liberté.</p> $P\left(c_1 \leq \frac{ns_n^2}{\sigma^2} \leq c_2\right) = 1 - \alpha$ <p>Ceci donne :</p> $P\left(\frac{ns_n^2}{c_2} \leq \sigma^2 \leq \frac{ns_n^2}{c_1}\right) = 1 - \alpha$ <p>Et donc un intervalle de confiance pour <math>\sigma^2</math> de niveau de confiance <math>1 - \alpha</math> est :</p> $\left[\frac{ns_n^2}{c_2}, \frac{ns_n^2}{c_1}\right]$ <p>Où <math>c_1</math> (respectivement <math>c_2</math>) est le fractile d'ordre <math>1 - \frac{\alpha}{2}</math> (respectivement <math>\frac{\alpha}{2}</math>) d'une loi de Khi-deux à n degrés de liberté.</p>	<p>L'EMV de <math>\sigma^2</math> est :</p> $s_n'^2 = \frac{1}{n-1} \sum_{i=0}^n (X_i - \overline{X}_n)^2$ <p>On a : <math>\frac{(n-1)s_n'^2}{\sigma^2} \sim \chi_{n-1}^2</math></p> <p>Nous allons donc utiliser les fractiles d'une loi de Khi-deux à n-1 degrés de liberté.</p> $P\left(c_1 \leq \frac{(n-1)s_n'^2}{\sigma^2} \leq c_2\right) = 1 - \alpha$ <p>Ceci donne :</p> $P\left(\frac{(n-1)s_n'^2}{c_2} \leq \sigma^2 \leq \frac{(n-1)s_n'^2}{c_1}\right) = 1 - \alpha$ <p>Et donc un intervalle de confiance pour <math>\sigma^2</math> de niveau de confiance <math>1 - \alpha</math> est :</p> $\left[\frac{(n-1)s_n'^2}{c_2}, \frac{(n-1)s_n'^2}{c_1}\right]$ <p>Où <math>c_1</math> (respectivement <math>c_2</math>) est le fractile d'ordre <math>1 - \frac{\alpha}{2}</math> (respectivement <math>\frac{\alpha}{2}</math>) d'une loi de Khi-deux à n-1 degrés de liberté.</p>	<p>L'EMV de p est <math>\hat{p} = \frac{N}{n}</math>, où <math>N \sim B(n, p)</math> (loi binomiale).</p> <p>Pour n assez grand (<math>n \geq 30</math>), on a : <math>N \rightarrow N(np, np(1-p))</math></p> <p>Donc :</p> $\sqrt{n} \frac{\hat{p} - p}{\sqrt{p(1-p)}} \sim N(0,1)$ $P\left(-u_{1-\frac{\alpha}{2}} \leq \sqrt{n} \frac{\hat{p} - p}{\sqrt{p(1-p)}} \leq u_{1-\frac{\alpha}{2}}\right) = 1 - \alpha$ <p>Ceci donne :</p> $P\left(\hat{p} - u_{1-\frac{\alpha}{2}} \sqrt{\frac{p(1-p)}{n}} \leq p \leq \hat{p} + u_{1-\frac{\alpha}{2}} \sqrt{\frac{p(1-p)}{n}}\right) = 1 - \alpha$ <p>Comme p (à l'intérieur de l'intervalle) est inconnu, on l'estime avec son équivalent empirique <math>\hat{p}</math>.</p> <p>Un intervalle de confiance pour p de niveau de confiance <math>1 - \alpha</math> est :</p> $\left[\hat{p} - u_{1-\frac{\alpha}{2}} \sqrt{\frac{p(1-p)}{n}}, \hat{p} + u_{1-\frac{\alpha}{2}} \sqrt{\frac{p(1-p)}{n}}\right]$ <p>Où <math>u_{1-\frac{\alpha}{2}}</math> est le fractile d'ordre <math>1 - \frac{\alpha}{2}</math> d'une loi normale centrée réduite.</p>

## Avoir l'aide

### Accéder à l'aide

**?fonction** : Accéder à l'aide d'une fonction.  
**Help.search("fonction")** : Chercher de l'aide d'une fonction.  
**Help (package = 'MASS')** : Trouvez de l'aide pour un package.

## Packages

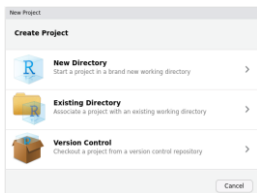
**install.packages ('MASS')** : Télécharger et installer un package.  
**library ('MASS')** : Importer un package et rendre toutes ses fonctions accessibles.  
**data (iris)** : Importer un dataset de R (ici iris).

## Répertoire

**getwd ()** : Trouver le répertoire de travail.  
**setwd ('C://fichier/chemin')** : Changer le répertoire de travail courant.

### Organiser ses scripts !

Afin d'organiser son travail et de faciliter l'accès à l'ensemble de fichiers constitutifs d'une analyse, on commence notre travail sur R par la création d'Un projet !  
 Pour créer un projet, il faut aller dans le menu File puis sélectionner New project.



Selon que le dossier du projet existe déjà ou pas, on choisira Existing directory ou New directory

## Importer et exporter les données

### Import de fichiers textes

On utilise L'extension readr, qui fait partie de tidyverse  
 Library (tidyverse)  
 Library (readr)  
 - **d <- read\_csv("fichier.csv")** : pour un fichier Excel séparées par des points virgule  
 - **d <- read\_csv2("fichier.csv")** : pour un fichier Excel séparées par des points virgule  
 - **fread** de l'extension data.table : pour importer le plus rapidement possible des fichiers textes très volumineux.

### Import depuis un fichier Excel

L'extension readxl, qui fait partie du tidyverse, permet d'importer des données directement depuis un fichier au format xls ou.xlsx.  
 Library (readxl)  
 - **d <- read\_excel("fichier.xls")**

### Import de fichiers SAS, SPSS et Stata

Library(haven)  
 - **read\_sas** ou **read\_xpt** : Pour les fichiers provenant de SAS  
 - **read\_sav** ou **read\_por** Pour les fichiers provenant de SPSS,  
 - **read\_dta** : Pour les fichiers provenant de Stata.

### Export de données

- **write\_csv, write\_csv2, read\_tsv** permettent d'enregistrer un data frame ou un tibble dans un fichier au format texte délimité  
 - **write\_sas** export au format SAS  
 - **write\_sav** export au format SPSS  
 - **write\_dt** exporter au format Stata



**NB :**  
**Il n'existe pas de fonctions permettant d'enregistrer directement au format xls ou.xlsx. On peut dans ce cas passer par un fichier CSV.**

## Programmation

### BOUCLE for

**for (variable in sequence) {**  
 instructions  
**}**

**EXEMPLE**  

```
for (i in 1:4) {
  j <- i + 10
  print (j) }
#> 11 12 13 14
```

### CONDITION if

**if (condition) {**  
 instructions  
**}** else {  
 Instructions différentes}

**EXEMPLE**  

```
while (i < 5) {
  print (i)
  i <- i + 1 }
```

### Condition While

**while (condition) {**  
 instructions  
**}**

**EXEMPLE**  

```
while (i < 5) {
  print (i)
  i <- i + 1 }
```

### Fonctions

**nom\_de\_fonction** ← **function (variable){**  
 instructions  
**return (nouvelle\_variable) }**

**EXEMPLE**  

```
square <- function (x) {
  squared <- x * x
  return (squared) }
```

## Fonctions mathématiques

**log (x)** : Logarithme de x.  
**exp (x)** : Exponentielle de x.  
**max (x)** : Plus grand élément de x.  
**min (x)** : Plus petit élément de x.  
**round (x, n)** : Arrondit x à n décimales.  
**signif (x, n)** : Arrondit x à n valeurs significatives.  
**corr (x, y)** : Coefficient de corrélation entre x et y.

**sum (x)** : Somme des éléments de x.  
**median (x)** : Médiane des éléments de x.  
**quantile (x)** : Quantiles.  
**rank (x)** : Rangs des éléments de x.  
**var (x)** : Variance des éléments de x.  
**sd (x)** : Ecart-type des éléments de x

## Conditions

**a == b** Est-ce que a est égal à b ?

**a != b** Est-ce que a est différent à b ?

**a > b** Est-ce que a est strictement supérieur à b ?

**a < b** Est-ce que a est strictement inférieur à b ?

**is.na(a)** Y'a-t-il une valeur manquante dans a ?

**is.null(a)** Y'a-t-il du contenu dans a ?

**a >= b** Est-ce que a est supérieur ou égale à b ?

**a <= b** Est-ce que a est inférieur ou égal à b ?

## Vecteurs

### Créer des vecteurs

Commande	Retour	Commentaire
<code>c(2,4,6)</code>	2 4 6	Rassembler des éléments dans un vecteur
<code>2:6</code>	2 3 4 5 6	Créer une séquence d'entiers
<code>seq(2,3, by=0.5)</code>	2.0 2.5 3.0	Créer une séquence entre 2 et 3 avec un pas de 0.5
<code>rep(1:2, times = 3)</code>	1 2 1 2 1 2	Répéter le vecteur 1:2 3 fois.
<code>rep(1:2, each = 3)</code>	1 1 1 2 2 2	Répéter chaque élément du vecteur 1:2 3 fois.

## Listes

`lst ← list ( x = 1:5, y = c('a', 'b'))`

Une liste est une collection de données qui peuvent ne pas être du même type.

`lst[[2]]` : Retourne le 2ème élément de lst.

`lst[1]` : Retourne une nouvelle liste avec le 1er élément de lst.

`lst$x` : Retourne l'élément x de lst.

`lst['y']` : Retourne une nouvelle liste avec l'élément y de lst.

## Matrices

`m ← matrix ( x, nrow = 3, ncol = 3)`

Créer une matrice 3x3 à partir des éléments de x.



`m[2,]` : Sélectionner la 2ème ligne.

`m[,1]` : Sélectionner la 1ère colonne.

`m[2,3]` : Sélectionner le croisement de la 2ème ligne et la 3ème colonne.

`t(m)` : Transposée de m.

`m%*%n` : Multiplication matricielle.

`solve(m, n)` : Résoudre l'équation  $mx = n$ .

`solve(m)` : Inverser m.

## Chaînes de Caractères

`paste(x, y, sep = " ")` :

Joint les deux vecteurs x et y.

`grep(pattern, x)` :

Trouve les occurrences d'une expression dans x.

`gsub(pattern, replace, x)` :

Remplacer les occurrences de «pattern» par «replace» dans x.

`toupper(x)` : Transforme en majuscules.

`tolower(x)` : Transforme en minuscules.

`toupper(x)` : Transforme en majuscules.

`nchar(x)` : Compte le nombre de caractères dans une chaîne.

## Data Frames

`df ← data.frame (x= 1:3, y = c('a','b','c'))`

Un dataframe est une liste particulière où tous les éléments ont la même longueur.

x	y
1	a
2	b
3	c

### Extraction (de listes)



Il y a aussi des fonctions d'extraction comme `subset()` et la structure `attach()`, `detach()` ..



### Combinaison



## Manipuler les data frames

### Ajouter une colonne à un data frame

```
df$nouvelle_variable <- x
df[["nouvelle_variable"]] <- x
df[, "nouvelle_variable"] <- x
```

### Supprimer les colonnes

```
df$col <- NULL
df[, "col"] <- NULL
df[["col"]] <- NULL
```

### Mieux connaître le df

**View (df)** : Voir le dataframe.

**Head (df)** : Voir les 6 premières lignes.

**nrow (df)** : Nombre de lignes dans df.

**ncol (df)** : Nombre de colonnes dans df.

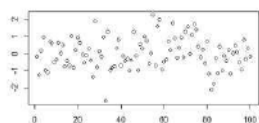
**dim (df)** : Taille de la matrice df.

**attributes(df)** : voir les attributs internes

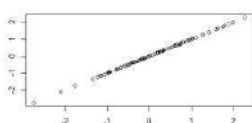
## Distributions

	Fonction densité	Fonction de rép.	Quantile
Normale	<code>dnorm</code>	<code>pnorm</code>	<code>qnorm</code>
Poisson	<code>dpois</code>	<code>ppois</code>	<code>qpois</code>
Binomiale	<code>dbinom</code>	<code>pbinom</code>	<code>qbinom</code>
Uniforme	<code>dunif</code>	<code>punif</code>	<code>qunif</code>
Student	<code>dt</code>	<code>pt</code>	<code>qt</code>
Khi-deux	<code>dchisq</code>	<code>pchisq</code>	<code>qchisq</code>

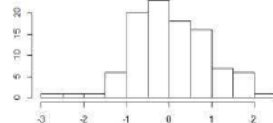
## Graphique



**plot ( x )**  
Nuage de points x



**plot ( x, y )**  
Nuage des points y en fonction de x



**hist ( x )**  
Histogramme de x

## Liens utiles

- ✓ <https://www.linkedin.com/learning/r-pour-les-data-scientists/transformer-un-data-frame>
- ✓ <https://openclassrooms.com/fr/courses/1393696-effectuez-vos-etudes-statistiques-avec-r/1394024-decouvrons-de-nouvelles-fonctions>
- ✓ <https://cran.r-project.org/>
- ✓ <https://courses.edx.org/courses/course-v1:HarvardX+PH125.1x+1T2020/course/>



## Variables et Types

### Affectation

```
x = 5
```

### Calculs

```
x + 2
```

 Addition

```
>>> 7
```

```
x - 2
```

 Soustraction

```
>>> 3
```

```
x * 2
```

 Multiplication

```
>>> 10
```

```
x ** 2
```

 Puissance

```
>>> 25
```

```
x % 2
```

 Reste

```
>>> 1
```

```
x / float(2)
```

 Division

```
>>> 2.5
```

### Types et conversion

```
str ()
```

 Convertit en ch de caractères

```
int ()
```

 Convertit en entiers

```
float ()
```

 Convertit en réels

```
bool ()
```

 Convertit en booléens

### Aide

```
>>> help (str)
```

## Listes

```
a = 'is'
```

```
b = 'nice'
```

```
my_list1 = ['my', 'list', a, b]
```

```
my_list2 = [[4, 5, 6, 7], [3, 4, 5, 6]]
```

### Extraire les éléments d'une liste

```
my_list1[1]
```

 Extraire l'élément d'indice 1.

```
my_list1[-3]
```

 Extraire le troisième dernier élément.

```
my_list1[1:3]
```

 Extraire les éléments 'indices 1 et 2.

```
my_list1[1:]
```

 Extraire les éléments après l'indice 0.

```
my_list1[:3]
```

 Extraire les éléments avant l'indice 3.

```
my_list1[:]
```

 Copier la liste.

```
my_list1[1][0]
```

 Extraire le premier élément de la sous-liste d'indice 1.

### Opérations sur les listes

```
my_list
```

 Concatène la liste deux fois.

```
my_list * 5
```

 : Concatène la liste cinq fois.

### Opérations avancées

#### Par position

```
my_list.index(a)
```

 Extraire l'indice d'un élément.

```
my_list.count ( a )
```

 Compter les occurrences d'un élément.

```
my_list.append ( '!' )
```

 Ajouter un élément. **my\_list.remove ( '!' )** Supprimer un élément.

```
del ( my_list [ 0 : 1 ] )
```

 Supprimer les éléments d'indices 0 et 1.

```
my_list.reverse ( )
```

 Renverser la liste ( sens inverse ).

```
my_list.pop ( -1 )
```

 Supprimer le premier élément.

```
my_list.insert ( 0 , '!' )
```

 Ajouter un élément en première position.

```
my_list.sort ( -1 )
```

 Trier la liste.

## Librairie

### Importer les bibliothèques

```
import numpy
```

 Importe la librairie Numpy.

```
Import numpy as np
```

 Importer NumPy avec un nom raccourci.

### Importation de fonctions particulières

```
from math import pi
```

pandas



Data Analysis



Machine Learning



NumPy

Calcul scientifique



Graphiques en 2D

## Installation de python



Plateforme ouverte dédiée aux sciences des données développée sous Python



Environnement de développement inclus dans Anaconda



Application web pour le développement

## Tableaux

```
my_list = [1, 2, 3, 4]
```

```
my_array = np.array (my_list)
```

```
my_2darray = np.array ([1, 2, 3], [4, 5, 6])
```

### Extraction d'éléments

```
my_array[1]
```

 Extraire l'élément d'indice 1.

```
my_array[0:2]
```

 Extraire les éléments 'indices 0 et 1.

```
my_2darray[:, 0]
```

 Extraire les éléments d'indice 1 de chaque axe du tableau.

### Opérations sur les tableaux

```
my_array > 3
```

 Tester si chaque élément du tableau est strictement supérieur à 3.

```
array([FALSE, FALSE, FALSE, TRUE]).dtype = bool )
```

```
my_array * 2
```

 : Calculer le double de chaque élément du tableau.

```
array([2, 4, 6, 8])
```

```
my_array + np.array ([5, 6, 7, 8])
```

 : Additionner les tableaux terme à terme.

```
array([6, 8, 10, 12])
```

### Fonctions de tableaux

#### Par position

```
my_array.shape
```

 Retourne les dimensions du tableau.

```
np.append (other_array)
```

 Ajouter de nouveaux éléments au tableau.

```
np.insert(my_array, 1, 5) np.delete(my_array, [1])
```

 Insérer/Supp des éléments au tableau.

```
np.mean(my_array) np.median(my_array) np.std(my_array)
```

 Calculer la moyenne/médiane/écart type de éléments du tableau ).

```
my_array.corrcoef ()
```

 Calculer le coefficient de corrélation.

## Chaînes de caractères

```
my_string = 'ThisStringIsAwesome'
```

```
my_string
```

```
'ThisStringIsAwesome'
```

### Opérations

```
my_string * 2
```

```
'ThisStringIsAwesomeThisStringIsAwesome'
```

```
my_string + 'Innit'
```

```
'ThisStringIsAwesomeInnit'
```

```
'm' in my_string
```

```
TRUE
```

```
my_string [3]
```

```
's'
```

```
my_string [4:9]
```

```
'String'
```

### Méthodes

```
my_string.upper
```

 Transformer en majuscules.

```
my_string.lower
```

 Transformer en minuscules.

```
my_string.count ('w')
```

 Compter le nombre d'apparitions du caractère précisé ( ).

```
my_string.replace ('e', 'i')
```

 Remplacer des caractères.

```
my_string.strip
```

 Enlever des espaces.

## Pandas

Pandas est une librairie basée sur NumPy. Elle fournit des structures de données faciles à l'usage et des outils pour le traitement et la fouille des données.  
**Import pandas as pd**

## Structure de Données

### Series

Ils équivalent les listes sous R. Une série est une structure de données unidimensionnelle et indexée qui supporte tous les types de données.

```
s = pd.Series([3, -5, 7, 4],
              index = ['a', 'b', 'c', 'd'])
```

### Data frame

```
data = {'Country': ['Belgium', 'China', 'France'],
        'Capital': ['Brussels', 'Pekin', 'Paris'],
        'Pop': [11190846, 1303171035, 207847528]}
df = pd.DataFrame(data, columns = ['Country', 'Capital', 'Pop'])
```

a	3
b	-5
c	7
d	4

Index

	Country	Capital	Pop
0	Belgium	Brussels	11190846
1	China	Pekin	1303171035
2	France	Paris	87847528

## Extraction

### Accéder à un élément

```
s['b']
-5
df[1:]
```

	Country	Capital	Pop
1	China	Pekin	1303171035
2	France	Paris	87847528

### Extraire un élément

```
df.iloc[[0], [0]] Extraire un élément par ses coordonnées.
'Belgium'
df.iat[[0], [0]] Idem.
'Belgium'
df.iloc[[0], ['Country']] Extraire un élément par son label.
'Belgium'
df.at[[0], ['Country']] Idem.
'Belgium'

df.ix[2] Extraire une ligne à partir d'une série de lignes.
df.ix[:, 'Country'] Extraire une colonne à partir d'une série de colonnes.
df.ix[1:,'Country'] Extraire un croisement ligne/colonne.
```

```
s[-(s>1)] Extraire les séries où s n'est pas > 1.
s[(s<-1)|(s>2)] Extraire les séries où s est < -1 ou > 2.
df[df['Pop']>12000000] Mettre un filtre au dataframe.
```

## Manipulations de Base

### Supprimer des éléments

```
s.drop(['a', 'c']) Supprimer une valeur à partir d'une ligne.
s.drop('Country', axis = 1) Supprimer une valeur à partir d'une colonne.
```

### Trier & Ranger

```
df.sort_index() Trier en fonction des indices / labels.
df.sort_values(by = 'Country') Trier en fonction des modalités de la variable 'Country'.
df.rank() Affecter un rang à chaque élément.
```

### Récupérer des données

```
df.shape Retourne le nombre de lignes et de colonnes.
df.index Retourne le vecteur des indices.
df.columns Retourne le vecteur des colonnes du dataframe.
df.info() Retourne des informations sur le dataframe.
df.count() Compter le nombre de données non-manquantes.

df.sum() Calculer la somme des valeurs.
df.cumsum() Calculer les sommes cumulées des valeurs.
df.min() / df.max Calculer la valeur minimale / maximale.
df.idmin() / df.idmax Retourne l'indice de la valeur minimale / maximale.
df.describe() Retourne des statistiques descriptives de base.
df.mean() Calculer la moyenne des valeurs.
df.median() Calculer la médiane des valeurs.
```

## Fonctions

```
f = lambda x : x*2
df.apply ( f ) Appliquer une fonction.
df.applymap ( f ) Appliquer la fonction élément par élément.
```

## Liens Utiles

<https://openclassrooms.com/fr/courses/4262331-demarrez-votre-projet-avec-python/4262506-installez-python>  
installer python  
<https://openclassrooms.com/fr/courses/4452741-decouvrez-les-librairies-python-pour-la-data-science/5559646-installez-jupyter-sur-votre-propre-ordinateur>  
installer jupyter  
<https://www.linkedin.com/learning/python-l-analyse-de-donnees/bienvenue-dans-python-l-analyse-de-donnees>  
analyse de donnée sous Pandas  
<https://www.datacamp.com/courses/preprocessing-for-machine-learning-in-python>  
Apprentissage automatique sous python

## Ecrire et lire les données

### Lire/Ecrire sur un csv

```
pd.read_csv('fichier.csv', header = None, nrow = 5)
df.to_csv('monDataFrame.csv')
```

### Lire/Ecrire sur un excel

```
pd.read_excel('fichier.xlsx')
pd.to_excel(df.xlsx, sheet_name = 'Feuil1')
```

### lire plusieurs feuilles du même fichier

```
xlsx = pd.ExcelFile ('fichier.xls')
df = pd.read_excel(xlsx, 'Feuille1')
```

### Lire /Ecrire sur SQL

```
from sqlalchemy import create_engine
engine = create_engine ('sqlite:/// :memory:')
pd.read_sql('select * from my_table;', engine)
pd.read_sql_table('my_table', engine)
pd.read_sql_query ('select * from my_table;', engine)
```

```
pd.to_sql ('myDF', engine)
```



## Un programme SAS

- Un programme SAS se compose
  - d'étapes DATA et PROC
  - dans chaque étape, on écrit des instructions
- Pour exécuter un programme SAS, nous pouvons utiliser l'icône RUN ou la touche F3.

## Lecture des données à partir de datalines

```
Data names;
Infile datalines delimiter = ',';
Length first last $25.;
Input first $ last $;
datalines;
    Amine,Ahmidouch
    Narjisse,Cheddadi
;run;
```

Obs.	first	last
1	Amine	Ahmidouch
2	Narjisse	Cheddadi

Infile : identifie un fichier externe à lire.  
 Input : spécifie les variables de la nouvelle base.  
 Length : précise le nombre de bytes pour stocker les variables  
 Datalines : indique que des lignes de données suivent.  
 Run : exécute l'étape Data.

Différentes options de infile : delimiter, missover, dlm, firstobs

## Lecture des données à partir d'un fichier

```
Proc import
Datafile = "C:\file.csv"
Out = outdata
Dbms = csv
Replace;
Getnames = yes;
```

## Utiliser : proc import

Datafile est l'emplacement du fichier qu'on veut lire.  
 DBMS : spécifie le type de la base à importer.

```
Data names;
Infile "file-path" delimiter = ',';
Length first last $25.;
Input first $ last $;
```

## Utiliser : étape data

Utiliser le infile pour identifier le fichier externe à lire par un input.

```
Run;
```

## Lecture des données brutes

Informats : comment lire la data.    Formats : comment afficher la data  
 Dans la code data, on pourrait rajouter :  
 Informat first \$15. last \$15. birthday ddmmyy10.;  
 format birthday date9.;

## Exemples d'informats

\$w. lit des chaînes de caractères de longueur w.  
 w.d lit des données numériques de longueur w avec d chiffres en décimales  
 MMDDYYw. Lit des données date dans la forme 04-11-80

## Sélection des colonnes

## Sélection d'une colonne par nom

```
Data selected ;
Set sas.help.cars ;
Keep model type ;
```

## Supprimer d'une colonne par nom

```
Data selected ;
Set sas.help.cars ;
Drop model type ;
Changer de label
Data selected ;
Set sas.help.cars ;
Label model = 'car model'
Type = 'car type';
```

## Sélection d'une colonne par sa position

```
%Let to_keep(1:3) ;
Pour sélectionner les
variables de 1 à 3.
```

```
%Let to_keep(1,3, 5) ;
Pour sélectionner les
variables de 1, 3 et 5.
```

## Sélection des lignes

## Ignore les N premières observations

```
Data want ;
Set sas.help.buy (firstobs = 5) ;
```

## Limiter le nombre de lignes à lire

```
Data want ;
Set sas.help.buy (obs = 5);
```

## Sélectionner les lignes avec les conditions if

```
Data want ;
Set sas.help.buy;
If amount >= -1000;
```

## Trier

Utiliser la procédure proc sort :  
 Proc sort data = Score out = Sorted;  
 By descending score;  
 Run;  
 Si on ne précise pas l'ordre, c'est croissant par défaut.

La procédure Sort trie la base de données SAS par les valeurs d'une ou de plusieurs variables numériques ou des chaînes de caractères.

## Conditions if

```
Data titles ;
Set names ;
If name = 'Ali' then
do;
    title = 'Etudiant' ;
end;
else if name ='Karam' then
do;
    title = 'Professeur';
end;
else
    title = 'Salarié';
```

## Les boucles

```
Data df ;
Do x=1 to 3 ;
    y = x**2;
    output;
end;
run;

Data df ;
x = 1;
Do while (x<4);
    y = x**2;
    output;
    x = x+1;
end;
run;

Data df ;
Do x=1, 2, 3 ;
    y = x**2;
    output;
end;
run;

Data df ;
x = 1;
Do until (x>3);
    y = x**2;
    output;
    x = x+1;
end;
run;
```

Obs.	x	y
1	1	1
2	2	4
3	3	9

L'instruction end indique la fin de la boucle, comme c'est visible dans les exemples ci-après.

## Affichage du contenu de la base

## Afficher les 5 premières observations

```
Proc print data = sashelp.class (obs = 5);
```

## Table des fréquences

```
Proc freq data = sashelp.class;
Tables sex/ plots = freqplot;
```

## Un résumé statistique de la base

```
Proc means data = sashelp.iris n mean std min max q1
Median q3;
```





## Exercices d'applications

### Exercice 1:

On considère les matrices suivantes :

$$A = (1 \ 2 \ 3), B = \begin{pmatrix} 1 \\ -2 \end{pmatrix}, C = \begin{pmatrix} 2 & 1 \\ -3 & 0 \\ 1 & 2 \end{pmatrix}, D = \begin{pmatrix} -2 & 5 \\ 5 & 0 \end{pmatrix}, E = \begin{pmatrix} -1 & 1 & 3 \\ -1 & -4 & 0 \\ 0 & 2 & 5 \end{pmatrix}$$

Quels sont les produits matriciels possibles ? Quelles sont les matrices carrées et les matrices symétriques ?

### Exercice 2:

1) Soit  $A = \begin{pmatrix} -1 & 1 & 1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \end{pmatrix}$ . Montrer que  $A^2 = 2I_3 - A$ , en déduire que A est inversible et calculer  $A^{-1}$ .

2) Soit  $A = \begin{pmatrix} 1 & 0 & 2 \\ 0 & -1 & 1 \\ 1 & -2 & 0 \end{pmatrix}$ . Calculer  $A^3 - A$ . En déduire que A est inversible puis déterminer  $A^{-1}$ .

### Exercice 3:

Soit A une matrice carrée d'ordre n telle que  $A^2 = A$ .

# Algèbre linéaire

## Corrigés des exercices d'applications



### Corrigé 1 :

Les produits matriciels possibles sont : AC, AE, BD, CD, DC. La matrice carrée est E, C et D sont des matrices symétriques.

### Corrigé 2 :

Le calcul ne pose pas de problèmes. Il mène à :

$$\frac{A^2 + A}{2} = I_3 \Rightarrow A \frac{A + I_3}{2} = \frac{A + I_3}{2} A = I_3$$

A est inversible, et son inverse est :

$$A^{-1} = \frac{A + I_3}{2}$$

Un calcul donne.  $A^3 - A = 4I_3$ . Donc  $A * \frac{1(A^2 - I_3)}{4} = I_3$ , ainsi A est inversible et

$$A^{-1} = \frac{1}{4}(A^2 - I_3) = \frac{1}{4} \begin{pmatrix} 2 & -4 & 2 \\ 1 & -2 & -1 \\ 1 & 2 & -1 \end{pmatrix}$$

### Corrigé 3 :

1. Si A était inversible, on pourrait multiplier à gauche par  $A^{-1}$  les deux membres de l'égalité  $A^2 = A$ , ainsi  $A = I_n$
2. Posons  $B = I_n - A$ , alors  $B^2 = (I_n - A)^2 = I_n - 2A + A = I_n - A = B$ . Donc d'après la question précédente, si  $B \neq I_n$  alors B n'est pas inversible.  
Autrement dit, si  $A \neq I_n$ , alors  $I_n - A$  n'est pas inversible.

# Fondements de probabilités



## Exercices d'application

### Exercice 1 :

Dans la salle des profs 60% sont des femmes ; une femme sur trois porte des lunettes et un homme sur deux porte des lunettes : quelle est la probabilité pour qu'un porteur de lunettes pris au hasard soit une femme ?

### Exercice 2 :

Dans une entreprise deux ateliers fabriquent les mêmes pièces. L'atelier 1 fabrique en une journée deux fois plus de pièces que l'atelier 2. Le pourcentage de pièces défectueuses est 3% pour l'atelier 1 et 4% pour l'atelier 2. On prélève une pièce au hasard dans l'ensemble de la production d'une journée.

Déterminer :

- la probabilité que cette pièce provienne de l'atelier 1 ;
- la probabilité que cette pièce provienne de l'atelier 1 et est défectueuse ;
- la probabilité que cette pièce provienne de l'atelier 1 sachant qu'elle est défectueuse.

### Exercice 3 :

Soit F la fonction définie par :

$$f(x) = \begin{cases} 0 & \text{si } x < 0 \\ 0,29 & \text{si } -1 \leq x \leq 1 \\ 0,37 & \text{si } 1 \leq x \leq 7 \\ 0,69 & \text{si } 7 \leq x \leq 11 \\ 1 & \text{si } x \geq 11 \end{cases}$$

- a- Vérifier que F est bien une fonction de répartition
- b- Soit X la variable aléatoire admettant F pour fonction de répartition ; quelle est la loi de X ?

### Exercice 4 :

Soit X une variable aléatoire de loi de Poisson de paramètre  $\lambda > 0$ . Calculer  $E(1 + X)^{-1}$

### Exercice 5 :

Montrer que  $\text{Var}(X) = E(X^2) - 2(E(X))^2$

### Exercice 6 :

Le nombre X de kg de tomates récoltés dans un jardin en une semaine, est une variable aléatoire dont la loi de probabilité est la suivante :

- a- Quelle est l'espérance mathématique de X et quelle est sa variance ?
- b- Pendant les six semaines de la saison de récolte, la distribution de probabilité ne varie pas. Calculer l'espérance mathématique et la variance de la variable aléatoire Y donnant la récolte totale en kg durant les six semaines.

# Fondements de probabilités



## Corrigé des exercices d'application

### Corrigé 1 :

Notons les différents événements : Fe : « être femme », Lu : « porter des lunettes », H : « être homme ».

Alors on a  $P(\text{Fe}) = 0.6$ ,  $P(\text{Lu}|\text{Fe}) = 1/3$  ; il s'agit de la probabilité conditionnelle probabilité de « porter des lunettes » sachant que la personne est une femme.

De même, on a  $P(\text{Lu}|\text{H}) = 0.5$ .

On cherche la probabilité conditionnelle  $P(\text{Fe}|\text{Lu})$ .

D'après la formule des probabilités totales on a :  $P(\text{Fe}|\text{Lu}) P(\text{Lu}) = P(\text{Lu}|\text{Fe}) P(\text{Fe})$  avec  $P(\text{Lu}) = P(\text{Lu}|\text{Fe}) P(\text{Fe}) + P(\text{Lu}|\text{H}) P(\text{H})$ .

Application numérique :  $P(\text{Lu}) = 0.4$ , donc  $P(\text{Fe}|\text{Lu}) = P(\text{Lu}|\text{Fe}) P(\text{Fe}) / P(\text{Lu}) = 0.5$ .

### Corrigé 2

Notons A l'événement « la pièce provient de l'atelier 1 », B l'événement « la pièce provient de l'atelier 2 » et D l'événement « la pièce est défectueuse ».

L'énoncé nous dit que les  $2/3$  des pièces produites proviennent de l'atelier 1. Donc  $P(A) = 2/3$ .

On cherche  $P(A \cap D) = P(A)P(D) = 0,03 \times 23 = 150$ . On démontre de même que  $P(B \cap D) = 175$  et donc que  $P(D) = P(A \cap D) + P(B \cap D) = 130$ .

Ainsi, on a  $P(D|A) = P(A \cap D) / P(D) = 35$ .

### Corrigé 3

1- F est croissante, de limite nulle en  $-\infty$ , de limite égale à 1 en  $+\infty$  et continue à droite, il s'agit donc bien d'une fonction de répartition.

2-

$x \in X(\Omega)$	-1	1	7	11
$P(X = x)$	0,29	0,08	0,32	0,31

### Corrigé 4

$$E((1 + X)^{-1}) = \sum_{k=0}^{+\infty} (1 + k)^{-1} e^{-\lambda} \frac{\lambda^k}{k!} = \frac{e^{-\lambda}}{\lambda} \sum_{k=0}^{\infty} \frac{\lambda^{k+1}}{(k+1)!}$$

$$= \frac{e^{-\lambda}}{\lambda} \left( \sum_{k=0}^{+\infty} \frac{\lambda^k}{k!} - 1 \right) = \frac{e^{-\lambda}}{\lambda} (e^{\lambda} - 1) = (1 - e^{-\lambda}) / \lambda$$

# Fondements de probabilités



## Corrigé des exercices d'application

### Corrigé 5

$$\begin{aligned}\text{Var}(X) &= E[(X-E(X))^2] = E(X^2 - 2E(X)X + (E(X))^2) \\ &= E(X^2) - 2E(X)E(X) + (E(X))^2 = E(X^2) - 2(E(X))^2\end{aligned}$$

### Corrigé 6 :

- a-  $E(X) = 0 \times 0,1 + 1 \times 0,5 + 2 \times 0,3 + 3 \times 0,1 = 1,4$  ;  $E(X^2) = 0^2 \times 0,1 + 1^2 \times 0,5 + 2^2 \times 0,3 + 3^2 \times 0,1 = 2,6$  ;  $\text{Var}(X) = E(X^2) - (E(X))^2 = 0,64$ .
- b- Y étant la somme de 6 variables aléatoires. i.i.d. on a :  $E(Y) = 6E(X) = 8,4$  et  $\text{Var}(Y) = 6\text{Var}(X) = 3,84$

# Statistique inférentielle



## Exercices d'application

### Exercice 1 :

Soit  $X$  une variable aléatoire dont la densité de probabilité  $f$  est définie par : où  $\theta$  est un paramètre réel strictement positif.

$$f(x) = \begin{cases} -\frac{1}{\theta} \exp \frac{-x}{\theta}, & x < 0 \\ 0, & x \geq 0 \end{cases}$$

1. Déterminer l'estimateur du maximum de vraisemblance  $\hat{\theta}$  de  $\theta$  d'un  $n$ -échantillon de variable parente  $X$ .
2. Calculer l'espérance mathématique et la variance de  $\hat{\theta}$ .

Que peut-on conclure ?

3. Calculer la quantité d'information de Fisher.
4. En déduire que  $\hat{\theta}$  est efficace.

### Exercice 2 :

Une entreprise fabrique des sacs en plastique pour les enseignes de distribution. Elle s'intéresse au poids maximal que ces sacs peuvent supporter sans se déchirer. On suppose ici que le poids maximal que ces sacs peuvent supporter suit une loi normale d'espérance mathématique 58 Kg et d'écart-type 3 Kg.

1. Sur 200 sacs reçus, une grande enseigne de distribution constate un poids moyen de 57,7 Kg.
  - 1.1. Donner un intervalle de confiance bilatéral de la moyenne des poids sur un échantillon de taille 200, au seuil de risque 1 %.
  - 1.2. Quelle est votre conclusion sur le poids moyen constaté ?
2. Donner le poids moyen dépassé dans 97 % des cas, sur un échantillon de taille 200.



# Statistique inférentielle



## Corrigé des exercices d'application

### Corrigé 1 :

On a :

$$E(X) = \theta \text{ et } V(X) = \theta^2$$

Considérons un r-échantillon de cette structure.

Sa fonction de vraisemblance est définie pour tout  $\theta, \theta > 0$ , et tout  $(x_1, \dots, x_r) \in R^r$ , tous strictement positifs, par :

$$L(\theta ; x_1, \dots, x_r) = \prod_{i=1}^r f(x_i)$$

$$= \frac{1}{\theta^r} \frac{\sum_{i=1}^r x_i}{\theta}$$

D'où :

$$\ln L(\theta ; x_1, \dots, x_r) = -\ln \theta + \frac{\sum_{i=1}^r x_i}{\theta^2}$$

il en résulte que :

$$\frac{\partial}{\partial \theta} \ln L(\theta ; x_1, \dots, x_r) = -\frac{r}{\theta} + \frac{\sum_{i=1}^r x_i}{\theta^2}$$

D'où :

$$\frac{\partial}{\partial \theta} \ln L(\theta ; x_1, \dots, x_r) = 0 \Rightarrow \theta = \frac{1}{r} \sum_{i=1}^r x_i$$

Et comme  $\frac{\partial^2}{\partial \theta^2} \ln L(\theta ; x_1, \dots, x_r) < 0$

Donc l'estimateur du maximum de vraisemblance d'un r-échantillon d'une structure exponentielle est :

$$\hat{\theta} = \frac{1}{r} \sum_{i=1}^r x_i$$

C'est la moyenne empirique du r-échantillon.

# Statistique inférentielle



## Corrigé des exercices d'application

2. Comme

$$\hat{\theta} = \frac{1}{r} \sum_{i=1}^r x_i$$

Alors :

$$E(\hat{\theta}) = E(X) = \theta \quad \text{et} \quad V(\hat{\theta}) = \frac{V(X)}{r} = \frac{\theta^2}{r}$$

On en déduit que  $\hat{\theta}$  est un estimateur sans biais et convergent de  $\theta$ .

3. Calculons la quantité d'information de Fisher,  $I[X, \theta]$ , concernant  $\theta$ .

On a :

$$\begin{aligned} I[X, \theta] &= -E\left[\frac{\partial^2}{\partial \theta^2} \ln f(\theta; X)\right] \\ &= -E\left[\frac{\partial^2}{\partial \theta^2} \left(-\ln \theta - \frac{X}{\theta}\right)\right] \\ &= E\left[-\frac{1}{\theta^2} + \frac{2X}{\theta^3}\right] \\ &= \frac{1}{\theta^2} \end{aligned}$$

Donc la quantité d'information de Fisher,  $I[X_1, \dots, X_r, \theta]$ , concernant  $\theta$  fournie par le  $r$ -échantillon est :

$$I[X_1, \dots, X_r, \theta] = r I[X, \theta] = \frac{r}{\theta^2}$$

Calculons l'efficacité :

$$e[\hat{\theta}] = \frac{1}{I[X_1, \dots, X_r, \theta] V(\hat{\theta})} = 1$$

Donc  $\hat{\theta}$  optimal ou efficace.

# Statistique inférentielle



## Corrigé des exercices d'application

### Corrigé 2 :

1. Ici on travaille sur un échantillon de taille 200 (la taille de la population étant considérée comme infinie).

1.1. La variable aléatoire  $\bar{X}$  égale à la moyenne des poids maximaux sur tout échantillon de taille 200 suit la loi normale  $N(58 ; \frac{3}{\sqrt{200}}) = N(58 ; 0,212)$ .

On cherche  $P(58 - a \leq \bar{X} \leq 58 + a) = 0,99$ . Après avoir posé  $T = \frac{\bar{X} - 58}{0,212}$  et lu sur la table de la loi normale centrée réduite, on obtient  $a = 0,55$ .

Donc l'intervalle de confiance sur tout échantillon de taille 200, de la moyenne des poids, est  $[57,45 ; 58,55]$ .

1.2. Le poids moyen constaté sur l'échantillon ci-dessus est conforme aux attentes (57,7 Kg appartient à l'intervalle).

2. On cherche  $P(\bar{X} > b) = 0,97$  donc après calculs on obtient  $b = 57,6$ .

Donc le poids moyen dépassé dans 97 % des cas est 57,6 Kg.



# Prérequis de l'Économie

## CPP

### Qu'est-ce qu'un modèle de CPP ?

Le modèle de concurrence pure et parfaite CPP stipule que :

- ✓ Les agents économiques sont très nombreux et très petits de sorte que aucun agent ne peut avoir à lui seul une influence sur le marché ;
- ✓ La transparence et la fluidité ;
- ✓ L'information complète ;
- ✓ La liberté d'accès ;

## Optimisation sous contraintes

En économie, le multiplicateur de Lagrange permet de déterminer une situation optimale (par exemple comment maximiser son profit, minimiser ses dépenses, ou encore maximiser bien-être) sous une contrainte quelconque (budget limité, bien-être minimum requis...).

La maximisation d'une fonction  $f(x,y)$  sous la contrainte  $g(x,y)$  revient à maximiser la fonction de Lagrange suivante :

$$L(x,y,\lambda) = f(x,y) + \lambda \times g(x,y)$$

Avec  $\lambda$  est le multiplicateur de Lagrange associé à la contrainte  $g$ .

### Condition nécessaires :

$$\frac{\partial L}{\partial x} = l_x = 0 \quad \frac{\partial L}{\partial y} = l_y = 0 \quad \frac{\partial L}{\partial \lambda} = l_\lambda = 0$$

## Qu'est-ce qu'un panier de biens ?

Un panier de biens est un ensemble composé d'un ou de plusieurs produits. Ce panier peut être préféré à un autre contenant une combinaison différente de biens. En effet Les individus peuvent classer certains paniers de biens en fonction de leurs préférences (goûts).

On définit pour chaque individu une relation de préférence sur les paniers de biens : préféré, non préféré, indifférent.

Exemple : soient 2 paniers de biens A et B : Le consommateur peut les classer du point de vue de la satisfaction qu'ils lui procurent :

$A \sim B$  : il est indifférent entre les deux paniers. Les deux paniers sont donc équivalents pour lui.

$A \succsim B$  : il préfère faiblement A à B.

## Préférence

### Relation de préférence "complète" :

Signifie que pour tous les paniers de consommation A et B, le consommateur est toujours capable de dire s'il préfère A à B ou B à A ou si A et B sont équivalents.

$$\text{Soit } A \succsim B, \text{ Soit } B \succsim A \text{ soit } A \sim B$$

### Relation de préférence "réflexive" :

Signifie qu'un panier est toujours équivalent à lui-même.

$$A \succsim A \Leftrightarrow A \sim A$$

### Relation de préférence "transitive" :

Signifie que si le panier A est préféré ou indifférent au panier B et si le panier B est préféré ou indifférent au panier C, alors le panier A est préféré ou indifférent au panier C.

$$A \succsim B \text{ et } B \succsim C \Leftrightarrow A \succsim C$$

# English

## Lexique Anglais -Francais

Frequency	Fréquence
Intervalle	Étendue
Path	Chemin
Trend	Tendance
Pending...	En Cours...
Skewness	Asymétrie
Mean Or Average	Moyenne
Split	Diviser
Treshold	Seuil
Degree Of Freedom	Degré De Liberté
Dash Board	Tableau De Bord
Tail	Queue
Merge	Fusion
Outlier	Valeur Extrême
Cdf	Fonction De Répartition
Data Set	Ensemble De Données
Estimator	Estimateur
Feature	Caractéristique
Label	Étiquette
Layer	Couche
Mean Squared Error (MSE)	Erreur Quadratique
Metric	Moyenne (MSE)
Parameter	Statistique
Target	Paramètre
	Cible

Application Programming Interface
Decision Tree
Cluster
Data visualisation
Structured And Unstructured Data
Machine Learning
Data Cleansing
Data Mining
Accuracy
Auc (Area Under The Roc Curve)
Baseline
Bias
Binary Classification
Centroid
Checkpoint
Class
Confusion Matrix
Data Analysis
Normalization
Overfitting
Recall
Training
Training Set
Weight

Interface De Programmation
Arbres De Décision
Une Grappe De Serveurs Ou Nœuds
Visualisation Des Données
Données Structurées Et Non-Structurées
Apprentissage Automatique
Nettoyage Des Données
Fouille De Données
Justesse
AUC (Aire Sous La Courbe ROC)
Référence
Biais
Classification Binaire
Centroïde
Point De Contrôle
Classe
Matrice De Confusion
Analyse De Données
Normalisation
Surapprentissage
Rappel
Apprentissage
Ensemble d'apprentissage
Pondération