# Algorithmique, structures informatiques et cryptologie

**Loïck Lhote**
loick.lhote@unicaen.fr

UNIVERSITÉ
CAEN
NORMANDIE

# Introduction à la théorie de l'information

UNIVERSITÉ CAEN NORMANDIE

La théorie des communications étudie les moyens de transmettre des informations d'une source (voix, vidéo, lumière d'étoile, séquence binaire d'un ordinateur,. . . ) à un utilisateur.

Les sources d'information traitées sont génériques du moment qu'elles puissent avoir une représentation numérique.

# Introduction à la théorie de l'information

UNIVERSITÉ CAEN NORMANDIE

La théorie des communications étudie les moyens de transmettre des informations d'une source (voix, vidéo, lumière d'étoile, séquence binaire d'un ordinateur,...) à un utilisateur.

Les sources d'information traitées sont génériques du moment qu'elles puissent avoir une représentation numérique.

**Théorie de l'information** : théorie qui décrit les principes fondamentaux des systèmes de communication.

# Introduction à la théorie de l'information

UNIVERSITÉ CAEN NORMANDIE

La théorie des communications étudie les moyens de transmettre des informations d'une source (voix, vidéo, lumière d'étoile, séquence binaire d'un ordinateur,. . . ) à un utilisateur.

Les sources d'information traitées sont génériques du moment qu'elles puissent avoir une représentation numérique.

**Théorie de l'information** : théorie qui décrit les principes fondamentaux des systèmes de communication.

Discipline créée par Claude Shannon (années 40) qui s'est très largement développée depuis.

**Applications principales** : compression des données, codes correcteurs d'erreurs, cryptographie (un peu), algorithmique, ...

# Introduction à la théorie de l'information

UNIVERSITÉ CAEN NORMANDIE

La théorie des communications étudie les moyens de transmettre des informations d'une source (voix, vidéo, lumière d'étoile, séquence binaire d'un ordinateur,. . . ) à un utilisateur.

Les sources d'information traitées sont génériques du moment qu'elles puissent avoir une représentation numérique.

**Théorie de l'information :** théorie qui décrit les principes fondamentaux des systèmes de communication.

Discipline créée par Claude Shannon (années 40) qui s'est très largement développée depuis.

**Applications principales :** compression des données, codes correcteurs d'erreurs, cryptographie (un peu), algorithmique, ...

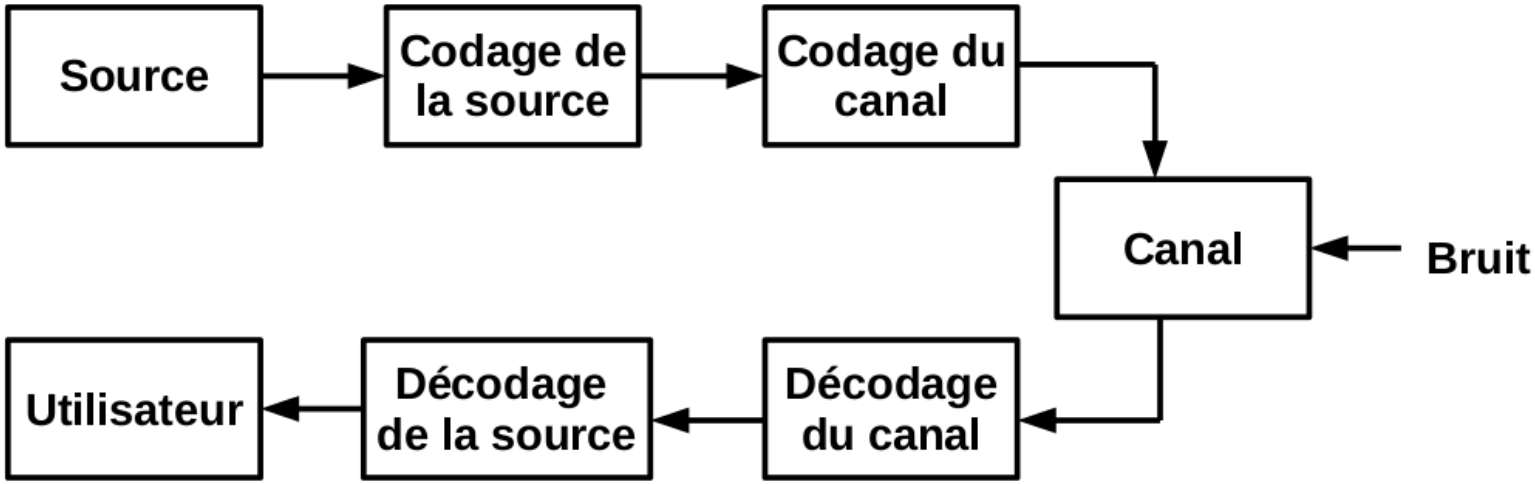
**Organisation du cours :** 4h de cours (et 4h de TP), combinées de manière indépendante avec la partie optimisation.

UNIVERSITÉ
CAEN
NORMANDIE

# 1- Introduction à la Théorie de l'Information

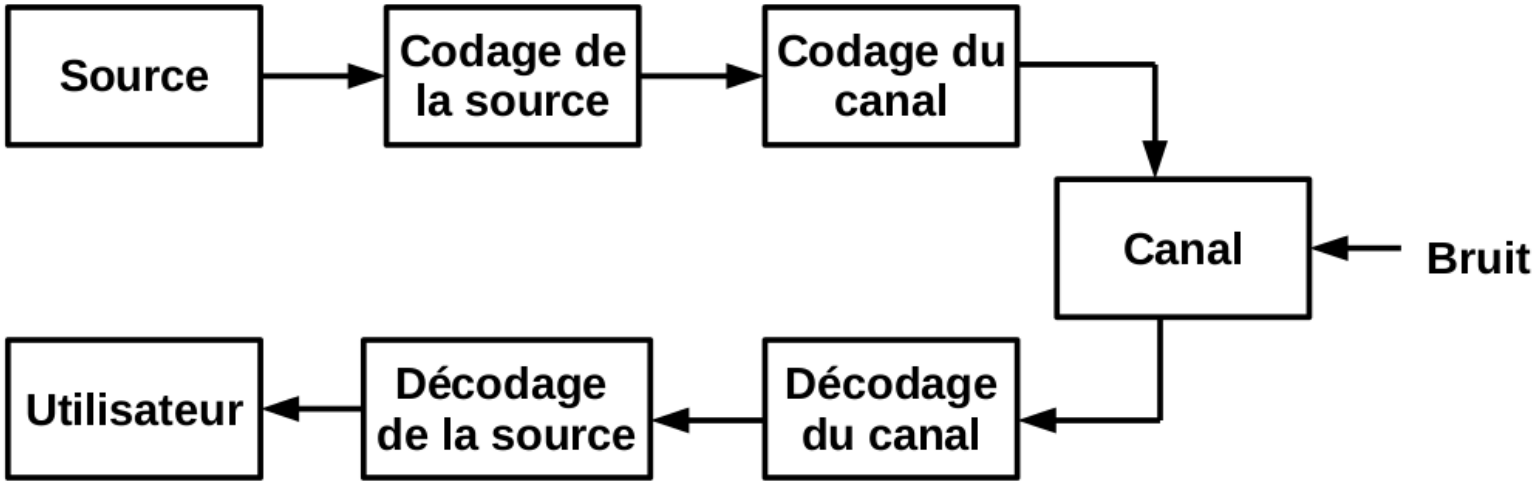
**(Transparents de Patrick Lacharme)**

# Codage de source et codage de canal




Dans ce cours, la source est un processus discret qui émet des symboles sur un alphabet fixé

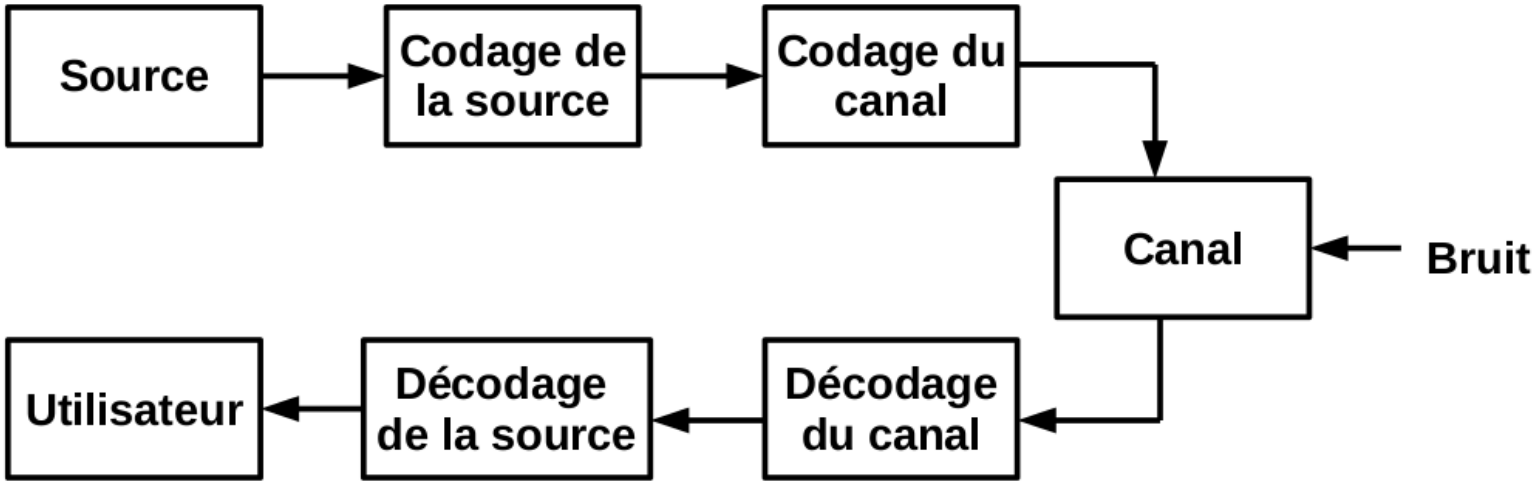
# Codage de source et codage de canal




Dans ce cours, la source est un processus discret qui émet des symboles sur un alphabet fixé

Le codage de la source consiste à transformer efficacement (à priori compresser) une source donnée en tenant compte par exemple de la fréquence des lettres/mots.

# Codage de source et codage de canal

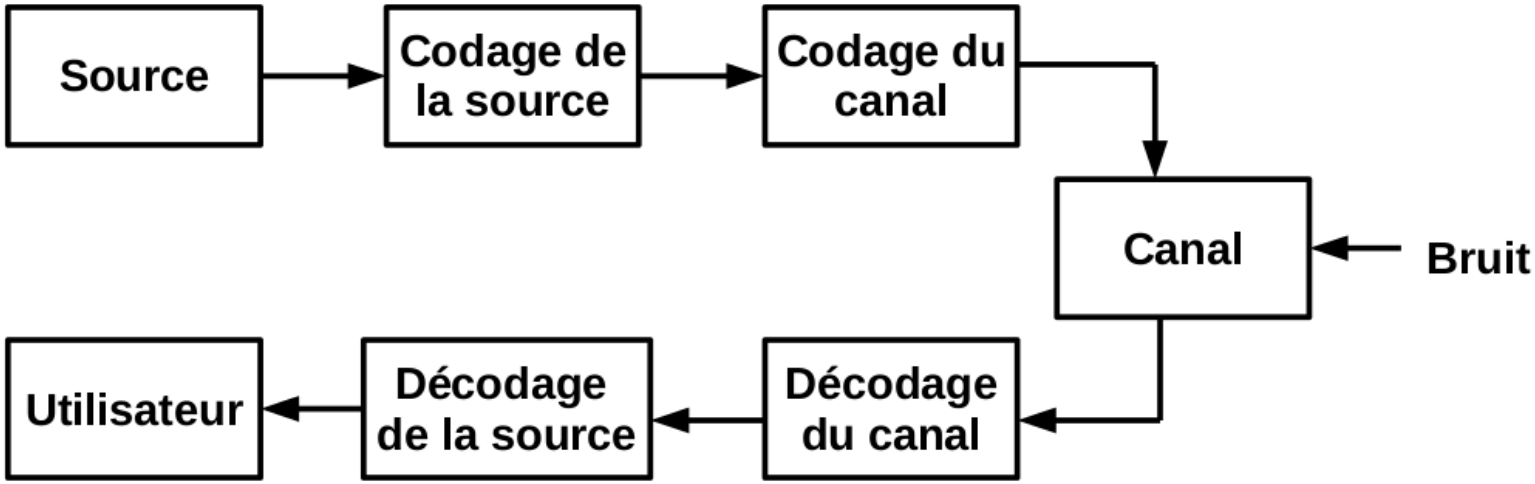



Dans ce cours, la source est un processus discret qui émet des symboles sur un alphabet fixé

Le codage de la source consiste à transformer efficacement (à priori compresser) une source donnée en tenant compte par exemple de la fréquence des lettres/mots.

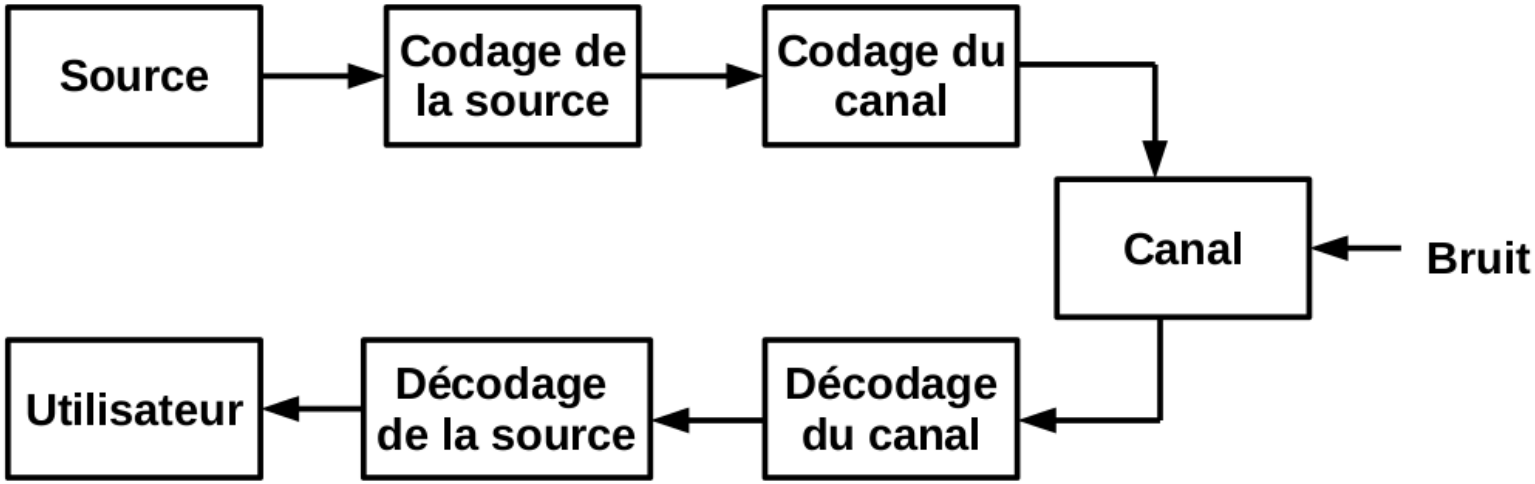
Le codage du canal consiste à transmettre efficacement le plus d'information possible à travers un canal bruité (comportement probabiliste, introduction d'erreurs).

# Codage de source et codage de canal




La séparation source/canal permet une modélisation fine et indépendante de la source et du canal.

# Codage de source et codage de canal




La séparation source/canal permet une modélisation fine et indépendante de la source et du canal.

Remarque : il n'y a pas de cryptographie ici.

# Exemples de codages de sources

Une source émet 4 symboles $\{a, b, c, d\}$. Voici deux codages binaires :

| | $a$ | $b$ | $c$ | $d$ |
|---|---|---|---|---|
| code 1 | 00 | 01 | 10 | 11 |
| code 2 | 1 | 01 | 001 | 000 |

Quelle est le nombre moyen de bits nécessaires pour encoder un symbole

1. si $\mathbb{P}[a] = \mathbb{P}[b] = \mathbb{P}[c] = \mathbb{P}[d] = \frac{1}{4}$ ?

# Exemples de codages de sources

Une source émet 4 symboles $\{a, b, c, d\}$. Voici deux codages binaires :

| | $a$ | $b$ | $c$ | $d$ |
|---|---|---|---|---|
| code 1 | 00 | 01 | 10 | 11 |
| code 2 | 1 | 01 | 001 | 000 |

Quelle est le nombre moyen de bits nécessaires pour encoder un symbole

1. si $\mathbb{P}[a] = \mathbb{P}[b] = \mathbb{P}[c] = \mathbb{P}[d] = \frac{1}{4}$ ?

$$\text{longueur moyenne code 1} \quad = \quad 4 \times \frac{1}{4} \times 2 = 2$$

# Exemples de codages de sources

Une source émet 4 symboles $\{a, b, c, d\}$. Voici deux codages binaires :

| | $a$ | $b$ | $c$ | $d$ |
|---|---|---|---|---|
| code 1 | 00 | 01 | 10 | 11 |
| code 2 | 1 | 01 | 001 | 000 |

Quelle est le nombre moyen de bits nécessaires pour encoder un symbole

1. si $\mathbb{P}[a] = \mathbb{P}[b] = \mathbb{P}[c] = \mathbb{P}[d] = \frac{1}{4}$ ?

$$\begin{aligned} \text{longueur moyenne code 1} &= 4 \times \frac{1}{4} \times 2 = 2 \\ \text{longueur moyenne code 2} &= \frac{1}{4} \times (1+2+3+3) = \frac{9}{4} > 2 \end{aligned}$$

# Exemples de codages de sources

Une source émet 4 symboles $\{a, b, c, d\}$. Voici deux codages binaires :

| | $a$ | $b$ | $c$ | $d$ |
|---|---|---|---|---|
| code 1 | 00 | 01 | 10 | 11 |
| code 2 | 1 | 01 | 001 | 000 |

Quelle est le nombre moyen de bits nécessaires pour encoder un symbole

1. si $\mathbb{P}[a] = \mathbb{P}[b] = \mathbb{P}[c] = \mathbb{P}[d] = \frac{1}{4}$ ?

$$\begin{aligned} \text{longueur moyenne code 1} &= 4 \times \frac{1}{4} \times 2 = 2 \\ \text{longueur moyenne code 2} &= \frac{1}{4} \times (1+2+3+3) = \frac{9}{4} > 2 \end{aligned}$$

2. si $\mathbb{P}[a] = \frac{1}{2},\ \mathbb{P}[b] = \frac{1}{4},\ \mathbb{P}[c] = \mathbb{P}[d] = \frac{1}{8}$ ?

# Exemples de codages de sources

Une source émet 4 symboles $\{a, b, c, d\}$. Voici deux codages binaires :

| | $a$ | $b$ | $c$ | $d$ |
|---|---|---|---|---|
| code 1 | 00 | 01 | 10 | 11 |
| code 2 | 1 | 01 | 001 | 000 |

Quelle est le nombre moyen de bits nécessaires pour encoder un symbole

1. si $\mathbb{P}[a] = \mathbb{P}[b] = \mathbb{P}[c] = \mathbb{P}[d] = \frac{1}{4}$ ?

$$\text{longueur moyenne code 1} = 4 \times \frac{1}{4} \times 2 = 2$$

$$\text{longueur moyenne code 2} = \frac{1}{4} \times (1 + 2 + 3 + 3) = \frac{9}{4} > 2$$

2. si $\mathbb{P}[a] = \frac{1}{2},\ \mathbb{P}[b] = \frac{1}{4},\ \mathbb{P}[c] = \mathbb{P}[d] = \frac{1}{8}$ ?

$$\text{longueur moyenne code 1} = 2 \times \left(\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{8}\right) = 2$$

# Exemples de codages de sources

Une source émet 4 symboles $\{a, b, c, d\}$. Voici deux codages binaires :

| | $a$ | $b$ | $c$ | $d$ |
|---|---|---|---|---|
| code 1 | 00 | 01 | 10 | 11 |
| code 2 | 1 | 01 | 001 | 000 |

Quelle est le nombre moyen de bits nécessaires pour encoder un symbole

1. si $\mathbb{P}[a] = \mathbb{P}[b] = \mathbb{P}[c] = \mathbb{P}[d] = \frac{1}{4}$ ?

$$\text{longueur moyenne code 1} = 4 \times \frac{1}{4} \times 2 = 2$$

$$\text{longueur moyenne code 2} = \frac{1}{4} \times (1 + 2 + 3 + 3) = \frac{9}{4} > 2$$

2. si $\mathbb{P}[a] = \frac{1}{2},\ \mathbb{P}[b] = \frac{1}{4},\ \mathbb{P}[c] = \mathbb{P}[d] = \frac{1}{8}$ ?

$$\text{longueur moyenne code 1} = 2 \times \left(\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{8}\right) = 2$$

$$\text{longueur moyenne code 2} = 1 \cdot \frac{1}{2} + 2 \cdot \frac{1}{4} + 3 \cdot \frac{1}{8} + 3 \cdot \frac{1}{8} = \frac{7}{4} < 2$$

# Exemples de codages de sources

UNIVERSITÉ CAEN NORMANDIE

## Point important

Un codage efficace d'une source doit tenir compte des propriétés statistiques de la source.

# Exemples de codages de sources

UNIVERSITÉ CAEN NORMANDIE

## Point important

Un codage efficace d'une source doit tenir compte des propriétés statistiques de la source.

Question :

- Pour une source donnée, quel est le codage optimal ?
- Peut-on construire un codage optimal d'une source ?
- Y-a-t'il des limites qui ne peuvent être dépassées ?
- Efficacité des algorithmes ?

# Exemples de codages de sources

UNIVERSITÉ CAEN NORMANDIE

## Point important

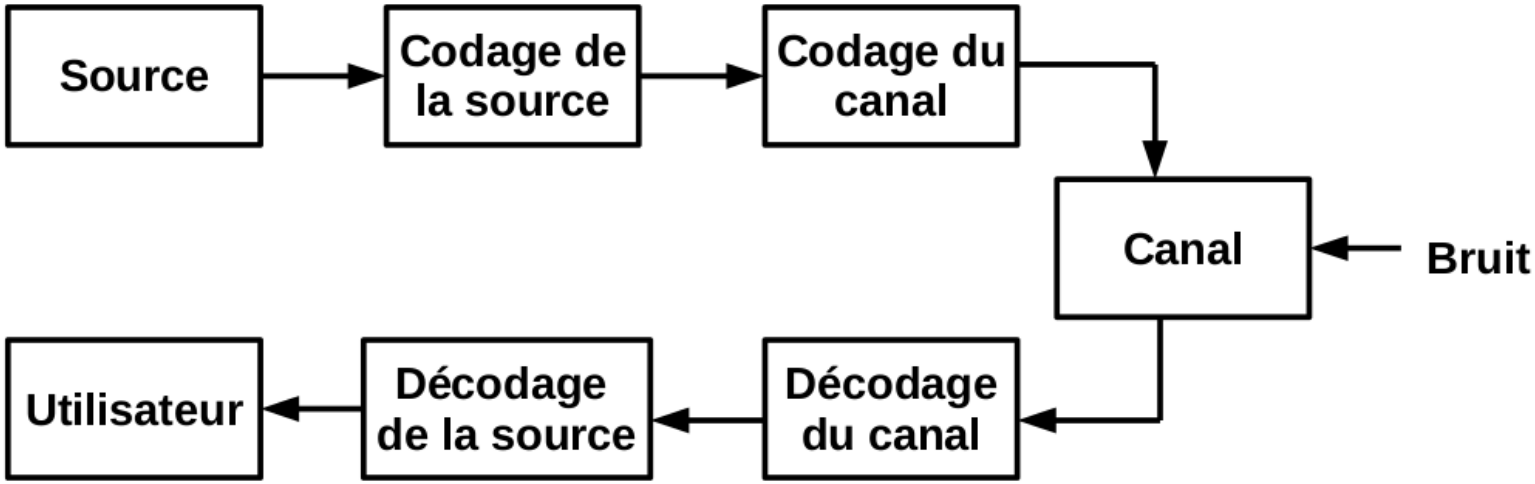
Un codage efficace d'une source doit tenir compte des propriétés statistiques de la source.

Question :

- Pour une source donnée, quel est le codage optimal ?
- Peut-on construire un codage optimal d'une source ?
- Y-a-t'il des limites qui ne peuvent être dépassées ?
- Efficacité des algorithmes ?

$\rightarrow$ réponses données grâce à la Théorie de l'Information.

# Codage de source et codage de canal




La séparation source/canal permet une modélisation fine et indépendante de la source et du canal.

# Exemples de codages de canaux discrets

UNIVERSITÉ CAEN NORMANDIE

Un canal discret sans-mémoire est défini par

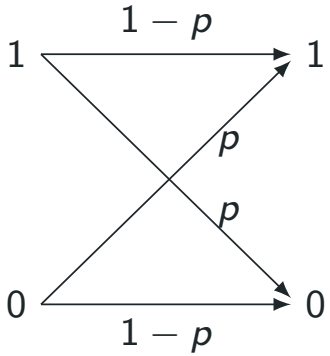
- un alphabet d'entrée $X = \{a_1, \ldots, a_K\}$
- un alphabet de sortie $Y = \{b_1, \ldots, b_J\}$ (on peut avoir $J \neq K$)
- une loi de transition définie par les probabilités conditionnelles $\mathbb{P}(b_j|a_k)$ (matrice stochastique de transition)

# Exemples de codages de canaux discrets

Un canal discret sans-mémoire est défini par

- un alphabet d'entrée $X = \{a_1, \ldots, a_K\}$
- un alphabet de sortie $Y = \{b_1, \ldots, b_J\}$ (on peut avoir $J \neq K$)
- une loi de transition définie par les probabilités conditionnelles $\mathbb{P}(b_j|a_k)$ (matrice stochastique de transition)

Canal binaire symétrique




Matrice stochastique du canal

$$\Pi = \begin{pmatrix} 1-p & p \\ p & 1-p \end{pmatrix}$$

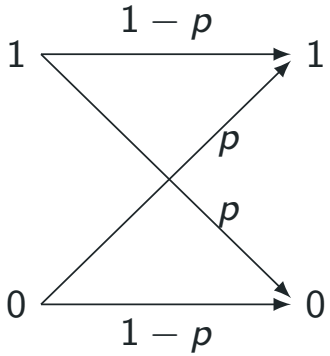
**Remarques :** Si $p$ est "faible", il est "peu probable" d'avoir beaucoup d'erreurs

# Exemples de codages de canaux discrets

Un canal discret sans-mémoire est défini par

- un alphabet d'entrée $X = \{a_1, \ldots, a_K\}$
- un alphabet de sortie $Y = \{b_1, \ldots, b_J\}$ (on peut avoir $J \neq K$)
- une loi de transition définie par les probabilités conditionnelles $\mathbb{P}(b_j|a_k)$ (matrice stochastique de transition)

Canal binaire symétrique




Matrice stochastique du canal

$$\Pi = \begin{pmatrix} 1-p & p \\ p & 1-p \end{pmatrix}$$

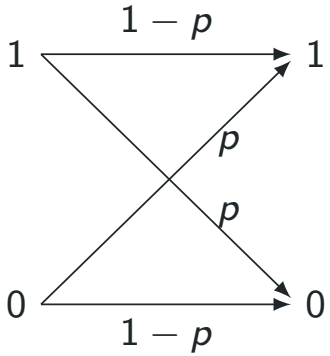
**Remarques :** Si $p$ est "faible", il est "peu probable" d'avoir beaucoup d'erreurs
$\rightarrow$ comment détecter ou corriger des erreurs ?

# Exemples de codages de canaux discrets

Un canal discret sans-mémoire est défini par

- un alphabet d'entrée $X = \{a_1, \ldots, a_K\}$
- un alphabet de sortie $Y = \{b_1, \ldots, b_J\}$ (on peut avoir $J \neq K$)
- une loi de transition définie par les probabilités conditionnelles $\mathbb{P}(b_j|a_k)$ (matrice stochastique de transition)

Canal binaire symétrique



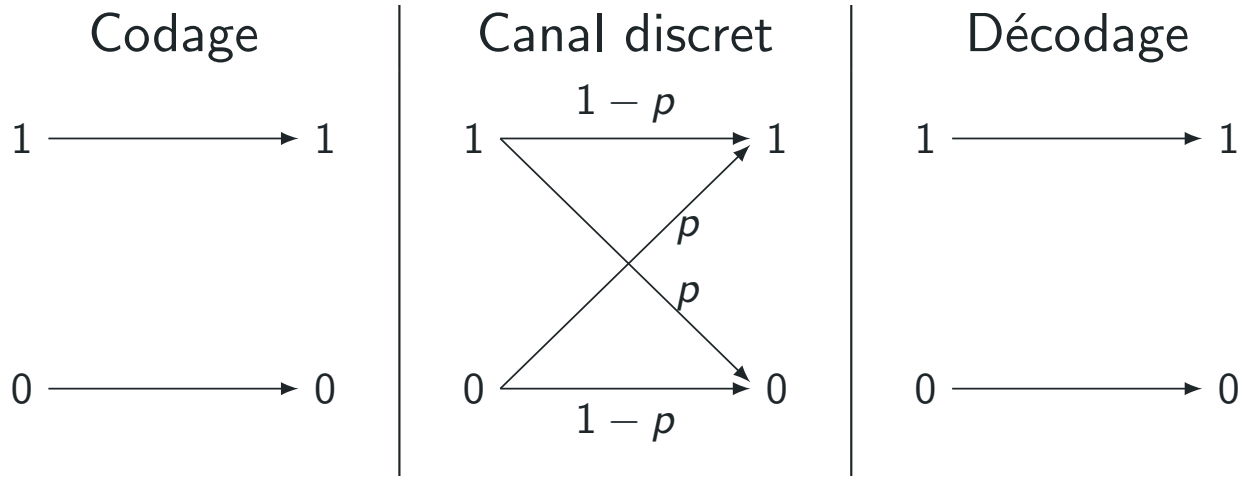

Matrice stochastique du canal

$$\Pi = \begin{pmatrix} 1-p & p \\ p & 1-p \end{pmatrix}$$

**Remarques :** Si $p$ est "faible", il est "peu probable" d'avoir beaucoup d'erreurs
$\rightarrow$ comment détecter ou corriger des erreurs ?
$\rightarrow$ comment optimiser le transfert d'informations à travers ce canal ?

# Exemples de codages de canaux discrets

On utilise des codes pour détecter/corriger des erreurs




- Ce code ne permet pas de corriger/détecter des erreurs ! ! !

# Exemples de codages de canaux discrets

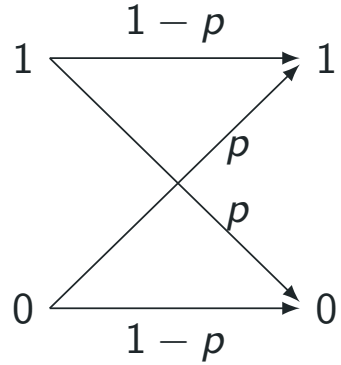
On utilise des codes pour détecter/corriger des erreurs

**Codage**

$1 \longrightarrow 111$

$0 \longrightarrow 000$

**Canal discret ($p < 0.5$)**



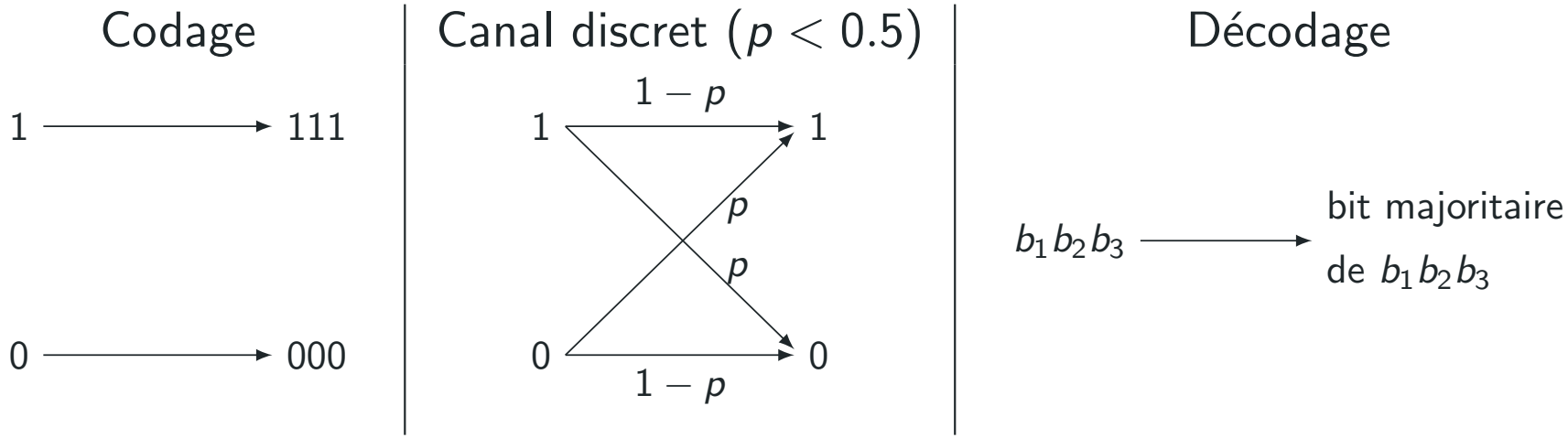

**Décodage**

$b_1 b_2 b_3 \longrightarrow$ bit majoritaire de $b_1 b_2 b_3$

- On applique le canal discret sur chaque bit du codage (canal de $\{0,1\}^3 \to \{0,1\}^3$)

# Exemples de codages de canaux discrets

On utilise des codes pour détecter/corriger des erreurs




- On applique le canal discret sur chaque bit du codage (canal de $\{0,1\}^3 \to \{0,1\}^3$)
- Ce code permet de corriger jusqu'à un bit d'erreur par mot de code

# Exemples de codages de canaux discrets

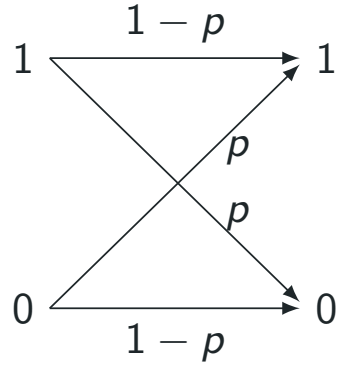
On utilise des codes pour détecter/corriger des erreurs

**Codage**

$1 \longrightarrow 111$

$0 \longrightarrow 000$

**Canal discret ($p < 0.5$)**




**Décodage**

$b_1 b_2 b_3 \longrightarrow$ bit majoritaire de $b_1 b_2 b_3$

- On applique le canal discret sur chaque bit du codage (canal de $\{0,1\}^3 \to \{0,1\}^3$)
- Ce code permet de corriger jusqu'à un bit d'erreur par mot de code
- Le processus étant probabiliste, on n'est jamais certain à 100% du décodage

UNIVERSITÉ
CAEN
NORMANDIE

# 2- Notions d'entropies

# Entropie (au sens de Shannon) d'une source

Soit $X$ une variable aléatoire discrète (**la source**) à valeurs dans un alphabet $A = \{x_1, \ldots, x_n\}$, de distribution de probabilité $p(x_i) = P(X = x_i)$.

**Exemple :** $A = \{0, 1\}$, $p(0) = \mathbb{P}[X = 0] = \frac{1}{3}$, $p(1) = \mathbb{P}[X = 1] = \frac{2}{3}$.

# Entropie (au sens de Shannon) d'une source

Soit $X$ une variable aléatoire discrète (**la source**) à valeurs dans un alphabet $A = \{x_1, \ldots, x_n\}$, de distribution de probabilité $p(x_i) = P(X = x_i)$.

**Exemple :** $A = \{0, 1\}$, $p(0) = \mathbb{P}[X = 0] = \frac{1}{3}$, $p(1) = \mathbb{P}[X = 1] = \frac{2}{3}$.

## Entropie de Shannon

L'**entropie** (au sens de Shannon) de la variable aléatoire $X$ est définie par :

$$H(X) = -\sum_{x \in A} p(x) \log_2 p(x) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i).$$

*Convention :* on utilise la convention $0 \log_2 0 = 0$.

# Entropie (au sens de Shannon) d'une source

Soit $X$ une variable aléatoire discrète (**la source**) à valeurs dans un alphabet $A = \{x_1, \ldots, x_n\}$, de distribution de probabilité $p(x_i) = P(X = x_i)$.

**Exemple :** $A = \{0, 1\}$, $p(0) = \mathbb{P}[X = 0] = \frac{1}{3}$, $p(1) = \mathbb{P}[X = 1] = \frac{2}{3}$.

## Entropie de Shannon

L'**entropie** (au sens de Shannon) de la variable aléatoire $X$ est définie par :

$$H(X) = -\sum_{x \in A} p(x) \log_2 p(x) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i).$$

*Convention :* on utilise la convention $0 \log_2 0 = 0$.

**Exemple :** L'entropie de $X$ est

$$H(X) = -\frac{1}{3} \log_2(\frac{1}{3}) - \frac{2}{3} \log_2(\frac{2}{3}) \simeq 0.917 \text{ bits.}$$

# Entropie (au sens de Shannon) d'une source

- L'entropie mesure l'incertitude d'une source.
- Plus l'entropie est grande, plus l'incertitude est grande.

# Entropie (au sens de Shannon) d'une source

UNIVERSITÉ CAEN NORMANDIE

- L'entropie mesure l'incertitude d'une source.
- Plus l'entropie est grande, plus l'incertitude est grande.

## Propriétés

Soit $X$ une variable aléatoire binaire sur un alphabet $A = \{x_1, \ldots, x_n\}$, à $n$ valeurs. Alors on a :

$$0 \leq H(X) \leq \log_2 n.$$

De plus, $H(X) = 0$ si et seulement si $X$ est constante et $H(X) = \log_2 n$ si et seulement si $X$ est uniformément distribuée.

# Entropie (au sens de Shannon) d'une source

- L'entropie mesure l'incertitude d'une source.
- Plus l'entropie est grande, plus l'incertitude est grande.

## Propriétés

Soit $X$ une variable aléatoire binaire sur un alphabet $A = \{x_1, \ldots, x_n\}$, à $n$ valeurs. Alors on a :

$$0 \leq H(X) \leq \log_2 n.$$

De plus, $H(X) = 0$ si et seulement si $X$ est constante et $H(X) = \log_2 n$ si et seulement si $X$ est uniformément distribuée.

Pour une source binaire $X$ sur $A = \{0, 1\}$

- si $p(0) = 1 - p(1) = 1$ (source certaine), alors $H(X) = -p(0)\log_2 p(0) - p(1)\log_2 p(1) = 0$

# Entropie (au sens de Shannon) d'une source

UNIVERSITÉ CAEN NORMANDIE

- L'entropie mesure l'incertitude d'une source.
- Plus l'entropie est grande, plus l'incertitude est grande.

## Propriétés

Soit $X$ une variable aléatoire binaire sur un alphabet $A = \{x_1, \ldots, x_n\}$, à $n$ valeurs. Alors on a :

$$0 \leq H(X) \leq \log_2 n.$$

De plus, $H(X) = 0$ si et seulement si $X$ est constante et $H(X) = \log_2 n$ si et seulement si $X$ est uniformément distribuée.

Pour une source binaire $X$ sur $A = \{0, 1\}$

- si $p(0) = 1 - p(1) = 1$ (source certaine), alors $H(X) = -p(0)\log_2 p(0) - p(1)\log_2 p(1) = 0$
- si $p(0) = p(1) = \frac{1}{2}$ (source uniforme), alors $H(X) = -p(0)\log_2 p(0) - p(1)\log_2 p(1) = 1 = \log_2 2$.

# Entropie (au sens de Shannon) d'une source

- L'entropie mesure l'incertitude d'une source.
- Plus l'entropie est grande, plus l'incertitude est grande.

# Entropie (au sens de Shannon) d'une source

UNIVERSITÉ CAEN NORMANDIE

- L'entropie mesure l'incertitude d'une source.
- Plus l'entropie est grande, plus l'incertitude est grande.

**Remarque fondamentale**

L'entropie mesure aussi l'information moyenne d'un symbole émit par la source

$\rightarrow$ Plus l'entropie est grande, plus il faudra de bits (en moyenne) pour coder un symbole.

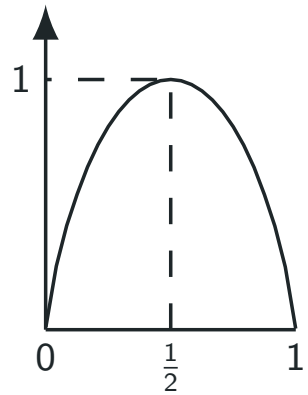
# Entropie (au sens de Shannon) d'une source

- L'entropie mesure l'incertitude d'une source.
- Plus l'entropie est grande, plus l'incertitude est grande.

## Remarque fondamentale

L'entropie mesure aussi l'information moyenne d'un symbole émit par la source

$\rightarrow$ Plus l'entropie est grande, plus il faudra de bits (en moyenne) pour coder un symbole.




- Ci-contre, l'entropie d'une source binaire sans mémoire en fonction de $p \in [0, 1]$

$H(X) = -p \log_2 p - (1-p) \log_2 (1-p)$

# Entropie (au sens de Shannon) d'une source

UNIVERSITÉ CAEN NORMANDIE

**Application 1 : compression**

Pour toute source discrète sans mémoire $X$ d'entropie $H(X)$ codée en binaire au moyen d'un code de longueur moyenne $\overline{m}$, on a

$$H(X) \leq \overline{m}.$$

Autrement dit, on ne peut pas compresser en dessous de l'entropie sans riquer de perdre de l'information.

# Entropie (au sens de Shannon) d'une source

UNIVERSITÉ CAEN NORMANDIE

**Application 1 : compression**

Pour toute source discrète sans mémoire $X$ d'entropie $H(X)$ codée en binaire au moyen d'un code de longueur moyenne $\overline{m}$, on a

$$H(X) \leq \overline{m}.$$

Autrement dit, on ne peut pas compresser en dessous de l'entropie sans riquer de perdre de l'information.

On peut s'approcher autant que souhaité de l'entropie en regardant des blocs de symboles (voir partie codage)

# Entropie (au sens de Shannon) d'une source

UNIVERSITÉ CAEN NORMANDIE

**Application 1 : compression**

Pour toute source discrète sans mémoire $X$ d'entropie $H(X)$ codée en binaire au moyen d'un code de longueur moyenne $\overline{m}$, on a

$$H(X) \leq \overline{m}.$$

Autrement dit, on ne peut pas compresser en dessous de l'entropie sans riquer de perdre de l'information.

On peut s'approcher autant que souhaité de l'entropie en regardant des blocs de symboles (voir partie codage)

**Application 2 : transmission**

L'entropie représente la quantité moyenne d'information par symbole

$\rightarrow$ la capacité d'un canal doit être suffisante pour transmettre cette information

$\rightarrow$ donne une borne sur la capacité minimale d'un canal pour une source

UNIVERSITÉ
CAEN
NORMANDIE

# 3- Codage d'une source discrète

# Exemple de codage d'une source : le code ASCII

UNIVERSITÉ CAEN NORMANDIE

| Lettres maj. | ASCII | Lettres maj. | ASCII |
|---|---|---|---|
| A | 41 | N | 4E |
| B | 42 | O | 4F |
| C | 43 | P | 50 |
| D | 44 | Q | 51 |
| E | 45 | R | 52 |
| F | 46 | S | 53 |
| G | 47 | T | 54 |
| H | 48 | U | 55 |
| I | 49 | V | 56 |
| J | 4A | W | 57 |
| K | 4B | X | 58 |
| L | 4C | Y | 59 |
| M | 4D | Z | 5A |

Le code ASCII (7 bits pour les caractères anglosaxons, sinon 8 bits) n'est pas optimal car toutes les lettres sont codées par un nombre identique de symboles.

# Exemple de codage d'une source : le code Morse

| Lettres | Morse | Lettres | Morse |
|---|---|---|---|
| A | ·— | N | —· |
| B | — ··· | O | — — — |
| C | — · —· | P | · — —· |
| D | — ·· | Q | — — ·— |
| E | · | R | · — · |
| F | ·· —· | S | ··· |
| G | — — · | T | — |
| H | ···· | U | ·· — |
| I | ·· | V | ···— |
| J | · — ——— | W | · — — |
| K | — · — | X | — ··— |
| L | · — ·· | Y | — · —— |
| M | —— | Z | — — ·· |

**Remarque :** bien qu'apparemment chaque lettre de l'alphabet soit codée par une combinaison de deux symboles (. et −), un troisième symbole (l'espace) est nécessaire pour pouvoir lire le message sans ambiguïté.

# Codes et codage d'une source discrète

UNIVERSITÉ CAEN NORMANDIE

## Code

Soit $X$ une variable aléatoire (la source) sur un alphabet $A$ fini. Un **code** binaire pour $A$ est une application $\phi$ de $A$ vers $\{0,1\}^*$.
Les éléments de $\phi(A)$ sont appelés **mots de code**.

Quelle propriété simple doit satisfaire un code pour être utile ?

# Codes et codage d'une source discrète

UNIVERSITÉ CAEN NORMANDIE

## Code

Soit $X$ une variable aléatoire (la source) sur un alphabet $A$ fini. Un **code** binaire pour $A$ est une application $\phi$ de $A$ vers $\{0,1\}^*$.
Les éléments de $\phi(A)$ sont appelés **mots de code**.

Quelle propriété simple doit satisfaire un code pour être utile ?

## Code régulier

Un code $\phi$ est **régulier** si $\phi$ est injective (deux symboles différents sont codés par deux mots de code différents).

# Codes et codage d'une source discrète

UNIVERSITÉ CAEN NORMANDIE

## Code

Soit $X$ une variable aléatoire (la source) sur un alphabet $A$ fini. Un **code** binaire pour $A$ est une application $\phi$ de $A$ vers $\{0,1\}^*$.
Les éléments de $\phi(A)$ sont appelés **mots de code**.

Quelle propriété simple doit satisfaire un code pour être utile ?

## Code régulier

Un code $\phi$ est **régulier** si $\phi$ est injective (deux symboles différents sont codés par deux mots de code différents).

## Codage

Un **codage** binaire $\psi$ pour l'alphabet $A$ est une application de $A^*$ vers $\{0,1\}^*$.

**Exemple :** étant donné un code $\phi : A \to \{0,1\}^*$, la concaténation est un codage possible (il en existe d'autres - modes opératoires) :

$$\psi : a_1 \parallel \ldots \parallel a_n \to \phi(a_1) \parallel \ldots \parallel \phi(a_n).$$

# Longueur moyenne et efficacité d'un code

UNIVERSITÉ CAEN NORMANDIE

Soit $X$ une v.a. (la source) sur l'alphabet $\{x_1, \ldots, x_n\}$, où $p(x_i)$ est leur probabilité d'occurrence.

## Longueur moyenne d'un code

La **longueur moyenne $\ell$ d'un code** $\phi$ est définie par

$$\ell(\phi) = \sum_{i=1}^{n} \ell(\phi(x_i))p(x_i),$$

où $\ell(\phi(x_i))$ est la longueur du mot de code $\phi(x_i)$.

Remarque : cette longueur est en bits si le code est binaire.

# Longueur moyenne et efficacité d'un code

UNIVERSITÉ CAEN NORMANDIE

Soit $X$ une v.a. (la source) sur l'alphabet $\{x_1, \ldots, x_n\}$, où $p(x_i)$ est leur probabilité d'occurrence.

## Longueur moyenne d'un code

La **longueur moyenne $\ell$ d'un code** $\phi$ est définie par

$$\ell(\phi) = \sum_{i=1}^{n} \ell(\phi(x_i))p(x_i),$$

où $\ell(\phi(x_i))$ est la longueur du mot de code $\phi(x_i)$.

Remarque : cette longueur est en bits si le code est binaire.

## Efficacité d'un code

L'**efficacité d'un code** $\phi$ est définie par

$$E(\phi) = \frac{H(X)}{\ell(\phi)}.$$

# Longueur moyenne et efficacité d'un code

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i)$$

$$\ell(\phi) = \sum_{i=1}^{n} \ell(\phi(x_i)) p(x_i), \qquad E(\phi) = \frac{H(X)}{\ell(\phi)}$$

**Exercice :** on considère une source $X$ sur l'alphabet $\{A, \ldots, Z\}$, muni de la distribution de probabilité uniforme et $\phi$ le code ASCII codé sur un octet.

Montrer que la longueur moyenne de $\phi$, $\ell(\phi)$,vaut 8 bits.
Montrer que l'efficacité de $\phi$, $E(\phi)$, vaut $\simeq 0.58$.

# Longueur moyenne et efficacité d'un code

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i)$$

$$\ell(\phi) = \sum_{i=1}^{n} \ell(\phi(x_i)) p(x_i), \qquad E(\phi) = \frac{H(X)}{\ell(\phi)}$$

**Exercice :** on considère une source $X$ sur l'alphabet $\{A, \ldots, Z\}$, muni de la distribution de probabilité uniforme et $\phi$ le code ASCII codé sur un octet.

Montrer que la longueur moyenne de $\phi$, $\ell(\phi)$,vaut 8 bits.
Montrer que l'efficacité de $\phi$, $E(\phi)$, vaut $\simeq 0.58$.

Pourquoi l'efficacité est elle si basse ?

# Longueur moyenne et efficacité d'un code

UNIVERSITÉ CAEN NORMANDIE

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i)$$

$$\ell(\phi) = \sum_{i=1}^{n} \ell(\phi(x_i)) p(x_i), \qquad E(\phi) = \frac{H(X)}{\ell(\phi)}$$

**Exercice :** on considère une source $X$ sur l'alphabet $\{A, \ldots, Z\}$, muni de la distribution de probabilité uniforme et $\phi$ le code ASCII codé sur un octet.

Montrer que la longueur moyenne de $\phi$, $\ell(\phi)$,vaut 8 bits.
Montrer que l'efficacité de $\phi$, $E(\phi)$, vaut $\simeq 0.58$.

Pourquoi l'efficacité est elle si basse ?
Trop de bits sont utilisés pour encoder une lettre ! 5 bits suffisent...

# Codes de longueur fixe

UNIVERSITÉ CAEN NORMANDIE

## Code de longueur fixe

Un **code de longueur fixe** est un code où tous les mots de code ont une même longueur $n$.

# Codes de longueur fixe

UNIVERSITÉ CAEN NORMANDIE

## Code de longueur fixe

Un **code de longueur fixe** est un code où tous les mots de code ont une même longueur $n$.

## Théorème

Pour toute source d'alphabet $\{x_1, \ldots, x_K\}$ de cardinal $K$, il existe un code régulier de longueur $n$ vérifiant

$$\log_2 K \leq n < 1 + \log_2 K.$$

**Remarque :** le code étant régulier, on a nécessairement $K \leq 2^n$.

# Codes de longueur fixe

UNIVERSITÉ CAEN NORMANDIE

## Code de longueur fixe

Un **code de longueur fixe** est un code où tous les mots de code ont une même longueur $n$.

## Théorème

Pour toute source d'alphabet $\{x_1, \ldots, x_K\}$ de cardinal $K$, il existe un code régulier de longueur $n$ vérifiant

$$\log_2 K \leq n < 1 + \log_2 K.$$

**Remarque :** le code étant régulier, on a nécessairement $K \leq 2^n$.

**Preuve :** soit $n$ le plus petit entier vérifiant $2^{n-1} < K \leq 2^n$. Il suffit de coder chaque lettre $x_i$ de l'alphabet (où $i = 1, \ldots, K$) par l'écriture de $i$ en base 2 (code de longueur fixe).

# Codes de longueur fixe

UNIVERSITÉ CAEN NORMANDIE

## Code de longueur fixe

Un **code de longueur fixe** est un code où tous les mots de code ont une même longueur $n$.

## Théorème

Pour toute source d'alphabet $\{x_1, \ldots, x_K\}$ de cardinal $K$, il existe un code régulier de longueur $n$ vérifiant

$$\log_2 K \leq n < 1 + \log_2 K.$$

**Remarque :** le code étant régulier, on a nécessairement $K \leq 2^n$.

**Preuve :** soit $n$ le plus petit entier vérifiant $2^{n-1} < K \leq 2^n$. Il suffit de coder chaque lettre $x_i$ de l'alphabet (où $i = 1, \ldots, K$) par l'écriture de $i$ en base 2 (code de longueur fixe).

**Conséquence :** Soit $\phi$ un code fixe de longueur $n$ pour la source $X$ de cardinal $K$. Comme $H(X) \leq \log_2 K$, l'efficacité de $\phi$ est $E(\phi) = H(X)/n \leq 1$. On a $E(\phi) = 1$ si et seulement si $H(X) = n = \log_2 K$ ($X$ suit une loi uniforme sur son alphabet).

# Exemple : le codage des chiffres

UNIVERSITÉ CAEN NORMANDIE

- Soit $X$ la source dont l'alphabet est $\{0, \ldots, 9\}$, muni de la loi uniforme.

Considérons le code de longueur fixe 4 suivant :

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| 0000 | 0001 | 0010 | 0011 | 0100 | 0101 | 0110 | 0111 | 1000 | 1001 |

L'efficacité de ce code est

# Exemple : le codage des chiffres

UNIVERSITÉ CAEN NORMANDIE

- Soit $X$ la source dont l'alphabet est $\{0, \ldots, 9\}$, muni de la loi uniforme.

Considérons le code de longueur fixe 4 suivant :

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| 0000 | 0001 | 0010 | 0011 | 0100 | 0101 | 0110 | 0111 | 1000 | 1001 |

L'efficacité de ce code est $H(X)/4 = (\log_2 10)/4 \simeq 0.83$.

# Exemple : le codage des chiffres

UNIVERSITÉ CAEN NORMANDIE

■ Soit $X$ la source dont l'alphabet est $\{0, \ldots, 9\}$, muni de la loi uniforme. Considérons le code de longueur fixe 4 suivant :

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| 0000 | 0001 | 0010 | 0011 | 0100 | 0101 | 0110 | 0111 | 1000 | 1001 |

L'efficacité de ce code est $H(X)/4 = (\log_2 10)/4 \simeq 0.83$.

■ On considère la source $X^2$ sur l'alphabet $\{00, 01, \ldots, 99\}$ de cardinal 100. Son entropie est $H(X^2) = \log_2(100) = 2H(X)$.

# Exemple : le codage des chiffres

UNIVERSITÉ CAEN NORMANDIE

■ Soit $X$ la source dont l'alphabet est $\{0, \ldots, 9\}$, muni de la loi uniforme.
Considérons le code de longueur fixe 4 suivant :

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| 0000 | 0001 | 0010 | 0011 | 0100 | 0101 | 0110 | 0111 | 1000 | 1001 |

L'efficacité de ce code est $H(X)/4 = (\log_2 10)/4 \simeq 0.83$.

■ On considère la source $X^2$ sur l'alphabet $\{00, 01, \ldots, 99\}$ de cardinal 100.
Son entropie est $H(X^2) = \log_2(100) = 2H(X)$.

Il existe un code régulier de longueur 7 (car $2^6 < 100 \leq 2^7$) tel que
$\log_2 100 \leq n < 1 + \log_2 100$.

# Exemple : le codage des chiffres

UNIVERSITÉ CAEN NORMANDIE

■ Soit $X$ la source dont l'alphabet est $\{0, \ldots, 9\}$, muni de la loi uniforme.
Considérons le code de longueur fixe 4 suivant :

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| 0000 | 0001 | 0010 | 0011 | 0100 | 0101 | 0110 | 0111 | 1000 | 1001 |

L'efficacité de ce code est $H(X)/4 = (\log_2 10)/4 \simeq 0.83$.

■ On considère la source $X^2$ sur l'alphabet $\{00, 01, \ldots, 99\}$ de cardinal 100.
Son entropie est $H(X^2) = \log_2(100) = 2H(X)$.

Il existe un code régulier de longueur 7 (car $2^6 < 100 \leq 2^7$) tel que
$\log_2 100 \leq n < 1 + \log_2 100$.

L'efficacité de ce code est donc $2\log_2(10)/7 \simeq 0.95$.

# Exemple : le codage des chiffres

UNIVERSITÉ CAEN NORMANDIE

■ Soit $X$ la source dont l'alphabet est $\{0, \ldots, 9\}$, muni de la loi uniforme.
Considérons le code de longueur fixe 4 suivant :

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| 0000 | 0001 | 0010 | 0011 | 0100 | 0101 | 0110 | 0111 | 1000 | 1001 |

L'efficacité de ce code est $H(X)/4 = (\log_2 10)/4 \simeq 0.83$.

■ On considère la source $X^2$ sur l'alphabet $\{00, 01, \ldots, 99\}$ de cardinal 100.
Son entropie est $H(X^2) = \log_2(100) = 2H(X)$.

Il existe un code régulier de longueur 7 (car $2^6 < 100 \leq 2^7$) tel que
$\log_2 100 \leq n < 1 + \log_2 100$.

L'efficacité de ce code est donc $2\log_2(10)/7 \simeq 0.95$.

■ On peut continuer le processus avec des triplets en considérant $X^3$.
On obtient un code d'efficacité $\log_2(1000)/10 \simeq 0.996$.

# Exemple : le codage des chiffres

UNIVERSITÉ CAEN NORMANDIE

■ Soit $X$ la source dont l'alphabet est $\{0, \ldots, 9\}$, muni de la loi uniforme.
Considérons le code de longueur fixe 4 suivant :

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| 0000 | 0001 | 0010 | 0011 | 0100 | 0101 | 0110 | 0111 | 1000 | 1001 |

L'efficacité de ce code est $H(X)/4 = (\log_2 10)/4 \simeq 0.83$.

■ On considère la source $X^2$ sur l'alphabet $\{00, 01, \ldots, 99\}$ de cardinal 100.
Son entropie est $H(X^2) = \log_2(100) = 2H(X)$.

Il existe un code régulier de longueur 7 (car $2^6 < 100 \leq 2^7$) tel que
$\log_2 100 \leq n < 1 + \log_2 100$.

L'efficacité de ce code est donc $2\log_2(10)/7 \simeq 0.95$.

■ On peut continuer le processus avec des triplets en considérant $X^3$.
On obtient un code d'efficacité $\log_2(1000)/10 \simeq 0.996$.

Soit $X$ une source de cardinal $K$. Alors il existe un code régulier dont l'efficacité est aussi proche que souhaité de $H(X)/\log_2 K$.

# Codes de longueur variables et codes préfixes

UNIVERSITÉ CAEN NORMANDIE

**Remarques :**

- Un code de longueur fixe ne tient pas compte de la fréquence des symboles
- Un code de longueur fixe est donc adapté pour les distributions uniformes
- On peut faire mieux lorsque la source n'est pas uniforme en codant les symboles les plus fréquents sur peu de bits.
- C'est le principe d'un code à **longueur variable**.

# Codes de longueur variables et codes préfixes

UNIVERSITÉ CAEN NORMANDIE

**Remarques :**

- Un code de longueur fixe ne tient pas compte de la fréquence des symboles
- Un code de longueur fixe est donc adapté pour les distributions uniformes
- On peut faire mieux lorsque la source n'est pas uniforme en codant les symboles les plus fréquents sur peu de bits.
- C'est le principe d'un code à **longueur variable**.

## Codes déchiffrables et préfixes

Un code est **déchiffrable** si son codage associé est injectif, c'est-à-dire que toute sequence binaire finie correspond à au plus une sequence de lettre de la source.

Un code est **préfixe** (ou instantané) si aucun mot de code n'est le début d'un autre mot de code.

# Codes de longueur variables et codes préfixes

UNIVERSITÉ CAEN NORMANDIE

## Codes déchiffrables et préfixes

Un code est **déchiffrable** si son codage associé est injectif, c'est-à-dire que toute sequence binaire finie correspond à au plus une sequence de lettre de la source.

Un code est **préfixe** (ou instantané) si aucun mot de code n'est le début d'un autre mot de code.

## Proposition :

Un code préfixe est déchiffrable. La réciproque est fausse.

# Codes de longueur variables et codes préfixes

UNIVERSITÉ CAEN NORMANDIE

## Codes déchiffrables et préfixes

Un code est **déchiffrable** si son codage associé est injectif, c'est-à-dire que toute sequence binaire finie correspond à au plus une sequence de lettre de la source.

Un code est **préfixe** (ou instantané) si aucun mot de code n'est le début d'un autre mot de code.

## Proposition :

Un code préfixe est déchiffrable. La réciproque est fausse.

**Exemple :** on considère les codes de la source $\{a, b, c\}$ suivants :
$a \to 1$, $b \to 10$, $c \to 110$ et $a \to 1$, $b \to 10$, $c \to 100$ et $a \to 1$, $b \to 01$, $c \to 001$.

Le premier code n'est ni préfixe, ni déchiffrable (110 peut signifier $c$ ou $ab$), le second est déchiffrable mais pas préfixe et le troisième code est préfixe (donc déchiffrable).

# Représentation d'un code binaire par un arbre

UNIVERSITÉ CAEN NORMANDIE

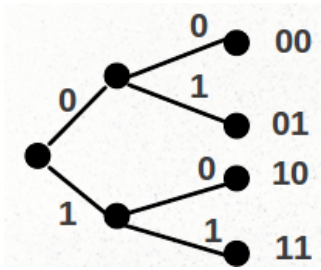
On considère l'arbre binaire où

- chaque branche a pour attribut 0 ou 1
- et où chaque nœud a pour attribut la concaténation des attributs des branches qui le relie à la racine.

## Arbre d'un code

Le plus petit arbre binaire contenant tout les mots d'un code donné sur ses nœuds est appelé arbre du code.

**Exemple :** l'arbre du code de la source $\{a, b, c, d\}$ défini par l'application $a \to 00$, $b \to 01$, $c \to 10$, $d \to 11$ est l'arbre binaire complet :




Les quatre mots du codes se lisent sur les quatre feuilles de l'arbre binaire ci-dessus.

# Représentation d'un code préfixe

UNIVERSITÉ CAEN NORMANDIE

## Propriété

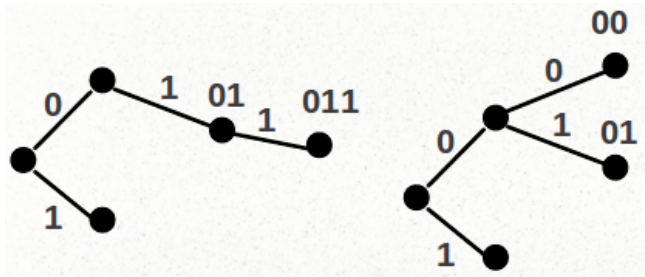
Un code est préfixe si et seulement si les feuilles de son arbre sont exactement ses mots de code.

**Preuve :** par définition d'un code préfixe un mot est sur une feuille signifie qu'il n'est pas le début d'un autre mot de code.

**Exemple :** on considère les deux codes de la source $\{a, b, c\}$ suivants :

- $a \to 1,\ b \to 01,\ c \to 011$
- $a \to 1,\ b \to 01,\ c \to 00$

Lequel est un code préfixe ?



# Représentation d'un code préfixe

UNIVERSITÉ CAEN NORMANDIE

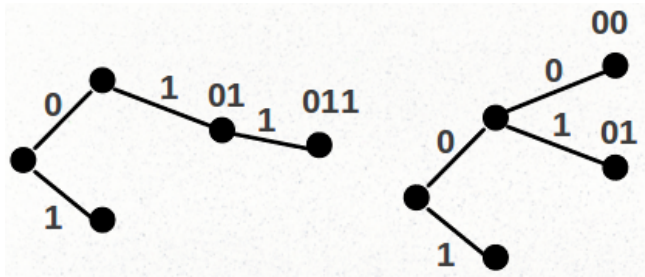
## Propriété

Un code est préfixe si et seulement si les feuilles de son arbre sont exactement ses mots de code.

**Preuve :** par définition d'un code préfixe un mot est sur une feuille signifie qu'il n'est pas le début d'un autre mot de code.

**Exemple :** on considère les deux codes de la source $\{a, b, c\}$ suivants :

- $a \to 1,\ b \to 01,\ c \to 011$
- $a \to 1,\ b \to 01,\ c \to 00$




Lequel est un code préfixe ?

Les mots de l'arbre de gauche ne sont pas tous sur les feuilles de l'arbre. Le code associé n'est donc pas préfixe ($\phi(b) = 01$ est le préfixe de $\phi(c) = 011$)

Le second code est préfixe (tous les mots sont aux feuilles).

# Premier théorème de Shannon

UNIVERSITÉ CAEN NORMANDIE

## Premier théorème de Shannon

Pour toute variable aléatoire $X$ d'entropie $H(X)$, il existe un code préfixe, de longueur moyenne $\ell$, vérifiant

$$H(X) \leq \ell < H(X) + 1.$$

**Preuve :** Utilise l'inégalité de Kraft.

**Exemples de codages non triviaux :**

- Codage de Shannon-Fano : quasiment plus utilisé aujourd'hui (non optimal)
- Codage de Huffman : JPEG, MPEG, MP3, GZIP, PKZIP, ... (optimal)
- Codage arithmétique : JPEG 2000, JBIG2 (optimal mais complexe)
- ...

# Codage de Shannon-Fano

UNIVERSITÉ CAEN NORMANDIE

Algorithme de compression sans perte publié par Shannon en 1948.

Les probabilités d'apparition de chaque symbole sont triées par ordre décroissant dans un tableau.

**Algorithme :**

1. Diviser le tableau en deux de telle manière que la somme des probabilités de chaque partie soit le plus proche possible.
2. Associer un 0 à la partie gauche du tableau et un 1 à la partie droite.
3. Appliquer récursivement les pas 1 et 2 à chacune des parties du tableau jusqu'à obtenir un codage différent pour chaque symbole.

Le code de chaque symbole se lit en partant de la racine jusqu'au symbole en question.

# Codage de Shannon-Fano

UNIVERSITÉ CAEN NORMANDIE

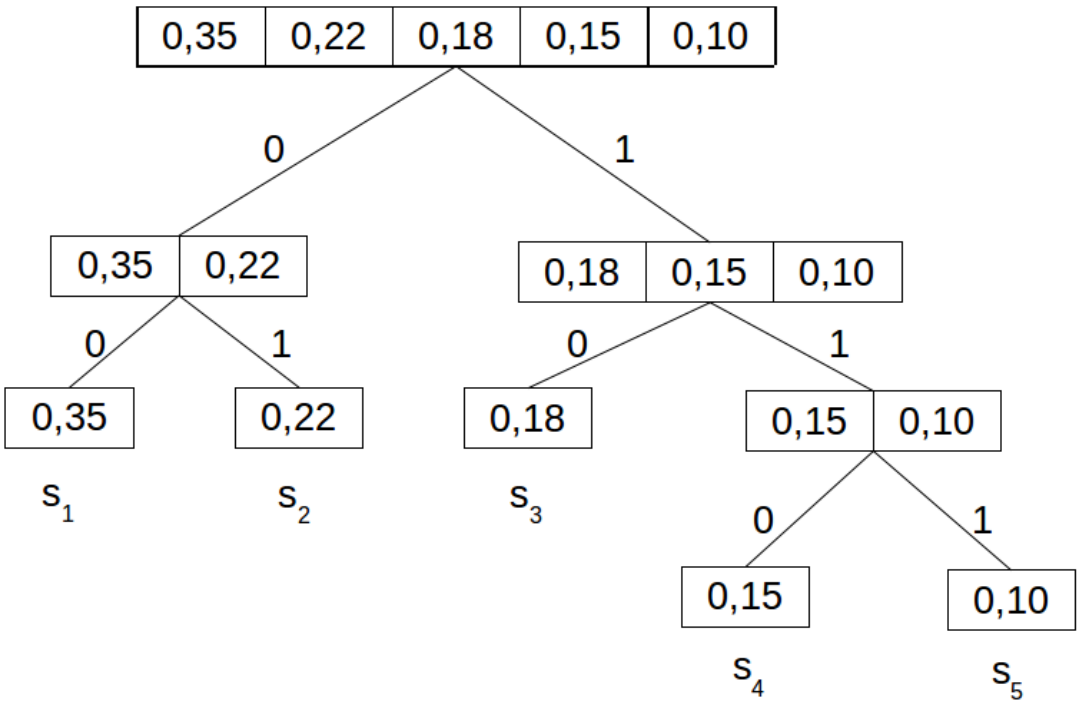
Considérons la source $X$ sur l'alphabet $\{s_1, s_2, s_3, s_4, s_5\}$ dont les fréquences sont données par :

$$p(s_1) = 0.35, \quad p(s_2) = 0.22, \quad p(s_3) = 0.18, \quad p(s_4) = 0.15, \quad p(s_5) = 0.1.$$

# Codage de Shannon-Fano

Considérons la source $X$ sur l'alphabet $\{s_1, s_2, s_3, s_4, s_5\}$ dont les fréquences sont données par :

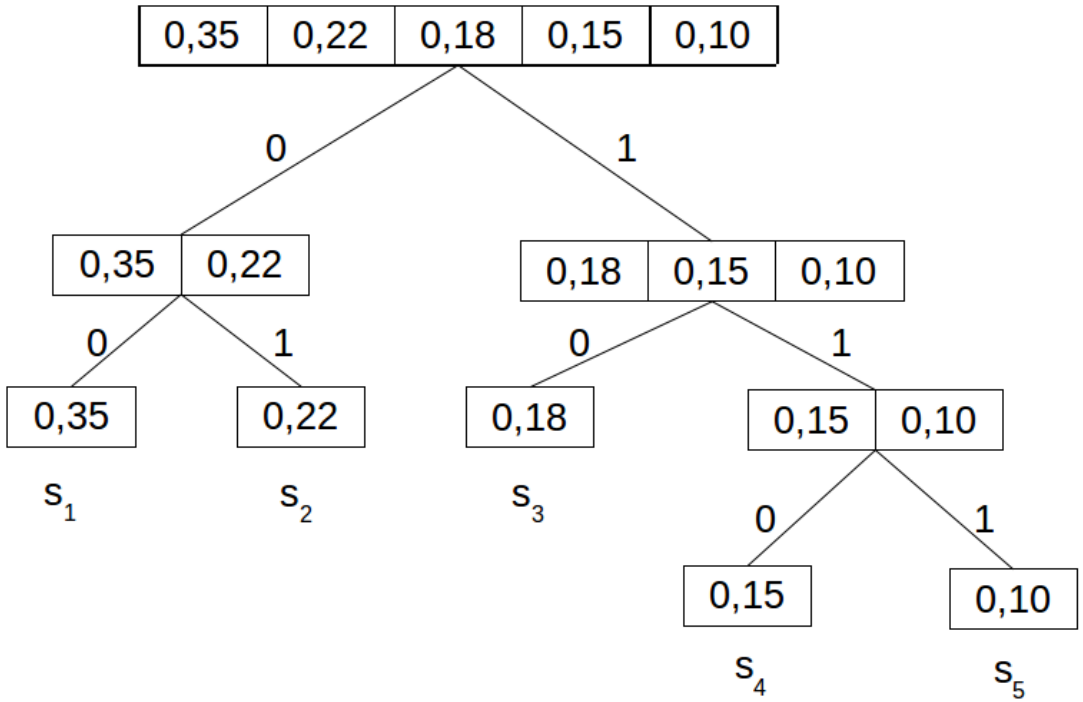
$$p(s_1) = 0.35, \quad p(s_2) = 0.22, \quad p(s_3) = 0.18, \quad p(s_4) = 0.15, \quad p(s_5) = 0.1.$$



# Codage de Shannon-Fano

Considérons la source $X$ sur l'alphabet $\{s_1, s_2, s_3, s_4, s_5\}$ dont les fréquences sont données par :

$$p(s_1) = 0.35, \quad p(s_2) = 0.22, \quad p(s_3) = 0.18, \quad p(s_4) = 0.15, \quad p(s_5) = 0.1.$$




$$\phi(s_1) = 00, \quad \phi(s_2) = 01, \quad \phi(s_3) = 10, \quad \phi(s_4) = 110, \quad \phi(s_5) = 111.$$

# Optimalité du code Shannon-Fano

UNIVERSITÉ CAEN NORMANDIE

## Propriété

Le code de Shannon-Fano associé à une source sans mémoire vérifie le premier Théorème de Shannon.
Cependant, d'autres codes préfixes ont une longueur moyenne plus faible.

Autrement dit, le code de Shannon-Fano n'est pas optimal.

# Code de Huffman (compression sans perte, 1952)

UNIVERSITÉ CAEN NORMANDIE

Publié par David Albert Huffman en 1952 lors de sa thèse au MIT dans l'article :
*A Method for the Construction of Minimum-Redundancy Codes.*

Le code de Huffman construit itérativement un arbre binaire de la manière suivante :

1. Au départ, il y a un nœud isolé par symbole, trié par ordre décroissant.
2. Construire un nouveau noeud dont la probabilité est la somme des probabilités des deux noeuds de plus faible probabilité.
3. Réitérer cette opération jusqu'à obtenir un seul arbre dont les feuilles sont les symboles initiaux.
4. Associer un 0 à chaque branche de gauche et un 1 à chaque branche de droite.

Le code de chaque symbole se lit en partant de la racine jusqu'au symbole en question.

# Code de Huffman (compression sans perte, 1952)

UNIVERSITÉ CAEN NORMANDIE

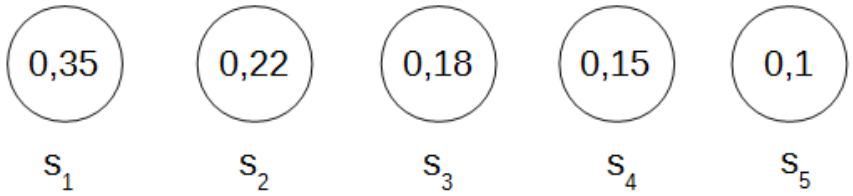
Considérons la source $X$ sur l'alphabet $\{s_1, s_2, s_3, s_4, s_5\}$ dont les fréquences sont données par :

$$p(s_1) = 0.35, \quad p(s_2) = 0.22, \quad p(s_3) = 0.18, \quad p(s_4) = 0.15, \quad p(s_5) = 0.1.$$

# Code de Huffman (compression sans perte, 1952)

Considérons la source $X$ sur l'alphabet $\{s_1, s_2, s_3, s_4, s_5\}$ dont les fréquences sont données par :

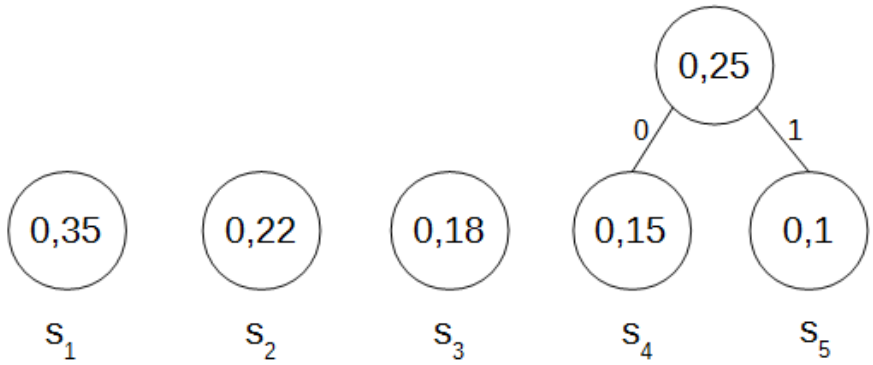
$$p(s_1) = 0.35, \quad p(s_2) = 0.22, \quad p(s_3) = 0.18, \quad p(s_4) = 0.15, \quad p(s_5) = 0.1.$$



# Code de Huffman (compression sans perte, 1952)

Considérons la source $X$ sur l'alphabet $\{s_1, s_2, s_3, s_4, s_5\}$ dont les fréquences sont données par :

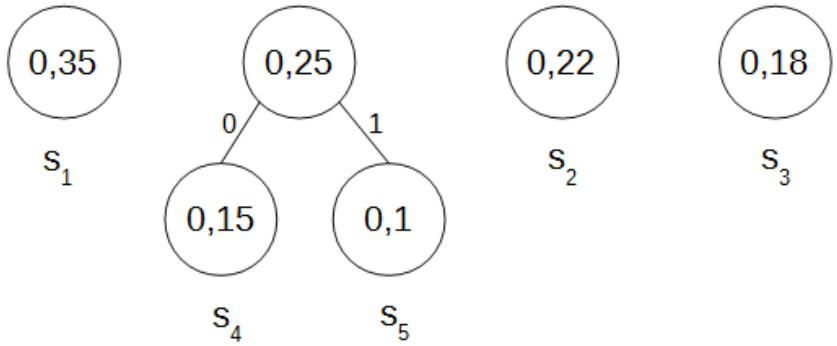
$$p(s_1) = 0.35, \quad p(s_2) = 0.22, \quad p(s_3) = 0.18, \quad p(s_4) = 0.15, \quad p(s_5) = 0.1.$$



# Code de Huffman (compression sans perte, 1952)

Considérons la source $X$ sur l'alphabet $\{s_1, s_2, s_3, s_4, s_5\}$ dont les fréquences sont données par :

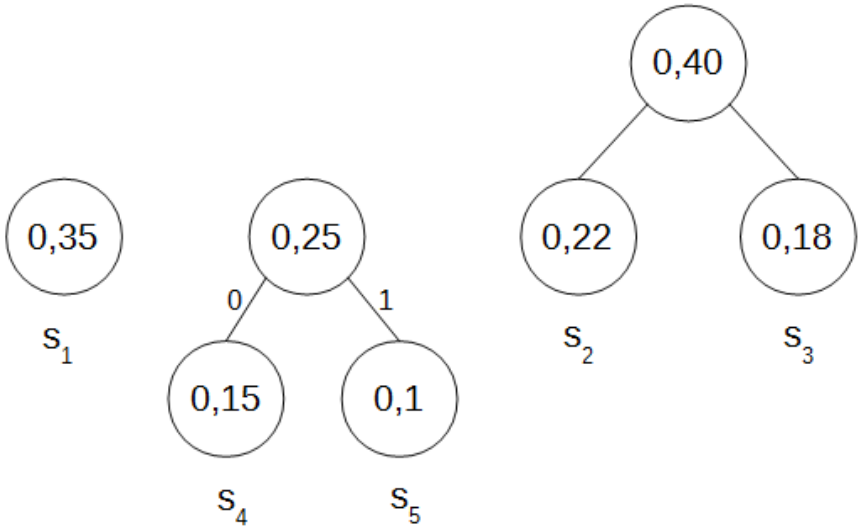
$$p(s_1) = 0.35, \quad p(s_2) = 0.22, \quad p(s_3) = 0.18, \quad p(s_4) = 0.15, \quad p(s_5) = 0.1.$$



# Code de Huffman (compression sans perte, 1952)

Considérons la source $X$ sur l'alphabet $\{s_1, s_2, s_3, s_4, s_5\}$ dont les fréquences sont données par :

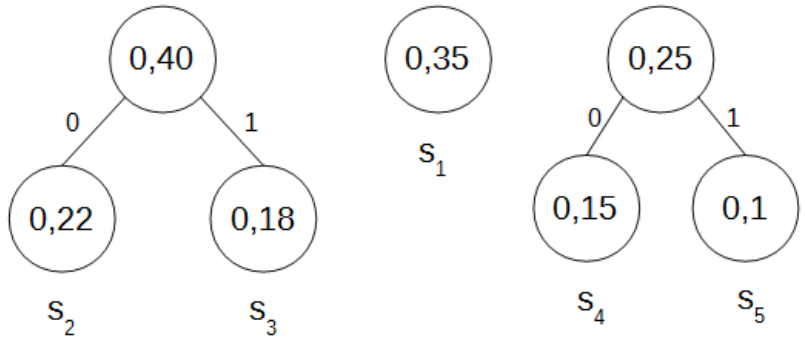
$$p(s_1) = 0.35, \quad p(s_2) = 0.22, \quad p(s_3) = 0.18, \quad p(s_4) = 0.15, \quad p(s_5) = 0.1.$$



# Code de Huffman (compression sans perte, 1952)

Considérons la source $X$ sur l'alphabet $\{s_1, s_2, s_3, s_4, s_5\}$ dont les fréquences sont données par :

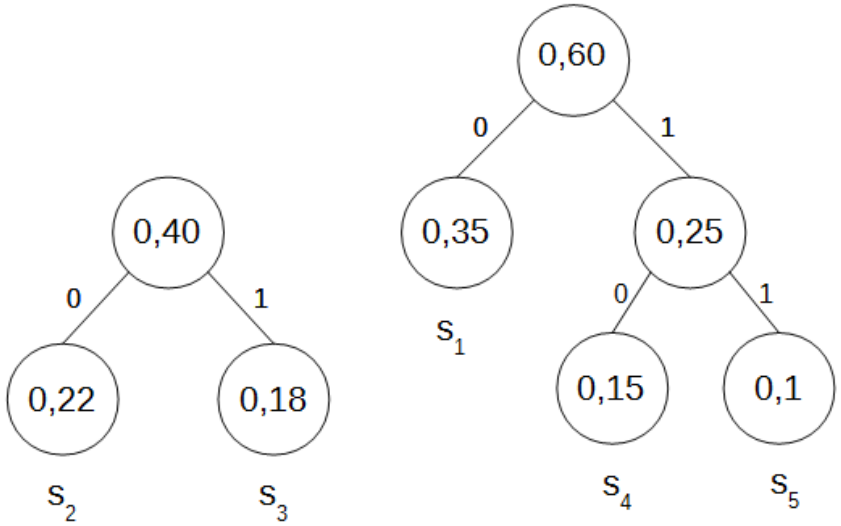
$$p(s_1) = 0.35, \quad p(s_2) = 0.22, \quad p(s_3) = 0.18, \quad p(s_4) = 0.15, \quad p(s_5) = 0.1.$$



# Code de Huffman (compression sans perte, 1952)

Considérons la source $X$ sur l'alphabet $\{s_1, s_2, s_3, s_4, s_5\}$ dont les fréquences sont données par :

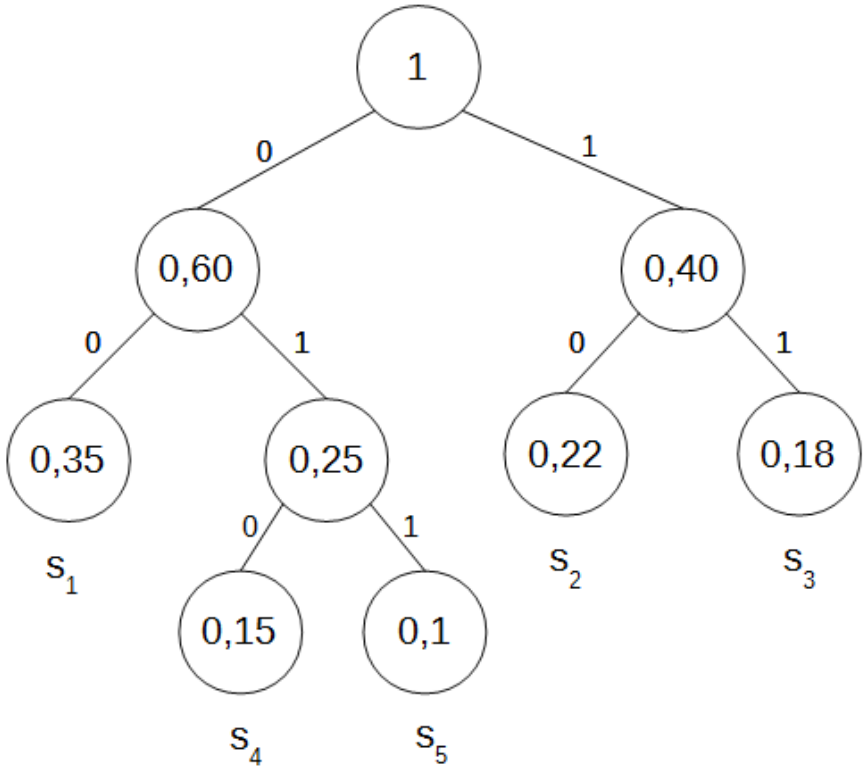
$$p(s_1) = 0.35, \quad p(s_2) = 0.22, \quad p(s_3) = 0.18, \quad p(s_4) = 0.15, \quad p(s_5) = 0.1.$$



# Code de Huffman (compression sans perte, 1952)

Considérons la source $X$ sur l'alphabet $\{s_1, s_2, s_3, s_4, s_5\}$ dont les fréquences sont données par :

$$p(s_1) = 0.35, \quad p(s_2) = 0.22, \quad p(s_3) = 0.18, \quad p(s_4) = 0.15, \quad p(s_5) = 0.1.$$




$$\phi(s_1) = 00, \quad \phi(s_2) = 10, \quad \phi(s_3) = 11, \quad \phi(s_4) = 010, \quad \phi(s_5) = 011.$$

# Optimalité du code de Huffman

UNIVERSITÉ CAEN NORMANDIE

## Théorème

Le code de Huffman associé à une source sans mémoire vérifie le premier Théorème de Shannon. De plus, il est le code préfixe dont la longueur moyenne est optimale (les autres codes font moins bien).

**Preuve :** admise...

# Optimalité du code de Huffman

UNIVERSITÉ CAEN NORMANDIE

## Théorème

Le code de Huffman associé à une source sans mémoire vérifie le premier Théorème de Shannon. De plus, il est le code préfixe dont la longueur moyenne est optimale (les autres codes font moins bien).

**Preuve :** admise...

Avantages et inconvénients du code de Huffman

(+) Il est optimal : il n'existe pas de code plus efficace

# Optimalité du code de Huffman

UNIVERSITÉ CAEN NORMANDIE

## Théorème

Le code de Huffman associé à une source sans mémoire vérifie le premier Théorème de Shannon. De plus, il est le code préfixe dont la longueur moyenne est optimale (les autres codes font moins bien).

**Preuve :** admise...

Avantages et inconvénients du code de Huffman

(+) Il est optimal : il n'existe pas de code plus efficace

(−) On a besoin du dictionnaire pour décompresser

# Optimalité du code de Huffman

UNIVERSITÉ CAEN NORMANDIE

## Théorème

Le code de Huffman associé à une source sans mémoire vérifie le premier Théorème de Shannon. De plus, il est le code préfixe dont la longueur moyenne est optimale (les autres codes font moins bien).

**Preuve :** admise...

Avantages et inconvénients du code de Huffman

(+) Il est optimal : il n'existe pas de code plus efficace

(−) On a besoin du dictionnaire pour décompresser

- Soit on ajoute le dictionnaire au codage mais cela peut fortement diminuer l'efficacité de la "compression".

# Optimalité du code de Huffman

UNIVERSITÉ CAEN NORMANDIE

## Théorème

Le code de Huffman associé à une source sans mémoire vérifie le premier Théorème de Shannon. De plus, il est le code préfixe dont la longueur moyenne est optimale (les autres codes font moins bien).

**Preuve :** admise...

Avantages et inconvénients du code de Huffman

(+) Il est optimal : il n'existe pas de code plus efficace

(−) On a besoin du dictionnaire pour décompresser

- Soit on ajoute le dictionnaire au codage mais cela peut fortement diminuer l'efficacité de la "compression".
- Soit on utilise un dictionnaire externe mais qui ne sera pas optimal pour tous les fichiers (probabilités différentes).

# Optimalité du code de Huffman

UNIVERSITÉ CAEN NORMANDIE

## Théorème

Le code de Huffman associé à une source sans mémoire vérifie le premier Théorème de Shannon. De plus, il est le code préfixe dont la longueur moyenne est optimale (les autres codes font moins bien).

**Preuve :** admise...

Avantages et inconvénients du code de Huffman

(+) Il est optimal : il n'existe pas de code plus efficace

(−) On a besoin du dictionnaire pour décompresser

- ☐ Soit on ajoute le dictionnaire au codage mais cela peut fortement diminuer l'efficacité de la "compression".
- ☐ Soit on utilise un dictionnaire externe mais qui ne sera pas optimal pour tous les fichiers (probabilités différentes).
- ☐ Soit on utilise des dictionnaires construits dynamiquement.

# Conclusion : codage de source

UNIVERSITÉ CAEN NORMANDIE

Nous avons vu (parmi d'autres choses) :

- les notions d'entropie, d'entropie conjointe et conditionnelle
- la notion de codes de longueur fixe
- la notion de code (préfixe) de longueur variable
- le fait que la longueur moyenne d'un code est toujours plus grande que l'entropie de la source (borne inférieure de compression)
- que le code de Huffman est un code optimal

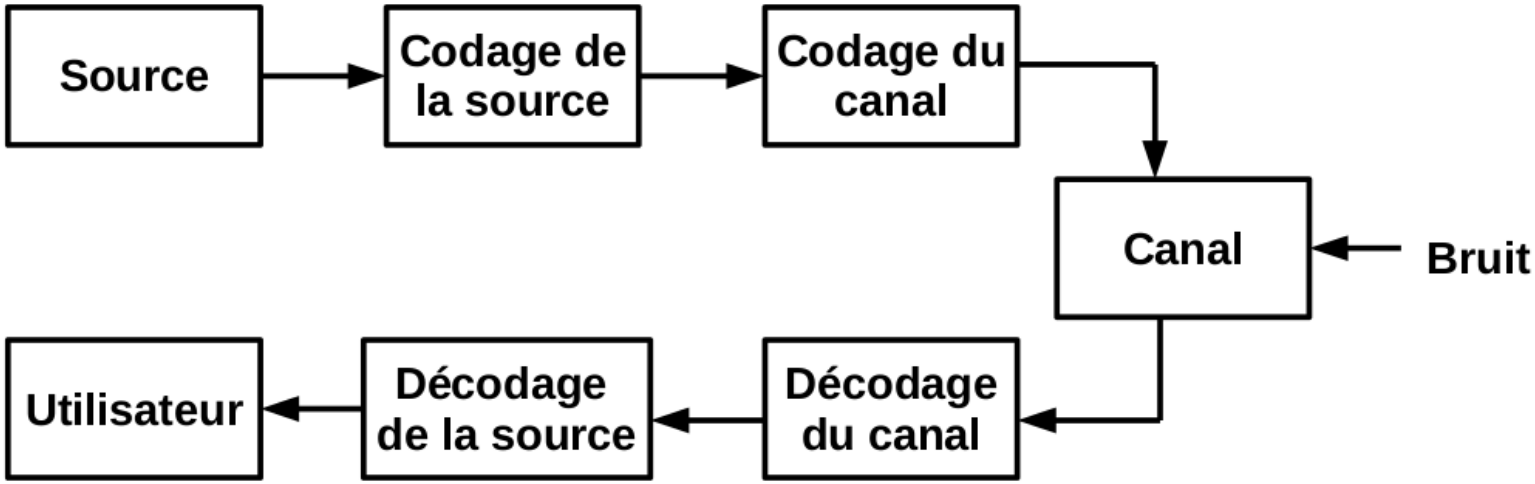
Une application fondamentale du codage de source est la compression...

Nous allons voir maintenant le codage de canal dont l'application principale est la transmission efficace d'informations avec correction d'erreurs.

UNIVERSITÉ
CAEN
NORMANDIE

# 4- Codage d'un canal

# Codage de source et codage de canal




La séparation source/canal permet une modélisation fine et indépendante de la source et du canal.

# Exemples de codes par parité : ISBN-10 et ISBN-13

UNIVERSITÉ CAEN NORMANDIE

La norme ISO 2108 spécifie le codage ISBN (*International Standard Book Number*) pour identifier de manière unique chaque livre publié.

Un code ISBN-10 comporte 10 chiffres $(c_{10}, \ldots, c_1)$, où $c_{10}, \ldots c_2$ sont des codes linguistiques, éditeurs et numéro de livres. $c_1$ est un caractère de contrôle calculé de la manière suivante :

$$c_1 = 11 - (\sum_{i=2}^{10} i * c_i \bmod 11) \bmod 11$$

.

Le code ISBN-13 comporte 13 chiffres $(c_{13}, \ldots, c_1)$ ou les trois premiers sont 978, suivis des 9 premiers chiffres de ISBN-10. $c_1$ est un caractère de contrôle calculé de la manière suivante :

$$c_1 = 10 - (\sum_{i=2}^{13} c_{2i+1} + 3 * c_{2i} \bmod 10) \bmod 10.$$

# Exemples de codes par parité : ISBN-10 et ISBN-13

UNIVERSITÉ CAEN NORMANDIE

Le code EAN-13 correspond au code ISBN-13 sans tirets entre certains chiffres pour pouvoir être identifié par un code barre.

Une extension du code EAN-13 est le code EAN-128 qui permet de coder aussi des lettres.

Ces codes permettent de détecter une erreur dans le code mais pas de la corriger.

# Exemples de codes par parité : ISBN-10 et ISBN-13

UNIVERSITÉ CAEN NORMANDIE

Le code EAN-13 correspond au code ISBN-13 sans tirets entre certains chiffres pour pouvoir être identifié par un code barre.

Une extension du code EAN-13 est le code EAN-128 qui permet de coder aussi des lettres.

Ces codes permettent de détecter une erreur dans le code mais pas de la corriger.

Certains codes barre permettent aussi de corriger les erreurs comme les codes PDF417 (portable data file), Aztec, datamatrix ou encore les QR codes.
Un QR code est bidimensionnel et multi lignes et peut coder jusqu'à 3000 caractères et corriger des erreurs grâce a un code de Reed Solomon.

UNIVERSITÉ
CAEN
NORMANDIE

# 4- Codage d'un canal

## Codage d'un canal - un peu de théorie

# Entropie conjointe

UNIVERSITÉ CAEN NORMANDIE

- Soit $X$ une variable aléatoire à valeurs dans $\{x_1, \ldots, x_n\}$.
- Soit $Y$ une variable aléatoire à valeurs dans $\{y_1, \ldots, y_m\}$.
- Pour tout $1 \leq i \leq n$ et $1 \leq j \leq m$, on note

$$p(x_i) = \mathbb{P}[X = x_i], \quad p(y_j) = \mathbb{P}[Y = y_j], \quad p(x_i, y_j) = \mathbb{P}[X = x_i \cap Y = y_j].$$

# Entropie conjointe

UNIVERSITÉ CAEN NORMANDIE

- Soit $X$ une variable aléatoire à valeurs dans $\{x_1, \ldots, x_n\}$.
- Soit $Y$ une variable aléatoire à valeurs dans $\{y_1, \ldots, y_m\}$.
- Pour tout $1 \leq i \leq n$ et $1 \leq j \leq m$, on note

$$p(x_i) = \mathbb{P}[X = x_i], \quad p(y_j) = \mathbb{P}[Y = y_j], \quad p(x_i, y_j) = \mathbb{P}[X = x_i \cap Y = y_j].$$

## Entropie conjointe

L'**entropie conjointe** $H(X, Y)$ de $X$ et $Y$ est

$$H(X, Y) = -\sum_{i,j} p(x_i, y_j) \log_2 p(x_i, y_j) = H(Y, X).$$

# Entropie conjointe

UNIVERSITÉ CAEN NORMANDIE

- Soit $X$ une variable aléatoire à valeurs dans $\{x_1, \ldots, x_n\}$.
- Soit $Y$ une variable aléatoire à valeurs dans $\{y_1, \ldots, y_m\}$.
- Pour tout $1 \leq i \leq n$ et $1 \leq j \leq m$, on note

$$p(x_i) = \mathbb{P}[X = x_i], \quad p(y_j) = \mathbb{P}[Y = y_j], \quad p(x_i, y_j) = \mathbb{P}[X = x_i \cap Y = y_j].$$

## Entropie conjointe

L'**entropie conjointe** $H(X, Y)$ de $X$ et $Y$ est

$$H(X, Y) = -\sum_{i,j} p(x_i, y_j) \log_2 p(x_i, y_j) = H(Y, X).$$

- Il s'agit de l'entropie de la variable aléatoire discrète $Z = (X, Y)$ à valeurs dans $\{x_1, \ldots, x_n\} \times \{y_1, \ldots, y_m\}$.
- Cette définition se généralise naturellement à $n$ variables aléatoires $X_1, \ldots, X_n$.

# Entropie conjointe

$$H(X,Y) = -\sum_{i,j} p(x_i, y_j) \log_2 p(x_i, y_j) = H(Y,X).$$

**Exemple :**

| $Y$ \ $X$ | 0 | 1 |
|---|---|---|
| 0 | 1/8 | 1/4 |
| 1 | 1/8 | 1/8 |
| 2 | 1/4 | 1/8 |

1. Calculer $\mathbb{P}[X = i]$ pour $i = 1, 2$
2. Calculer $\mathbb{P}[Y = j]$ pour $j = 1, 2, 3$.
3. Calculer $H(X)$ et $H(Y)$ .
4. Calculer $H(X, Y)$.
5. Comparer $H(X, Y)$ avec $H(X)$ et $H(Y)$
6. Comparer $H(X, Y)$ avec $H(X) + H(Y)$

# Entropie conjointe

$$H(X,Y) = -\sum_{i,j} p(x_i, y_j) \log_2 p(x_i, y_j) = H(Y,X).$$

**Exemple :**

| $Y$ \ $X$ | 0 | 1 |
|---|---|---|
| 0 | 1/8 | 1/4 |
| 1 | 1/8 | 1/8 |
| 2 | 1/4 | 1/8 |

1. Calculer $\mathbb{P}[X = i]$ pour $i = 1, 2$ $(\mathbb{P}[X = i] = \frac{1}{2})$
2. Calculer $\mathbb{P}[Y = j]$ pour $j = 1, 2, 3$.
   $(\mathbb{P}[Y = 0] = \mathbb{P}[Y = 2] = \frac{3}{8}, \quad \mathbb{P}[Y = 1] = \frac{2}{8})$
3. Calculer $H(X)$ $(H(X) = 1)$ et $H(Y)$ $(H(Y) \approx 1.56)$.
4. Calculer $H(X, Y)$. $(H(X, Y) = 2.5)$
5. Comparer $H(X, Y)$ avec $H(X)$ et $H(Y)$
   $(H(X, Y) \geq H(X), H(Y))$
6. Comparer $H(X, Y)$ avec $H(X) + H(Y)$
   $(H(X, Y) \leq H(X) + H(Y))$

# Entropie conjointe

UNIVERSITÉ CAEN NORMANDIE

## Propriétés

Soit $X, Y$ deux variables aléatoires discrètes. Alors

$$H(X,Y) \geq H(X), \quad H(X,Y) \geq H(Y), \quad H(X,Y) \leq H(X) + H(Y).$$

La dernière inégalité est une égalité si et seulement si $X$ et $Y$ sont des variables aléatoires indépendantes.

# Entropie conjointe

UNIVERSITÉ CAEN NORMANDIE

## Propriétés

Soit $X, Y$ deux variables aléatoires discrètes. Alors

$$H(X,Y) \geq H(X), \quad H(X,Y) \geq H(Y), \quad H(X,Y) \leq H(X) + H(Y).$$

La dernière inégalité est une égalité si et seulement si $X$ et $Y$ sont des variables aléatoires indépendantes.

- Les deux premières inégalités indiquent que l'incertitude de deux sources conjointes est plus grande que l'incertitude d'une des deux sources.
- La dernière inégalité indique que l'incertitude de deux sources conjointes est moins grande que la somme des incertitudes des deux sources ("séparées").
- En pratique, $X$ sera la source (codée) à l'entrée du canal et $Y$ sera la sortie du canal.

# Entropie conjointe

UNIVERSITÉ CAEN NORMANDIE

## Propriétés

Soit $X, Y$ deux variables aléatoires discrètes. Alors

$$H(X,Y) \geq H(X), \quad H(X,Y) \geq H(Y), \quad H(X,Y) \leq H(X) + H(Y).$$

La dernière inégalité est une égalité si et seulement si $X$ et $Y$ sont des variables aléatoires indépendantes.

- Les deux premières inégalités indiquent que l'incertitude de deux sources conjointes est plus grande que l'incertitude d'une des deux sources.
- La dernière inégalité indique que l'incertitude de deux sources conjointes est moins grande que la somme des incertitudes des deux sources ("séparées").
- En pratique, $X$ sera la source (codée) à l'entrée du canal et $Y$ sera la sortie du canal.

**Preuve :** Utiliser $p(x_i, y_j) = p(y_j|x_i)p(x_i) = p(x_i|y_j)p(y_j)$ pour les deux premières inégalités. Pour la dernière, il faut utiliser une inégalité de convexité.

# Entropie conditionnelle

UNIVERSITÉ CAEN NORMANDIE

- Soit $X$ une variable aléatoire à valeurs dans $\{x_1, \ldots, x_n\}$.
- Soit $Y$ une variable aléatoire à valeurs dans $\{y_1, \ldots, y_m\}$.
- Pour tout $1 \leq i \leq n$ et $1 \leq j \leq m$, on note

$$p(x_i) = \mathbb{P}[X = x_i], \quad p(y_j) = \mathbb{P}[Y = y_j], \quad p(x_i, y_j) = \mathbb{P}[X = x_i \cap Y = y_j].$$

- En pratique, $X$ sera la source (codée) à l'entrée du canal et $Y$ sera la sortie du canal.

# Entropie conditionnelle

UNIVERSITÉ CAEN NORMANDIE

- Soit $X$ une variable aléatoire à valeurs dans $\{x_1, \ldots, x_n\}$.
- Soit $Y$ une variable aléatoire à valeurs dans $\{y_1, \ldots, y_m\}$.
- Pour tout $1 \leq i \leq n$ et $1 \leq j \leq m$, on note

$$p(x_i) = \mathbb{P}[X = x_i], \quad p(y_j) = \mathbb{P}[Y = y_j], \quad p(x_i, y_j) = \mathbb{P}[X = x_i \cap Y = y_j].$$

- En pratique, $X$ sera la source (codée) à l'entrée du canal et $Y$ sera la sortie du canal.

## Entropie conditionnelle

L'**entropie conditionnelle** de $X$ sachant $Y$ représente l'incertitude que l'on a sur $X$ (ou encore l'information apportée par $X$), lorsque l'on connait la variable $Y$ :

$$H(X \mid Y) = -\sum_{i,j} p(x_i, y_j) \log_2 p(x_i \mid y_j) = \sum_{i,j} p(x_i, y_j) \log_2 \frac{p(y_j)}{p(x_i, y_j)}.$$

# Entropie conditionnelle

UNIVERSITÉ CAEN NORMANDIE

- Soit $X$ une variable aléatoire à valeurs dans $\{x_1, \ldots, x_n\}$.
- Soit $Y$ une variable aléatoire à valeurs dans $\{y_1, \ldots, y_m\}$.
- Pour tout $1 \leq i \leq n$ et $1 \leq j \leq m$, on note

$$p(x_i) = \mathbb{P}[X = x_i], \quad p(y_j) = \mathbb{P}[Y = y_j], \quad p(x_i, y_j) = \mathbb{P}[X = x_i \cap Y = y_j].$$

- En pratique, $X$ sera la source (codée) à l'entrée du canal et $Y$ sera la sortie du canal.

## Entropie conditionnelle

L'**entropie conditionnelle** de $X$ sachant $Y$ représente l'incertitude que l'on a sur $X$ (ou encore l'information apportée par $X$), lorsque l'on connait la variable $Y$ :

$$H(X \mid Y) = -\sum_{i,j} p(x_i, y_j) \log_2 p(x_i \mid y_j) = \sum_{i,j} p(x_i, y_j) \log_2 \frac{p(y_j)}{p(x_i, y_j)}.$$

L'égalité vient de la relation $p(x_i, y_j) = p(x_i|y_j)p(y_j)$.

# Entropie conditionnelle

$$H(Y) = -\sum_j p(y_j) \log_2 p(y_j).$$
$$H(X,Y) = -\sum_{i,j} p(x_i, y_j) \log_2 p(x_i, y_j) = H(Y,X).$$
$$H(X \mid Y) = -\sum_{i,j} p(x_i, y_j) \log_2 p(x_i \mid y_j).$$

**Exemple :**

| $Y$ \ $X$ | 0 | 1 |
|---|---|---|
| 0 | 1/8 | 1/4 |
| 1 | 1/8 | 1/8 |
| 2 | 1/4 | 1/8 |

1. Calculer $H(X,Y)$.
2. Calculer $H(X \mid Y)$. .
3. Calculer $p(y_j) = \mathbb{P}[Y = j]$ pour $j = 1, 2, 3$.
4. Calculer $H(Y)$ .
5. Que constatez-vous ?

# Entropie conditionnelle

$$H(Y) = -\sum_j p(y_j) \log_2 p(y_j).$$
$$H(X,Y) = -\sum_{i,j} p(x_i, y_j) \log_2 p(x_i, y_j) = H(Y,X).$$
$$H(X \mid Y) = -\sum_{i,j} p(x_i, y_j) \log_2 p(x_i \mid y_j).$$

**Exemple :**

| $Y$ \ $X$ | 0 | 1 |
|---|---|---|
| 0 | 1/8 | 1/4 |
| 1 | 1/8 | 1/8 |
| 2 | 1/4 | 1/8 |

1. Calculer $H(X,Y)$. $(H(X,Y) = 2.5)$
2. Calculer $H(X \mid Y)$. $(H(X \mid Y) \approx 0.94)$.
3. Calculer $p(y_j) = \mathbb{P}[Y = j]$ pour $j = 1, 2, 3$.
   $(\mathbb{P}[Y = 0] = \mathbb{P}[Y = 2] = \frac{3}{8}, \quad \mathbb{P}[Y = 1] = \frac{2}{8})$
4. Calculer $H(Y)$ $(H(Y) \approx 1.56)$.
5. Que constatez-vous ? $H(X \mid Y) = H(X,Y) - H(Y)$

# Lien entre entropie conjointe et conditionnelle

UNIVERSITÉ CAEN NORMANDIE

## Propriétés

$$H(X \mid Y) = H(X, Y) - H(Y).$$

# Lien entre entropie conjointe et conditionnelle

UNIVERSITÉ CAEN NORMANDIE

## Propriétés

$$H(X \mid Y) = H(X, Y) - H(Y).$$

**Preuve :**

$$\begin{aligned}
H(X,Y) &= -\sum_{i,j} p(x_i, y_j) \log_2 p(x_i, y_j) \\
&= -\sum_{i,j} p(x_i, y_j) \log_2 (p(x_i \mid y_j) p(y_j)) \\
&= -\sum_{i,j} p(x_i, y_j) (\log_2 p(x_i \mid y_j) + \log_2 p(y_j)) \\
&= H(X \mid Y) - \sum_j (\sum_i p(x_i, y_j) \log_2 p(y_j)) \\
&= H(X \mid Y) - \sum_j p(y_j) \log_2 p(y_j) = H(X \mid Y) + H(Y).
\end{aligned}$$

# Conséquence

UNIVERSITÉ CAEN NORMANDIE

## Proposition

L'incertitude contenue dans $X$ connaissant $Y$ est toujours inférieure à celle sans la connaissance de $Y$ : $H(X|Y) \leq H(X)$, avec égalité si et seulement si les variables $X$ et $Y$ sont indépendantes.

**Preuve :** à l'aide du théorème précédent, on a :
$H(X) - H(X \mid Y) = H(X) + H(Y) - H(X, Y) \geq 0.$

# Application : règle de la chaine (*chain rule*)

UNIVERSITÉ CAEN NORMANDIE

## Théorème

Soit $(X_1, \ldots, X_n)$, la variable aléatoire de distribution de probabilité $p(x_1, \ldots, x_n)$. Alors on a :

$$H(X_1, \ldots, X_n) = \sum_{i=1}^{n} H(X_i \mid X_{i-1} \ldots X_1).$$

**Preuve :** par récurrence sur la formule précédente.

# Exemple d'application : la cryptographie

UNIVERSITÉ CAEN NORMANDIE

On considère $m$ et $c$ deux variables aléatoires vérifiant la relation $c = E_k(m)$, où $E_k$ est une fonction de chiffrement de clé (aléatoire) $k$. On peut retrouver $m$ à partir de $c$ et $k$ par une fonction de déchiffrement $m = D_k(c)$. Ainsi $H(m \mid k, c) = 0$.

# Exemple d'application : la cryptographie

UNIVERSITÉ CAEN NORMANDIE

On considère $m$ et $c$ deux variables aléatoires vérifiant la relation $c = E_k(m)$, où $E_k$ est une fonction de chiffrement de clé (aléatoire) $k$. On peut retrouver $m$ à partir de $c$ et $k$ par une fonction de déchiffrement $m = D_k(c)$. Ainsi $H(m \mid k, c) = 0$.

## Système de chiffrement parfait

Le système est *parfaitement sûr* si $H(m \mid c) = H(m)$.

# Exemple d'application : la cryptographie

UNIVERSITÉ CAEN NORMANDIE

On considère $m$ et $c$ deux variables aléatoires vérifiant la relation $c = E_k(m)$, où $E_k$ est une fonction de chiffrement de clé (aléatoire) $k$. On peut retrouver $m$ à partir de $c$ et $k$ par une fonction de déchiffrement $m = D_k(c)$. Ainsi $H(m \mid k, c) = 0$.

## Système de chiffrement parfait

Le système est *parfaitement sûr* si $H(m \mid c) = H(m)$.

**Théorème :** le système vérifie $H(k \mid c) = H(m \mid c) + H(k \mid m, c)$.

**Preuve :**

$$\begin{aligned} H(m \mid c) &= H(m, c) - H(c) = H(m, k, c) - H(k \mid m, c) - H(c). \\ H(k \mid c) &= H(k, c) - H(c) = H(m, k, c) - H(m \mid k, c) - H(c) \\ &= H(m, k, c) - H(c) = H(m \mid c) + H(k \mid m, c). \end{aligned}$$

# Exemple d'application : la cryptographie

UNIVERSITÉ CAEN NORMANDIE

On considère $m$ et $c$ deux variables aléatoires vérifiant la relation $c = E_k(m)$, où $E_k$ est une fonction de chiffrement de clé (aléatoire) $k$. On peut retrouver $m$ à partir de $c$ et $k$ par une fonction de déchiffrement $m = D_k(c)$. Ainsi $H(m \mid k, c) = 0$.

## Système de chiffrement parfait

Le système est *parfaitement sûr* si $H(m \mid c) = H(m)$.

**Théorème :** le système vérifie $H(k \mid c) = H(m \mid c) + H(k \mid m, c)$.

**Preuve :**

$$\begin{aligned} H(m \mid c) &= H(m, c) - H(c) = H(m, k, c) - H(k \mid m, c) - H(c). \\ H(k \mid c) &= H(k, c) - H(c) = H(m, k, c) - H(m \mid k, c) - H(c) \\ &= H(m, k, c) - H(c) = H(m \mid c) + H(k \mid m, c). \end{aligned}$$

**Corolaire :** On a $H(k \mid c) \geq H(m \mid c)$.

# Exemple d'application : la cryptographie

UNIVERSITÉ CAEN NORMANDIE

On considère $m$ et $c$ deux variables aléatoires vérifiant la relation $c = E_k(m)$, où $E_k$ est une fonction de chiffrement de clé (aléatoire) $k$. On peut retrouver $m$ à partir de $c$ et $k$ par une fonction de déchiffrement $m = D_k(c)$. Ainsi $H(m \mid k, c) = 0$.

## Système de chiffrement parfait

Le système est *parfaitement sûr* si $H(m \mid c) = H(m)$.

**Théorème :** le système vérifie $H(k \mid c) = H(m \mid c) + H(k \mid m, c)$.

**Corolaire :** On a $H(k \mid c) \geq H(m \mid c)$.

# Exemple d'application : la cryptographie

UNIVERSITÉ CAEN NORMANDIE

On considère $m$ et $c$ deux variables aléatoires vérifiant la relation $c = E_k(m)$, où $E_k$ est une fonction de chiffrement de clé (aléatoire) $k$. On peut retrouver $m$ à partir de $c$ et $k$ par une fonction de déchiffrement $m = D_k(c)$. Ainsi $H(m \mid k, c) = 0$.

## Système de chiffrement parfait

Le système est *parfaitement sûr* si $H(m \mid c) = H(m)$.

**Théorème :** le système vérifie $H(k \mid c) = H(m \mid c) + H(k \mid m, c)$.

**Corolaire :** On a $H(k \mid c) \geq H(m \mid c)$.

**Théorème :** le système parfaitement sûr vérifie $H(k) \geq H(m)$.

**Preuve :** $H(k) \geq H(k \mid c) \geq H(m \mid c) = H(m)$.

# Exemple d'application : la cryptographie

UNIVERSITÉ CAEN NORMANDIE

On considère $m$ et $c$ deux variables aléatoires vérifiant la relation $c = E_k(m)$, où $E_k$ est une fonction de chiffrement de clé (aléatoire) $k$. On peut retrouver $m$ à partir de $c$ et $k$ par une fonction de déchiffrement $m = D_k(c)$. Ainsi $H(m \mid k, c) = 0$.

## Système de chiffrement parfait

Le système est *parfaitement sûr* si $H(m \mid c) = H(m)$.

**Théorème :** le système vérifie $H(k \mid c) = H(m \mid c) + H(k \mid m, c)$.

**Corolaire :** On a $H(k \mid c) \geq H(m \mid c)$.

**Théorème :** le système parfaitement sûr vérifie $H(k) \geq H(m)$.

**Preuve :** $H(k) \geq H(k \mid c) \geq H(m \mid c) = H(m)$.
$\rightarrow$ Autrement dit, le nombre de bits pour encoder la clé est plus grand que le nombre de bits pour encoder un message

# Exemple de calculs d'entropie

Soient $(X, Y)$ une paire de variables aléatoires, définies sur le même alphabet $\{1, 2, 3, 4\}$, de distribution de probabilité conjointe :

| $Y \backslash X$ | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 1 | 1/8 | 1/16 | 1/32 | 1/32 |
| 2 | 1/16 | 1/8 | 1/32 | 1/32 |
| 3 | 1/16 | 1/16 | 1/16 | 1/16 |
| 4 | 1/4 | 0 | 0 | 0 |

La distribution de probabilité de $X$ est donc $(\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8})$ et celle de $Y$ est $(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$.

On obtient ainsi $H(X) = 7/4$ et $H(Y) = 2$.

On a $p(x = 1 \mid y = 1) = 1/2$, $p(x = 2 \mid y = 1) = 1/4$, $p(x = 3 \mid y = 1) = 1/8$ et $p(x = 4 \mid y = 1) = 1/8$

Alors on a $H(X \mid y = 1) = 1/2 + 2/4 + 3/8 + 3/8 = 7/4$. De même $H(X \mid y = 2) = 2/4 + 1/2 + 3/8 + 3/8 = 7/4$, $H(X \mid y = 3) = 2/4 + 2/4 + 2/4 + 2/4 = 2$, et $H(X \mid y = 4) = 0$.

On obtient donc
$H(X \mid Y) = \sum_j p(y = j) H(X \mid y = j) = \frac{1}{4} \times \frac{7}{4} + \frac{1}{4} \times \frac{7}{4} + \frac{1}{4} \times 2 + \frac{1}{4} \times 0 = \frac{11}{8}$.

De manière similaire on calcule $H(Y \mid X) = \frac{13}{8}$ ou encore $H(X, Y) = \frac{27}{8}$.

On retrouve bien $H(X) + H(Y \mid X) = \frac{7}{4} + \frac{13}{8} = \frac{27}{8} = H(X, Y) = 2 + \frac{11}{8} = H(Y) + H(X \mid Y)$.

# Notion d'information mutuelle

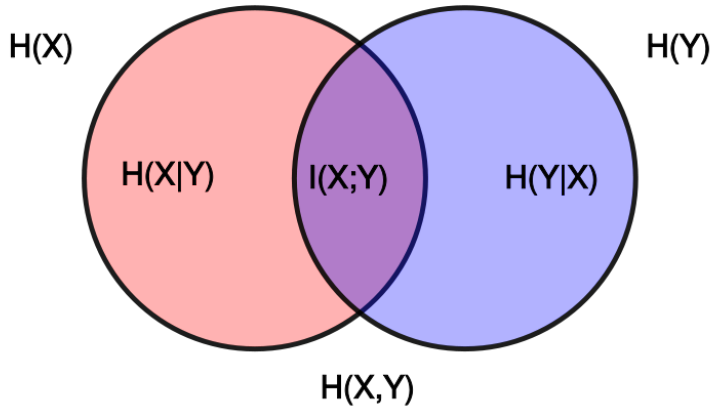
Soit $X$ et $Y$ deux variables aléatoires de distributions de probabilité $p(x)$ et $p(y)$ et soit $p(x, y)$ la distribution de la variable $(X, Y)$.

## Information mutuelle

L'**information mutuelle** $I(X, Y)$ est définie par

$$I(X, Y) = \sum_{i,j} p(x_i, y_j) \log_2 \frac{p(x_i, y_j)}{p(x_i)p(y_j)}.$$

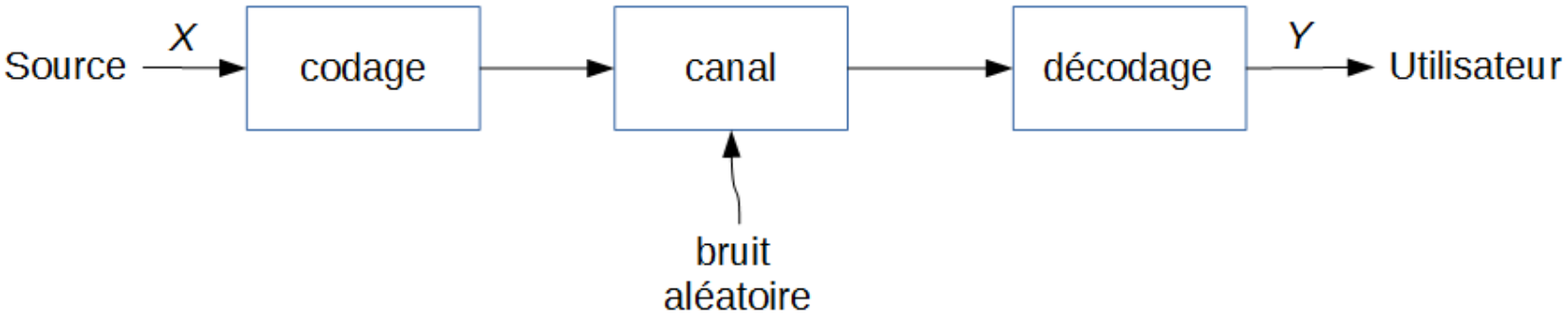
- L'information mutuelle mesure la corrélation entre $X$ et $Y$




[wikipedia]

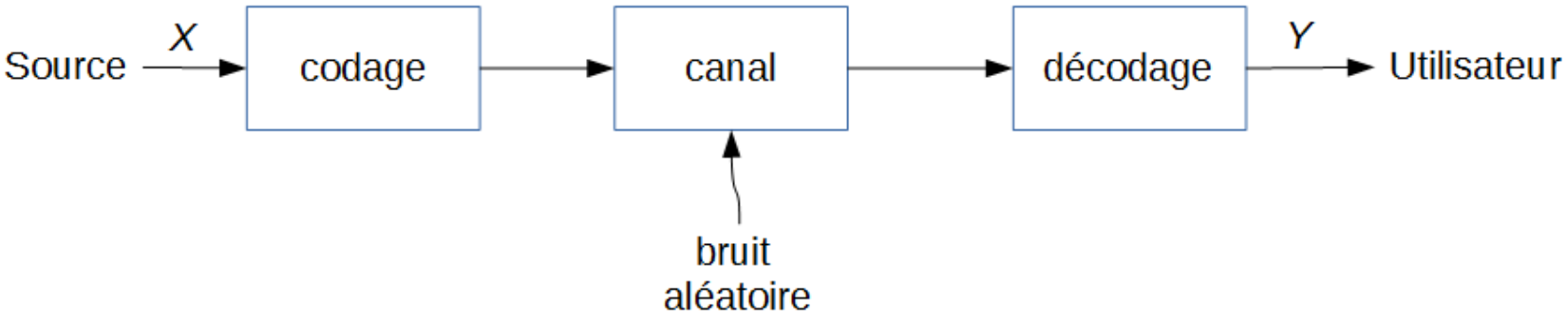
# Notion d'information mutuelle

**Application :**

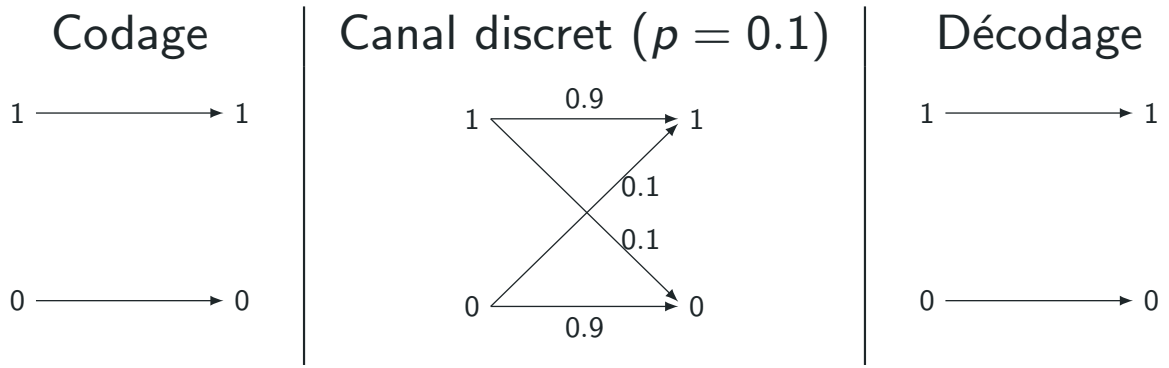


# Notion d'information mutuelle

**Application :**




**Exemple :** On considère la source $X$ sur $\{0,1\}$ définie par $\mathbb{P}[X=1] = 1 - \mathbb{P}[X=0] = \frac{1}{4}$.
On considère le canal suivant :

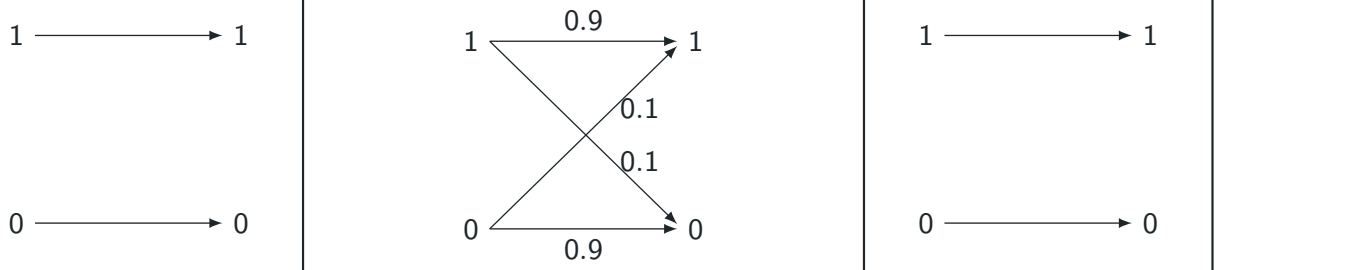



$Y$ dépend de $X$ et du canal (+codage). Quelle est la loi de probabilité de $Y$ ?

# Notion d'information mutuelle

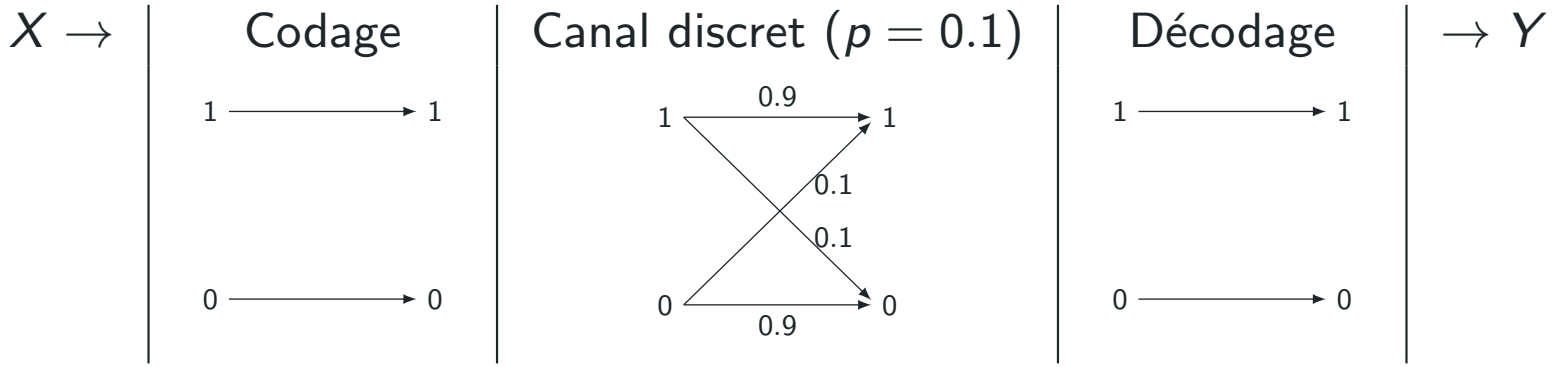
**Exemple :** On considère la source $X$ sur $\{0,1\}$ définie par $\mathbb{P}[X=1] = 1 - \mathbb{P}[X=0] = \frac{1}{4}$.
On considère le canal suivant :

$X \rightarrow$ | Codage | Canal discret ($p = 0.1$) | Décodage | $\rightarrow Y$



# Notion d'information mutuelle

**Exemple :** On considère la source $X$ sur $\{0,1\}$ définie par $\mathbb{P}[X=1] = 1 - \mathbb{P}[X=0] = \frac{1}{4}$.
On considère le canal suivant :




$Y$ dépend de $X$ et du canal (+codage). Quelle est la loi de probabilité de $Y$ ?

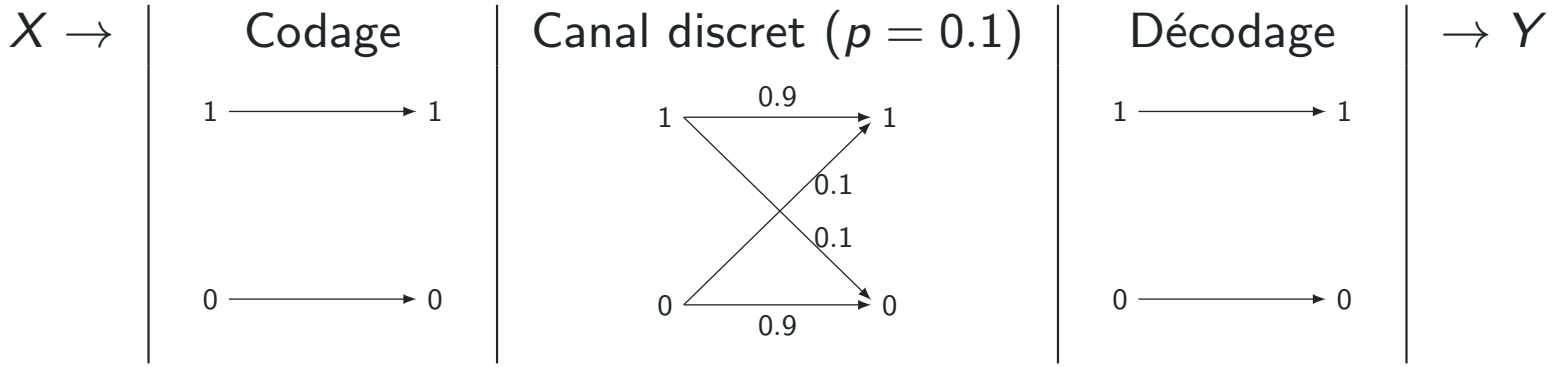
$$\begin{aligned}
\mathbb{P}[Y=1] &= \mathbb{P}[Y=1|X=0]\mathbb{P}[X=0] + \mathbb{P}[Y=1|X=1]\mathbb{P}[X=1] \\
&= 0.1 \times \frac{3}{4} + 0.9 \times \frac{1}{4} = \frac{3}{10} \\
\mathbb{P}[Y=0] &= \frac{7}{10}
\end{aligned}$$

Les lois de $X$ et $Y$ sont différentes !

# Notion d'information mutuelle

UNIVERSITÉ CAEN NORMANDIE

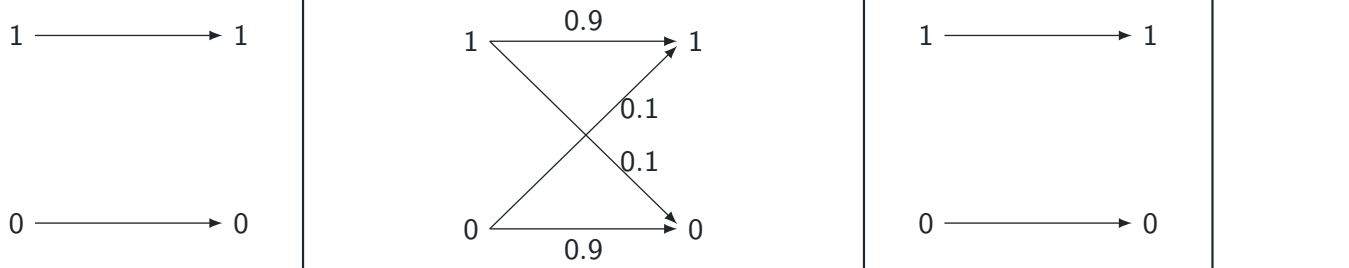
**Exemple :** On considère la source $X$ sur $\{0,1\}$ définie par $\mathbb{P}[X=1] = 1 - \mathbb{P}[X=0] = \frac{1}{4}$.
On considère le canal suivant :



# Notion d'information mutuelle

**Exemple :** On considère la source $X$ sur $\{0,1\}$ définie par $\mathbb{P}[X=1] = 1 - \mathbb{P}[X=0] = \frac{1}{4}$.
On considère le canal suivant :

$X \to$ | Codage | Canal discret ($p = 0.1$) | Décodage | $\to Y$




Allons plus loin, nous avons :

$$\mathbb{P}[X=0,Y=0] = \frac{27}{40}, \quad \mathbb{P}[X=0,Y=1] = \frac{3}{40}, \quad \mathbb{P}[X=1,Y=0] = \frac{1}{40},$$

$$\mathbb{P}[X=1,Y=1] = \frac{9}{40}, \qquad \mathbb{P}[Y=1] = 1 - \mathbb{P}[Y=0] = \frac{3}{10}$$

$$I(X,Y) = \sum_{i,j} p(x_i,y_j) \log_2 \frac{p(x_i,y_j)}{p(x_i)p(y_j)} \approx 0.41$$

# Notion d'information mutuelle

Soit $X$ et $Y$ deux variables aléatoires de distributions de probabilité $p(x)$ et $p(y)$ et soit $p(x, y)$ la distribution de la variable $(X, Y)$.

## Information mutuelle

L'**information mutuelle** $I(X, Y)$ est définie par

$$I(X, Y) = \sum_{i,j} p(x_i, y_j) \log_2 \frac{p(x_i, y_j)}{p(x_i)p(y_j)}.$$

# Notion d'information mutuelle

Soit $X$ et $Y$ deux variables aléatoires de distributions de probabilité $p(x)$ et $p(y)$ et soit $p(x, y)$ la distribution de la variable $(X, Y)$.

## Information mutuelle

L'**information mutuelle** $I(X, Y)$ est définie par

$$I(X, Y) = \sum_{i,j} p(x_i, y_j) \log_2 \frac{p(x_i, y_j)}{p(x_i)p(y_j)}.$$

## Théorème

$$\begin{aligned} I(X, Y) = I(Y, X) &= H(X) - H(X \mid Y) \\ &= H(Y) - H(Y \mid X) \\ &= H(X) + H(Y) - H(X, Y). \end{aligned}$$

$I(X, Y) \geq 0$, avec égalité si et seulement si $X$ et $Y$ sont indépendants.

**Preuve :** facile...

# Notion d'information mutuelle

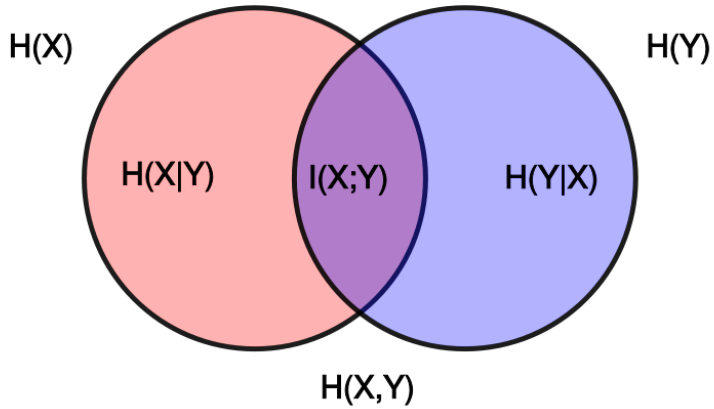
Soit $X$ et $Y$ deux variables aléatoires de distributions de probabilité $p(x)$ et $p(y)$ et soit $p(x, y)$ la distribution de la variable $(X, Y)$.

## Information mutuelle

L'**information mutuelle** $I(X, Y)$ est définie par

$$I(X, Y) = \sum_{i,j} p(x_i, y_j) \log_2 \frac{p(x_i, y_j)}{p(x_i)p(y_j)}.$$

- L'information mutuelle mesure la corrélation entre $X$ et $Y$




[wikipedia]

# Capacité d'un canal

UNIVERSITÉ CAEN NORMANDIE

On se fixe un canal $\mathcal{C}$.

Pour toute source $X$, le canal $\mathcal{C}$ transforme $X$ en une variable aléatoire $Y = \mathcal{C}(X)$.

## Capacité d'un canal

La **capacité $C$ d'un canal** $\mathcal{C}$ est le maximum de l'information mutuelle $I(X, Y)$, prise sur les distributions possibles de $X$ :

$$C = \sup_X I(X, Y) = \sup_X I(X, \mathcal{C}(X)) \geq 0.$$

# Capacité d'un canal

UNIVERSITÉ CAEN NORMANDIE

On se fixe un canal $\mathcal{C}$.

Pour toute source $X$, le canal $\mathcal{C}$ transforme $X$ en une variable aléatoire $Y = \mathcal{C}(X)$.

## Capacité d'un canal

La **capacité $C$ d'un canal** $\mathcal{C}$ est le maximum de l'information mutuelle $I(X, Y)$, prise sur les distributions possibles de $X$ :

$$C = \sup_X I(X, Y) = \sup_X I(X, \mathcal{C}(X)) \geq 0.$$

La capacité d'un canal est la quantité d'information maximale pouvant transiter par ce canal (notion similaire à l'entropie pour la compression sans perte).

**Remarque :** Le calcul de l'efficacité d'un canal est un calcul généralement difficile

# Capacité d'un canal

UNIVERSITÉ CAEN NORMANDIE

On se fixe un canal $\mathcal{C}$.

Pour toute source $X$, le canal $\mathcal{C}$ transforme $X$ en une variable aléatoire $Y = \mathcal{C}(X)$.

## Capacité d'un canal

La **capacité $C$ d'un canal** $\mathcal{C}$ est le maximum de l'information mutuelle $I(X, Y)$, prise sur les distributions possibles de $X$ :

$$C = \sup_X I(X, Y) = \sup_X I(X, \mathcal{C}(X)) \geq 0.$$

La capacité d'un canal est la quantité d'information maximale pouvant transiter par ce canal (notion similaire à l'entropie pour la compression sans perte).

**Remarque :** Le calcul de l'efficacité d'un canal est un calcul généralement difficile

**Exemple :** nous allons voir que la capacité du canal symétrique binaire est $C = \max_X I(X, Y) = 1 + p \log_2 p + (1 - p) \log_2(1 - p) = 1 - H(p)$.

# Canal discret sans mémoire

UNIVERSITÉ CAEN NORMANDIE

Un **canal de communication discret** est défini par un alphabet d'entrée $\{x_1, \ldots, x_n\}$, un alphabet de sortie $\{y_1, \ldots, y_m\}$, et une matrice de transition à $n$ lignes et $m$ colonnes :

$$\mathcal{M} = \begin{pmatrix} P(y_1 \mid x_1) & \ldots & P(y_m \mid x_1) \\ \vdots & & \vdots \\ P(y_1 \mid x_n) & \ldots & P(y_m \mid x_n) \end{pmatrix}$$

## Définitions

Le canal est **symétrique** si les lignes et les colonnes de sa matrice de transition sont identiques à permutation près.

Le canal est **sans mémoire** si les éléments envoyés sont indépendants. En d'autres termes, si pour tout $(x_1, \ldots, x_n)$ transmis et $(y_1, \ldots, y_n)$ reçu, alors on a $P(y_1, \ldots, y_n \mid x_1, \ldots, x_n) = P(y_1 \mid x_1) \ldots P(y_n|x_n)$.

# Exemple : le canal symétrique binaire

Un **canal symétrique binaire** est un canal de communication discret sans mémoire par lequel transite l'information bit à bit.

La probabilité d'erreur lors de la transmission est $p$ (pour 0 et pour 1). $p$ est donc le paramètre du canal.

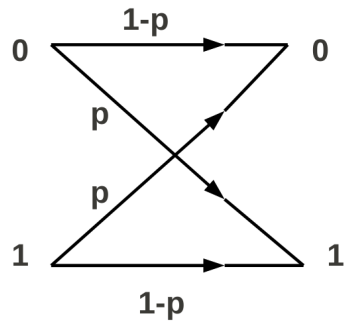



Figure – Canal symétrique binaire de paramètre $p$

**Exercice :** quelle est la matrice de transition de ce canal ?

**Remarque :** on peut supposer que $p < 0.5$. En effet, si $p > 0.5$, on prend le canal complémentaire de paramètre $1 - p < 0.5$. Enfin, si $p = 0.5$, le canal devient complètement aléatoire.

# Information mutuelle dans un canal symétrique

UNIVERSITÉ CAEN NORMANDIE

L'information mutuelle permet de mesurer l'information apportée sur $X$ par la connaissance de $Y$, transmise à travers un canal.

$I(X, Y) = H(Y) - H(Y \mid X)$

# Information mutuelle dans un canal symétrique

UNIVERSITÉ CAEN NORMANDIE

L'information mutuelle permet de mesurer l'information apportée sur $X$ par la connaissance de $Y$, transmise à travers un canal.

$$I(X,Y) = H(Y) - H(Y \mid X) = H(Y) + \sum_i \sum_j p(x_i, y_j) \log_2 p(y_j \mid x_i)$$

# Information mutuelle dans un canal symétrique

UNIVERSITÉ CAEN NORMANDIE

L'information mutuelle permet de mesurer l'information apportée sur $X$ par la connaissance de $Y$, transmise à travers un canal.

$I(X,Y) = H(Y) - H(Y \mid X) = H(Y) + \sum_i \sum_j p(x_i, y_j) \log_2 p(y_j \mid x_i)$
$= H(Y) + \sum_i \sum_j p(y_j \mid x_i) p(x_i) \log_2 p(y_j \mid x_i)$

# Information mutuelle dans un canal symétrique

UNIVERSITÉ CAEN NORMANDIE

L'information mutuelle permet de mesurer l'information apportée sur $X$ par la connaissance de $Y$, transmise à travers un canal.

$I(X,Y) = H(Y) - H(Y \mid X) = H(Y) + \sum_i \sum_j p(x_i, y_j) \log_2 p(y_j \mid x_i)$
$= H(Y) + \sum_i \sum_j p(y_j \mid x_i) p(x_i) \log_2 p(y_j \mid x_i)$
$= H(Y) + \sum_i p(x_i) \sum_j p(y_j \mid x_i) \log_2 p(y_j \mid x_i).$

# Information mutuelle dans un canal symétrique

UNIVERSITÉ CAEN NORMANDIE

L'information mutuelle permet de mesurer l'information apportée sur $X$ par la connaissance de $Y$, transmise à travers un canal.

$I(X,Y) = H(Y) - H(Y \mid X) = H(Y) + \sum_i \sum_j p(x_i, y_j) \log_2 p(y_j \mid x_i)$
$= H(Y) + \sum_i \sum_j p(y_j \mid x_i) p(x_i) \log_2 p(y_j \mid x_i)$
$= H(Y) + \sum_i p(x_i) \sum_j p(y_j \mid x_i) \log_2 p(y_j \mid x_i).$

**Si le canal est symétrique**, les lignes de la matrice $\mathcal{M}$ sont toutes égales à ordre près.

# Information mutuelle dans un canal symétrique

UNIVERSITÉ CAEN NORMANDIE

L'information mutuelle permet de mesurer l'information apportée sur $X$ par la connaissance de $Y$, transmise à travers un canal.

$I(X,Y) = H(Y) - H(Y \mid X) = H(Y) + \sum_i \sum_j p(x_i, y_j) \log_2 p(y_j \mid x_i)$
$= H(Y) + \sum_i \sum_j p(y_j \mid x_i) p(x_i) \log_2 p(y_j \mid x_i)$
$= H(Y) + \sum_i p(x_i) \sum_j p(y_j \mid x_i) \log_2 p(y_j \mid x_i).$

**Si le canal est symétrique**, les lignes de la matrice $\mathcal{M}$ sont toutes égales à ordre près. Ainsi la quantité $\sum_j p(y_j \mid x_i) \log_2 p(y_j \mid x_i)$ ne dépend pas de $i$ et est notée $-H(\mathcal{M})$.

# Information mutuelle dans un canal symétrique

UNIVERSITÉ CAEN NORMANDIE

L'information mutuelle permet de mesurer l'information apportée sur $X$ par la connaissance de $Y$, transmise à travers un canal.

$I(X,Y) = H(Y) - H(Y \mid X) = H(Y) + \sum_i \sum_j p(x_i, y_j) \log_2 p(y_j \mid x_i)$
$= H(Y) + \sum_i \sum_j p(y_j \mid x_i) p(x_i) \log_2 p(y_j \mid x_i)$
$= H(Y) + \sum_i p(x_i) \sum_j p(y_j \mid x_i) \log_2 p(y_j \mid x_i).$

**Si le canal est symétrique**, les lignes de la matrice $\mathcal{M}$ sont toutes égales à ordre près. Ainsi la quantité $\sum_j p(y_j \mid x_i) \log_2 p(y_j \mid x_i)$ ne dépend pas de $i$ et est notée $-H(\mathcal{M})$. Dans le cas d'un canal symétrique, on a donc $I(X;Y) = H(Y) - H(\mathcal{M})$.

**Exemple :** l'information mutuelle transmise à travers un canal symétrique binaire est : $I(X,Y) = H(Y) - H(p)$, où $H(p) = -p \log_2 p + (1-p) \log_2 (1-p)$.

# Information mutuelle dans un canal symétrique

UNIVERSITÉ CAEN NORMANDIE

L'information mutuelle permet de mesurer l'information apportée sur $X$ par la connaissance de $Y$, transmise à travers un canal.

$I(X,Y) = H(Y) - H(Y \mid X) = H(Y) + \sum_i \sum_j p(x_i, y_j) \log_2 p(y_j \mid x_i)$
$= H(Y) + \sum_i \sum_j p(y_j \mid x_i) p(x_i) \log_2 p(y_j \mid x_i)$
$= H(Y) + \sum_i p(x_i) \sum_j p(y_j \mid x_i) \log_2 p(y_j \mid x_i).$

**Si le canal est symétrique**, les lignes de la matrice $\mathcal{M}$ sont toutes égales à ordre près. Ainsi la quantité $\sum_j p(y_j \mid x_i) \log_2 p(y_j \mid x_i)$ ne dépend pas de $i$ et est notée $-H(\mathcal{M})$. Dans le cas d'un canal symétrique, on a donc $I(X;Y) = H(Y) - H(\mathcal{M})$.

**Exemple :** l'information mutuelle transmise à travers un canal symétrique binaire est : $I(X,Y) = H(Y) - H(p)$, où $H(p) = -p \log_2 p + (1-p) \log_2 (1-p)$.
Il faut maximiser $H(Y)$ pour maximiser $I(X,Y)$.

# Information mutuelle dans un canal symétrique

UNIVERSITÉ CAEN NORMANDIE

L'information mutuelle permet de mesurer l'information apportée sur $X$ par la connaissance de $Y$, transmise à travers un canal.

$I(X,Y) = H(Y) - H(Y \mid X) = H(Y) + \sum_i \sum_j p(x_i, y_j) \log_2 p(y_j \mid x_i)$
$= H(Y) + \sum_i \sum_j p(y_j \mid x_i) p(x_i) \log_2 p(y_j \mid x_i)$
$= H(Y) + \sum_i p(x_i) \sum_j p(y_j \mid x_i) \log_2 p(y_j \mid x_i)$.

**Si le canal est symétrique**, les lignes de la matrice $\mathcal{M}$ sont toutes égales à ordre près. Ainsi la quantité $\sum_j p(y_j \mid x_i) \log_2 p(y_j \mid x_i)$ ne dépend pas de $i$ et est notée $-H(\mathcal{M})$. Dans le cas d'un canal symétrique, on a donc $I(X;Y) = H(Y) - H(\mathcal{M})$.

**Exemple :** l'information mutuelle transmise à travers un canal symétrique binaire est : $I(X,Y) = H(Y) - H(p)$, où $H(p) = -p \log_2 p + (1-p) \log_2 (1-p)$.
Il faut maximiser $H(Y)$ pour maximiser $I(X,Y)$.
En prenant la distribution uniforme pour $X$, $Y$ suit une loi uniforme, $H(Y) = 1$ est maximum et $I(X,Y) = 1 - H(p)$.

# Information mutuelle dans un canal symétrique

UNIVERSITÉ CAEN NORMANDIE

L'information mutuelle permet de mesurer l'information apportée sur $X$ par la connaissance de $Y$, transmise à travers un canal.

$I(X,Y) = H(Y) - H(Y \mid X) = H(Y) + \sum_i \sum_j p(x_i, y_j) \log_2 p(y_j \mid x_i)$
$= H(Y) + \sum_i \sum_j p(y_j \mid x_i) p(x_i) \log_2 p(y_j \mid x_i)$
$= H(Y) + \sum_i p(x_i) \sum_j p(y_j \mid x_i) \log_2 p(y_j \mid x_i)$.

**Si le canal est symétrique**, les lignes de la matrice $\mathcal{M}$ sont toutes égales à ordre près. Ainsi la quantité $\sum_j p(y_j \mid x_i) \log_2 p(y_j \mid x_i)$ ne dépend pas de $i$ et est notée $-H(\mathcal{M})$. Dans le cas d'un canal symétrique, on a donc $I(X;Y) = H(Y) - H(\mathcal{M})$.

**Exemple :** l'information mutuelle transmise à travers un canal symétrique binaire est : $I(X,Y) = H(Y) - H(p)$, où $H(p) = -p \log_2 p + (1-p) \log_2 (1-p)$.
Il faut maximiser $H(Y)$ pour maximiser $I(X,Y)$.
En prenant la distribution uniforme pour $X$, $Y$ suit une loi uniforme, $H(Y) = 1$ est maximum et $I(X,Y) = 1 - H(p)$.
La capacité du canal symétrique binaire est donc $C = 1 + p \log_2 p + (1-p) \log_2 (1-p)$

# Codage et codes (binaires)

UNIVERSITÉ CAEN NORMANDIE

## Code en blocs et taux de transmission

Un **code en blocs** est un ensemble composé de $M$ mots de même longueur $n$ fixée ($\{0,1\}^n$ en pratique).
Le **taux de transmission** d'un code est défini par $R = (\log_2 M)/n$.

# Codage et codes (binaires)

UNIVERSITÉ CAEN NORMANDIE

## Code en blocs et taux de transmission

Un **code en blocs** est un ensemble composé de $M$ mots de même longueur $n$ fixée ($\{0,1\}^n$ en pratique).
Le **taux de transmission** d'un code est défini par $R = (\log_2 M)/n$.

L'algorithme de *codage* associe à un élément à envoyer un mot de code $c_1$ qui est ensuite transmis sur un canal.

A la réception, on *décode* l'élément reçu en un mot de code $c_2$ (si possible).

Tout s'est bien passé si $c_1 = c_2$.

# Codage et codes (binaires)

UNIVERSITÉ CAEN NORMANDIE

## Code en blocs et taux de transmission

Un **code en blocs** est un ensemble composé de $M$ mots de même longueur $n$ fixée ($\{0,1\}^n$ en pratique).
Le **taux de transmission** d'un code est défini par $R = (\log_2 M)/n$.

L'algorithme de *codage* associe à un élément à envoyer un mot de code $c_1$ qui est ensuite transmis sur un canal.

A la réception, on *décode* l'élément reçu en un mot de code $c_2$ (si possible).

Tout s'est bien passé si $c_1 = c_2$.

**Exemple :** on considère le *code à répétition* $0 \to 000,\ 1 \to 111$. On peut alors détecter et corriger une erreur. ($R = (\log_2 2)/3 = 0.33$)

# Codage et codes (binaires)

UNIVERSITÉ CAEN NORMANDIE

## Code en blocs et taux de transmission

Un **code en blocs** est un ensemble composé de $M$ mots de même longueur $n$ fixée ($\{0,1\}^n$ en pratique).
Le **taux de transmission** d'un code est défini par $R = (\log_2 M)/n$.

L'algorithme de *codage* associe à un élément à envoyer un mot de code $c_1$ qui est ensuite transmis sur un canal.

A la réception, on *décode* l'élément reçu en un mot de code $c_2$ (si possible).

Tout s'est bien passé si $c_1 = c_2$.

**Exemple :** on considère le *code à répétition* $0 \to 000,\ 1 \to 111$. On peut alors détecter et corriger une erreur. ($R = (\log_2 2)/3 = 0.33$)

## Taux d'erreur d'un code

Le **taux d'erreur** d'un code est la probabilité maximale $p(x)$ que le décodage ne s'effectue pas correctement quand $x$ est transmis, pour un canal donné.

Le taux d'erreur dépend du codage, du décodage et du canal ! ! !

# Deuxième théorème de Shannon

UNIVERSITÉ CAEN NORMANDIE

## Deuxième théorème de Shannon

On considère un canal discret sans mémoire, de capacité $C$. Alors pour tout $\delta > 0$ et pour tout $R < C$, il existe un code en bloc de $M$ mots de longueur $n$ et de taux de transmission $R = \log_2 M/n$ tel que la probabilité d'erreur est inférieure à $\delta$.

**Preuve (non constructive) :** admise.

Ce théorème dit qu'il existe des codes en bloc de taux d'erreur aussi faible que possible, dont le taux de transmission est proche (mais inférieure) de la capacité du canal.

**Exercice :** l'espace des mots avant encodage est $\{0,1\}^k$ et le code utilisé est le mot initial, complété avec $r$ symboles binaires de redondance. Donner la borne minimum sur $r$ en fonction de $C$ et $k$, à l'aide du second théorème de Shannon.

# Deuxième théorème de Shannon

UNIVERSITÉ CAEN NORMANDIE

## Deuxième théorème de Shannon

On considère un canal discret sans mémoire, de capacité $C$. Alors pour tout $\delta > 0$ et pour tout $R < C$, il existe un code en bloc de $M$ mots de longueur $n$ et de taux de transmission $R = \log_2 M/n$ tel que la probabilité d'erreur est inférieure à $\delta$.

**Preuve (non constructive) :** admise.

Ce théorème dit qu'il existe des codes en bloc de taux d'erreur aussi faible que possible, dont le taux de transmission est proche (mais inférieure) de la capacité du canal.

**Exercice :** l'espace des mots avant encodage est $\{0,1\}^k$ et le code utilisé est le mot initial, complété avec $r$ symboles binaires de redondance. Donner la borne minimum sur $r$ en fonction de $C$ et $k$, à l'aide du second théorème de Shannon.

**Réponse :** par définition du taux de transmission on a : $R = k/(k+r) < C$ donc $k + r > \frac{k}{C}$ donc $r > k(\frac{1}{C} - 1)$.

# Codes correcteurs d'erreurs et distance minimale

UNIVERSITÉ CAEN NORMANDIE

## Distance de Hamming

La **distance de Hamming** entre deux mots $x = (x_1, \ldots, x_n)$ et $y = (y_1, \ldots, y_n)$, notée $d_H(x, y)$, est le nombre d'indices $i$ tels que $x_i \neq y_i$.

**Remarque :** la distance de Hamming est une distance : $d_H(x, y) = 0 \Leftrightarrow x = y$, $d_H(x, y) = d_H(y, x)$ et $d_H(x, z) \leq d_H(x, y) + d_H(y, z)$.

# Codes correcteurs d'erreurs et distance minimale

UNIVERSITÉ CAEN NORMANDIE

## Distance de Hamming

La **distance de Hamming** entre deux mots $x = (x_1, \ldots, x_n)$ et $y = (y_1, \ldots, y_n)$, notée $d_H(x, y)$, est le nombre d'indices $i$ tels que $x_i \neq y_i$.

**Remarque :** la distance de Hamming est une distance : $d_H(x, y) = 0 \Leftrightarrow x = y$, $d_H(x, y) = d_H(y, x)$ et $d_H(x, z) \leq d_H(x, y) + d_H(y, z)$.

## $(n, K, d)$-code correcteur d'erreur

Un $(n, K, d)$-**code correcteur d'erreur** (binaire) est un code de longueur $n$, de cardinal $K$ et tel que pour tout mots de code $x, y$ la distance de Hamming vérifie $d_H(x, y) \geq d$. L'entier $d$ est appelé **distance minimale** du code.

# Codes correcteurs d'erreurs et distance minimale

UNIVERSITÉ CAEN NORMANDIE

## Distance de Hamming

La **distance de Hamming** entre deux mots $x = (x_1, \ldots, x_n)$ et $y = (y_1, \ldots, y_n)$, notée $d_H(x, y)$, est le nombre d'indices $i$ tels que $x_i \neq y_i$.

**Remarque :** la distance de Hamming est une distance : $d_H(x, y) = 0 \Leftrightarrow x = y$, $d_H(x, y) = d_H(y, x)$ et $d_H(x, z) \leq d_H(x, y) + d_H(y, z)$.

## $(n, K, d)$-code correcteur d'erreur

Un $(n, K, d)$-**code correcteur d'erreur** (binaire) est un code de longueur $n$, de cardinal $K$ et tel que pour tout mots de code $x, y$ la distance de Hamming vérifie $d_H(x, y) \geq d$. L'entier $d$ est appelé **distance minimale** du code.

**Exemple :** soit $C = \{(00000), (11000), (10110), (11111)\}$ un code de longueur 5 et de cardinal 4. Alors sa distance minimale est 2 (atteinte par exemple par $d_H((00000), (11000)) = 2$ et $d_H((10110), (11111)) = 2$).

# Capacité de correction d'un code

UNIVERSITÉ CAEN NORMANDIE

## Capacité de correction

Soit $C$ un code de distance minimale $d$. Alors on peut détecter au plus $d-1$ erreurs et corriger au plus $t = \lfloor (d-1)/2 \rfloor$ erreurs. $t$ est appelé **capacité de correction** du code.

**Preuve :**

# Capacité de correction d'un code

UNIVERSITÉ CAEN NORMANDIE

## Capacité de correction

Soit $C$ un code de distance minimale $d$. Alors on peut détecter au plus $d-1$ erreurs et corriger au plus $t = \lfloor (d-1)/2 \rfloor$ erreurs. $t$ est appelé **capacité de correction** du code.

**Preuve :** deux boules de rayon $\lfloor (d-1)/2 \rfloor$ centrées en deux mots de code distincts sont disjointes. Cela implique qu'un code de distance minimale $d$ peut corriger au plus $\lfloor (d-1)/2 \rfloor$ erreurs. Toute boule de rayon $d-1$ centrée en un mot de code ne contient aucun autre mot de code. Cela implique qu'un code de distance minimale $d$ peut détecter jusqu'à $d-1$ erreurs.

# Capacité de correction d'un code

UNIVERSITÉ CAEN NORMANDIE

## Capacité de correction

Soit $C$ un code de distance minimale $d$. Alors on peut détecter au plus $d-1$ erreurs et corriger au plus $t = \lfloor (d-1)/2 \rfloor$ erreurs. $t$ est appelé **capacité de correction** du code.

**Preuve :** deux boules de rayon $\lfloor (d-1)/2 \rfloor$ centrées en deux mots de code distincts sont disjointes. Cela implique qu'un code de distance minimale $d$ peut corriger au plus $\lfloor (d-1)/2 \rfloor$ erreurs. Toute boule de rayon $d-1$ centrée en un mot de code ne contient aucun autre mot de code. Cela implique qu'un code de distance minimale $d$ peut détecter jusqu'à $d-1$ erreurs.

**Rappel (dénombrement) :** Soit $x \in \{0,1\}^n$ et $B(x,r) = \{y \in \{0,1\}^n, \mid d_H(x,y) \leq r\}$ la boule centrée en $x$ de rayon $r$. Alors le nombre d'éléments contenus dans $B(x,r)$ est $\sum_{i=1}^{r} \binom{n}{i}$, où $\binom{n}{i}$ est un coefficient binomial.

En particulier, pour $x \in \{0,1\}^n$, on retrouve que le cardinal de $B(x,n) = 2^n$ (formule du binôme de Newton).

# Exemples de codes

UNIVERSITÉ CAEN NORMANDIE

**Exemple 1 :** soit $C = \{(00000), (11000), (10110), (11111)\}$ un code de longueur 5 et de cardinal 4. Alors sa distance minimale est 2 (atteinte par exemple par $d_H((00000), (11000)) = 2$ et $d_H((10110), (11111)) = 2$).

Sa capacité de correction est donc

# Exemples de codes

UNIVERSITÉ CAEN NORMANDIE

**Exemple 1 :** soit $C = \{(00000), (11000), (10110), (11111)\}$ un code de longueur 5 et de cardinal 4. Alors sa distance minimale est 2 (atteinte par exemple par $d_H((00000), (11000)) = 2$ et $d_H((10110), (11111)) = 2$).

Sa capacité de correction est donc $\lfloor (2-1)/2 \rfloor = 0$.

# Exemples de codes

UNIVERSITÉ CAEN NORMANDIE

**Exemple 1 :** soit $C = \{(00000), (11000), (10110), (11111)\}$ un code de longueur 5 et de cardinal 4. Alors sa distance minimale est 2 (atteinte par exemple par $d_H((00000), (11000)) = 2$ et $d_H((10110), (11111)) = 2$).

Sa capacité de correction est donc $\lfloor (2-1)/2 \rfloor = 0$.
En effet, si on reçoit 11110, il est impossible de savoir si le mot envoyé était 10110 ou 11111.

# Exemples de codes

UNIVERSITÉ CAEN NORMANDIE

**Exemple 1 :** soit $C = \{(00000), (11000), (10110), (11111)\}$ un code de longueur 5 et de cardinal 4. Alors sa distance minimale est 2 (atteinte par exemple par $d_H((00000), (11000)) = 2$ et $d_H((10110), (11111)) = 2$).

Sa capacité de correction est donc $\lfloor (2-1)/2 \rfloor = 0$.
En effet, si on reçoit 11110, il est impossible de savoir si le mot envoyé était 10110 ou 11111.

**Exemple 2 :** Soit $C$ le code à répétitions $C = \{(000), (111)\}$.
Longueur=
Distance minimale=
Capacité de correction=

# Exemples de codes

**Exemple 1 :** soit $C = \{(00000), (11000), (10110), (11111)\}$ un code de longueur 5 et de cardinal 4. Alors sa distance minimale est 2 (atteinte par exemple par $d_H((00000), (11000)) = 2$ et $d_H((10110), (11111)) = 2$).

Sa capacité de correction est donc $\lfloor (2-1)/2 \rfloor = 0$.
En effet, si on reçoit 11110, il est impossible de savoir si le mot envoyé était 10110 ou 11111.

**Exemple 2 :** Soit $C$ le code à répétitions $C = \{(000), (111)\}$.
Longueur$= 3$
Distance minimale$= 3$
Capacité de correction$= \lfloor (3-1)/2 \rfloor = 1$

# Ex : le code de Hamming (7, 4)

UNIVERSITÉ CAEN NORMANDIE

Le code de Hamming (7, 4) est un code en blocs, binaire, de longueur 7 qui peut coder $2^4 = 16$ messages (taux de transmission$= 4/7 \simeq 0.57$).

# Ex : le code de Hamming (7, 4)

UNIVERSITÉ CAEN NORMANDIE

Le code de Hamming (7, 4) est un code en blocs, binaire, de longueur 7 qui peut coder $2^4 = 16$ messages (taux de transmission$= 4/7 \simeq 0.57$).

Le code de Hamming (7,4) permet de détecter deux erreurs et de corriger une erreur survenue lors de la transmission d'un bloc.

# Ex : le code de Hamming (7, 4)

Le code de Hamming (7, 4) est un code en blocs, binaire, de longueur 7 qui peut coder $2^4 = 16$ messages (taux de transmission$= 4/7 \simeq 0.57$).

Le code de Hamming (7,4) permet de détecter deux erreurs et de corriger une erreur survenue lors de la transmission d'un bloc.

**Codage :** Pour coder le vecteur binaire $(m_1, m_2, m_3, m_4)$, on construit le vecteur binaire $x = (x_1, \ldots, x_7)$, tel que $x_3 = m_1$, $x_5 = m_2$, $x_6 = m_3$, $x_7 = m_4$.

Les bits $x_1, x_2, x_4$ sont calculés par les sommes de contrôle suivantes :
$x_1 + x_3 + x_5 + x_7 = 0 \bmod 2$, $x_2 + x_3 + x_6 + x_7 = 0 \bmod 2$ et
$x_4 + x_5 + x_6 + x_7 = 0 \bmod 2$.

# Ex : le code de Hamming (7, 4)

UNIVERSITÉ CAEN NORMANDIE

Le code de Hamming (7, 4) est un code en blocs, binaire, de longueur 7 qui peut coder $2^4 = 16$ messages (taux de transmission$= 4/7 \simeq 0.57$).

Le code de Hamming (7,4) permet de détecter deux erreurs et de corriger une erreur survenue lors de la transmission d'un bloc.

**Codage :** Pour coder le vecteur binaire $(m_1, m_2, m_3, m_4)$, on construit le vecteur binaire $x = (x_1, \ldots, x_7)$, tel que $x_3 = m_1$, $x_5 = m_2$, $x_6 = m_3$, $x_7 = m_4$.

Les bits $x_1, x_2, x_4$ sont calculés par les sommes de contrôle suivantes :
$x_1 + x_3 + x_5 + x_7 = 0 \bmod 2$, $x_2 + x_3 + x_6 + x_7 = 0 \bmod 2$ et
$x_4 + x_5 + x_6 + x_7 = 0 \bmod 2$.

Les vecteurs $x$ envoyés ont donc un nombre de 1 dans les positions 1,3,5,7 qui est pair, de même pour les positions 2,3,6,7 et 4,5,6,7.

L'ensemble des $2^4 = 16$ vecteurs $x$ possibles forment le code de Hamming (7,4).

# Décodage du code de Hamming (7, 4)

UNIVERSITÉ CAEN NORMANDIE

Soit $y = (y_1, \ldots, y_7)$ le mot binaire reçu après la transmission de $x$. On suppose qu'il y a une erreur dans la composante d'indice $i$ (et $i = i_0 + 2i_1 + 4i_2$ la décomposition de $i$ en base 2).

# Décodage du code de Hamming (7, 4)

Soit $y = (y_1, \ldots, y_7)$ le mot binaire reçu après la transmission de $x$. On suppose qu'il y a une erreur dans la composante d'indice $i$ (et $i = i_0 + 2i_1 + 4i_2$ la décomposition de $i$ en base 2).

S'il y a un nombre impair de 1 dans les composantes d'indices 1,3,5,7, alors l'erreur se situe sur un indice $i$ tel que $i_0 = 1$.
De même si il y a un nombre impair de 1 dans les composantes d'indices respectifs 2,3,6,7, et 4,5,6,7, alors l'erreur se situe sur un indice $i$ tel que $i_1 = 1$ et $i_2 = 1$ respectivement.

# Décodage du code de Hamming (7, 4)

UNIVERSITÉ CAEN NORMANDIE

Soit $y = (y_1, \ldots, y_7)$ le mot binaire reçu après la transmission de $x$. On suppose qu'il y a une erreur dans la composante d'indice $i$ (et $i = i_0 + 2i_1 + 4i_2$ la décomposition de $i$ en base 2).

S'il y a un nombre impair de 1 dans les composantes d'indices 1,3,5,7, alors l'erreur se situe sur un indice $i$ tel que $i_0 = 1$.
De même si il y a un nombre impair de 1 dans les composantes d'indices respectifs 2,3,6,7, et 4,5,6,7, alors l'erreur se situe sur un indice $i$ tel que $i_1 = 1$ et $i_2 = 1$ respectivement.

Ainsi pour décoder, on calcule le *syndrome* $s = (s_0, s_1, s_2)$ de $y$ défini par
$s_0 = y_1 + y_3 + y_5 + y_7 \mod 2$, $s_1 = y_2 + y_3 + y_6 + y_7 \mod 2$, $s_2 = y_4 + y_5 + y_6 + y_7 \mod 2$.
Si $s = (0, 0, 0)$ alors il n'y a pas d'erreur et si il y a une erreur, son indice est $s_0 + 2s_1 + 4s_2$.

# Décodage du code de Hamming (7, 4)

Soit $y = (y_1, \ldots, y_7)$ le mot binaire reçu après la transmission de $x$. On suppose qu'il y a une erreur dans la composante d'indice $i$ (et $i = i_0 + 2i_1 + 4i_2$ la décomposition de $i$ en base 2).

S'il y a un nombre impair de 1 dans les composantes d'indices 1,3,5,7, alors l'erreur se situe sur un indice $i$ tel que $i_0 = 1$.
De même si il y a un nombre impair de 1 dans les composantes d'indices respectifs 2,3,6,7, et 4,5,6,7, alors l'erreur se situe sur un indice $i$ tel que $i_1 = 1$ et $i_2 = 1$ respectivement.

Ainsi pour décoder, on calcule le *syndrome* $s = (s_0, s_1, s_2)$ de $y$ défini par
$s_0 = y_1 + y_3 + y_5 + y_7 \mod 2$, $s_1 = y_2 + y_3 + y_6 + y_7 \mod 2$, $s_2 = y_4 + y_5 + y_6 + y_7 \mod 2$.
Si $s = (0, 0, 0)$ alors il n'y a pas d'erreur et si il y a une erreur, son indice est $s_0 + 2s_1 + 4s_2$.

**Exemple :** Soit $m = (1, 0, 0, 0)$ le message à coder. Alors $x = (1, 1, 1, 0, 0, 0, 0)$.
Supposons que l'on reçoit le mot $y = (1, 1, 1, 0, 1, 0, 0)$. Son syndrome est $s = (1, 0, 1)$, l'erreur se trouve donc à la $1 + 0 + 4 = 5$ ème place.

# Borne d'empilement des spheres et exercice

## Borne d'empilement des spheres

Soit $C$ un $(n, K, d)$ code binaire de capacité de correction $t = \lfloor (d-1)/2 \rfloor$. Alors on a

$$K \sum_{r=0}^{t} \binom{n}{r} \leq 2^n.$$

**Preuve :** par définition de la capacité de correction d'un code, toutes les boules de rayon $t$ centrées en un mot de code sont deux à deux disjointes et il y a en tout $2^n$ éléments de longueur $n$.

**Exercice :**

# Borne d'empilement des spheres et exercice

UNIVERSITÉ CAEN NORMANDIE

## Borne d'empilement des spheres

Soit $C$ un $(n, K, d)$ code binaire de capacité de correction $t = \lfloor (d-1)/2 \rfloor$. Alors on a

$$K \sum_{r=0}^{t} \binom{n}{r} \leq 2^n.$$

**Preuve :** par définition de la capacité de correction d'un code, toutes les boules de rayon $t$ centrées en un mot de code sont deux à deux disjointes et il y a en tout $2^n$ éléments de longueur $n$.

**Exercice :** Combien de mots de code un code de longueur 11 et de distance minimale 3 peut il contenir au maximum ? Quelle est la longueur minimale d'un code de 32 mots pouvant corriger 2 erreurs ? Quelle est la plus grande distance minimale possible pour un code de longueur 13 et de cardinal 64 ?

# Borne d'empilement des spheres et exercice

UNIVERSITÉ CAEN NORMANDIE

## Borne d'empilement des spheres

Soit $C$ un $(n, K, d)$ code binaire de capacité de correction $t = \lfloor (d-1)/2 \rfloor$. Alors on a

$$K \sum_{r=0}^{t} \binom{n}{r} \leq 2^n.$$

**Preuve :** par définition de la capacité de correction d'un code, toutes les boules de rayon $t$ centrées en un mot de code sont deux à deux disjointes et il y a en tout $2^n$ éléments de longueur $n$.

**Exercice :** Combien de mots de code un code de longueur 11 et de distance minimale 3 peut il contenir au maximum ? Quelle est la longueur minimale d'un code de 32 mots pouvant corriger 2 erreurs ? Quelle est la plus grande distance minimale possible pour un code de longueur 13 et de cardinal 64 ?

**Réponses :** 170, 12 et 3.

# Rayon de recouvrement et codes parfaits

UNIVERSITÉ CAEN NORMANDIE

## Rayon de recouvrement

le **rayon de recouvrement** d'un code binaire de longueur $n$ est le plus petit entier $r$ tel que l'ensemble des boules de rayon $r$ centrées en chaque mot de code forment un recouvrement de $\{0,1\}^n$.

**Remarque :** le rayon de recouvrement est toujours supérieur ou égal à la capacité de correction d'un code.

# Rayon de recouvrement et codes parfaits

UNIVERSITÉ CAEN NORMANDIE

## Rayon de recouvrement

le **rayon de recouvrement** d'un code binaire de longueur $n$ est le plus petit entier $r$ tel que l'ensemble des boules de rayon $r$ centrées en chaque mot de code forment un recouvrement de $\{0,1\}^n$.

**Remarque :** le rayon de recouvrement est toujours supérieur ou égal à la capacité de correction d'un code.

## Code parfait

Si le rayon de recouvrement d'un code est égal à sa capacité de correction, alors le code est dit **parfait**.

**Remarque :** pour un code parfait, on obtient une égalité dans la borne d'empilement des sphères (immédiat)

# Rayon de recouvrement et codes parfaits

UNIVERSITÉ CAEN NORMANDIE

## Rayon de recouvrement

le **rayon de recouvrement** d'un code binaire de longueur $n$ est le plus petit entier $r$ tel que l'ensemble des boules de rayon $r$ centrées en chaque mot de code forment un recouvrement de $\{0,1\}^n$.

**Remarque :** le rayon de recouvrement est toujours supérieur ou égal à la capacité de correction d'un code.

## Code parfait

Si le rayon de recouvrement d'un code est égal à sa capacité de correction, alors le code est dit **parfait**.

**Remarque :** pour un code parfait, on obtient une égalité dans la borne d'empilement des sphères (immédiat)

**Exemple (voir TP) :** le code de Hamming (7,4) binaire est parfait. En effet, il y a $2^4 = 16$ mots de codes et la capacité de correction est 1. On a donc une égalité pour la borne d'empilement des sphères car $16 \times (1+7) = 128 = 2^7$.

Le code à répétition $\{(000),(111)\}$ est lui aussi parfait.

# Conclusion du cours de théorie de l'information

UNIVERSITÉ CAEN NORMANDIE

La théorie de l'information est utilisée dans de nombreuses branches de l'informatique avec des applications fortes comme la compression des données ou les codes correcteurs d'erreurs (construits à l'aide de l'algèbre linéaire et des corps finis), mais aussi en algorithmique, en biométrie, ...

Plus de détails et d'applications dans le cours de Transmission d'information : compression, codage en 3A (tronc commun).

**Examen :** un partiel indépendant de la partie optimisation. Session 2 : commun avec la partie optimisation.

**TP** (1/2 de la note de TP de mathématiques pour l'informatique), programmation en langage C :

- Premier TP (2h) : codes de Huffman
- Deuxième TP (2h) : codes de Hamming