# Théorie de l'information et du codage

Marc Lelarge
(revu par Anne Bouillard)

Notes de cours pour l'année 2014-2015

## Table des matières

**1 Entropie, suites typiques et compression de données avec pertes** **4**
1.1 Définition de l'entropie . . . . . 4
1.2 Ensemble typique . . . . . 4
1.3 Quelques propriétés de l'entropie . . . . . 7
1.4 Entropie conditionnelle et information mutuelle . . . . . 8

**2 Codage pour des sources discrètes** **10**
2.1 Mots code de longueur variable . . . . . 10
2.2 Un théorème de codage de source . . . . . 12
2.3 Un codage optimal : le codage de Huffman . . . . . 13

# Introduction

La théorie de l'information répond à deux questions fondamentales :

1. Quel est le taux de compression maximal pour des données ? (notion d'entropie $H$)
2. Quel est le taux de transmission maximal d'un canal bruité ? (notion de capacité de canal $C$).

La théorie de l'information et du codage est donc une part de la théorie de la communication.

Mais d'autres domaines utilisent les notions de la théorie de l'information :

- **Théorie de la complexité :** la complexité de Kolmogorov s'intéresse à la question fondamentale « étant donné un mot, quel est le programme le plus court (en binaire) pour calculer ce mot ?». C'est le problème de la description la plus courte du mot. Cette complexité $K$ est proche de $H$.
- **Thermodynamique :** la notion d'entropie joue un rôle central ;
- **Statistique :** On utilise la notion d'information mutuelle pour faire par exemple des tests d'hypothèse.

### Quelques exemples

**Nombre de questions pour deviner une valeur** Soit $\Omega$ un ensemble fini. le but est de deviner une valeur $x \in \Omega$ en posant des questions (dont la réponse est `oui` ou `non`).

Par exemple, si $\Omega = [\![1, 32]\!]$, par une recherche dichotomique, on peut trouver un nombre en $\log_2(32) = 5$ questions.

Si maintenant, on connaît la distribution avec laquelle $x$ est choisi. Par exemple, $\Omega = [\![1, 8]\!]$ avec la distribution $p = (1/2, 1/4, 1/8, 1/16, 1/64, 1/64, 1/64, 1/64)$. En faisant une dichotomie indépendante de la distribution, on pose 3 questions.

On peut aussi choisir de poser ces questions (dans l'ordre jusqu'à ce que la valeur soit devinée) :

1. « Est-ce 1 ? » (avec probabilité 1/2, la réponse est `oui` et la valeur devinée) ;
2. « Est-ce 2 ? »
3. « Est-ce 3 ? »
4. « Est-ce 4 ? »
5. « Est-ce 5 ou 6 ? »
6. « Est-ce 5 ? » (si la dernière réponse est `oui`) et « Est-ce 7 ? » sinon.

Il y a un maximum de 6 questions, mais en moyenne, le nombre de questions posées est

$$\frac{1}{2} + 2\frac{1}{4} + 3\frac{1}{8} + 4\frac{1}{16} + 4 \cdot 6\frac{1}{64} = 2$$

La formule utilisée pour ce calcul est

$$\sum_{x \in \Omega} p(x) \log \frac{1}{p(x)} \triangleq H(X),$$

et $H(X)$ est l'**entropie** d'une variable aléatoire $X$ distribuée selon $p$.

Si on veut encoder les différentes valeurs de $\Omega$ en binaire, on peut utiliser le codage suivant pour les valeurs respectives de $x \in [\![1, 8]\!]$ : $0, 10, 110, 1110, 111100, 111101, 111110, 111111$, et la longueur moyenne du code d'un entier est 2.

**Canal de communication** Un canal de communication prend un mot en entrée et retourne un mot en sortie. Si le canal est parfait, il ne fait que retransmettre les caractères reçus. Mais le canal est bruité, ce qui signifie que le mot retourné n'est pas nécessairement le mot attendu. On veut définir une notion de capacité : la quantité d'information que l'on est capable de reconstruire en fonction du message reçu.

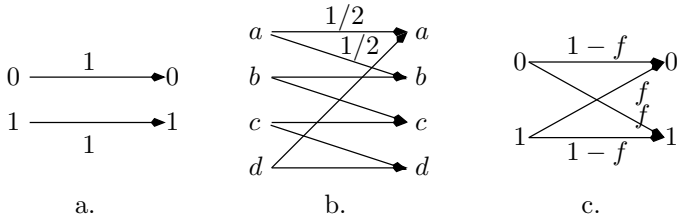


FIGURE 1 – Canals de communication : a) canal non bruité ; b) canal bruité à décalage ; c) canal bruité symétrique

Si le canal n'est pas bruité (figure 1-a), alors la capacité est 1. Si le canal est sur l'alphabet $\{a, b, c, d\}$ et que la transmission se fait selon le schéma de la figure 1-b, c'est-à-dire que si lorsque $a$ est reçu alors $a$ ou $b$ est renvoyé, chacun avec probabilité $1/2$, et ainsi de suite, on peut n'envoyer que les lettres $a$ et $c$. Ainsi, si le canal renvoie $a$ ou $b$, on sait que $a$ a été envoyé, et si $c$ ou $d$ est reçu, on sait que c'était $c$. Ainsi, la capacité du canal est aussi 1.

Considérons maintenant le canal de la figure 1-c. Il s'agit du canal binaire bruité symétrique. La probabilité d'erreur de transmission est $f = 0.1$ : $\mathbf{P}(Y = 0 \mid X = 1) = \mathbf{P}(Y = 1 \mid X = 0) = f$.

Quelle est la probabilité d'erreur si on utilise un code à répétition ? (chaque caractère est envoyé 3 fois, et on décode le terme majoritaire).

La probabilité qu'il y ait deux erreurs de transmission parmi les trois fois où est envoyé un caractère est $f^3 + 3f^2(1 - f)$. Donc on a une erreur de probabilité de $\approx 0.03$ pour une capacité de $1/3$ dans ce cas. On va montrer que l'on peut faire beaucoup mieux. En fait, on peut atteindre la capacité

$$1 + f \log f + (1 - f) \log(1 - f) = 1 - H(f)$$

avec une probabilité d'erreur arbitrairement faible. Cette quantité correspond à l'**information mutuelle** maximale entre le mot en entrée et le mot en sortie.

À titre d'exemple, si $f = 0$, alors on retrouve une capacité 1, mais si $f = 0.5$, on trouve une capacité 0 : le mot en sortie est indépendant du mot en entrée.

# 1 Entropie, suites typiques et compression de données avec pertes

## 1.1 Définition de l'entropie

Intuitivement, l'entropie mesure l'incertitude d'une variable aléatoire (v.a.). Une source (discrète) émet une suite de v.a. $\{U_i\}_{i=1}^{\infty}$ à valeurs dans un ensemble fini $\mathcal{U}$ appelé l'**alphabet de la source**. Si les $U_i$ sont indépendants et identiquement distribués (i.i.d.) de loi $P$, la source est dite **sans mémoire de distribution** $P$.

**Définition 1.** *Soit $U$ une variable aléatoire à valeurs dans un ensemble fini $\mathcal{U}$, de distribution de probabilité :*

$$p(u) = \mathbf{P}(U = u),\ u \in \mathcal{U}\,.$$

*Son entropie est*

$$H(U) \stackrel{\Delta}{=} -\mathbf{E}\left[\log(p(U))\right] = -\sum_{u\in\mathcal{U}} p(u)\log p(u)\,,$$

*avec la convention $0\log 0 = 0$.*

On remarque que $H(U) \geq 0$ : pour tout $u \in \mathcal{U}$, $p(u) \leq 1$.

**Exemple (Entropie d'une variable Bernoulli).** Si $U \sim \mathcal{B}\mathrm{er}(p)$, alors

$$H(U) = -p\log p - (1-p)\log(1-p) \stackrel{\Delta}{=} H(p).$$

Si $p \in \{0,1\}$, alors $H(p) = 0$ et $H(p)$ est maximisée pour $p = 1/2$, avec $H(1/2) = 1$.

Le choix de la base du logarithme correspond à un choix d'unité. Sauf mention du contraire, on choisit par défaut la base 2. L'entropie s'exprime alors en bits.

## 1.2 Ensemble typique

Soient $U_1, U_2, \ldots, U_n$ des v.a. i.i.d. de distribution $p$. On va montrer que l'on peut séparer les différentes suites de valeurs en deux ensembles : les suites **typiques**, dont la probabilité est proche de $2^{-H(U_1,\ldots,U_n)}$ et les autres. En effet, une suite $(u_1,\ldots,u_n)$ caractéristique de la distribution aura un nombre d'occurrences de $u \in \mathcal{U}$ proche de $np(u)$. La probabilité d'un tel élément est donc, grossièrement,

$$p(u_1,\ldots,u_n) = p(u_i)\cdots p(u_n) \approx \prod_{u\in\mathcal{U}} p(u)^{np(u)} = 2^{-nH(U)}.$$

**Définition 2.** *Pour $n \in \mathbb{N}$ et $\delta > 0$, l'ensemble typique $A_\delta^{(n)}$ par rapport à la distribution $p(u)$ est l'ensemble des suites $(u_1,\ldots,u_n) \in \mathcal{U}^n$ telles que :*

$$2^{-n(H(U)+\delta)} \leq p(u_1,\ldots,u_n) \leq 2^{-n(H(U)-\delta)}.$$

Toutes les suites typiques ont donc « à peu près » la même probabilité.

**Théorème 1.** *Pour tout $n \in \mathbb{N}$ et tout $\delta > 0$, l'ensemble typique satisfait les propriétés suivantes.*

*(i) Pour tout* $(u_1, \ldots, u_n) \in A_\delta^{(n)}$ *alors*

$$H(U) - \delta \leq -\frac{1}{n} \log p(u_1, \ldots, u_n) \leq H(U) + \delta.$$

*(ii) Pour tout* $\epsilon > 0$ *et pour* $n$ *suffisamment grand,*

$$\mathbf{P}(A_\delta^{(n)}) = \mathbf{P}\big((U_1, \ldots, U_n) \in A_\delta^{(n)}\big) \geq 1 - \epsilon.$$

*(iii) Le cardinal de l'ensemble* $A_\delta^{(n)}$ *est borné par*

$$\left|A_\delta^{(n)}\right| \leq 2^{n(H(U)+\delta)},$$

*et pour tout* $\epsilon > 0$*, pour* $n$ *suffisamment grand, ce cardinal est minoré par*

$$\left|A_\delta^{(n)}\right| \geq (1-\epsilon)2^{n(H(U)-\delta)}.$$

**Remarque 1.** *Le point 3. du théorème entraîne directement*

$$H(U) - \delta \leq \liminf_{n\to\infty} \frac{\log\left|A_\delta^{(n)}\right|}{n} \leq \limsup_{n\to\infty} \frac{\log\left|A_\delta^{(n)}\right|}{n} \leq H(U) + \delta.$$

*Démonstration.* $(i)$ c'est une application directe de la définition.
$(ii)$ On rappelle la **loi faible des grands nombres** : si $(X_i)$ est une suite de v.a. i.i.d. d'espérance finie, et $\bar{X}_n = \sum_{i=1}^n X_i$, alors pour tout $\epsilon > 0$,

$$\mathbf{P}\left(\left|\frac{\bar{X}_n}{n} - \mathbf{E}[X_1]\right| > \epsilon\right) \overset{n\to\infty}{\longrightarrow} 0.$$

Or, grâce à l'indépendance des variables $U_i$, on peut écrire :

$$-\frac{1}{n}\log p\,(U_1, U_2, \ldots, U_n) = -\frac{1}{n}\sum_{i=1}^n \log p(U_i),$$

qui est une somme de v.a. i.i.d. d'espérance $-\mathbf{E}\left[\log p(U)\right] = H(U)$. En posant $X_i = \log p(U_i)$,
Alors, pour tout $\epsilon > 0$

$$\begin{aligned}\mathbf{P}\big((U_1, \ldots, U_n) \in A_\delta^{(n)}\big) &= \mathbf{P}\left(|\bar{X}_n/n - H(U)| \leq \delta\right) \\ &= 1 - \mathbf{P}\left(|\bar{X}_n/n - H(U)| > \delta\right) \geq 1 - \epsilon\end{aligned}$$

pour $n$ assez grand.
$(iii)$ Tout d'abord,

$$1 = \sum_{(u_1,\ldots,u_n)\in\mathcal{U}^n} p(u_1, \ldots, u_n) \geq \sum_{(u_1,\ldots,u_n)\in A_\delta^{(n)}} p(u_1, \ldots, u_n) \geq \left|A_\delta^{(n)}\right| 2^{-n(H(U)+\delta)},$$

où la dernière inégalité provient de la définition de l'ensemble typique, ce qui prouve la première inégalité.

D'autre part, en appliquant $(ii)$ pour $n$ suffisamment grand :

$$1 - \epsilon \leq \mathbf{P}\left(A_\delta^{(n)}\right) \leq \left|A_\delta^{(n)}\right| 2^{-n(H(U)-\delta)},$$

ce qui prouve la seconde inégalité. □

### 1.2.1 Application : codage de source avec perte

Dans cette section, nous considérons une notion très générale de codage qui sera précisée un peu plus loin.

Un **codage binaire** (ou $(n,k)$-code) est une paire de fonctions $(f,\phi)$

$$\begin{array}{ll} f : \mathcal{U}^k \to \{0,1\}^n, & \text{fonction de codage} \\ \phi : \{0,1\}^n \to \mathcal{U}^k, & \text{fonction de décodage.} \end{array}$$

Pour une source donnée, la **probabilité d'erreur du code** $(f,\phi)$ est

$$e(f,\phi) := \mathbf{P}(\phi(f(U^{(k)})) \neq U^{(k)}),$$

où $U^{(k)} = (U_1, \ldots, U_k)$ est la suite des $k$ premiers symboles émis par la source. C'est donc la probabilité que la suite encodée puis décodée soit différente de la suite d'origine.

On souhaite trouver des codes dont le rapport $n/k$ est petit et dont probabilité d'erreur est également petite. Plus précisément, pour tout $k$, soit $n(k,\epsilon)$ le plus petit entier $n$ tel qu'il existe un $(k,n)$-code satisfaisant $e(f,\phi) \leq \epsilon$.

**Théorème 2.** *Considérons une source discrète sans mémoire de distribution* $\mathbf{P}(U = u) = p(u)$*. Alors, pour tout* $\epsilon \in (0,1)$ *:*

$$\lim_{k\to\infty} \frac{n(k,\epsilon)}{k} = H(U) = -\sum_{u\in\mathcal{U}} p(u)\log p(u).$$

*Démonstration.* Soit $(f,\phi)$ un $(n,k)$-code. On peut définir l'ensemble $A = \{u^{(k)} \in \mathcal{U}^k \mid \phi(f(u^{(k)})) = u^{(k)}\}$. On a alors nécessairement $|A| \leq 2^n$ et l'équivalence $e(f,\phi) \leq \epsilon \Rightarrow \mathbf{P}(A) \geq 1-\epsilon$.

On peut donc rechercher un ensemble $A \subseteq \mathcal{U}^k$ de taille tel que $\mathbf{P}(A) \geq 1-\epsilon$. On pourra alors déduire du cardinal de cet ensemble $n(k,\epsilon)$ : on aura $n(k,\epsilon) = \lceil \log |A| \rceil$.

Soit $s(k,\epsilon)$ la taille minimale d'un ensemble $A \subseteq \mathcal{U}^k$ avec $P(A) \geq 1-\epsilon$ :

$$s(k,\epsilon) = \min\{|A| \mid P(A) \geq 1-\epsilon\}.$$

Pour prouver le théorème, il suffit de montrer que pour $\epsilon \in (0,1)$,

$$\lim_{k\to\infty} \frac{\log s(k,\epsilon)}{k} = H(U). \tag{1}$$

Pour tout $\delta > 0$, en prenant $A = A_\delta^{(k)}$ l'ensemble typique pour la source $p(u)$, on a pour $k$ suffisamment grand $\mathbf{P}\left(A_\delta^{(k)}\right) \geq 1-\epsilon$ et donc :

$$s(k,\epsilon) \leq |A_\delta^{(k)}| \leq 2^{k(H(U)+\delta)},$$

donc

$$\limsup_{k\to\infty} \frac{\log s(k,\epsilon)}{k} \leq H(U). \tag{2}$$

Inversement, pour tout $A \subseteq \mathcal{U}^k$ tel que $\mathbf{P}(A) \geq 1-\epsilon > 0$, le point $(ii)$ du théorème 1 implique que pour $k$ suffisamment grand $\mathbf{P}\left(A_\delta^{(k)}\right) \geq \frac{1+\epsilon}{2}$ et donc

$$\mathbf{P}(A \cap A_\delta^{(k)}) \geq \mathbf{P}(A) - (1 - \mathbf{P}\left(A_\delta^{(k)}\right)) \geq \frac{1-\epsilon}{2}.$$

On a donc par définition de $A_\delta^{(k)}$,

$$|A| \geq |A \cap A_\delta^{(k)}| \geq \sum_{u^{(k)} \in A \cap A_\delta^{(k)}} p(u^{(k)}) 2^{k(H(U)-\delta)} \geq \frac{1-\epsilon}{2} 2^{k(H(U)-\delta)},$$

et donc pour tout $\delta > 0$,

$$\liminf_{k\to\infty} \frac{1}{k} \log s(k,\epsilon) \geq H(U) - \delta.$$

Ceci, avec (2), implique (1). □

### 1.3 Quelques propriétés de l'entropie

**Lemme 1.** *Soit $(p_i)_{1\leq i\leq n}$ une distribution de probabilité sur $[\![1,n]\!]$. Pour toute distribution $(q_i)_{1\leq i\leq n}$ sur $[\![1,n]\!]$, on définit*

$$G(q_1,\ldots,q_n) = -\sum_{1\leq i\leq n} p_i \log q_i.$$

*Le minimum de cette fonction est atteint pour la distribution $p$ uniquement.*

*Démonstration.* On utilise la concavité de la fonction log, ce qui entraîne $\log z \leq (z-1)\log e$. Cette inégalité est une égalité uniquement lorsque $z = 1$. Alors pour tout $i \in [\![1,n]\!]$,

$$\log\left(\frac{q_i}{p_i}\right) \leq \left(\frac{q_i}{p_i} - 1\right) \log e,$$

avec égalité si et seulement si $q_i = p_i$. On a donc

$$G(p_1,\ldots,p_n) - G(q_1,\ldots,q_n) = \sum_i p_i \log\left(\frac{q_i}{p_i}\right) \leq \log e \sum_i p_i(\frac{q_i}{p_i} - 1) = \log e \sum_i (q_i - p_i) = 0.$$

□

On en déduit facilement les théorème suivants :

**Théorème 3.** *Pour tout $n \in \mathbb{N}$, $H(p_1,\ldots,p_n) \leq \log n$, avec égalité si et seulement si $p_1 = p_2 = \cdots = p_n = 1/n$.*

*Démonstration.* On applique le lemme précédent à la distribution $p$ et à la distribution uniforme :

$$H(p_1,\ldots,p_n) = G(p_1,\ldots,p_n) \leq G(1/n,\ldots,1/n) = \log n,$$

avec égalité si et seulement si $p_i = 1/n$ pour tout $1 \leq i \leq n$. □

**Théorème 4.** *Si $X$ et $Y$ sont des v.a. (discrètes) alors $H(X,Y) \leq H(X) + H(Y)$, avec égalité si et seulement si $X$ et $Y$ sont indépendantes.*

L'entropie d'une paire $(X,Y)$ ne nécessite pas de nouvelle définition. On notera $H((X,Y)) = H(X,Y)$.

*Démonstration.* On a

$$\begin{aligned} H(X)+H(Y) &= -\sum_x p(x)\log p(x) - \sum_y p(y)\log p(y) \\ &= -\sum_{x,y} p(x,y)\log p(x) - \sum_{x,y} p(x,y)\log p(y) \\ &= -\sum_{x,y} p(x,y)\log p(x)p(y). \end{aligned}$$

On applique le lemme 1 à la distribution de $(X,Y)$ et à la distribution $(p(x)p(y)$, ce qui donne

$$H(X,Y) = -\sum_{x,y} p(x,y)\log p(x,y) \leq -\sum_{x,y} p(x,y)\log p(x)p(y) = H(X)+H(Y),$$

avec égalité si et seulement si $p(x,y) = p(x)p(y)$, c'est-à-dire si $X$ et $Y$ sont indépendantes. □

### 1.4 Entropie conditionnelle et information mutuelle

Étant donné une v.a. $X$ sur un espace de probabilité $\Omega$ et $A$ un événement dans $\Omega$, on définit l'**entropie conditionnelle de $X$ sachant $A$** par

$$H(X|A) \stackrel{\Delta}{=} -\sum_{x\in\Omega} \mathbf{P}(X=x \mid A)\log \mathbf{P}(X=x \mid A).$$

De la même manière si $Y$ est une autre v.a. sur $\Omega$, on définit l'**entropie conditionnelle de $X$ sachant $Y$** par

$$\begin{aligned} H(X|Y) &\stackrel{\Delta}{=} \sum_{y\in\Omega} H(X \mid Y=y)\mathbf{P}(Y=y) \\ &= -\sum_{y\in\Omega} \mathbf{P}(X=x \mid Y=y)\mathbf{P}(Y=y)\log \mathbf{P}(X=x \mid Y=y) \\ &= -\sum_{x,y\in\Omega} p(x,y)\log p(x|y), \end{aligned}$$

avec la notation $p(x,y) = \mathbf{P}(X=x, Y=y)$ et $p(x|y) = \mathbf{P}(X=x \mid Y=y)$.

**Proposition 1.** *$H(X|Y) = 0$ si et seulement si $X = g(Y)$ pour une fonction $g$. En particulier, $H(X|X) = 0$.*

*Démonstration.* On peut écrire $H(X|Y) = -\sum_{x,y\in\Omega} p(x,y)\log p(x|y)$. Pour chaque terme, soit $x = g(y)$ et $\log p(x|y) = \log 1 = 0$, soit $x \neq g(y)$ et $p(x|y) = p(x,y) = 0$ et on a pris la convention $0\log 0 = 0$.

Réciproquement, s'il n'existe pas $g$ tel que $X = g(Y)$, alors il existe $x, y$ tel que $p(x|y) \notin \{0,1\}$ et au moins un des termes de la somme précédente est strictement positif. □

La différence $H(X,Y) - H(X)$ mesure la quantité d'information supplémentaire sur le couple $(X,Y)$ donnée par $Y$ si $X$ est déjà connu. Comme montré dans le théorème suivant, cette différence est l'entropie conditionnelle de $Y$ sachant $X$.

**Théorème 5.** *Pour toute paire de v.a. $X, Y$, on a $H(X,Y) = H(Y) + H(X|Y)$.*

*Démonstration.* Il suffit d'écrire la définition de l'entropie et la définition des probabilité conditionnelles.

$$\begin{aligned} H(X,Y) &= -\sum_{x,y} p(x,y)\log p(x,y) \\ &= -\sum_{x,y} p(x,y)\log p(y)p(x|y) \\ &= -\sum_{y} p(y)\log p(y) - \sum_{x,y} p(x,y)\log p(x|y), \end{aligned}$$

ce qui est l'égalité souhaitée. □

**Corollaire 1.** *Pour toute paire de v.a. $X, Y$, $H(X|Y) \leq H(X)$ avec égalité si et seulement si $X$ et $Y$ sont indépendantes.*

*Démonstration.* On a $H(X|Y) = H(X,Y) - H(Y)$ et $H(X,Y) \leq H(X) + H(Y)$ avec égalité si et seulement si $X$ et $Y$ sont indépendantes. Le résultat en découle. □

**Définition 3.** *L'information mutuelle entre $X$ et $Y$ est définie par*

$$I(X;Y) \triangleq H(Y) - H(Y|X) = H(X) - H(X|Y) = H(X) + H(Y) - H(X,Y).$$

L'information mutuelle entre $X$ et $Y$ correspond à la diminution d'incertitude sur $Y$ causée par la connaissance de $X$, c'est-à-dire la quantité d'information sur $Y$ contenue *dans* $X$. Elle est symétrique en $X$ et $Y$.

# 2 Codage pour des sources discrètes

## 2.1 Mots code de longueur variable

On pose $\mathcal{D} = \{0, 1, \ldots, D-1\}$ un alphabet, et $\mathcal{D}^*$ est l'ensemble des mots finis sur sur $\mathcal{D}$.

**Définition 4.** *Un* **code de source** $C$ *pour une variable aléatoire* $U \in \mathcal{U}$ *est une fonction de* $\mathcal{U}$ *vers* $\mathcal{D}^*$. $C(u)$ *est le mot code correspondant à* $u$, *et* $\ell(u)$ *sa longueur.*

**Définition 5.** *La longueur moyenne* $L(C)$ *d'un code pour la variable aléatoire* $U$ *de distribution de probabilité* $p(u)$ *est*

$$L(C) = \mathbf{E}[\ell(U)] = \sum_{u \in \mathcal{U}} p(u)\ell(u)$$

**Exemple (Code binaire).** Sur un alphabet binaire, si :

$$\begin{aligned} P(U=1) &= 1/2 \quad C(1) = 0 \\ P(U=2) &= 1/4 \quad C(2) = 10 \\ P(U=3) &= 1/8 \quad C(3) = 110 \\ P(U=4) &= 1/8 \quad C(4) = 111 \end{aligned}$$

On a $H(U) = 1.75$ bits et $L(C) = 1.75$ bits. Par exemple, 0110111100110 se décode en 134213.

**Définition 6.** *Un code* $C$ *est dit* **non-ambigu** *si*

$$x \neq y \Rightarrow C(x) \neq C(y).$$

**Définition 7.** *L'extension* $C^*$ *du code* $C$ *est la fonction des mots finis de* $\mathcal{U}$ *vers les mots finis de* $\mathcal{D}$ *définie par*

$$C(u_1 u_2 \ldots u_n) := C(u_1) \cdot c(u_2) \ldots C(u_n) \text{ (concaténation)}$$

**Définition 8.** *Un code* $C$ *est dit* **uniquement décodable** *si son extension* $C^*$ *est non-ambigüe.*

**Définition 9.** *Un code est dit* **instantané** *(ou* **prefixe***)si aucun mot code n'est le préfixe d'un autre mot code.*

**Exemple (Différents types de codes).** Parmi les codes suivants :

| U | code 1 | code 2 | code 3 |
|---|---|---|---|
| 1 | 0 | 10 | 0 |
| 2 | 010 | 00 | 10 |
| 3 | 01 | 11 | 110 |
| 4 | 10 | 110 | 111 |

– Le code 3 est instantané.

– Le code 2 est uniquement décodable (il suffit de regarder la parité du nombre de 0 après 11).
– Le code 1 est non-ambigu, mais non uniquement décodable, par exemple 010 peut se décoder par $2, 14$ ou $31$.

Un code instantané est uniquement décodable. De plus un code instantané peut être décodé sans référence aux mots code future puisque la fin d'un mot code est reconnaissable immédiatement.

**Théorème 6** (Inégalité de Kraft)**.** *Pour un code instantané sur un alphabet de taille $D$, les longueurs des mots code $\ell_1, \ldots, \ell_m$ doivent vérifier :*

$$\sum_i D^{-l_i} \leq 1$$

*Inversement, étant donné une suite de longueurs vérifiant cette inégalité, il existe un code instantané avec des mots code ayant ces longueurs.*

*Démonstration.* Pour prouver le premier point, on peut considérer l'arbre de codage du code $C$.

Soit $\ell_{max}$ la longueur du plus long mot code. Un mot code de longueur $\ell_i$ a $D^{\ell_{max}-\ell_i}$ descendants à la profondeur $\ell_{max}$ qui doivent être disjoints des descendants des autres mots code par la propriété du préfixe. On a donc

$$\sum_i D^{\ell_{max}-\ell_i} \leq D^{\ell_{max}}.$$

Inversement étant donné des longueurs $\ell_1 \leq \cdots \leq \ell_m$ satisfaisant l'inégalité de Kraft, on peut toujours construire un arbre de codage comme précédemment. On attribue $0^{\ell_1}$ au mot-code 1, et on continue en attibuant toujours le premier mot possible dans l'ordre lexicographique au prochain mot-code. Comme on prend les longueurs des mots l'ordre croissants, on est assuré de pouvoir toujours prendre les mots dans l'ordre lexicographique « sans laisser de trou ». □

**Théorème 7** (McMillan)**.** *Les longueurs des mots code d'un code $D$-aire uniquement décodable doivent satisfaire l'inégalité de Kraft.*

Une conséquence immédiate est que les codes instantanés seront tout aussi performants (pour ce qui concerne leurs longueurs) que les codes uniquement décodables (et pas moins, comme on aurait pu le penser).

*Démonstration.* Soit $k$ un entier.

On a

$$\begin{aligned}\left(\sum_{u\in\mathcal{U}} D^{-\ell(u)}\right)^k &= \sum_{u_1\in\mathcal{U}} \cdots \sum_{u_k\in\mathcal{U}} D^{-\ell(u_1)-\ell(u_2)-\cdots-\ell(u_k)} \\ &= \sum_{(u_1,\ldots,u_k)\in\mathcal{U}^k} D^{-\ell(u_1\ldots u_k)} \\ &= \sum_{m=1}^{k\ell_{max}} A(m)D^{-m}\end{aligned}$$

où

$$A(m) = |\{(u_1 \dots u_k) \in \mathcal{U}^k, \ell(u_1 \dots u_k) = m\}|$$

On a $A(m) \leq D^m$ car le code est uniquement décodable, et donc pour tout $k \geq 0$,

$$\sum_{u \in \mathcal{U}} D^{-l(u)} \leq (kl_{max})^{1/k}$$

Or $(kl_{max})^{1/k} \xrightarrow[k\to\infty]{} 1$, ce qui conclut la preuve. □

## 2.2 Un théorème de codage de source

**Théorème 8.** *Etant donné une source discrète à valeurs dans $\mathcal{U}$ et d'entropie $H(U)$, et étant donné un alphabet de $D$ symboles pour le code, il est possible de coder chaque lettre de la source de manière instantanée et telle que la longueur moyenne des mots satisfasse $L(C) < \frac{H(U)}{\log D} + 1$.*

*De plus, pour tout code uniquement décodable, $L(C) \geq H(U)/\log D$.*

*Démonstration.* Soit $C$ un code uniquement décodable. On a alors

$$\begin{aligned} H(U) - L(C)\log D &= \sum_u p(u) \log \frac{1}{p(u)} - \sum_u p(u)\ell(u)\log D \\ &= \sum_u p(u) \log \frac{D^{-\ell(u)}}{p(u)} \end{aligned}$$

On sait par ailleurs que pour $z > 0$, $\log z \leq (z-1)\log e$, on a donc :

$$\begin{aligned} H(U) - L(C)\log D &\leq (\log e)\Big(\sum_u D^{-\ell(u)} - \underbrace{\sum_u p(u)}_{=1}\Big) \\ &\leq 0 \quad \text{par (Mcmillan)} \end{aligned}$$

Montrons maintenant que l'on peut trouver un code instantané qui satisfait l'autre inégalité. On choisit $\ell(u)$ tel que $D^{-\ell(u)} \leq p(u) < D^{-\ell(u)+1}$.

On a donc

$$\sum_u D^{-\ell(u)} \leq 1.$$

D'après l'inégalité de Kraft, il existe donc un code instantané avec ces longueurs. De plus

$$\begin{aligned} \log p(u) &< (-\ell(u)+1)\log D \\ l(u) &< \frac{-\log p(u)}{\log D} + 1 \end{aligned}$$

et

$$L(C) = \sum p(u)\ell(u) < \frac{H(U)}{\log D} + 1.$$

□

**Théorème 9.** *Pour une source discrète sans mémoire d'entropie $H(U)$ et un alphabet à $D$ symboles, il est possible de coder les suites de $k$ lettres de la source de sorte que*

1. *La propriété du préfixe soit satisfaite*

*2. La longueur moyenne des mots code par lettre source vérifie :*

$$H(U)/\log D \leq L^k/k < H(U)/\log D + 1/k$$

*où* $L^k = \sum_{u_1 \dots u_k} \ell(u_1 \dots u_k) p(u_1 \dots u_k)$

*Démonstration.* Comme la source est sans mémoire, $U_1, \dots, U_k$ sont des variables aléatoires i.i.d, et donc $H(U^{(k)}) = kH(U)$. Il suffit alors d'appliquer le théorème précédent au mots de $k$ lettres. □

## 2.3 Un codage optimal : le codage de Huffman

On présente ici le codage de Huffman. Il est optimal en ce sens qu'il n'existe pas de code uniquement décodable avec une longueur moyenne inférieure.

Dans toute la suite, on se restreint aux codes instantnés sans perte de généralité par l'inégalité de McMillan.

### 2.3.1 Cas du code binaire : $D = 2$

On suppose que $\mathcal{U} = \{U_1, \dots, U_k\}$, avec $p(u_1) \geq p(u_2) \geq \dots \geq p(u_k)$.

**Lemme 2.** *Pour tout $k \geq 2$, un code binaire optimal existe pour lequel les mots code les moins probables $C(u_k)$ et $C(u_{k-1})$ ont même longueur maximale et diffèrent par le dernier bit.*

*Démonstration.* Supposons qu'il existe $i$ et $j$ tels que $\ell_{u_i} > \ell(u_j)$ et $p(u_i) > p(u_j)$. Alors, on obtient un meilleur code en interchangeant les mots codant $u_i$ et $u_j$. En effet, on obtient un nouveau code $C'$ avec

$$\begin{aligned} L(C') - L(C) &= p(u_i)\ell(u_j) + p(u_j)\ell(u_i) - p(u_j)\ell(u_j) - p(u_i)\ell(u_i) \\ &= (p(u_i) - p(u_j))(\ell(u_j) - \ell(u_i)) < 0. \end{aligned}$$

On peut donc supposer que les longueurs des mots-code sont ordonnées inversement à leur probabilité. Ainsi, $C(u_k)$ est de longueur maximale. Si $C(u_{k-1})$ n'était de la même longueur que $C(u_k)$, alors $C(u_k)$ serait le seul mot de sa longueur. Mais alors, on obtiendrait un meilleur code instantané en enlevant le dernier bit de $C(u_k)$ (le mot obtenu en changeant sa dernière lettre n'est ni un mot-code, ni un préfixe d'un mot code). Donc nécessairement, si $C$ eest optimal, $C(u_k)$ et $C(u_{k-1})$ sont tous les deux de la meme longueur, maximale. En ce cas, on peut choisir le code de manière à ce que ces deux mots ne diffèrent que par leur dernier bit. □

On a donc réduit le problème de construction d'un code optimal à celui de construire $C(u_1), \dots, C(u_{k-2})$ et trouver les $l(u_k) - 1$ premiers digits de $C(u_k)$.

On définit maintenant l'ensemble réduit : $\mathcal{U}' = \{u'_1, \dots, u'_{k-1}\}$ avec la v.a $U'$ associée : $p(u'_j) = p(u_j)$ si $j \leq k-2$ et $p(u'_{k-1}) = p(u_k) + p(u_{k-1})$.

Il y a une bijection entre les codes instantanés pour $U'$ et les codes instantanés pour $U$ pour lesquels $C(u_k)$ et $C(u_{k-1})$ ne diffèrent que par le dernier digit, $C(u_k)$ finissant par un 1 et $C(u_{k-1})$ par un 0.

**Lemme 3.** *Si un code instantané est optimal pour $U'$, le code instantané correspondant pour $U$ est optimal.*

*Démonstration.*

$$\ell(u_j) = \begin{cases} \ell(u'_j) & \text{si } j \leq k-2 \\ \ell(u'_{k-1}) + 1 & \text{si } j \geq k-1 \end{cases}$$

Donc,

$$\begin{aligned} L(C) &= \sum p(u_j)\ell(u_j) \\ &= \sum_{j \leq k-2} p(u'_j)\ell(u'_j) + (p(u_{k-1}) + p(u_k))(\ell(u'_{k-1}) + 1) \end{aligned}$$

Or $p(u_{k-1}) + p(u_k) = p(u'_{k-1})$, donc :

$$L(C) = L(C') + p(u'_{k-1}).$$

Comme $p(u'_{k-1})$ ne dépend pas de $C'$, on peut minimiser $L(C)$ sur la classe des codes où $C(u_k)$ et $C(u_{k-1})$ ne diffèrent que sur le dernier digit en minimiasant $L(C')$. Par le lemme 2, un tel code minimise $L(C)$ sur tous les codes instantanés. □

**Application** On construit donc l'arbre de codage de Huffman de proche en proche en rassemblant à chaque étape les deux noeuds de plus faible probabilité et en affectant la somme de ces probabilités au noeud père.

Voici un exemple :

| mot code | message | $p(u_k)$ |
|---|---|---|
| 00 | $u_1$ | 0.3 |
| 01 | $u_2$ | 0.25 |
| 10 | $u_3$ | 0.25 |
| 110 | $u_4$ | 0.1 |
| 111 | $u_5$ | 0.1 |

### 2.3.2 Extension au cas $D > 2$

On définit un arbre de codage *complet* comme un arbre de codage pour lequel tous les noeuds intermédiaires ont $D$ enfants.

**Lemme 4.** *Le nombre de feuilles dans un arbre de codage complet est de la forme $D+m(D-1)$ pour un certain entier $m$.*

*Démonstration.* Le plus petit arbre complet a $D$ feuilles, le second plus petit en a $D-1+D$ (on remplace une feuille de l'arbre précédent par un noeud à $D$ enfants), d'où le résultat par récurrence. □

Pour un code instantané, nous complétons son arbre de codage en rajoutant $B$ feuilles (non utilisées par le code).

Pour un code optimal, toutes les feuilles non utilisées doivent être au même niveau que le mot code le plus long, et ne diffèrent que par le dernier digit.

Un code optimal doit donc avoir au plus $D-2$ feuilles inutilisées.

Si $K$ le nombre de mots code et $B$ le nombre de feuilles inutilisées, on doit avoir :

$$B + K = m(D-1) + D \text{ et } B \leq D-2,$$

donc $K-2 = m(D-1) + (D-2-B)$ et $0 \leq D-2-B \leq D-2$

ainsi, $B = D-2-((K-2)mod(D-1))$.

En suivant le Lemme 2, un code optimal existe pour lequel les $B$ feuilles inutilisées et les $D-B$ mots code les moins probables diffèrent par le dernier digit. Donc la première étape consiste à grouper les $D-B$ noeuds les moins probables. Ensuite à chaque itération, l'ensemble réduit est de cardinal $D + m(D-1)$ et on regroupe les $D$ noeuds les moins probables.

**Exemple ().** Pour $D = 3$ et $K = 6$, il faut rajouté 1 feuille inutilisée et on obtient dans cet exemple :

| mot code | message | $p(u_k)$ |
|---|---|---|
| 0 | $u_1$ | 0.4 |
| 1 | $u_2$ | 0.3 |
| 20 | $u_3$ | 0.2 |
| 21 | $u_4$ | 0.05 |
| 220 | $u_5$ | 0.03 |
| 221 | $u_6$ | 0.02 |