Université Paris 8 Année 2022–2023

M1 Mathématiques et applications, parcours ACC

Théorie de l'information – Solutions feuille de TD 4

12/10/2022

Retrouvez le sujet du TD et d'autres exercices à l'adresse :

`www.math.univ-paris13.fr/~lavauzelle/teaching/2022-23/theorie-information.html`

($\star$) exercice fondamental ($\star\star$) pour s'entraîner ($\star\star\star$) pour aller plus loin 💻 sur machine

Exercice 1. ($\star$) Codes de Huffman et de Shannon–Fano.

Soit X une variable aléatoire donnée par la distribution de probabilité $(\frac{1}{3}, \frac{1}{3}, \frac{1}{4}, \frac{1}{12})$.

Question 1.– Quel est le code de Shannon–Fano associé à cette variable aléatoire ?

Question 2.– Trouver deux codes de Huffman distincts (c'est-à-dire, avec des longueurs de mots différentes) pour la source X. Comparer leur longueur moyenne à celle du code de Shannon–Fano.

Solutions de l'Exercice 1.

Solution Q1. Dans le code de Shannon–Fano, un symbole d'occurence p_i sera codé en un mot de longueur $n_i = \lceil \log_2 \frac{1}{p_i} \rceil$. Dans notre cas, on obtient donc les longueurs suivantes : $(2, 2, 2, 4)$. Un code associé est alors :

$$\{00, 01, 10, 1111\}.$$

Solution Q2. On applique l'algorithme de Huffman sur la distribution $p = (\frac{1}{3}, \frac{1}{3}, \frac{1}{4}, \frac{1}{12})$.

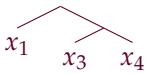
— Première étape : on construit l'arbre associé aux deux probabilités les plus faibles :

La distributiion passée en argument de l'appel récursif est alors $(\frac{1}{3}, \frac{1}{3}, \frac{1}{4} + \frac{1}{12} = \frac{1}{3})$. Pour la deuxième étape, nous auront donc essentiellement deux choix.

— Deuxième étape :

— *Choix* 1 : dans la sélection de deux probabilités, on choisit celle associées à x_1 et x_2. L'arbre est alors :

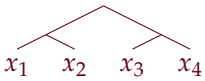
— *Choix* 2 : la probabilité de x_3 ou x_4 est choisie, avec celle de x_1 (ou celle de x_2, c'est équivalent). On a alors l'arbre



Dans les deux cas, la distribution utilisée dans le dernier appel récursif est $(\frac{2}{3}, \frac{1}{3})$.

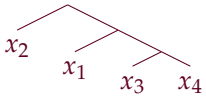
— Troisième étape : SUivant le choix de l'étape précédente, nous obtenons deux arbres différents.

— *Choix* 1 : l'arbre



donne la séquence de longueurs $(2, 2, 2, 2)$.

— *Choix* 2 : l'arbre



donne la séquence de longueurs $(1, 2, 3, 3)$.

On peut vérifier que l'entropie de la source vaut approximativement $H(X) \simeq 1.99$. La longueur moyenne de l'arbre de Shannon–Fano est d'environ 2.17. Ceux produits par l'algorithme d'Huffman sont bien meilleurs, de longueur moyenne égale à 2.

Exercice 2. ($\star\star$) Code sur la loi conjointe.

Soit $\mathcal{X} = \{a, b\}$ et X, Y deux variables indépendantes sur $\mathcal{X}$ de même loi de Bernoulli de paramètre λ. On note $Z = (X, Y)$ la variable produit, définie sur $\mathcal{X}^2$, de loi conjointe $p_{X,Y}$.

Question 1.– Calculer $\mathbb{P}(Z = z)$ pour tout $z = (x, y) \in \mathcal{X}^2$.

Question 2.– Quelle est l'entropie de Z ?

Question 3.– Décrire le code de Huffman de source Z. Selon la valeur de λ, on pourra distinguer plusieurs formes pour l'arbre binaire associé au code.

Question 4.– Tracer le graphe de la longueur moyenne du code de Huffman en fonction de λ. Sous quelle condition sur λ le code de Huffman est-il strictement meilleur que le code de longueur fixe égale à 2 ?

Question 5.– Décrire le code de Shannon–Fano de source Z. On donnera les longueurs des mots en fonction de λ.

Question 6.– Sous quelle condition sur λ le code de Shannon–Fano est-il strictement meilleur que le code de longueur fixe égale à 2 ? On pourra s'aider d'un logiciel pour les résolutions numériques.

Solutions de l'Exercice 2.

Solution Q1. Si $\mathbb{P}(X = 1) = \mathbb{P}(Y = 1) = \lambda$, on a

z	00	01	10	11
$\mathbb{P}(Z = z)$	$(1-\lambda)^2$	$\lambda(1-\lambda)$	$\lambda(1-\lambda)$	λ^2

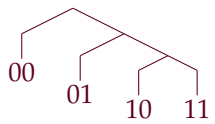
Solution Q2. Comme X et Y sont indépendantes et de loi de Bernoulli de paramètre λ, on a $H(Z) = 2h(\lambda)$.

Solution Q3. On suppose $\lambda < 1/2$ et on raisonnera symétriquement par rapport à $1/2$ pour $\lambda > 1/2$. À la première étape de l'algorithme de Huffman, les probabilités λ^2 et $\lambda(1-\lambda)$ sont sélectionnées. La nouvelle loi de probabilité obtenue est donc :

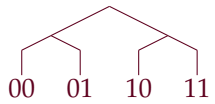
$$(\lambda^2 + \lambda(1-\lambda) = \lambda, \lambda(1-\lambda), (1-\lambda)^2)\,.$$

On distingue donc deux cas :

1. Si $\lambda(1-\lambda) < \lambda \leq (1-\lambda)^2$, alors on obtient l'arbre



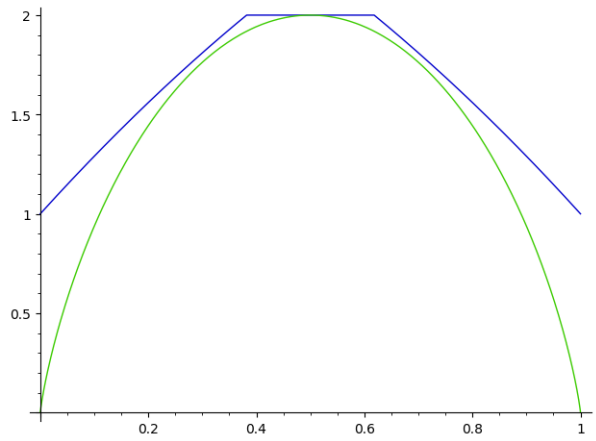
   Ce cas intervient pour $\lambda \leq (1-\lambda)^2$, c'est-à-dire $\lambda \leq \frac{3-\sqrt{5}}{2}$.
2. Si $\lambda(1-\lambda) < (1-\lambda)^2 < \lambda$, alors on obtient l'arbre



Solution Q4. Le code de Huffman a pour longueur moyenne 2 dans le cas où $\lambda \in [\frac{3-\sqrt{5}}{2}, \frac{1}{2} - \frac{3-\sqrt{5}}{2}]$.

Dans le cas contraire, on a une longueur moyenne $\overline{\ell} = 1 + 3\lambda - \lambda^2$ si $\lambda < \frac{3-\sqrt{5}}{2}$ et $\overline{\ell} = 3 - \lambda - \lambda^2$ si $\lambda > 1 - \frac{3-\sqrt{5}}{2}$.

Dans le graphe suivant, on représente en abscisse le paramètre λ. La courbe bleue donne la longueur moyenne du code de Huffman, et la courbe verte représente la fonction d'entropie binaire.



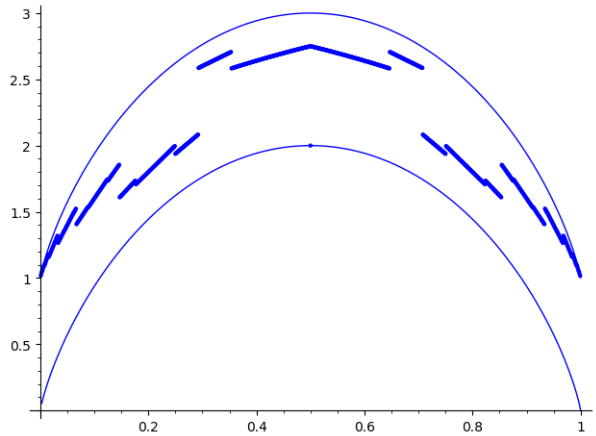
Solution Q5. Le code de Shannon-Fano associé à Z a

— un mot de longueur $\lceil -\log_2(1-\lambda)^2 \rceil$,
— deux mots de longueur $\lceil -\log_2 \lambda(1-\lambda) \rceil$,
— un mot de longueur $\lceil -\log_2 \lambda^2 \rceil$.

Solution Q6. La longueur moyenne du code de Shannon-Fano associé à Z est

$$\overline{\ell}(\lambda) = (1-\lambda)^2 \lceil -\log_2(1-\lambda)^2 \rceil + 2\lambda(1-\lambda) \lceil -\log_2 \lambda(1-\lambda) \rceil + \lambda^2 \lceil -\log_2 \lambda^2 \rceil .$$

L'analyse théorique est plus complexe. On remarque notamment que la fonction $\overline{\ell}(\lambda)$ n'est pas continue, si croissante sur $[0,1/2]$. Le graphe suivant, où $\overline{\ell}(\lambda)$ est représentée entre $h_2(\lambda)$ et $h_2(\lambda)+1$, en atteste.



Exercice 3. ($\star\star$) Mots prépondérants dans un code de Huffman.

Considérons le code de Huffman sur une source X de distribution $p_1 \geq \cdots \geq p_m$.

Question 1.– Démontrer que si la probabilité d'occurence la plus forte vérifie $p_1 > 2/5$, alors le symbole associé à cette probabilité est encodé par un mot de longueur 1.

Question 2.– Démontrer que s'il existe un mot de longueur 1, alors la probabilité $p_1 \geq 1/3$.

Solutions de l'Exercice 3.

Solution Q1. Considérons le premier appel récursif de l'algorithme d'Huffman pour lequel la somme des probabilités minimales dépasse p_1. Si un tel appel n'existe pas, alors il est clair que la longueur du mot de probabilité p_1 est 1, car cette probabilité sera sélectionnée en dernier appel récursif. Sans perte de généralité, notons

$p_1 \geq p'_2 \geq \cdots \geq p'_{i-1} \geq p'_i$ les probabilités lors de cet appel, où $i \geq 3$. Alors par hypothèse on a $p'_i + p'_{i-1} \geq p_1 > \frac{2}{5}$. Par conséquent, $2p'_{i-1} \geq p'_{i-1} + p'_i > \frac{2}{5}$ implique que $p'_{i-1} > \frac{1}{5}$. Si $i > 3$, alors on obtient la contradiction

$$\frac{1}{5} < p'_{i-1} \leq p'_2 \leq 1 - (p_1 + p'_{i-1} + p'_i) < 1 - \frac{4}{5} = \frac{1}{5}\,.$$

Ainsi, $i = 3$, ce qui impose que la longueur du mot associé à p_1 est 1.

Solution Q2. Considérons l'avant-dernier appel récursif dans l'algorithme de Huffman. Comme la longueur du mot le plus probable est 1, la probabilité p_1 n'est pas l'une des deux plus petites lors de cete appel. On a donc $p_1 \geq p'_2 \geq p'_3$, ce qui mène à

$$p_1 \geq \frac{1}{3}(p_1 + p'_2 + p'_3) = \frac{1}{3}\,.$$

Exercice 4. ($\star\star$) Code gamma.

Dans cet exercice, on souhaite coder efficacement une source à valeur dans l'ensemble des entiers naturels, sans avoir d'information préalable sur les entiers émis par la source.

Usuellement, pour coder un entier $n \in \mathbb{N}$ sous forme de chaîne de bits, on décompose l'entier en base 2 :

$$n = \sum_{i=0}^{k} n_i 2^i\,,$$

et on retourne la séquence $B(n) = (n_0, \ldots, n_k) \in \{0,1\}^{k+1}$ de longueur $k+1$, où $k = \lfloor \log_2(n) \rfloor$ (excepté pour $n = 0$, pour lequel on a $k = 0$).

Question 1.– Le code $B : \mathbb{N} \to \{0,1\}^+$ est-il préfixe ? Est-il uniquement décodable ? Pourquoi ?

Le code Gamma, introduit par Elias, propose une solution au problème précédent. Avant d'encoder l'entier n sous la forme de sa décomposition en base 2, on précise à l'aide d'un code unaire la longueur de n. Ainsi, le code $\Gamma : \mathbb{N} \to \{0,1\}^+$ est défini par :

$$\Gamma(n) = \underbrace{0 \cdots\cdots\cdots 0}_{\leftarrow 1+\lfloor \log_2(n) \rfloor \rightarrow} 1\, B(n) \quad \text{pour } n \geq 1,$$

et $\Gamma(0) = 10$.

Question 2.– Le code Γ est-il préfixe ?

Question 3.– Quelle est la longueur de $\Gamma(n)$ pour $n \in \mathbb{N}$? Donner une condition sur la distribution (p_n) de la source pour que la longueur moyenne du code Γ soit finie.

Question 4.– Calculer numériquement des valeurs approchées de la longueur moyenne du code Γ lorsque :

- X suit une loi uniforme sur $\{0, 1, \ldots, N\}$ (et $p_X(n)$ est nulle pour $n \geq N$),
- X suit une loi géométrique,
- X suit une loi de Poisson.

Question 5.– ($\star\star\star$) Le code Gamma peut être vu comme le codage unaire de la taille en bits de n, concaténé avec la représentation binaire de n. Peut-on itérer ce procédé pour obtenir un code encore plus court ? Quelle est la longueur du codage de l'entier n obtenu ?

Solutions de l'Exercice 4.

Solution Q1. Le code B n'est pas préfixe, car par exemple, $B(1) = 1$ est un préfixe de $B(2) = 11$. Il n'est pas non plus uniquement décodable, car

- par exemple, on peut voir que la chaîne de caractère 110 est l'image par B^+ des messages $(1,1,0)$, $(1,3)$, $(3,0)$ et 6,

— on observe aussi que la séquence de longueurs de B ne satisfait pas l'inégalité de Kraft : la quantité

$$\begin{aligned}\sum_{n=0}^{2^k-1} 2^{-\ell(B(n))} &= 2^{-\ell(B(0))} + \sum_{i=0}^{k-1}\sum_{j=0}^{2^i-1} 2^{-\ell(B(2^i+j))} \\ &= 1 + \sum_{i=0}^{k-1}\sum_{j=0}^{2^i-1} 2^{-(i+1)} \\ &= 1 + \sum_{i=0}^{k-1} \frac{1}{2} = \frac{k}{2} + 1\end{aligned}$$

diverge lorsque $k \to +\infty$.

Solution Q2. Soit n et m deux entiers, et supposons que $\Gamma(n)$ soit un préfixe de $\Gamma(m)$. Alors, $\Gamma(n)$ et $\Gamma(m)$ débutent par le même nombre de zéros, donc n et m sont encadrés par les mêmes puissances de 2. Autrement dit : $2^k \leq n, m \leq 2^{k+1} - 1$ où $k = \lfloor \log_2 n \rfloor$.

Puis, nécéssairement $B(n)$ est un préfixe de $B(m)$. Mais d'après ce qui précède, $B(n)$ et $B(m)$ doivent avoir la même longueur. Ceci induit que $B(n) = B(m)$, puis que $n = m$.

Solution Q3. Le mot $\Gamma(0)$ a longueur 2, et les mots $\Gamma(n)$ ont longueur $3 + 2\lfloor \log_2(n) \rfloor$ pour $n \geq 1$.

On a :

$$\overline{\ell}(\Gamma) = \sum_{n\in\mathbb{N}} p_n\, \ell(\Gamma(n)) = 2p_0 + \sum_{n\geq 1} p_n(3 + 2\lfloor \log_2 n \rfloor) = 3 - p_0 + 2\sum_{n\geq 1} p_n \lfloor \log_2 n \rfloor .$$

De manière générale, la longueur moyenne de Γ est donc finie si et seulement si la série de terme général $p_n \lfloor \log_2 n \rfloor$ converge.

Solution Q4. À venir

Solution Q5. À venir

Exercice 5. ($\star\star$) Implantation du code de Shannon–Fano.

Le but de cet exercice est d'implanter un algorithme quei permet, à partir d'une distribution $p = (p_x)_{x\in\mathcal{X}}$, de construire le code de Shannon–Fano associé.

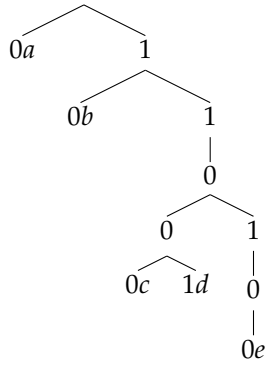
Rappel : en `python`, un dictionnaire `D` est une structure de données qui, à des clés `key` associent des élements `D[key]`. Pour construire un dictionnaire, on utilise la syntaxe `D = { key1: D[key1], ..., keyN: D[keyN] }`.

Question 1.– Écrire une fonction `compute_length(p)` qui, à partir d'une distribution $(p_x)_x$ représentée par un dictionnaire `p = { x: p[x] }`, retourne le dictionnaire des longueurs associées. Autrement dit, si `n[x]` représente $\lceil -\log_2(p_x) \rceil$, alors le dictionnaire retourné sera `{ x: n[x] }`.

Pour construire le code de Shannon–Fano, on peut avoir recours à une structure d'arbre binaire. En `python3`, il existe des bibliothèques externes (à installer) qui permettent de manipuler ces arbres. L'une d'entre elles s'appelle `binarytree`.

Remarque : vous pouvez également traiter cet exercice avec votre propre implantation des arbres binaires, ou avec d'autres bibliothèques/logiciels. Par exemple, Sagemath manipule très bien les arbres binaires.

À titre d'exemple, on considère l'arbre binaire ci-contre, qu'on appelle `T`.



Question 2.– Construire l'arbre `T`, puis l'afficher.

Question 3.– Écrire une fonction `get_paths_leaf_to_root(T)` qui prend en entrée un arbre binaire `T`, et retourne le dictionnaire dont les clés sont les étiquettes des feuilles de `T`, et la valeur associée à une clé `x` est la concaténation des étiquettes des nœuds menant de la feuille à la racine de l'arbre.

Puis, tester votre fonction avec l'exemple ci-contre.

Pour construire l'arbre binaire du code de Shannon–Fano, on va partir de l'arbre vide, puis ajouter petit à petit des feuilles à la profondeur souhaitée. Par définition du code de Shannon–Fano, cela sera toujours possible, mais il faudra comprendre où placer cette feuille.

Question 4.– Écrire une fonction `add_leaf_at_depth(T, x, d)` qui ajoute une feuille d'étiquette `x` à la profondeur `d` dans l'arbre `T`. Lors de l'ajout, si des nœuds internes sont créés, on leur assignera une étiquette 0 ou 1 (suivant si c'est un descendant gauche ou droit).

Question 5.– Déduire des questions précédentes une fonction `shannonfano` qui, à partir d'une distribution $(p_x)_x$ représentée par un dictionnaire `p = { x: p[x] }`, construit un code de Shannon–Fano associé.

Solutions de l'Exercice 5.

Scripts à retrouver sur la page web du cours.

Exercice 6. ($\star\star\star$) Borne entropique et codes optimaux.

Question 1.– Pour quelle valeur de λ la variable de Bernoulli de paramètre λ donne un code de Shannon–Fano optimal ?

Question 2.– Soit m un entier ≥ 3 quelconque. Trouver une variable aléatoire sur $\{x_1, \ldots, x_m\}$ telle que le code de Shannon–Fano associé est optimal.

Question 3.– Soit $\epsilon > 0$. Déterminer une distribution sur une variable aléatoire X telle que, pour tout code C sur X, la longueur moyenne $\overline{\ell}(C) \geq H(X) + 1 - \epsilon$.

Question 4.– Soit $\epsilon > 0$. Déterminer une distribution sur une variable aléatoire X telle que le code de Shannon–Fano sur X a une longueur moyenne $\overline{\ell}(C_{SF}) \geq H(X) + 1 - \epsilon$, et le code de Huffman sur X a une longueur moyenne $\overline{\ell}(C_H) \leq H(X) + \epsilon$.

Solutions de l'Exercice 6.

Solution Q1. La variable de Bernoulli de paramètre λ a pour distribution $(\lambda, 1-\lambda)$. Sans perte de généralité supposons $\lambda \leq 1/2$. La séquence de longueurs de son code de Shannon-Fano est alors

$$\left(1, \left\lceil \log_2 \frac{1}{\lambda} \right\rceil\right).$$

Un code optimal pour une variable binaire est nécessairement constitué de deux mots de longueur 1. Ainsi, le code de Shannon-Fano est optimal si et seulement si $\lceil \log_2 \frac{1}{\lambda} \rceil = 1$, c'est-à-dire $\lambda = \frac{1}{2}$.

Solution Q2. Il suffit de trouver des probabilités p_i telles que $\log_2(\frac{1}{p_i}) = \lceil \log_2(\frac{1}{p_i}) \rceil$. Autrement dit, les p_i doivent être des puissances de $\frac{1}{2}$ et vérifier $\sum_{i=1}^{m} p_i = 1$. Une des solutions est alors la suivante : si $m = 2^k + r$ avec $0 \leq r < 2^k$,

on prend $p_1 = \cdots = p_{2r} = 2^{-(k+1)}$ et $p_{2r+1} = \cdots = p_m = 2^{-k}$. On vérifie alors que :

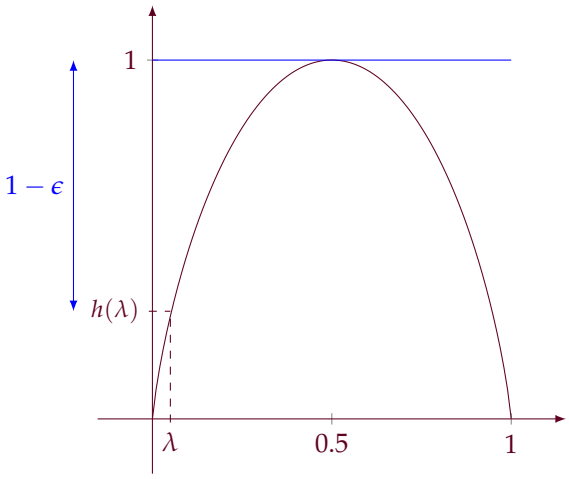
$$\sum_{i=1}^{m} p_i = 2r \cdot 2^{-(k+1)} + (m-2r) \cdot 2^{-k} = r \cdot 2^{-k} + (2^{-k} - r) \cdot 2^{-k} = 1\,.$$

Pour $m = 5$, par exemple, cela donne la distribution :

$$\left(\frac{1}{8}, \frac{1}{8}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)$$

qui produit un code de longueur moyenne $H(X) = 2 \times \frac{1}{8} \times 3 + 3 \times \frac{1}{4} \times 2 = \frac{3}{2}$.

Solution Q3. Considérons une variable de Bernoulli X de paramètre λ. Son entropie est $H(X) = \lambda \log_2 \frac{1}{\lambda} + (1-\lambda) \log_2 \frac{1}{1-\lambda} = h(\lambda)$. Or, tout code C sur X apour longueur moyenne $\overline{\ell}(C) \geq 1$. Il suffit donc de choisir λ tel que $H(X) \leq 1 - \epsilon$, par exemple l'unique valeur entre 0 et $\frac{1}{2}$ telle que $h(\lambda) = 1 - \epsilon$.



Solution Q4. Soit $n \geq 1 - \log_2(\epsilon)$ de sorte que $2^{-n} \leq \epsilon/2$. On choisit un alphabet à $2^n + 1$ éléments, et on prend X une variable aléatoire qui suit la loi uniforme sur cet alphabet.

Alors, $H(X) = \log(2^n + 1)$ vérifie

$$n \leq H(X) \leq n + \log(1 + 2^{-n}) \leq n + 2^{-n} \leq n + \epsilon/2\,.$$

Le code Shannon–Fano a pour longueur moyenne $n+1$, car chacun des mots du code a longueur $\lceil \log_2(2^n+1) \rceil = n+1$. Donc on a bien, $\overline{\ell}(C_{SF}) \geq H(X) + 1 - \epsilon$.

Enfin, l'algorithme de Huffman produit un arbre « presque » parfait : toutes les branches ont longueur n, sauf 2 qui ont longueur $n+1$. On a donc

$$\overline{\ell}(C_H) = \frac{1}{2^n+1}\left((2^n - 1) \times n + 2 \times (n+1)\right) = n + \frac{2}{2^n+1} < n + \epsilon\,.$$