None

Null

12 mars 2024

Table des matières

Première partie

TD3

1 Exercice 1

Soit c un code PF. Soit $x_1, \ldots, x_k, y_1, \ldots, y_l$ pour lesquels il y a égalité. Alors, soit $i = \min\{i \in \mathbb{N} \mid x_i \neq y_i\}$. Alors en ne comptant que les i premiers caractères du mot de départ, selon la longueur des codes, ou bien $c(x_1, \ldots x_{i-1})$ est un préfixe de $c(y_1, \ldots, y_{i-1})$ ou bien l'inverse.

2 Exercice 2

2.1 Question 1

c est un code PF.

2.2 Question 2

Pour déchiffrer le code, il suffit de regarder le premier 0 que l'on trouve. On choisit entre $1, 2$ et 3 pour le déchiffrer selon la valeur modulo 3 du nombre de 1 qui le suivent.

3 Exercice 3

J'ai une tête de cloche à fromage ouuuuuuuuuuu ?

4 Exercice 4

4.1 Question 1

On considère $\mathcal{U}_m = \{u \in \mathcal{U} \mid l(u) \leq m\}$. On a alors, puisque $\mathcal{U}_m$ est fini, l'existence de $l_{max,m} = \max\{l(u) \mid u \in \mathcal{U}_m\} \leq m$.
On considère ensuite c_m le code c tronqué à $\mathcal{U}_m$. On a : La suite des $\mathcal{U}_m$ est croissante et on a $\limsup \mathcal{U}_m = \mathcal{U}$. Donc, par passage à la limite supérieure, on a bien

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

4.2 Question 2

On retire, à partir de la profondeur $l(1)$ une proportion $\alpha_1 D^{-l(1)}$ des branches. On définit une mesure μ sur les fils de l'arbre comme la proportion des feuilles de l'arbre qu'ils recouvrent.
On note A_i l'union des fils enracinés à profondeur $l(i)$ qu'on a utilisés pour du codage de longueur $l(i)$. On a $\mu(A_i) = \alpha_i D^{-l(i)}$. Notre codage complet est l'union des A_i. On a alors :

$$\mu\left(\bigcup_{i \in \mathbb{N}^*} A_i\right) = \sum_{i=1}^{n} \mu(A_i) = \sum_{i=1}^{n} \alpha(i) D^{-l(i)} = \sum_{u \in \mathcal{U}} D^{-l(u)} \leq 1$$

Donc on n'a pas de recouvrement des arbres.
za

5 Exercice 5

5.1 Question 1

On code chaque nombre en base 2 sur k bits. On demande pour chaque bit si le nombre a la même valeur. Bah on a trouvé.

5.2 Question 2

On utilise le code $c_2(k)$ sur les K objets. Puisque le code est PF, on peut juste regarder pour tout i le i-ème symbole, et en fonction de ce qu'on a lu sur les $i-1$ premiers, lorsqu'on arrive à la fin du code, on a trouvé le résultat. On a $E[l(c_2(U))]$ questions en moyenne et donc on a bien :

$$H_2(U) \leq E[l(c_2)(U)] \leq H_2(U) + 1$$

Pour tout $u \in \{1, \ldots, K\}$, on utilise $\lceil \log_2(p_u) \rceil$ questions.

5.3 Question 3

Le nombre moyen de questions d'un questionnaire est équivalent à la longueur moyenne du codage UD qu'il représente. Par inégalité de Kraft et inégalité de Gibbs, on a :

$$\begin{aligned}
E[l(c)(k)] &= \sum_{k=1}^{K} p_k l(c(k)) \\
&= -\sum_{k=1}^{K} p_k \log_2(2^{-l(c(k))}) \\
&\geq -\sum p_k \log_2 q_k \\
&\geq -\sum p_k \log_2 p_k \\
&= H_2(U)
\end{aligned}$$

Donc le questionnaire a en moyenne au moins $H_2(U)$ questions.

5.4 Question 4

On considère $U_1, \ldots, U_m$ des réalisations i.i.d. de U.

1. Avec c_2, en moyenne, on va utiliser :

$$H_2(U) \leq \frac{1}{m} E[l(c_2(U_1)) + \ldots + l(c_2(U_m))] \leq \frac{H_2(U)+1}{m}$$

2. Avec un code arbitraire, on va utiliser :

$$H_2(U) \leq \frac{1}{m} E[l(c(U_1)) + \ldots l(c(U_m))]$$

5.5 Question 5

On passe à la limite dans la question précédente.

Deuxième partie
TD4

6 Exercice 1

On a :

$$\begin{aligned}
E[X] &= \int_0^\infty x f(x)\,\mathrm{d}x \\
&= \int_0^\infty [x f_X(t)]_0^t\,\mathrm{d}t \\
&= \int_0^\infty \int_0^t f_X(t)\,\mathrm{d}x\,\mathrm{d}t \\
&= \int_0^\infty \int_x^\infty f_X(t)\,\mathrm{d}t\,\mathrm{d}x \\
&= \int_0^\infty P(X \geq x)\,\mathrm{d}x
\end{aligned}$$

7 Exercice 2

On a :

$$\begin{aligned}
P(Y = y, Z = z | X = x) =& P(Y = y | Z = z, X = x) \times P(Z = z | X = x) \\
=& f(x, y) \times f(z, x)
\end{aligned}$$

Donc on a bien l'indépendance.

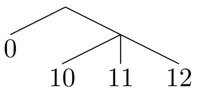
8 Exercice 3

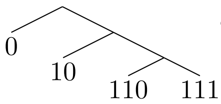
1. On sépare à chaque étape les pièces en 3. Puisqu'une unique pièce est plus lourde, une balance suffit à trouver dans quel groupe la pièce la plus lourde est : Si l'un des ensembles pesés est plus lourd, c'est dans celui là, sinon, s'il y a égalité c'est dans le troisième.
2. Chaque pesée correspond à une manière de séparer les pièces en 3. En particulier, l'arbre des possibilités correspondant à une stratégie arbitraire va être de hauteur au moins $\log_3(n)$. La hauteur de l'arbre correspondant bien au nombre moyen de pesées…
3. On utilise la stratégie de la première question. On n'utilise alors jamais plus de $\lceil \log_3(n) \rceil$ pesée. Si $n = 3^k$, on revient au premier cas avec $\lfloor \log_3(n) \rfloor = k$ pesées.
4. Pour $a = n/3$, cela revient à générer le code de Huffmann 3-aire. Celui-ci minimisant l'entropie.
5. On peut tester en deux pesées lors de la première étape si la pièce différente est plus lourde ou plus légère que les autres. En effet, si on sépare en trois ensembles A, B, C. Ou bien : $A < B$ et $B = C$ et la pièce est plus légère et dans A, ou bien $A < B$ et $B > C$ et la pièce est plus lourde et dans B. À renommage près, on applique la stratégie de 1) et on a le résultat. L'entropie dit qu'on nécessiterait $k + \log_3 2$ pesées en moyenne, ce qui n'est pas entier. On a donc une solution optimale.

9 Exercice 4

1. Il n'y a qu'à la première étape où on va regrouper moins de D fils et on va en regrouper $2 + (N - 2) \mod (D - 1)$. On doit finalement compléter l'arbre par $B = D - 2 - (N - 2) \mod (D - 1)$ feuilles.
2. Puisque tous les codes PF sont UD, le code de Huffman D-aire convient.

3. L'arbre ternaire qu'on obtient est :



On a alors $\bar{R_s} = 1.75$ et donc un taux de compression de $\frac{7/4}{\log_3 4} = 1.39$ Pour $D = 2$ on a l'arbre 

ce qui donne $\bar{R_s} = 9/4$ et donc un taux de compression de 1.125.

4. Pour la distribution $p = (0.3, 0.2, 0.2, 0.15, 0.08, 0.07)$, on trouve

Binaire $L_C = 2.45$ et donc un taux de compression de 1.225

Ternaire $L_C = 1.65$ et donc un taux de compression de 1.012 environ