

Série de travaux dirigés n°3 : Codage de source (2)

Exercice 1: Lesquels de ces codes n'est pas un code de Huffman ? Justifier votre réponse

- a. {1, 01, 00}
- b. {00,01, 10, 110}
- c. {01, 10}

Exercice 2:

a) En utilisant l'algorithme de Shannon-Fano, représentez la séquence suivante (sans tenir compte des espaces) par une séquence de bits:

INFORMATION CALCUL ET COMMUNICATION

- b) Combien de bits par lettre en moyenne sont-ils nécessaires pour représenter cette séquence?
- c) Calculez l'entropie de la séquence.
- d) Si vous vous restreignez à utiliser un code qui ne tient pas compte des probabilités d'apparition et qui utilise exactement le même nombre de bits pour chaque lettre, de combien de bits aurez-vous besoin pour représenter la séquence?

Voici une autre séquence de lettres (où on oublie à nouveau les espaces, les accents, les traits d'union, les virgules et les apostrophes!):

DIDON DINA, DIT-ON, DU DOS D'UN DODU DINDON

- e) A priori, pouvez-vous deviner laquelle des deux séquences ci-dessus a la plus faible entropie?
- f) Représentez à nouveau la séquence par une séquence de bits en utilisant l'algorithme de ShannonFano.
- g) Répondez à nouveau aux questions c) et d) pour cette deuxième séquence de lettres.

Exercice n°3 :L'union fait l'efficacité

Une source sans mémoire ne prend que deux valeurs x_0 et x_1 , de probabilités respectives $P(x_0) = 0.7$ et $P(x_1) = 0.3$.

- a) Calculer l'entropie de cette source. En déduire la quantité d'information par élément binaire si l'on code x_0 par 0 et x_1 par 1. Quelle est l'efficacité de ce codage ?
- b) On décide à présent de coder les symboles par paquets de 2. Calculer les probabilités de chacune des 4 combinaisons, construire un codage de Huffman. Quelle est l'efficacité de ce codage ?

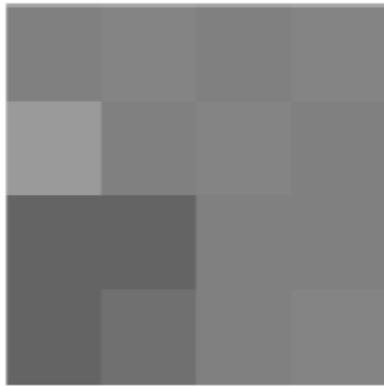
Exercice n°4 : Considérons la variable aléatoire avec la loi de probabilité dans le tableau suivant :

$x \in X$	x_1	x_2	x_3	x_4	x_5	x_6	x_7
-----------	-------	-------	-------	-------	-------	-------	-------

a) Quel est le codage RLE des trois images ci-dessus, et quelle est la longueur de ce code pour chaque image (i.e. combien épargne-t-on de bits par rapport à la taille originale de l'image qui est de 64 bits) ? (vos remarques sont les bienvenues)

Exercice n°7 : Compression d'images

Considérons l'image de taille 4×4 représentée ci-dessous. Les niveaux de gris de cette image sont indiqués dans le tableau de droite.



128	132	128	132
154	128	132	128
100	100	128	128
100	112	128	132

- En considérant que chaque pixel est codé sur un octet, quelle place occupe l'image non compressée ?
- Préciser les fréquences d'apparition de chacun des pixels. En déduire l'arbre binaire de Huffman, ainsi que les mots de code.
- Calculer l'entropie de l'image et déduire l'efficacité du codage obtenu.
- Quelle place occupe maintenant l'image compressée ? En déduire le taux de compression obtenu.