# Introduction au traitement du signal

## Amphi 8 - Théorie de l'information

MINES ParisTech, Tronc commun 1A

22 Juin 2020

# Sommaire

# Information

Figure: Illustration de la notion d'information. De gauche à droite: image originale, Image contenant $N = 8853$ pixels pris sur les contours, Image contenant $N = 8853$ pixels pris aléatoirement. On note bien que l'image centrale apporte nettement plus d'information que celle de droite.

# Modèle de source

- Source du signal modélisée par une variable aléatoire $X$ à valeurs dans un espace discret $\{x_1, \ldots, x_K\}$ de symboles

- Loi de probabilité connue

$$\forall k = 1, \ldots, K, p_k := P\{X = x_k\}$$

- Source supposée stationnaire: la loi de probabilité est constante au cours du temps

# Fonction d'information

1. La quantité d'information $H(x_k)$ apportée par l'évènement $\{X = x_k\}$ est nécessairement une fonction décroissante de la probabilité $p_k$ de survenue de l'évènement $\{X = x_k\}$.

2. L'information apportée par deux évènements indépendants doit être la somme des informations apportées par chacun des évènements. Si $X$ et $Y$ sont deux variables aléatoires discrètes indépendantes, le gain d'information $H(X, Y)$ apporté par l'observation conjointe de $X$ et $Y$ est
$$H(X, Y) = H(X) + H(Y).$$
Du fait de l'indépendance de $X$ et $Y$, la probabilité jointe $p(X, Y)$ est donnée par
$$p(X, Y) = p(X)p(Y).$$

# Fonction d'information (ii)

- Caractéristiques de la fonction d'information:

  1. fonction décroissante de la probabilité de survenue de cette réalisation
  2. transforme un produit en somme

- Expression:

  $$H(X) := -\log_a p(X),$$

  $\log_a$ : fonction logarithmique de base $a$, avec $a > 0$.

- Remarque: Le choix de la base $a$ peut être entièrement arbitraire. En pratique, étant donné que la majorité des données numériques sont encodées avec des 0 et des 1, c'est à dire en binaire, il est intéressant de choisir $a = 2$.

# Entropie de l'information

- La quantité moyenne d'information $\mathcal{H}(X)$ émise par la source est, par définition de l'espérance:

$$\mathcal{H}(X) := E[H(X)] = -\sum_{k=1}^{K} p(X = x_k) \log_a p(X = x_k).$$

Cette quantité fondamentale est appelée *l'entropie* de la variable aléatoire $X$.

- Interprétation: L'entropie mesure notre degré d'ignorance sur la source du signal

# Sommaire

# Codage binaire

## Definition (code)

Notons $\{0,1\}^*$ l'ensemble des séquences finies d'éléments de l'ensemble $\{0,1\}$. Un code sur l'ensemble de symboles discret ou *alphabet* $\mathcal{S}$ généré par la source est une application $\phi : \mathcal{S} \to \{0,1\}^*$ qui associe à tout élément $s_k$ de $\mathcal{S}$ une séquence finie constituée de 0 et de 1. On appelle *mot de code* tout élément de $\phi(\mathcal{S})$.

Exemple: Code de longueur fixe

$$\{a : 00, b : 01, c : 10, d : 11\}.$$

# Codage de longueur variable

- Approche alternative: code de longueur variable
- Nombre de bits moyen par symbole:

$$R_X = \sum_{k=1}^{K} l_k p_k.$$

Exemple: Alphabet $\{x_k\}_{1 \leq k \leq 5}$ de probabilités d'occurence

$$\{p_1 = 0.1, p_2 = 0.1, p_3 = 0.12, p_4 = 0.18, p_5 = 0.5\}$$

Codage:

$$\{w_1 = 111, w_2 = 110, w_3 = 10, w_4 = 01, w_5 = 00\}$$

Nombre de bits moyen par symbole:

$$R_X = 3 \times 0.1 + 3 \times 0.1 + 2 \times 0.12 + 2 \times 0.18 + 2 \times 0.5 = 2.2$$

# Codage de longueur variable (ii)

- Un code de longueur variable peut être ambigu
- Exemple: Symboles $\{a, b, c, d\}$ codés par

$$\{a : 0, b : 1, c : 10, d : 11\}.$$

- La séquence 0110 peut correspondre à *abba*, à *ada* ou encore à *abc*

# Code de préfixe

## Definition (code ambigu)

Un code (resp. codage) est dit non ambigu si deux symboles (resp. séquences de symboles) distincts sont codés par des mots distincts. Un code est dit à décodage unique si le codage associé est non ambigu.

## Definition (code de préfixe)

Un codage de préfixe est un type de codage pour lequel aucun mot ne peut être le préfixe d'un autre mot. On montre aisément que tout codage de préfixe est non ambigu.

# Code de préfixe et arbre binaire

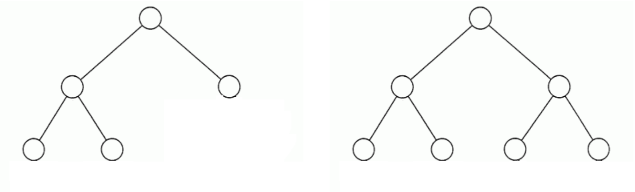


Figure: Arbre binaire quelconque de profondeur $P = 2$ (gauche). Arbre binaire complet de profondeur $P = 2$ (droite)

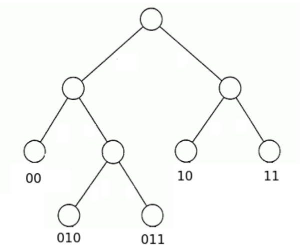



Figure: Codage associé à un arbre binaire

# Théorème d'entropie de Shannon

## Theorem (Théorème d'entropie de Shannon)

*Soit $X$ une variable aléatoire à valeur dans $\mathcal{S} := \{x_k\}_{1 \leq k \leq K}$. Pour tout $k \in \{1, \ldots, K\}$, on suppose que l'événement $\{X = x_k\}$ se produit avec une probabilité $p_k$. Alors, la taille moyenne $R_X$ du code utilisé pour un symbole en utilisant un code de préfixe est bornée inférieurement par l'entropie $\mathcal{H}(X)$:*

$$R_X \geq \mathcal{H}(X).$$

*Il existe de plus un code de préfixe tel que*

$$R_X \leq \mathcal{H}(X) + 1.$$

# Sommaire

# Algorithme de Huffman

1. Initialisation à partir des feuilles $\{x_1, ..., x_K\}$

2. On suppose que $p_1 \leq p_2 \leq \cdots \leq p_K$. L'algorithme aggrège $x_1$ et $x_2$ en un nouveau noeud, qui sera un parent commun aux feuilles $x_1$ et $x_2$ et se verra associé la probabilité $p_{1,2} = p_1 + p_2$.

3. A l'étape suivante, réindexation des noeuds $\{x_{1,2}, x_3, \ldots, x_K\}$ par probabilités d'occurence croissantes, et répétition de l'opération d'aggrégation.

4. L'algorithme stoppe lorsque tous les noeuds ont été aggrégés

# Exemple

Symboles $x_1, x_2, \ldots, x_5$, de probabilités d'occurence

$$\{p_1 = 0.1, p_2 = 0.1, p_3 = 0.12, p_4 = 0.18, p_5 = 0.5\}.$$

1. $x_{1,2}, x_3, x_4, x_5$ de probabilités d'occurence

$$\{p_{1,2} = 0.2, p_3 = 0.12, p_4 = 0.18, p_5 = 0.5\}.$$

2. $x_{1,2}, x_{3,4}, x_5$ de probabilités d'occurence

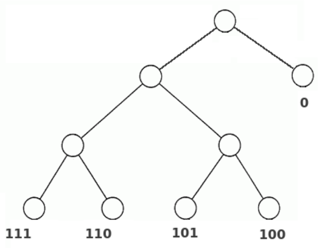
$$\{p_{1,2} = 0.2, p_{3,4} = 0.3, p_5 = 0.5\}.$$

3. $x_{1,2,3,4}, x_5$ de probabilités d'occurence

$$\{p_{1,2,3,4} = 0.5, p_5 = 0.5\}.$$

# Exemple (ii)

- Arbre binaire de codage




- Codage préfixe correspondant

$$\{x_1 : 111, x_2 : 110, x_3 : 101, x_4 : 100, x_5 : 0\}.$$

- Nombre de bits moyens nécessaire à la transmission du signal:

$$R_X = 2$$

# Sommaire

# Quantification

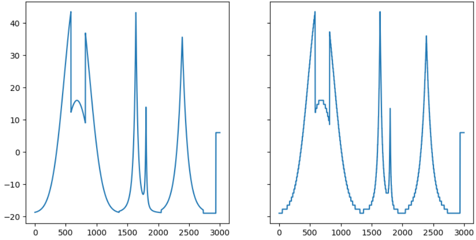



Figure: Quantification d'un signal

Erreur de quantification:

$$\epsilon := E\{|X - \tilde{X}|^2\}.$$

# Quantification scalaire

## Definition (Quantificateur scalaire)

Un quantificateur scalaire $Q$ découpe l'intervalle $[a, b]$ en une séquence de $K$ intervalles $[y_k, y_{k+1}]_{k=1,\dots,K}$ tels que $k_0 = a$ et $k_K = b$ et associe une valeur $a_k$ à la variable aléatoire $\tilde{X}$ si $X \in [y_k, y_{k+1}]$:

$$\tilde{X} := Q(X) = a_k \text{ si } X \in [y_k, y_{k+1}].$$

- Un quantificateur scalaire pour la variable aléatoire $X$ est dit à *haute définition* si, sur chaque intervalle $[y_k, y_{k+1}]$, la densité de probabilité $p(x)$ de $X$ peut être considérée comme constante:

$$\forall x \in [y_k, y_{k+1}], p(x) = \frac{p_k}{\Delta_k},$$

où $\Delta_k := y_{k+1} - y_k$.

# Quantification scalaire à haute définition

- Erreur de quantification:

$$\epsilon = \sum_{k=0}^{K-1} \int_{y_k}^{y_{k+1}} |x - a_k| \mathrm{d}x.$$

## Proposition

*La discrétisation $\{y_0, .., y_K\}$ étant fixée, l'erreur de quantification pour un quantificateur scalaire $Q$ à haute définition est minimale lorsque, pour tout $k = 0, ..., K-1$,*

$$a_k = \frac{y_k + y_{k+1}}{2}.$$

*Ce choix pour les valeurs $\{a_k\}_{k=0,\ldots,K-1}$ consiste simplement à prendre la moyenne des valeurs de l'intervalle $[y_k, y_{k+1}]$. L'erreur de quantification est alors:*

$$\epsilon = \frac{1}{12} \sum_{k=0}^{K-1} p_k \Delta_k^2.$$

# Quantification scalaire uniforme

## Definition (Quantificateur uniforme)

Un quantificateur uniforme est un quantificateur qui s'appuie sur un découpage en intervalles régulier de longueur $\Delta$ de l'espace de valeurs du signal. Dans l'hypothèse d'un quantificateur uniforme de haute définition, l'erreur de quantification minimale est donc

$$\epsilon = \frac{1}{12}\Delta^2 \sum_{k=0}^{K-1} p_k = \frac{\Delta^2}{12}.$$

- Remarque: On peut montrer que l'entropie d'un quantificateur à haute résolution est minimale lorsque ce dernier est uniforme.