Option : Théorie de l'information et Logique

5 février 2013

Joëlle Cohen

Voir `http://lacl.univ-paris12.fr//cohen/`
lien enseignement

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Chapitre I – Théorie de l'information

- I.Exemples
- II.Codes
- III.Mesure de l'information
- IV.Premier théorème de Shannon
- V.Canaux

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Chapitre I – Théorie de l'information

- I.Exemples
- II.Codes
- III.Mesure de l'information
- IV.Premier théorème de Shannon
- V.Canaux

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Chapitre I – Théorie de l'information

- I.Exemples
- II.Codes
- III.Mesure de l'information
- IV.Premier théorème de Shannon
- V.Canaux

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Chapitre I – Théorie de l'information

- I.Exemples
- II.Codes
- III.Mesure de l'information
- IV.Premier théorème de Shannon
- V.Canaux

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Chapitre I – Théorie de l'information

- I.Exemples
- II.Codes
- III.Mesure de l'information
- IV.Premier théorème de Shannon
- V.Canaux

Fax

Soit un document A4 à transmettre de format $21 \times 29,7$ cm, soit une surface de $623,7$ cm^2.

On transmet des points à imprimer par le codage suivant : $1 =$ point noir et $0 =$ point blanc.

On imprime 6200 points par cm^2 donc en tout

$6200 \times 623,7 = 3866948$ bits pour un document A4.

Si on utilise un débit de 14,4 Kbits par seconde alors une page A4 nécessite 268 s soit 4 min 28 s pour être transmis.

Grâce à des techniques de codage on peut réduire ce temps à une vingtaine de secondes.

Fax

Soit un document A4 à transmettre de format $21 \times 29,7$ cm, soit une surface de $623,7$ cm^2.
On transmet des points à imprimer par le codage suivant : $1 =$ point noir et $0 =$ point blanc.
On imprime 6200 points par cm^2 donc en tout
$6200 \times 623,7 = 3866948$ bits pour un document A4.
Si on utilise un débit de 14,4 Kbits par seconde alors une page A4 nécessite 268 s soit 4 min 28 s pour être transmis.
Grâce à des techniques de codage on peut réduire ce temps à une vingtaine de secondes.

Fax

Soit un document A4 à transmettre de format $21 \times 29,7$ cm, soit une surface de $623,7$ cm^2.
On transmet des points à imprimer par le codage suivant : $1 =$ point noir et $0 =$ point blanc.
On imprime 6200 points par cm^2 donc en tout
$6200 \times 623,7 = 3866948$ bits pour un document A4.
Si on utilise un débit de 14,4 Kbits par seconde alors une page A4 nécessite 268 s soit 4 min 28 s pour être transmis.
Grâce à des techniques de codage on peut réduire ce temps à une vingtaine de secondes.

Fax

Soit un document A4 à transmettre de format $21 \times 29,7$ cm, soit une surface de $623,7$ cm^2.
On transmet des points à imprimer par le codage suivant : $1 =$ point noir et $0 =$ point blanc.
On imprime 6200 points par cm^2 donc en tout
$6200 \times 623,7 = 3866948$ bits pour un document A4.
Si on utilise un débit de 14,4 Kbits par seconde alors une page A4 nécessite 268 s soit 4 min 28 s pour être transmis.
Grâce à des techniques de codage on peut réduire ce temps à une vingtaine de secondes.

Fax

Soit un document A4 à transmettre de format $21 \times 29,7$ cm, soit une surface de $623,7$ cm^2.
On transmet des points à imprimer par le codage suivant : $1 =$ point noir et $0 =$ point blanc.
On imprime 6200 points par cm^2 donc en tout
$6200 \times 623,7 = 3866948$ bits pour un document A4.
Si on utilise un débit de 14,4 Kbits par seconde alors une page A4 nécessite 268 s soit 4 min 28 s pour être transmis.
Grâce à des techniques de codage on peut réduire ce temps à une vingtaine de secondes.

Séquence vidéo sur CD-Rom

L'élément le plus élémentaire d'une image (pixel) est caractérisé par 3 composantes RVB en analogique et est codé numériquement selon 3 composantes Y, C_R et C_B.

Y est la luminance et C_R et C_B représentent la chrominance.

Y, C_R et C_B sont codées sur 8 bits : on a donc $2^8 = 256$ valeurs distinctes.

Soit une image composée de 480 lignes à 720 pixels par ligne.

Par combien de bits sera-t-elle codée ?

L'oeil humain ne distingue pas la chrominance de 2 pixels contigus donc le codage de la chrominance se fera à raison de 8 bits soit un octet pour 2 pixels.

Une image sera codée par

$$1 \times 720 \times 480\Big(\underbrace{\frac{1}{2}}_{C_R} + \underbrace{\frac{1}{2}}_{C_B} + \underbrace{1}_{Y}\Big) = 691200 \text{ octets.}$$

Séquence vidéo sur CD-Rom

L'élément le plus élémentaire d'une image (pixel) est caractérisé par 3 composantes RVB en analogique et est codé numériquement selon 3 composantes Y, C_R et C_B.
Y est la luminance et C_R et C_B représentent la chrominance.
Y, C_R et C_B sont codées sur 8 bits : on a donc $2^8 = 256$ valeurs distinctes.
Soit une image composée de 480 lignes à 720 pixels par ligne.
Par combien de bits sera-t-elle codée ?
L'oeil humain ne distingue pas la chrominance de 2 pixels contigus donc le codage de la chrominance se fera à raison de 8 bits soit un octet pour 2 pixels.
Une image sera codée par

$$1 \times 720 \times 480(\underbrace{\frac{1}{2}}_{C_R} + \underbrace{\frac{1}{2}}_{C_B} + \underbrace{1}_{Y}) = 691200 \text{ octets.}$$

Séquence vidéo sur CD-Rom

L'élément le plus élémentaire d'une image (pixel) est caractérisé par 3 composantes RVB en analogique et est codé numériquement selon 3 composantes Y, C_R et C_B.
Y est la luminance et C_R et C_B représentent la chrominance.
Y, C_R et C_B sont codées sur 8 bits : on a donc $2^8 = 256$ valeurs distinctes.

Soit une image composée de 480 lignes à 720 pixels par ligne.
Par combien de bits sera-t-elle codée ?
L'oeil humain ne distingue pas la chrominance de 2 pixels contigus donc le codage de la chrominance se fera à raison de 8 bits soit un octet pour 2 pixels.
Une image sera codée par

$$1 \times 720 \times 480\Big(\underbrace{\frac{1}{2}}_{C_R} + \underbrace{\frac{1}{2}}_{C_B} + \underbrace{1}_{Y}\Big) = 691200 \text{ octets.}$$

Séquence vidéo sur CD-Rom

L'élément le plus élémentaire d'une image (pixel) est caractérisé par 3 composantes RVB en analogique et est codé numériquement selon 3 composantes Y, C_R et C_B.
Y est la luminance et C_R et C_B représentent la chrominance.
Y, C_R et C_B sont codées sur 8 bits : on a donc $2^8 = 256$ valeurs distinctes.
Soit une image composée de 480 lignes à 720 pixels par ligne.
Par combien de bits sera-t-elle codée ?

L'oeil humain ne distingue pas la chrominance de 2 pixels contigus donc le codage de la chrominance se fera à raison de 8 bits soit un octet pour 2 pixels.
Une image sera codée par

$$1 \times 720 \times 480\Big(\underbrace{\frac{1}{2}}_{C_R} + \underbrace{\frac{1}{2}}_{C_B} + \underbrace{1}_{Y}\Big) = 691200 \text{ octets.}$$

Séquence vidéo sur CD-Rom

L'élément le plus élémentaire d'une image (pixel) est caractérisé par 3 composantes RVB en analogique et est codé numériquement selon 3 composantes Y, C_R et C_B.
Y est la luminance et C_R et C_B représentent la chrominance.
Y, C_R et C_B sont codées sur 8 bits : on a donc $2^8 = 256$ valeurs distinctes.
Soit une image composée de 480 lignes à 720 pixels par ligne.
Par combien de bits sera-t-elle codée ?
L'oeil humain ne distingue pas la chrominance de 2 pixels contigus donc le codage de la chrominance se fera à raison de 8 bits soit un octet pour 2 pixels.
Une image sera codée par

$$1 \times 720 \times 480(\underbrace{\frac{1}{2}}_{C_R} + \underbrace{\frac{1}{2}}_{C_B} + \underbrace{1}_{Y}) = 691200 \text{ octets.}$$

Séquence vidéo sur CD-Rom

L'élément le plus élémentaire d'une image (pixel) est caractérisé par 3 composantes RVB en analogique et est codé numériquement selon 3 composantes Y, C_R et C_B.
Y est la luminance et C_R et C_B représentent la chrominance.
Y, C_R et C_B sont codées sur 8 bits : on a donc $2^8 = 256$ valeurs distinctes.
Soit une image composée de 480 lignes à 720 pixels par ligne.
Par combien de bits sera-t-elle codée ?
L'oeil humain ne distingue pas la chrominance de 2 pixels contigus donc le codage de la chrominance se fera à raison de 8 bits soit un octet pour 2 pixels.
Une image sera codée par

$$1 \times 720 \times 480(\underbrace{\frac{1}{2}}_{C_R} + \underbrace{\frac{1}{2}}_{C_B} + \underbrace{1}_{Y}) = 691200 \text{ octets.}$$

Séquence vidéo sur CD-Rom

L'élément le plus élémentaire d'une image (pixel) est caractérisé par 3 composantes RVB en analogique et est codé numériquement selon 3 composantes Y, C_R et C_B.
Y est la luminance et C_R et C_B représentent la chrominance.
Y, C_R et C_B sont codées sur 8 bits : on a donc $2^8 = 256$ valeurs distinctes.
Soit une image composée de 480 lignes à 720 pixels par ligne.
Par combien de bits sera-t-elle codée ?
L'oeil humain ne distingue pas la chrominance de 2 pixels contigus donc le codage de la chrominance se fera à raison de 8 bits soit un octet pour 2 pixels.
Une image sera codée par

$$1 \times 720 \times 480(\underbrace{\frac{1}{2}}_{C_R} + \underbrace{\frac{1}{2}}_{C_B} + \underbrace{1}_{Y}) = 691200 \text{ octets.}$$

Séquence vidéo sur CD-Rom

Si on a une vidéo à 25 images par seconde alors une seconde d'images nécessite $25 \times 691200 = 17530000 = 17,53$ Moctets.

Un CD-Rom contient 650 Mo donc peut stocker $\frac{650}{17,53} = 37s$ de vidéo.

Avec un codage MPEG1 on atteint plus d'une heure d'image.

Séquence vidéo sur CD-Rom

Si on a une vidéo à 25 images par seconde alors une seconde d'images nécessite $25 \times 691200 = 17530000 = 17,53$ Moctets.

Un CD-Rom contient 650 Mo donc peut stocker $\frac{650}{17,53} = 37s$ de vidéo.

Avec un codage MPEG1 on atteint plus d'une heure d'image.

Séquence vidéo sur CD-Rom

Si on a une vidéo à 25 images par seconde alors une seconde d'images nécessite $25 \times 691200 = 17530000 = 17,53$ Moctets.

Un CD-Rom contient 650 Mo donc peut stocker $\frac{650}{17,53} = 37s$ de vidéo.

Avec un codage MPEG1 on atteint plus d'une heure d'image.

Fichier MP3

Un signal analogique stéréo a deux voies : une voie de gauche et une voie de droite chacune de fréquence comprise entre 0 et 20 Khz.

Le signal analogique est quasiment continu dans le temps.
Mais le passage au numérique suppose alors de rendre **discret** ce signal continu.
On va donc coder ponctuellement un certain nombre de sons par seconde : on appelle cela un échantillonnage.
Pour numériser en qualité CD Audio, on requiert une fréquence d'échantillonnage de 44,1Khz (c'est-à-dire on code 44100 échantillons de son par seconde) à raison de 16 bits par échantillon.
Donc une seconde de son sera codé par

$$\underbrace{44100}_{\text{échantillons}} \times \underbrace{16}_{\text{bits par échantillons}} \times \underbrace{2}_{\text{voies}} = 1,411 \text{ Mbits/seconde}$$

Fichier MP3

Un signal analogique stéréo a deux voies : une voie de gauche et une voie de droite chacune de fréquence comprise entre 0 et 20 Khz.

Le signal analogique est quasiment continu dans le temps.

Mais le passage au numérique suppose alors de rendre **discret** ce signal continu.

On va donc coder ponctuellement un certain nombre de sons par seconde : on appelle cela un échantillonnage.

Pour numériser en qualité CD Audio, on requiert une fréquence d'échantillonnage de 44,1Khz (c'est-à-dire on code 44100 échantillons de son par seconde) à raison de 16 bits par échantillon.

Donc une seconde de son sera codé par

$$\underbrace{44100}_{\text{échantillons}} \times \underbrace{16}_{\text{bits par échantillons}} \times \underbrace{2}_{\text{voies}} = 1,411 \text{ Mbits/seconde}$$

Fichier MP3

Un signal analogique stéréo a deux voies : une voie de gauche et une voie de droite chacune de fréquence comprise entre 0 et 20 Khz.
Le signal analogique est quasiment continu dans le temps.
Mais le passage au numérique suppose alors de rendre **discret** ce signal continu.
On va donc coder ponctuellement un certain nombre de sons par seconde : on appelle cela un échantillonnage.
Pour numériser en qualité CD Audio, on requiert une fréquence d'échantillonnage de 44,1Khz (c'est-à-dire on code 44100 échantillons de son par seconde) à raison de 16 bits par échantillon.
Donc une seconde de son sera codé par

$$\underbrace{44100}_{\textit{échantillons}} \times \underbrace{16}_{\textit{bits par échantillons}} \times \underbrace{2}_{\textit{voies}} = 1,411 \text{ Mbits/seconde}$$

Fichier MP3

Un signal analogique stéréo a deux voies : une voie de gauche et une voie de droite chacune de fréquence comprise entre 0 et 20 Khz.
Le signal analogique est quasiment continu dans le temps.
Mais le passage au numérique suppose alors de rendre **discret** ce signal continu.
On va donc coder ponctuellement un certain nombre de sons par seconde : on appelle cela un échantillonnage.
Pour numériser en qualité CD Audio, on requiert une fréquence d'échantillonnage de 44,1Khz (c'est-à-dire on code 44100 échantillons de son par seconde) à raison de 16 bits par échantillon.
Donc une seconde de son sera codé par

$$\underbrace{44100}_{\textit{échantillons}} \times \underbrace{16}_{\textit{bits par échantillons}} \times \underbrace{2}_{\textit{voies}} = 1,411 \text{ Mbits/seconde}$$

Fichier MP3

Un signal analogique stéréo a deux voies : une voie de gauche et une voie de droite chacune de fréquence comprise entre 0 et 20 Khz.
Le signal analogique est quasiment continu dans le temps.
Mais le passage au numérique suppose alors de rendre **discret** ce signal continu.
On va donc coder ponctuellement un certain nombre de sons par seconde : on appelle cela un échantillonnage.
Pour numériser en qualité CD Audio, on requiert une fréquence d'échantillonnage de 44,1Khz (c'est-à-dire on code 44100 échantillons de son par seconde) à raison de 16 bits par échantillon.
Donc une seconde de son sera codé par

$$\underbrace{44100}_{\text{échantillons}} \times \underbrace{16}_{\text{bits par échantillons}} \times \underbrace{2}_{\text{voies}} = 1,411 \text{ Mbits/seconde}$$

Fichier MP3

Un signal analogique stéréo a deux voies : une voie de gauche et une voie de droite chacune de fréquence comprise entre 0 et 20 Khz.
Le signal analogique est quasiment continu dans le temps.
Mais le passage au numérique suppose alors de rendre **discret** ce signal continu.
On va donc coder ponctuellement un certain nombre de sons par seconde : on appelle cela un échantillonnage.
Pour numériser en qualité CD Audio, on requiert une fréquence d'échantillonnage de 44,1Khz (c'est-à-dire on code 44100 échantillons de son par seconde) à raison de 16 bits par échantillon.
Donc une seconde de son sera codé par

$$\underbrace{44100}_{\textit{échantillons}} \times \underbrace{16}_{\textit{bits par échantillons}} \times \underbrace{2}_{\textit{voies}} = 1,411 \text{ Mbits/seconde}$$

Fichier MP3

Sans codage plus performant un CD-Rom contient alors $650 \times 8/1,411 = 3685$ secondes soit environ 1h de musique.

Le codage MP3 (MPEG 1 layer 3) permet de coder une seconde de musique par 128 Kbits.

Donc une minute de musique nécessite $\frac{128000 \times 60}{8} = 0,96$ Moctets.

Un CD-Rom peut alors contenir environ 650 minutes soit plus de 10h de musique.

Fichier MP3

Sans codage plus performant un CD-Rom contient alors $650 \times 8/1,411 = 3685$ secondes soit environ 1h de musique.

Le codage MP3 (MPEG 1 layer 3) permet de coder une seconde de musique par 128 Kbits.

Donc une minute de musique nécessite $\frac{128000 \times 60}{8} = 0,96$ Moctets.

Un CD-Rom peut alors contenir environ 650 minutes soit plus de 10h de musique.

Fichier MP3

Sans codage plus performant un CD-Rom contient alors $650 \times 8/1,411 = 3685$ secondes soit environ 1h de musique.

Le codage MP3 (MPEG 1 layer 3) permet de coder une seconde de musique par 128 Kbits.

Donc une minute de musique nécessite $\frac{128000 \times 60}{8} = 0,96$ Moctets.

Un CD-Rom peut alors contenir environ 650 minutes soit plus de 10h de musique.

Fichier MP3

Sans codage plus performant un CD-Rom contient alors $650 \times 8/1,411 = 3685$ secondes soit environ 1h de musique.

Le codage MP3 (MPEG 1 layer 3) permet de coder une seconde de musique par 128 Kbits.

Donc une minute de musique nécessite $\frac{128000 \times 60}{8} = 0,96$ Moctets.

Un CD-Rom peut alors contenir environ 650 minutes soit plus de 10h de musique.

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Qu'est-ce-que la théorie de l'information

La théorie de l'information est une théorie mathématique qui décrit les aspects fondamentaux des systèmes de communication. Elle a été initiée par C. Shannon dans les années 1940.
Un système de communication est la transmission d'une information depuis une source à travers un canal jusqu'à un récepteur.

Une source peut être

- une voix
- une suite de symboles binaires (bits)
- un signal électromagnétique ...

Le canal peut être

- une ligne téléphonique
- une liaison radio
- un support optique
- un support magnétique ...

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Une source peut être

- une voix
- une suite de symboles binaires (bits)
- un signal électromagnétique ...

Le canal peut être

- une ligne téléphonique
- une liaison radio
- un support optique
- un support magnétique ...

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Une source peut être

- une voix
- une suite de symboles binaires (bits)
- un signal électromagnétique ...

Le canal peut être

- une ligne téléphonique
- une liaison radio
- un support optique
- un support magnétique ...

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Une source peut être

- une voix
- une suite de symboles binaires (bits)
- un signal électromagnétique ...

Le canal peut être

- une ligne téléphonique
- une liaison radio
- un support optique
- un support magnétique ...

I.Exemples
II.Codes
III.Mesure de l'information
IV.Premier théorème de Shannon
V.Canaux

Une source peut être

- une voix
- une suite de symboles binaires (bits)
- un signal électromagnétique ...

Le canal peut être

- une ligne téléphonique
- une liaison radio
- un support optique
- un support magnétique ...

I.Exemples
II.Codes
III.Mesure de l'information
IV.Premier théorème de Shannon
V.Canaux

Une source peut être

- une voix
- une suite de symboles binaires (bits)
- un signal électromagnétique ...

Le canal peut être

- une ligne téléphonique
- une liaison radio
- un support optique
- un support magnétique ...

I.Exemples
II.Codes
III.Mesure de l'information
IV.Premier théorème de Shannon
V.Canaux

Une source peut être

- une voix
- une suite de symboles binaires (bits)
- un signal électromagnétique ...

Le canal peut être

- une ligne téléphonique
- une liaison radio
- un support optique
- un support magnétique ...

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Une source peut être

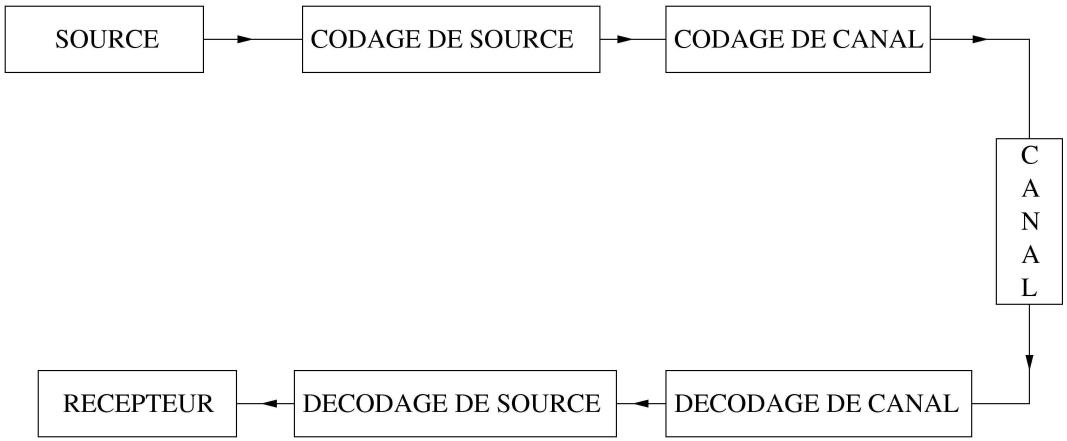
- une voix
- une suite de symboles binaires (bits)
- un signal électromagnétique ...

Le canal peut être

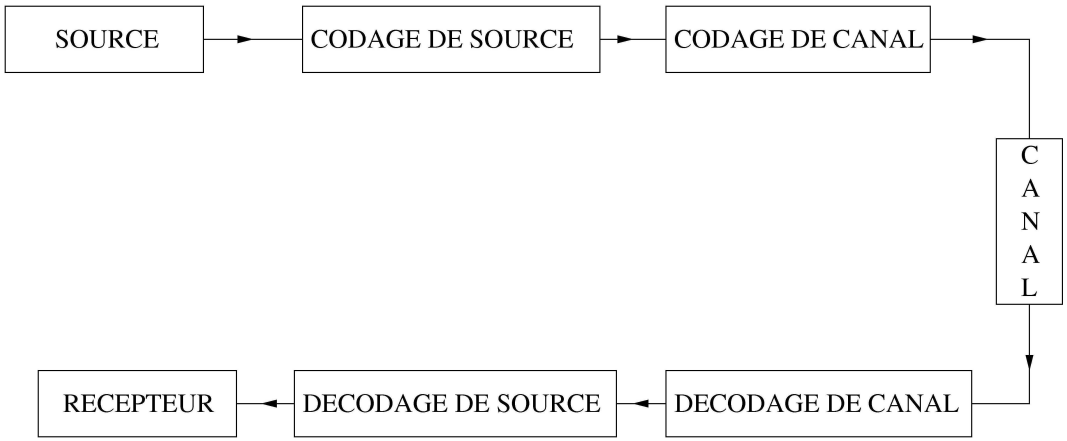
- une ligne téléphonique
- une liaison radio
- un support optique
- un support magnétique ...

Pour améliorer la transmission, on code la source pour réduire le débit de la source : cela peut se faire avec ou sans perte d'informations (on se limitera à sans perte) et consiste à transmettre **en moyenne** moins de symboles qu'il n'en provient de la source.

Le canal de transmission est sujet à diverses perturbations dues à l'environnement que l'on nommera bruit. Pour contrer ces perturbations qui peuvent engendrer soit perte soit déformation de l'information, on utilisera un codage de canal qui, contrairement au précédent, ajoutera des informations au message à transmettre ce qui augmentera le débit.



Pour améliorer la transmission, on code la source pour réduire le débit de la source : cela peut se faire avec ou sans perte d'informations (on se limitera à sans perte) et consiste à transmettre **en moyenne** moins de symboles qu'il n'en provient de la source.
Le canal de transmission est sujet à diverses perturbations dues à l'environnement que l'on nommera bruit. Pour contrer ces perturbations qui peuvent engendrer soit perte soit déformation de l'information, on utilisera un codage de canal qui, contrairement au précédent, ajoutera des informations au message à transmettre ce qui augmentera le débit.



I.Exemples
II.Codes
III.Mesure de l'information
IV.Premier théorème de Shannon
V.Canaux

Plusieurs modèles de sources peuvent être envisagés :

- sources discrètes sans mémoire
- sources discrètes stationnaires
- sources analogiques (onde caractérisée par un processus stochastique)...

On étudiera que les sources discrètes.
Une source sera modélisée par une séquence aléatoire de lettres choisies dans un alphabet fini $A = \{a_1 \dots a_n\}$.
Par exemple les données transmises par un ordinateur constituent une source sur l'alphabet $\{0, 1\}$.

codages

Soit A un ensemble fini dont les éléments seront appelés lettres. On note A^* l'ensemble des suites finies d'éléments de A. La suite vide est notée ε.
On munit A^* d'une opération appelée concaténation qui consiste à mettre bout à bout deux éléments de A^*.
Par exemple si $A = \{a, b, c, d\}$ alors $u = ababd$ et $v = cbdadcbc$ sont des éléments de A^* que l'on appellera mots. La concaténation de u et v est notée $u.v$ et on a $u.v = ababdcbdadcbc$.
Un langage sera alors une partie de A^* donc un ensemble des mots.

I.Exemples
II.Codes
III.Mesure de l'information
IV.Premier théorème de Shannon
V.Canaux

On va maintenant définir la notion de codage que l'on restreindra à un codage sur $\{0,1\}$.

Définition

un codage est une application injective γ de A dans $\{0,1\}^*$.

On étend γ à A^* en concaténant les images des lettres :

EXEMPLE : $\gamma : A \to \{0,1\}^*$

$a \to 000$

$b \to 01$

$c \to 10$

$d \to 1$

$\gamma(bac) = 0100010.$

I.Exemples
II.Codes
III.Mesure de l'information
IV.Premier théorème de Shannon
V.Canaux

On va maintenant définir la notion de codage que l'on restreindra à un codage sur $\{0,1\}$.

Définition

un codage est une application injective γ de A dans $\{0,1\}^*$.

On étend γ à A^* en concaténant les images des lettres :
EXEMPLE : $\gamma : A \to \{0,1\}^*$
$a \to 000$
$b \to 01$
$c \to 10$
$d \to 1$
$\gamma(bac) = 0100010.$

I.Exemples
II.Codes
III.Mesure de l'information
IV.Premier théorème de Shannon
V.Canaux

On va maintenant définir la notion de codage que l'on restreindra à un codage sur $\{0,1\}$.

Définition

un codage est une application injective γ de A dans $\{0,1\}^*$.

On étend γ à A^* en concaténant les images des lettres :
EXEMPLE : $\gamma : A \to \{0,1\}^*$
$a \to 000$
$b \to 01$
$c \to 10$
$d \to 1$
$\gamma(bac) = 0100010$.

code

On se restreindra à des codes sur $\{0,1\}$, soit aux codes binaires.

Définition

un langage L de $\{0,1\}^*$ est un code si pour tout mot de $\{0,1\}^*$ il n'existe pas deux factorisations de ce mot en mots de L .

EXEMPLE : $\gamma(A) = \{000, 01, 10, 1\}$ n'est pas un code .
$101 = (10)(1) = (1)(01)$.

Propriété

un ensemble de mots de même longueur est un code dit code de longueur fixe.

REMARQUE : dans un code binaire, le nombre de mots de longueur i est au plus 2^i puisque ces mots appartiennent à $\{0,1\}^i$.

code

On se restreindra à des codes sur $\{0,1\}$, soit aux codes binaires.

Définition

un langage L de $\{0,1\}^*$ est un code si pour tout mot de $\{0,1\}^*$ il n'existe pas deux factorisations de ce mot en mots de L .

EXEMPLE : $\gamma(A) = \{000, 01, 10, 1\}$ n'est pas un code .
$101 = (10)(1) = (1)(01)$.

Propriété

un ensemble de mots de même longueur est un code dit code de longueur fixe.

REMARQUE : dans un code binaire, le nombre de mots de longueur i est au plus 2^i puisque ces mots appartiennent à $\{0,1\}^i$.

code

On se restreindra à des codes sur $\{0,1\}$, soit aux codes binaires.

Définition

un langage L de $\{0,1\}^*$ est un code si pour tout mot de $\{0,1\}^*$ il n'existe pas deux factorisations de ce mot en mots de L .

EXEMPLE : $\gamma(A) = \{000, 01, 10, 1\}$ n'est pas un code .
$101 = (10)(1) = (1)(01)$.

Propriété

un ensemble de mots de même longueur est un code dit code de longueur fixe.

REMARQUE : dans un code binaire, le nombre de mots de longueur i est au plus 2^i puisque ces mots appartiennent à $\{0,1\}^i$.

code préfixe

Définition

un langage L est préfixe si aucun de ses mots n'est le début d'un autre de ses mots.

EXEMPLE : $= \{ab, abab, ababab\}$ n'est pas préfixe.

Propriété

tout langage préfixe est un code.

EXEMPLE : $C = \{11, 101, 00, 01\}$ est un code car préfixe.

code préfixe

Définition

un langage L est préfixe si aucun de ses mots n'est le début d'un autre de ses mots.

EXEMPLE : $= \{ab, abab, ababab\}$ n'est pas préfixe.

Propriété

tout langage préfixe est un code.

EXEMPLE : $C = \{11, 101, 00, 01\}$ est un code car préfixe.

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Le "décodage" est alors facilité par la propriété d'être préfixe. En effet, si un mot a une décomposition en mots de C alors cette décomposition se trouve en lisant de gauche à droite la suite de lettres et dès qu'un mot de C est reconnu alors il figure nécessairement dans l'unique factorisation recherchée.
EXEMPLE : pour $C = \{11, 101, 00, 01\}$,
$101110100101 = (101)(11)(01)(00)(101)$.

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

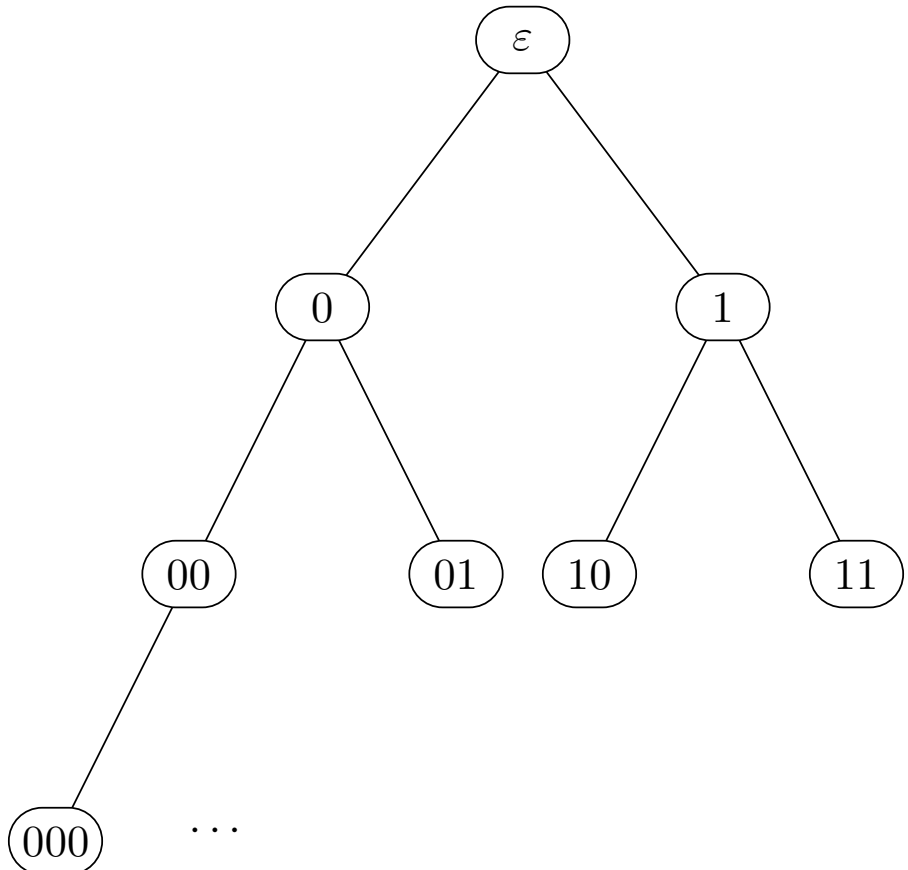
V.Canaux

Les propriétés suivantes vont nous permettre de nous concentrer sur ces codes préfixes.

Propriété

[inégalité de Kraft] tout code tel que $\sum_{c \in C} 2^{-|c|} \leq 1$ peut être transformé en un code préfixe équivalent (même nombre de mots, même distribution de longueur).

preuve : La construction du code préfixe équivalent se fera par des choix de chemins dans un arbre binaire dans lequel les branches de gauche seront étiquetées par 0 et celles de droite par 1.
Pour assurer un code préfixe, il suffit de choisir des chemins dont aucun n'est inclus dans un autre.
Chaque mot du code préfixe correspondra donc à un chemin aboutissant à un noeud de l'arbre.
REMARQUE : au niveau i de l'arbre il y a 2^i noeuds.


ε
0
1
00
01
10
11
000
...

Dans le code C, il y a des mots de différentes longueur $1, \ldots, p$. Soient $n_1, \ldots n_p$ les nombres de mots du code C de longueur respective $1, \ldots, p$. On a alors

$$\sum_{c \in C} 2^{-|c|} = \sum_{i=1}^{i=p} n_i 2^{-i} \leq 1$$

On en déduit

$n_1 2^{-1} \leq 1$

$n_1 2^{-1} + n_2 2^{-2} \leq 1$

$\ldots$

$n_1 2^{-1} + n_2 2^{-2} + \cdots + n_p 2^{-p} \leq 1$

Donc on a $n_1 \leq 2$, $n_2 \leq 4 - 2n_1$, $\ldots$,

$n_p \leq 2^p - \left(2^{p-1} n_1 + 2^{p-2} n_2 + \cdots + 2^1 n_{p-1}\right)$.

On a alors 3 cas :

- $n_1 = 2$: donc $n_2 = \cdots = n_p = 0$, les deux seuls mots de C sont 0 et 1 qui est un code préfixe
- $n_1 = 1$: C a un mot de longueur 1 et au plus 2 mots de longueur 2, il suffit alors de choisir pour le mot de longueur 1 le mot 0 et il reste deux mots de longueur 2 possibles 10 11
- $n_1 = 0$: C n'a pas de mot de longueur 1 et C a au plus 4 mots de longueur 2, il suffit de choisir parmi 00, 01, 10, 11.

Supposons construits les mots de longueur $1, \ldots, i-1$.
On a donc choisi dans l'arbre des chemins jusqu'au niveau $i-1$.
On sait que $n_i \leq 2^i - (2^{i-1}n_1 + 2^{i-2}n_2 + \cdots + 2^1 n_{i-1})$.
On va montrer que $(2^{i-1}n_1 + 2^{i-2}n_2 + \cdots + 2^1 n_{i-1})$ est le nombre de noeuds du niveau i qui sont sur les chemins poursuivant ceux que l'on a choisi (donc des noeuds descendants de ceux choisis précédemment).
En effet, les descendants des mots de longueur 1 sont au nombre de $2^{i-1}n_1$, les descendants des mots de longueur 2 sont au nombre de $2^{i-2}n_2, \ldots$, les descendants des mots de longueur $i-1$ sont au nombre de $2^{i-(i-1)}n_{i-1} = 2n_{i-1}$.
Il reste donc au plus $2^i - (2^{i-1}n_1 + 2^{i-2}n_2 + \cdots + 2^1 n_{i-1})$ noeuds disponibles. On peut donc choisir n_i mots au niveau i.
Par une récurrence finie on a ainsi construit un code équivalent à C. ■

ex $C = \{10, 11, 000, 101, 111, 1100, 1101\}$.
$\sum_{c \in C} 2^{-c} = 2 \times 2^{-2} + 3 \times 2^{-3} + 2 \times 2^{-4} = 1$.
On utilise un arbre de hauteur 4 (4 est la longueur maximale d'un mot de C). Sur chaque niveau k on choisit autant de chemins qui ne soient pas strictement inclus dans un autre chemin qu'il y a de mots de C de longueur k.

Théorème

[Mac Millan] tout code vérifie l'inégalité de Kraft.

conséquence : on pourra n'utiliser que des codes préfixes.

EXEMPLE : $C = \{000, 01, 10, 1\}$ n'est pas préfixe, n'est pas un code car il ne vérifie pas l'inégalité de Kraft.

$2^{-3} + 2^{-2} + 2^{-2} + 2^{-1} = \frac{1}{8} + \frac{1}{4} + \frac{1}{4} + \frac{1}{2} = 1,125$

EXEMPLE : $C = \{000, 010, 10, 1\}$ n'est pas préfixe, vérifie l'inégalité de Kraft mais n'est pas un code.

$1010 = 1010$

$2^{-3} + 2^{-3} + 2^{-2} + 2^{-1} = \frac{1}{8} + \frac{1}{8} + \frac{1}{4} + \frac{1}{2} = 1$

I.Exemples
II.Codes
III.Mesure de l'information
IV.Premier théorème de Shannon
V.Canaux

Théorème

[Mac Millan] tout code vérifie l'inégalité de Kraft.

conséquence : on pourra n'utiliser que des codes préfixes.

EXEMPLE : $C = \{000, 01, 10, 1\}$ n'est pas préfixe, n'est pas un code car il ne vérifie pas l'inégalité de Kraft.
$2^{-3} + 2^{-2} + 2^{-2} + 2^{-1} = \frac{1}{8} + \frac{1}{4} + \frac{1}{4} + \frac{1}{2} = 1,125$

EXEMPLE : $C = \{000, 010, 10, 1\}$ n'est pas préfixe, vérifie l'inégalité de Kraft mais n'est pas un code.
$1010 = 1010$
$2^{-3} + 2^{-3} + 2^{-2} + 2^{-1} = \frac{1}{8} + \frac{1}{8} + \frac{1}{4} + \frac{1}{2} = 1$

I.Exemples
II.Codes
III.Mesure de l'information
IV.Premier théorème de Shannon
V.Canaux

Théorème

[Mac Millan] tout code vérifie l'inégalité de Kraft.

conséquence : on pourra n'utiliser que des codes préfixes.

EXEMPLE : $C = \{000, 01, 10, 1\}$ n'est pas préfixe, n'est pas un code car il ne vérifie pas l'inégalité de Kraft.
$2^{-3} + 2^{-2} + 2^{-2} + 2^{-1} = \frac{1}{8} + \frac{1}{4} + \frac{1}{4} + \frac{1}{2} = 1,125$

EXEMPLE : $C = \{000, 010, 10, 1\}$ n'est pas préfixe, vérifie l'inégalité de Kraft mais n'est pas un code.
$1010 = 1010$
$2^{-3} + 2^{-3} + 2^{-2} + 2^{-1} = \frac{1}{8} + \frac{1}{8} + \frac{1}{4} + \frac{1}{2} = 1$

I.Exemples
II.Codes
III.Mesure de l'information
IV.Premier théorème de Shannon
V.Canaux

Théorème

[Mac Millan] tout code vérifie l'inégalité de Kraft.

conséquence : on pourra n'utiliser que des codes préfixes.

EXEMPLE : $C = \{000, 01, 10, 1\}$ n'est pas préfixe, n'est pas un code car il ne vérifie pas l'inégalité de Kraft.
$2^{-3} + 2^{-2} + 2^{-2} + 2^{-1} = \frac{1}{8} + \frac{1}{4} + \frac{1}{4} + \frac{1}{2} = 1,125$

EXEMPLE : $C = \{000, 010, 10, 1\}$ n'est pas préfixe, vérifie l'inégalité de Kraft mais n'est pas un code.
$1010 = 1010$
$2^{-3} + 2^{-3} + 2^{-2} + 2^{-1} = \frac{1}{8} + \frac{1}{8} + \frac{1}{4} + \frac{1}{2} = 1$

I.Exemples
II.Codes
III.Mesure de l'information
IV.Premier théorème de Shannon
V.Canaux

Théorème

[Mac Millan] tout code vérifie l'inégalité de Kraft.

conséquence : on pourra n'utiliser que des codes préfixes.

EXEMPLE : $C = \{000, 01, 10, 1\}$ n'est pas préfixe, n'est pas un code car il ne vérifie pas l'inégalité de Kraft.
$2^{-3} + 2^{-2} + 2^{-2} + 2^{-1} = \frac{1}{8} + \frac{1}{4} + \frac{1}{4} + \frac{1}{2} = 1,125$

EXEMPLE : $C = \{000, 010, 10, 1\}$ n'est pas préfixe, vérifie l'inégalité de Kraft mais n'est pas un code.
$1010 = 1010$
$2^{-3} + 2^{-3} + 2^{-2} + 2^{-1} = \frac{1}{8} + \frac{1}{8} + \frac{1}{4} + \frac{1}{2} = 1$

I.Exemples
II.Codes
III.Mesure de l'information
IV.Premier théorème de Shannon
V.Canaux

Théorème

[Mac Millan] tout code vérifie l'inégalité de Kraft.

conséquence : on pourra n'utiliser que des codes préfixes.

EXEMPLE : $C = \{000, 01, 10, 1\}$ n'est pas préfixe, n'est pas un code car il ne vérifie pas l'inégalité de Kraft.
$2^{-3} + 2^{-2} + 2^{-2} + 2^{-1} = \frac{1}{8} + \frac{1}{4} + \frac{1}{4} + \frac{1}{2} = 1,125$

EXEMPLE : $C = \{000, 010, 10, 1\}$ n'est pas préfixe, vérifie l'inégalité de Kraft mais n'est pas un code.
$1010 = 1010$
$2^{-3} + 2^{-3} + 2^{-2} + 2^{-1} = \frac{1}{8} + \frac{1}{8} + \frac{1}{4} + \frac{1}{2} = 1$

Exemples de code non binaire : code Morse

C'est un code ternaire : les trois symboles utilisés sont le point, le trait et la pause (permettant de séparer les lettres), chacun correspondant à une impulsion électrique de durée croissante.
Chaque lettre est codée selon sa fréquence d'utilisation en anglais.
Par exemple, A est codé par .– et E par ., etc.

Exemples de code non binaire : code arithmétique

Le message complet est codé par un nombre décimal en virgule flottante.
Par exemple, on doit coder MATT DAMON.
On va estimer les probabilités des lettres source en les assimilant à leur fréquence d'apparition dans le message. On a alors

A	D	M	N	O	T	espace
2/10	1/10	2/10	1/10	1/10	2/10	1/10

On attribue ensuite à chaque lettre un intervalle contenu dans $[0, 1]$: les intervalles doivent être fermés à gauche, ouverts à droite, disjoints 2 à 2 et leurs longueurs doivent correspondre à la fréquence d'apparition de la lettre associée.

A	D	M	N	O	T	esp
[0,1 ;0,3[	[0,3 ;0,4[	[0,4 ;0,6[	[0,6 ;0,7[	[0,7 ;0,8[	[0,8 ;1[	[0 ;0

Le nombre qui représentera le message va appartenir à l'intervalle correspondant à la première lettre soit $I_1 = [0,4; 0,6[$. Pour affiner la valeur cherchée, cet intervalle sera restreint par la deuxième lettre, ici A, de la façon suivante :

- on ajoute à la borne inférieure de I_1 la borne inférieure de l'intervalle de la deuxième lettre multipliée par la longueur de I_1.
- on ajoute à la borne inférieure de I_1 la borne supérieure de l'intervalle de la deuxième lettre multipliée par la longueur de I_1.

On obtient ainsi un nouvel intervalle que l'on notera I_2. On applique à I_2 la même restriction avec la troisième lettre et ainsi de suite jusqu'à la dernière restriction.

A	D	M	N	O	T	esp
[0,1 ;0,3[	[0,3 ;0,4[	[0,4 ;0,6[	[0,6 ;0,7[	[0,7 ;0,8[	[0,8 ;1[	[0 ;0

Le nombre qui représentera le message va appartenir à l'intervalle correspondant à la première lettre soit $I_1 = [0,4; 0,6[$. Pour affiner la valeur cherchée, cet intervalle sera restreint par la deuxième lettre, ici A, de la façon suivante :

- on ajoute à la borne inférieure de I_1 la borne inférieure de l'intervalle de la deuxième lettre multipliée par la longueur de I_1.
- on ajoute à la borne inférieure de I_1 la borne supérieure de l'intervalle de la deuxième lettre multipliée par la longueur de I_1.

On obtient ainsi un nouvel intervalle que l'on notera I_2. On applique à I_2 la même restriction avec la troisième lettre et ainsi de suite jusqu'à la dernière restriction.

A	D	M	N	O	T	esp
[0,1 ;0,3[	[0,3 ;0,4[	[0,4 ;0,6[	[0,6 ;0,7[	[0,7 ;0,8[	[0,8 ;1[	[0 ;0

Le nombre qui représentera le message va appartenir à l'intervalle correspondant à la première lettre soit $I_1 = [0,4; 0,6[$. Pour affiner la valeur cherchée, cet intervalle sera restreint par la deuxième lettre, ici A, de la façon suivante :

- on ajoute à la borne inférieure de I_1 la borne inférieure de l'intervalle de la deuxième lettre multipliée par la longueur de I_1.
- on ajoute à la borne inférieure de I_1 la borne supérieure de l'intervalle de la deuxième lettre multipliée par la longueur de I_1.

On obtient ainsi un nouvel intervalle que l'on notera I_2. On applique à I_2 la même restriction avec la troisième lettre et ainsi de suite jusqu'à la dernière restriction.

A	D	M	N	O	T	esp
[0,1 ;0,3[	[0,3 ;0,4[	[0,4 ;0,6[	[0,6 ;0,7[	[0,7 ;0,8[	[0,8 ;1[	[0 ;0

Le nombre qui représentera le message va appartenir à l'intervalle correspondant à la première lettre soit $I_1 = [0,4; 0,6[$. Pour affiner la valeur cherchée, cet intervalle sera restreint par la deuxième lettre, ici A, de la façon suivante :

- on ajoute à la borne inférieure de I_1 la borne inférieure de l'intervalle de la deuxième lettre multipliée par la longueur de I_1.
- on ajoute à la borne inférieure de I_1 la borne supérieure de l'intervalle de la deuxième lettre multipliée par la longueur de I_1.

On obtient ainsi un nouvel intervalle que l'on notera I_2. On applique à I_2 la même restriction avec la troisième lettre et ainsi de suite jusqu'à la dernière restriction.

- pour M, $I_1 = [0,4; 0,6[$
- pour MA,
  $I_2 = [0,4 + 0,1 \times 0,2; 0,4 + 0,3 \times 0,2[= [0,42; 0,46[$
- pour MAT,
  $I_3 = [0,42 + 0,8 \times 0,04; 0,42 + 1 \times 0,04[= [0,452; 0,46[$
- pour MATT, $I_4 =$
  $[0,452 + 0,8 \times 0,008; 0,452 + 1 \times 0,008[= [0,4584; 0,46[$
- pour MATT ,
  $I_5 = [0,4584 + 0 \times 0,0016; 0,4584 + 0,1 \times 0,0016[=$
  $[0,4584; 0,45856[$
- pour MATT D,
  $I_6 = [0,4584 + 0,3 \times 0,00016; 0,4584 + 0,4 \times 0,00016[=$
  $[0,458448; 0,458464[$

- pour M, $I_1 = [0,4; 0,6[$
- pour MA,
  $I_2 = [0,4 + 0,1 \times 0,2; 0,4 + 0,3 \times 0,2[= [0,42; 0,46[$
- pour MAT,
  $I_3 = [0,42 + 0,8 \times 0,04; 0,42 + 1 \times 0,04[= [0,452; 0,46[$
- pour MATT, $I_4 =$
  $[0,452 + 0,8 \times 0,008; 0,452 + 1 \times 0,008[= [0,4584; 0,46[$
- pour MATT ,
  $I_5 = [0,4584 + 0 \times 0,0016; 0,4584 + 0,1 \times 0,0016[=$
  $[0,4584; 0,45856[$
- pour MATT D,
  $I_6 = [0,4584 + 0,3 \times 0,00016; 0,4584 + 0,4 \times 0,00016[=$
  $[0,458448; 0,458464[$

- pour M, $I_1 = [0,4; 0,6[$
- pour MA,
  $I_2 = [0,4 + 0,1 \times 0,2; 0,4 + 0,3 \times 0,2[= [0,42; 0,46[$
- pour MAT,
  $I_3 = [0,42 + 0,8 \times 0,04; 0,42 + 1 \times 0,04[= [0,452; 0,46[$
- pour MATT, $I_4 =$
  $[0,452 + 0,8 \times 0,008; 0,452 + 1 \times 0,008[= [0,4584; 0,46[$
- pour MATT ,
  $I_5 = [0,4584 + 0 \times 0,0016; 0,4584 + 0,1 \times 0,0016[=$
  $[0,4584; 0,45856[$
- pour MATT D,
  $I_6 = [0,4584 + 0,3 \times 0,00016; 0,4584 + 0,4 \times 0,00016[=$
  $[0,458448; 0,458464[$

- pour M, $I_1 = [0,4; 0,6[$
- pour MA,
  $I_2 = [0,4 + 0,1 \times 0,2; 0,4 + 0,3 \times 0,2[= [0,42; 0,46[$
- pour MAT,
  $I_3 = [0,42 + 0,8 \times 0,04; 0,42 + 1 \times 0,04[= [0,452; 0,46[$
- pour MATT, $I_4 =$
  $[0,452 + 0,8 \times 0,008; 0,452 + 1 \times 0,008[= [0,4584; 0,46[$
- pour MATT ,
  $I_5 = [0,4584 + 0 \times 0,0016; 0,4584 + 0,1 \times 0,0016[=$
  $[0,4584; 0,45856[$
- pour MATT D,
  $I_6 = [0,4584 + 0,3 \times 0,00016; 0,4584 + 0,4 \times 0,00016[=$
  $[0,458448; 0,458464[$

- pour M, $I_1 = [0,4; 0,6[$
- pour MA,
  $I_2 = [0,4 + 0,1 \times 0,2; 0,4 + 0,3 \times 0,2[= [0,42; 0,46[$
- pour MAT,
  $I_3 = [0,42 + 0,8 \times 0,04; 0,42 + 1 \times 0,04[= [0,452; 0,46[$
- pour MATT, $I_4 =$
  $[0,452 + 0,8 \times 0,008; 0,452 + 1 \times 0,008[= [0,4584; 0,46[$
- pour MATT ,
  $I_5 = [0,4584 + 0 \times 0,0016; 0,4584 + 0,1 \times 0,0016[=$
  $[0,4584; 0,45856[$
- pour MATT D,
  $I_6 = [0,4584 + 0,3 \times 0,00016; 0,4584 + 0,4 \times 0,00016[=$
  $[0,458448; 0,458464[$

- pour M, $I_1 = [0,4; 0,6[$
- pour MA,
  $I_2 = [0,4 + 0,1 \times 0,2; 0,4 + 0,3 \times 0,2[= [0,42; 0,46[$
- pour MAT,
  $I_3 = [0,42 + 0,8 \times 0,04; 0,42 + 1 \times 0,04[= [0,452; 0,46[$
- pour MATT, $I_4 =$
  $[0,452 + 0,8 \times 0,008; 0,452 + 1 \times 0,008[= [0,4584; 0,46[$
- pour MATT ,
  $I_5 = [0,4584 + 0 \times 0,0016; 0,4584 + 0,1 \times 0,0016[=$
  $[0,4584; 0,45856[$
- pour MATT D,
  $I_6 = [0,4584 + 0,3 \times 0,00016; 0,4584 + 0,4 \times 0,00016[=$
  $[0,458448; 0,458464[$

- pour MATT DA,
  $I_7 = [0,458448 + 0,1 \times 0,000016; 0,458448 + 0,3 \times 0,000016[= [0,4584496; 0,4584528[$
- pour MATT DAM,
  $I_8 = [0,4584496 + 0,4 \times 0,0000032; 0,4584496 + 0,6 \times 0,0000032[= [0,45845088; 0,45845152[$
- pour MATT DAMO,
  $I_9 = [0,45845088 + 0,7 \times 0,00000064; 0,45845088 + 0,8 \times 0,00000064[= [0,458451328; 0,458451392[$
- pour MATT DAMON,
  $I_{10} = [0,458451328 + 0,6 \times 0,000000064; 0,458451328 + 0,7 \times 0,000000064[= [0,458451712; 0,458451774[$

La borne inférieure du dernier intervalle est choisie pour coder le message, soit $x = 0,458451712$.

- pour MATT DA,
  $I_7 = [0,458448 + 0,1 \times 0,000016; 0,458448 + 0,3 \times 0,000016[= [0,4584496; 0,4584528[$
- pour MATT DAM,
  $I_8 = [0,4584496 + 0,4 \times 0,0000032; 0,4584496 + 0,6 \times 0,0000032[= [0,45845088; 0,45845152[$
- pour MATT DAMO,
  $I_9 = [0,45845088 + 0,7 \times 0,00000064; 0,45845088 + 0,8 \times 0,00000064[= [0,458451328; 0,458451392[$
- pour MATT DAMON,
  $I_{10} = [0,458451328 + 0,6 \times 0,000000064; 0,458451328 + 0,7 \times 0,000000064[= [0,458451712; 0,458451774[$

La borne inférieure du dernier intervalle est choisie pour coder le message, soit $x = 0,458451712$.

- pour MATT DA,
  $I_7 = [0,458448 + 0,1 \times 0,000016; 0,458448 + 0,3 \times 0,000016[= [0,4584496; 0,4584528[$
- pour MATT DAM,
  $I_8 = [0,4584496 + 0,4 \times 0,0000032; 0,4584496 + 0,6 \times 0,0000032[= [0,45845088; 0,45845152[$
- pour MATT DAMO,
  $I_9 = [0,45845088 + 0,7 \times 0,00000064; 0,45845088 + 0,8 \times 0,00000064[= [0,458451328; 0,458451392[$
- pour MATT DAMON,
  $I_{10} = [0,458451328 + 0,6 \times 0,000000064; 0,458451328 + 0,7 \times 0,000000064[= [0,458451712; 0,458451774[$

La borne inférieure du dernier intervalle est choisie pour coder le message, soit $x = 0,458451712$.

- pour MATT DA,
  $I_7 = [0,458448 + 0,1 \times 0,000016; 0,458448 + 0,3 \times 0,000016[= [0,4584496; 0,4584528[$
- pour MATT DAM,
  $I_8 = [0,4584496 + 0,4 \times 0,0000032; 0,4584496 + 0,6 \times 0,0000032[= [0,45845088; 0,45845152[$
- pour MATT DAMO,
  $I_9 = [0,45845088 + 0,7 \times 0,00000064; 0,45845088 + 0,8 \times 0,00000064[= [0,458451328; 0,458451392[$
- pour MATT DAMON,
  $I_{10} = [0,458451328 + 0,6 \times 0,000000064; 0,458451328 + 0,7 \times 0,000000064[= [0,458451712; 0,458451774[$

La borne inférieure du dernier intervalle est choisie pour coder le message, soit $x = 0,458451712$.

- pour MATT DA,
  $I_7 = [0,458448 + 0,1 \times 0,000016; 0,458448 + 0,3 \times 0,000016[= [0,4584496; 0,4584528[$
- pour MATT DAM,
  $I_8 = [0,4584496 + 0,4 \times 0,0000032; 0,4584496 + 0,6 \times 0,0000032[= [0,45845088; 0,45845152[$
- pour MATT DAMO,
  $I_9 = [0,45845088 + 0,7 \times 0,00000064; 0,45845088 + 0,8 \times 0,00000064[= [0,458451328; 0,458451392[$
- pour MATT DAMON,
  $I_{10} = [0,458451328 + 0,6 \times 0,000000064; 0,458451328 + 0,7 \times 0,000000064[= [0,458451712; 0,458451774[$

La borne inférieure du dernier intervalle est choisie pour coder le message, soit $x = 0,458451712$.

- pour MATT DA,
  $I_7 = [0,458448 + 0,1 \times 0,000016; 0,458448 + 0,3 \times 0,000016[= [0,4584496; 0,4584528[$
- pour MATT DAM,
  $I_8 = [0,4584496 + 0,4 \times 0,0000032; 0,4584496 + 0,6 \times 0,0000032[= [0,45845088; 0,45845152[$
- pour MATT DAMO,
  $I_9 = [0,45845088 + 0,7 \times 0,00000064; 0,45845088 + 0,8 \times 0,00000064[= [0,458451328; 0,458451392[$
- pour MATT DAMON,
  $I_{10} = [0,458451328 + 0,6 \times 0,000000064; 0,458451328 + 0,7 \times 0,000000064[= [0,458451712; 0,458451774[$

La borne inférieure du dernier intervalle est choisie pour coder le message, soit $x = 0,458451712$.

Pour décoder le message, on commence par déterminer dans quel intervalle se trouve le nombre reçu.
Il appartient à $[0,4; 0,6[$ donc la première lettre est M.
Ensuite on modifie x en lui soustrayant la borne inférieure de la première lettre, soit $0,4$ et on divise le résultat par la longueur de l'intervalle de M : on obtient
$x = (0,458451712 - 0,4) \div 0,2 = 0,229225854$.
La deuxième lettre correspond à l'intervalle auquel appartient x, soit l'intervalle $[0,1; 0,3[$: c'est donc A.
Ainsi de proche en proche on décode le message.
Ici, on a choisi des intervalles déterminés par le message lui-même mais en pratique les intervalles sont fixés au préalable afin d'être connu du récepteur.

Exemple de codage

Supposons que l'on émette 4 lettres pour nos messages ; par exemple $\{a, b, c, d\}$.

Chaque lettre est émise avec une certaine fréquence : $p(a) = 1/2$, $p(b) = 1/4$, et $p(c) = p(d) = 1/8$.

On se propose de "coder" chaque lettre par un code binaire.

	C_1	C_2
a	00	1
b	01	01
c	10	001
d	11	000

La longueur moyenne pour coder une lettre sera

- $l(C_1) = 1/2 \times 2 + 1/4 \times 2 + 1/8 \times 2 + 1/8 \times 2 = 2$
- $l(C_2) = 1/2 \times 1 + 1/4 \times 2 + 1/8 \times 3 + 1/8 \times 3 = 1,75$

Sur un message de 1000 lettres le code C_1 émettra 2000 bits alors que C_2 émettra 1750 bits.

C_2 est meilleur que C_1 : c'est une compression.

Exemple de codage

Supposons que l'on émette 4 lettres pour nos messages ; par exemple $\{a, b, c, d\}$.
Chaque lettre est émise avec une certaine fréquence :
$p(a) = 1/2$, $p(b) = 1/4$, et $p(c) = p(d) = 1/8$.
On se propose de "coder" chaque lettre par un code binaire.

	C_1	C_2
a	00	1
b	01	01
c	10	001
d	11	000

La longueur moyenne pour coder une lettre sera

- $l(C_1) = 1/2 \times 2 + 1/4 \times 2 + 1/8 \times 2 + 1/8 \times 2 = 2$
- $l(C_2) = 1/2 \times 1 + 1/4 \times 2 + 1/8 \times 3 + 1/8 \times 3 = 1,75$

Sur un message de 1000 lettres le code C_1 émettra 2000 bits alors que C_2 émettra 1750 bits.
C_2 est meilleur que C_1 : c'est une compression.

Exemple de codage

Supposons que l'on émette 4 lettres pour nos messages ; par exemple $\{a, b, c, d\}$.
Chaque lettre est émise avec une certaine fréquence :
$p(a) = 1/2$, $p(b) = 1/4$, et $p(c) = p(d) = 1/8$.
On se propose de "coder" chaque lettre par un code binaire.

	C_1	C_2
a	00	1
b	01	01
c	10	001
d	11	000

La longueur moyenne pour coder une lettre sera

- $l(C_1) = 1/2 \times 2 + 1/4 \times 2 + 1/8 \times 2 + 1/8 \times 2 = 2$
- $l(C_2) = 1/2 \times 1 + 1/4 \times 2 + 1/8 \times 3 + 1/8 \times 3 = 1,75$

Sur un message de 1000 lettres le code C_1 émettra 2000 bits alors que C_2 émettra 1750 bits.
C_2 est meilleur que C_1 : c'est une compression.

Exemple de codage

Supposons que l'on émette 4 lettres pour nos messages ; par exemple $\{a, b, c, d\}$.
Chaque lettre est émise avec une certaine fréquence :
$p(a) = 1/2$, $p(b) = 1/4$, et $p(c) = p(d) = 1/8$.
On se propose de "coder" chaque lettre par un code binaire.

	C_1	C_2
a	00	1
b	01	01
c	10	001
d	11	000

La longueur moyenne pour coder une lettre sera

- $l(C_1) = 1/2 \times 2 + 1/4 \times 2 + 1/8 \times 2 + 1/8 \times 2 = 2$
- $l(C_2) = 1/2 \times 1 + 1/4 \times 2 + 1/8 \times 3 + 1/8 \times 3 = 1,75$

Sur un message de 1000 lettres le code C_1 émettra 2000 bits alors que C_2 émettra 1750 bits.
C_2 est meilleur que C_1 : c'est une compression.

Exemple de codage

Supposons que l'on émette 4 lettres pour nos messages ; par exemple $\{a, b, c, d\}$.
Chaque lettre est émise avec une certaine fréquence :
$p(a) = 1/2$, $p(b) = 1/4$, et $p(c) = p(d) = 1/8$.
On se propose de "coder" chaque lettre par un code binaire.

	C_1	C_2
a	00	1
b	01	01
c	10	001
d	11	000

La longueur moyenne pour coder une lettre sera

- $l(C_1) = 1/2 \times 2 + 1/4 \times 2 + 1/8 \times 2 + 1/8 \times 2 = 2$
- $l(C_2) = 1/2 \times 1 + 1/4 \times 2 + 1/8 \times 3 + 1/8 \times 3 = 1,75$

Sur un message de 1000 lettres le code C_1 émettra 2000 bits alors que C_2 émettra 1750 bits.
C_2 est meilleur que C_1 : c'est une compression.

Exemple de codage

Supposons que l'on émette 4 lettres pour nos messages ; par exemple $\{a, b, c, d\}$.
Chaque lettre est émise avec une certaine fréquence :
$p(a) = 1/2$, $p(b) = 1/4$, et $p(c) = p(d) = 1/8$.
On se propose de "coder" chaque lettre par un code binaire.

	C_1	C_2
a	00	1
b	01	01
c	10	001
d	11	000

La longueur moyenne pour coder une lettre sera

- $l(C_1) = 1/2 \times 2 + 1/4 \times 2 + 1/8 \times 2 + 1/8 \times 2 = 2$
- $l(C_2) = 1/2 \times 1 + 1/4 \times 2 + 1/8 \times 3 + 1/8 \times 3 = 1,75$

Sur un message de 1000 lettres le code C_1 émettra 2000 bits alors que C_2 émettra 1750 bits.
C_2 est meilleur que C_1 : c'est une compression.

Exemple de codage

Supposons que l'on émette 4 lettres pour nos messages ; par exemple $\{a, b, c, d\}$.
Chaque lettre est émise avec une certaine fréquence :
$p(a) = 1/2$, $p(b) = 1/4$, et $p(c) = p(d) = 1/8$.
On se propose de "coder" chaque lettre par un code binaire.

	C_1	C_2
a	00	1
b	01	01
c	10	001
d	11	000

La longueur moyenne pour coder une lettre sera

- $l(C_1) = 1/2 \times 2 + 1/4 \times 2 + 1/8 \times 2 + 1/8 \times 2 = 2$
- $l(C_2) = 1/2 \times 1 + 1/4 \times 2 + 1/8 \times 3 + 1/8 \times 3 = 1,75$

Sur un message de 1000 lettres le code C_1 émettra 2000 bits alors que C_2 émettra 1750 bits.
C_2 est meilleur que C_1 : c'est une compression.

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Un message n'apporte de l'information que si son contenu n'est pas connu à l'avance de son destinataire.

Par exemple, si je connais le prochain bit à recevoir, je n'ai pas besoin de le recevoir.

On va supposer que l'ensemble de tous les messages possibles est fini.

Fournir une information c'est lever une incertitude.

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Un message n'apporte de l'information que si son contenu n'est pas connu à l'avance de son destinataire.
Par exemple, si je connais le prochain bit à recevoir, je n'ai pas besoin de le recevoir.
On va supposer que l'ensemble de tous les messages possibles est fini.
Fournir une information c'est lever une incertitude.

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Un message n'apporte de l'information que si son contenu n'est pas connu à l'avance de son destinataire.
Par exemple, si je connais le prochain bit à recevoir, je n'ai pas besoin de le recevoir.
On va supposer que l'ensemble de tous les messages possibles est fini.
Fournir une information c'est lever une incertitude.

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Un message n'apporte de l'information que si son contenu n'est pas connu à l'avance de son destinataire.
Par exemple, si je connais le prochain bit à recevoir, je n'ai pas besoin de le recevoir.
On va supposer que l'ensemble de tous les messages possibles est fini.
Fournir une information c'est lever une incertitude.

Cette incertitude peut varier pour un même événement si on a connaissance d'une autre information : pour 2 événements E et F, si

- $p(E/F) < p(E)$ alors l'incertitude sur E augmente si on sait que F s'est réalisé
- $p(E/F) = p(E)$ alors E et F sont indépendants, l'information apportée par F n'influence pas l'incertitude sur la survenue de E
- $p(E/F) > p(E)$ alors E devient plus probable si on sait que F s'est réalisé

EXEMPLE : on a reçu la lettre q. La probabilité de recevoir la lettre u est maintenant plus importante.

EXEMPLE : on a tiré une carte rouge. La probabilité que ce soit D♡ passe de $\frac{1}{32}$ à $\frac{1}{8}$.

Cette incertitude peut varier pour un même événement si on a connaissance d'une autre information : pour 2 événements E et F, si

- $p(E/F) < p(E)$ alors l'incertitude sur E augmente si on sait que F s'est réalisé
- $p(E/F) = p(E)$ alors E et F sont indépendants, l'information apportée par F n'influence pas l'incertitude sur la survenue de E
- $p(E/F) > p(E)$ alors E devient plus probable si on sait que F s'est réalisé

EXEMPLE : on a reçu la lettre q. La probabilité de recevoir la lettre u est maintenant plus importante.

EXEMPLE : on a tiré une carte rouge. La probabilité que ce soit D♡ passe de $\frac{1}{32}$ à $\frac{1}{8}$.

Cette incertitude peut varier pour un même événement si on a connaissance d'une autre information : pour 2 événements E et F, si

- $p(E/F) < p(E)$ alors l'incertitude sur E augmente si on sait que F s'est réalisé
- $p(E/F) = p(E)$ alors E et F sont indépendants, l'information apportée par F n'influence pas l'incertitude sur la survenue de E
- $p(E/F) > p(E)$ alors E devient plus probable si on sait que F s'est réalisé

EXEMPLE : on a reçu la lettre q. La probabilité de recevoir la lettre u est maintenant plus importante.

EXEMPLE : on a tiré une carte rouge. La probabilité que ce soit D♡ passe de $\frac{1}{32}$ à $\frac{1}{8}$.

Cette incertitude peut varier pour un même événement si on a connaissance d'une autre information : pour 2 événements E et F, si

- $p(E/F) < p(E)$ alors l'incertitude sur E augmente si on sait que F s'est réalisé
- $p(E/F) = p(E)$ alors E et F sont indépendants, l'information apportée par F n'influence pas l'incertitude sur la survenue de E
- $p(E/F) > p(E)$ alors E devient plus probable si on sait que F s'est réalisé

EXEMPLE : on a reçu la lettre q. La probabilité de recevoir la lettre u est maintenant plus importante.

EXEMPLE : on a tiré une carte rouge. La probabilité que ce soit D♡ passe de $\frac{1}{32}$ à $\frac{1}{8}$.

Cette incertitude peut varier pour un même événement si on a connaissance d'une autre information : pour 2 événements E et F, si

- $p(E/F) < p(E)$ alors l'incertitude sur E augmente si on sait que F s'est réalisé
- $p(E/F) = p(E)$ alors E et F sont indépendants, l'information apportée par F n'influence pas l'incertitude sur la survenue de E
- $p(E/F) > p(E)$ alors E devient plus probable si on sait que F s'est réalisé

EXEMPLE : on a reçu la lettre q. La probabilité de recevoir la lettre u est maintenant plus importante.

EXEMPLE : on a tiré une carte rouge. La probabilité que ce soit D♡ passe de $\frac{1}{32}$ à $\frac{1}{8}$.

Cette incertitude peut varier pour un même événement si on a connaissance d'une autre information : pour 2 événements E et F, si

- $p(E/F) < p(E)$ alors l'incertitude sur E augmente si on sait que F s'est réalisé
- $p(E/F) = p(E)$ alors E et F sont indépendants, l'information apportée par F n'influence pas l'incertitude sur la survenue de E
- $p(E/F) > p(E)$ alors E devient plus probable si on sait que F s'est réalisé

EXEMPLE : on a reçu la lettre q. La probabilité de recevoir la lettre u est maintenant plus importante.

EXEMPLE : on a tiré une carte rouge. La probabilité que ce soit D♡ passe de $\frac{1}{32}$ à $\frac{1}{8}$.

Cette incertitude peut varier pour un même événement si on a connaissance d'une autre information : pour 2 événements E et F, si

- $p(E/F) < p(E)$ alors l'incertitude sur E augmente si on sait que F s'est réalisé
- $p(E/F) = p(E)$ alors E et F sont indépendants, l'information apportée par F n'influence pas l'incertitude sur la survenue de E
- $p(E/F) > p(E)$ alors E devient plus probable si on sait que F s'est réalisé

EXEMPLE : on a reçu la lettre q. La probabilité de recevoir la lettre u est maintenant plus importante.

EXEMPLE : on a tiré une carte rouge. La probabilité que ce soit D♡ passe de $\frac{1}{32}$ à $\frac{1}{8}$.

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Rappel : si E et F sont 2 événements la probabilité conditionnelle est égale à

$$p(E/F) = \frac{p(E \cap F)}{p(F)}$$

E et F sont indépendants *si et seulement si* $p(E \cap F) = p(E)p(F)$ ce qui équivaut à

$$p(E/F) = p(E)$$

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Rappel : si E et F sont 2 événements la probabilité conditionnelle est égale à

$$p(E/F) = \frac{p(E \cap F)}{p(F)}$$

E et F sont indépendants *si et seulement si* $p(E \cap F) = p(E)p(F)$ ce qui équivaut à

$$p(E/F) = p(E)$$

FIGURE: Claude Shannon – 1916 - 2001

L'idée de Shannon est de quantifier cette information sachant que plus le contenu du message est rare plus l'information apportée est importante.

A contrario, si on est sûr de recevoir un certain message il n'apporte aucune information et la mesure de l'information apportée devra alors être nulle.

On voit alors qu'il y a un lien entre la probabilité de recevoir une information et la mesure que l'on veut en donner : ce lien que l'on cherche à établir doit respecter les idées ci-dessus. En bonus on veut que la quantité d'information apportée par *2 événements indépendants* soit la *somme* des quantités d'information apportées par chacun.

FIGURE: Claude Shannon – 1916 - 2001

L'idée de Shannon est de quantifier cette information sachant que plus le contenu du message est rare plus l'information apportée est importante.

A contrario, si on est sûr de recevoir un certain message il n'apporte aucune information et la mesure de l'information apportée devra alors être nulle.

On voit alors qu'il y a un lien entre la probabilité de recevoir une information et la mesure que l'on veut en donner : ce lien que l'on cherche à établir doit respecter les idées ci-dessus. En bonus on veut que la quantité d'information apportée par *2 événements indépendants* soit la *somme* des quantités d'information apportées par chacun.

FIGURE: Claude Shannon – 1916 - 2001

L'idée de Shannon est de quantifier cette information sachant que plus le contenu du message est rare plus l'information apportée est importante.

A contrario, si on est sûr de recevoir un certain message il n'apporte aucune information et la mesure de l'information apportée devra alors être nulle.

On voit alors qu'il y a un lien entre la probabilité de recevoir une information et la mesure que l'on veut en donner : ce lien que l'on cherche à établir doit respecter les idées ci-dessus.

En bonus on veut que la quantité d'information apportée par *2 événements indépendants* soit la *somme* des quantités d'information apportées par chacun.

FIGURE: Claude Shannon – 1916 - 2001

L'idée de Shannon est de quantifier cette information sachant que plus le contenu du message est rare plus l'information apportée est importante.

A contrario, si on est sûr de recevoir un certain message il n'apporte aucune information et la mesure de l'information apportée devra alors être nulle.

On voit alors qu'il y a un lien entre la probabilité de recevoir une information et la mesure que l'on veut en donner : ce lien que l'on cherche à établir doit respecter les idées ci-dessus. En bonus on veut que la quantité d'information apportée par *2 événements indépendants* soit la *somme* des quantités d'information apportées par chacun.

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Quantité d'information

Définition

L'unité de quantité d'information est le bit (binary unit) défini comme la quantité d'information apportée par le choix entre deux valeurs équiprobables.

REMARQUE : si on a une variable E qui prend deux valeurs équiprobables (par exemple pile ou face pour une pièce non truquée) alors la quantité d'information apportée par la réalisation de $\{E = \mathit{pile}\}$ est de 1 bit.

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Quantité d'information

Définition

Soit E un événement. On appelle quantité d'information de E la valeur

$$I(E) = -\log_2 p(E) = -\frac{\ln p(E)}{\ln 2}$$

où $p(E)$ est la probabilité de E.

REMARQUE : $I(\{E = pile\}) = -\log_2(1/2) = 1.$

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Quantité d'information

Définition

Soit E un événement. On appelle quantité d'information de E la valeur

$$I(E) = -\log_2 p(E) = -\frac{\ln p(E)}{\ln 2}$$

où $p(E)$ est la probabilité de E.

REMARQUE : $I(\{E = pile\}) = -\log_2(1/2) = 1.$

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

I vérifie bien les requis exprimés plus haut :

si $p(E)$ diminue, $I(E)$ augmente

si $p(E) = 1$ alors $I(E) = 0$

Propriété

Si E et F sont 2 événements indépendants alors $I(E \cap F) = I(E) + I(F)$. La quantité d'information apportée par 2 événements indépendants est la somme de leur quantités d'information respectives.

I.Exemples
II.Codes
III.Mesure de l'information
IV.Premier théorème de Shannon
V.Canaux

I vérifie bien les requis exprimés plus haut :

si $p(E)$ diminue, $I(E)$ augmente

si $p(E) = 1$ alors $I(E) = 0$

Propriété

Si E et F sont 2 événements indépendants alors $I(E \cap F) = I(E) + I(F)$. La quantité d'information apportée par 2 événements indépendants est la somme de leur quantités d'information respectives.

I.Exemples
II.Codes
III.Mesure de l'information
IV.Premier théorème de Shannon
V.Canaux

I vérifie bien les requis exprimés plus haut :

si $p(E)$ diminue, $I(E)$ augmente

si $p(E) = 1$ alors $I(E) = 0$

Propriété

Si E et F sont 2 événements indépendants alors $I(E \cap F) = I(E) + I(F)$. La quantité d'information apportée par 2 événements indépendants est la somme de leur quantités d'information respectives.

I.Exemples
II.Codes
III.Mesure de l'information
IV.Premier théorème de Shannon
V.Canaux

I vérifie bien les requis exprimés plus haut :

si $p(E)$ diminue, $I(E)$ augmente

si $p(E) = 1$ alors $I(E) = 0$

Propriété

Si E et F sont 2 événements indépendants alors $I(E \cap F) = I(E) + I(F)$. La quantité d'information apportée par 2 événements indépendants est la somme de leur quantités d'information respectives.

Exemple

Dans un jeu de 32 cartes, on effectue des tirages.
$E = \{$la carte tirée est un valet de cœur$\}$
$F = \{$la carte tirée est un cœur$\}$
$p(E) = 1/32$ et $I(E) = 5$,
$p(F) = 1/4$ et $I(F) = 2$.
E et F ne sont pas indépendants

$$p(E/F) = \frac{p(E \cap F)}{p(F)} = \frac{1/32}{1/4} = 1/8$$

Exemple

Dans un jeu de 32 cartes, on effectue des tirages.
$E = \{$la carte tirée est un valet de cœur$\}$
$F = \{$la carte tirée est un cœur$\}$
$p(E) = 1/32$ et $I(E) = 5$,
$p(F) = 1/4$ et $I(F) = 2$.
E et F ne sont pas indépendants

$$p(E/F) = \frac{p(E \cap F)}{p(F)} = \frac{1/32}{1/4} = 1/8$$

Exemple

Dans un jeu de 32 cartes, on effectue des tirages.
$E = \{$la carte tirée est un valet de cœur$\}$
$F = \{$la carte tirée est un cœur$\}$
$p(E) = 1/32$ et $I(E) = 5$,
$p(F) = 1/4$ et $I(F) = 2$.
E et F ne sont pas indépendants

$$p(E/F) = \frac{p(E \cap F)}{p(F)} = \frac{1/32}{1/4} = 1/8$$

Exemple

Dans un jeu de 32 cartes, on effectue des tirages.
$E = \{$la carte tirée est un valet de cœur$\}$
$F = \{$la carte tirée est un cœur$\}$
$p(E) = 1/32$ et $I(E) = 5$,
$p(F) = 1/4$ et $I(F) = 2$.
E et F ne sont pas indépendants

$$p(E/F) = \frac{p(E \cap F)}{p(F)} = \frac{1/32}{1/4} = 1/8$$

Exemple

Dans un jeu de 32 cartes, on effectue des tirages.
$E = \{$la carte tirée est un valet de cœur$\}$
$F = \{$la carte tirée est un cœur$\}$
$p(E) = 1/32$ et $I(E) = 5$,
$p(F) = 1/4$ et $I(F) = 2$.
E et F ne sont pas indépendants

$$p(E/F) = \frac{p(E \cap F)}{p(F)} = \frac{1/32}{1/4} = 1/8$$

Exemple

Dans un jeu de 32 cartes, on effectue des tirages.
$E = \{\text{la carte tirée est un valet de cœur}\}$
$F = \{\text{la carte tirée est un cœur}\}$
$p(E) = 1/32$ et $I(E) = 5$,
$p(F) = 1/4$ et $I(F) = 2$.
E et F ne sont pas indépendants

$$p(E/F) = \frac{p(E \cap F)}{p(F)} = \frac{1/32}{1/4} = 1/8$$

Information mutuelle

Définition

Soient E et F 2 événements. L'information apportée par F sur E est défini par

$$I(F \to E) = \log_2 \frac{p(E/F)}{p(E)}$$

REMARQUE : $I(F \to E)$ n'est pas toujours positif.

Information mutuelle

Propriété

Soient E et F 2 événements.

$$I(F \to E) = I(E \to F) = \log_2 \frac{p(E \cap F)}{p(E)p(F)}$$

On notera alors $I(F \to E) = I(E, F) = I(F, E)$ et on l'appellera information mutuelle entre E et F.

EXEMPLE : $E = \{\text{la carte tirée est un valet}\}$
$F = \{\text{la carte tirée est un cœur}\}$
$I(E, F) = \log_2 \frac{p(E \cap F)}{p(E)p(F)} = \log_2 \frac{1/32}{1/8.1/4} = 0$

EXEMPLE : $E = \{\text{la carte tirée est rouge}\}$
$F = \{\text{la carte tirée est un atout cœur}\}$
$I(E, F) = \log_2 \frac{p(E \cap F)}{p(E)p(F)} = \log_2 \frac{4/32}{1/2.4/32} = \log_2(2) = 1$

Information mutuelle

Propriété

Soient E et F 2 événements.

$$I(F \to E) = I(E \to F) = \log_2 \frac{p(E \cap F)}{p(E)p(F)}$$

On notera alors $I(F \to E) = I(E, F) = I(F, E)$ et on l'appellera information mutuelle entre E et F.

EXEMPLE : $E = \{\text{la carte tirée est un valet}\}$
$F = \{\text{la carte tirée est un cœur}\}$
$I(E, F) = \log_2 \frac{p(E \cap F)}{p(E)p(F)} = \log_2 \frac{1/32}{1/8.1/4} = 0$

EXEMPLE : $E = \{\text{la carte tirée est rouge}\}$
$F = \{\text{la carte tirée est un atout cœur}\}$
$I(E, F) = \log_2 \frac{p(E \cap F)}{p(E)p(F)} = \log_2 \frac{4/32}{1/2.4/32} = \log_2(2) = 1$

- si $I(E,F) > 0$ alors la réalisation d'un des 2 événements augmente la probabilité de l'autre (diminue son incertitude)
- si $I(E,F) = 0$ alors E et F sont indépendants, l'information mutuelle est nulle
- si $I(E,F) < 0$ alors la réalisation d'un des 2 événements diminue la probabilité de l'autre (augmente son incertitude)
- si $p(E \cap F) = 0$ alors la réalisation d'un des 2 événements rend impossible la réalisation de l'autre et $I(E,F) = -\infty$

- si $I(E,F) > 0$ alors la réalisation d'un des 2 événements augmente la probabilité de l'autre (diminue son incertitude)
- si $I(E,F) = 0$ alors E et F sont indépendants, l'information mutuelle est nulle
- si $I(E,F) < 0$ alors la réalisation d'un des 2 événements diminue la probabilité de l'autre (augmente son incertitude)
- si $p(E \cap F) = 0$ alors la réalisation d'un des 2 événements rend impossible la réalisation de l'autre et $I(E,F) = -\infty$

- si $I(E,F) > 0$ alors la réalisation d'un des 2 événements augmente la probabilité de l'autre (diminue son incertitude)
- si $I(E,F) = 0$ alors E et F sont indépendants, l'information mutuelle est nulle
- si $I(E,F) < 0$ alors la réalisation d'un des 2 événements diminue la probabilité de l'autre (augmente son incertitude)
- si $p(E \cap F) = 0$ alors la réalisation d'un des 2 événements rend impossible la réalisation de l'autre et $I(E,F) = -\infty$

- si $I(E,F) > 0$ alors la réalisation d'un des 2 événements augmente la probabilité de l'autre (diminue son incertitude)
- si $I(E,F) = 0$ alors E et F sont indépendants, l'information mutuelle est nulle
- si $I(E,F) < 0$ alors la réalisation d'un des 2 événements diminue la probabilité de l'autre (augmente son incertitude)
- si $p(E \cap F) = 0$ alors la réalisation d'un des 2 événements rend impossible la réalisation de l'autre et $I(E,F) = -\infty$

Propriété

$$I(E \cap F) = I(E) + I(F) - I(E, F)$$

preuve : $I(E \cap F) = -\log_2 p(E \cap F)$
$I(E, F) = \log_2 p(E \cap F) - \log_2 p(E) - \log_2 p(F) =$
$\log_2 p(E \cap F) + I(E) + I(F)$ donc
$-\log_2 p(E \cap F) = I(E) + I(F) - I(E, F)$ ■

EXEMPLE : $E = \{$la carte tirée est rouge$\}$
$F = \{$la carte tirée est un atout cœur$\}$
$I(E) = -\log_2(p(E)) = -\log_2(1/2) = 1$
$I(F) = -\log_2(p(F)) = -\log_2(4/32) = -\log_2(1/8) = 3$
$I(E, F) = 1$
$I(E \cap F) = -\log_2(p(E \cap F)) = -\log_2(4/32) = -\log_2(1/8) = 3$

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Propriété

$$I(E \cap F) = I(E) + I(F) - I(E, F)$$

preuve : $I(E \cap F) = -\log_2 p(E \cap F)$
$I(E, F) = \log_2 p(E \cap F) - \log_2 p(E) - \log_2 p(F) =$
$\log_2 p(E \cap F) + I(E) + I(F)$ donc
$-\log_2 p(E \cap F) = I(E) + I(F) - I(E, F)$ ■

EXEMPLE : $E = \{$la carte tirée est rouge$\}$
$F = \{$la carte tirée est un atout cœur$\}$
$I(E) = -\log_2(p(E)) = -\log_2(1/2) = 1$
$I(F) = -\log_2(p(F)) = -\log_2(4/32) = -\log_2(1/8) = 3$
$I(E, F) = 1$
$I(E \cap F) = -\log_2(p(E \cap F)) = -\log_2(4/32) = -\log_2(1/8) = 3$

Propriété

$$I(E \cap F) = I(E) + I(F) - I(E, F)$$

preuve : $I(E \cap F) = -\log_2 p(E \cap F)$
$I(E, F) = \log_2 p(E \cap F) - \log_2 p(E) - \log_2 p(F) =$
$\log_2 p(E \cap F) + I(E) + I(F)$ donc
$-\log_2 p(E \cap F) = I(E) + I(F) - I(E, F)$ ■

EXEMPLE : $E = \{$la carte tirée est rouge$\}$
$F = \{$la carte tirée est un atout cœur$\}$
$I(E) = -\log_2(p(E)) = -\log_2(1/2) = 1$
$I(F) = -\log_2(p(F)) = -\log_2(4/32) = -\log_2(1/8) = 3$
$I(E, F) = 1$
$I(E \cap F) = -\log_2(p(E \cap F)) = -\log_2(4/32) = -\log_2(1/8) = 3$

Propriété

$$I(E \cap F) = I(E) + I(F) - I(E, F)$$

preuve : $I(E \cap F) = -\log_2 p(E \cap F)$
$I(E, F) = \log_2 p(E \cap F) - \log_2 p(E) - \log_2 p(F) =$
$\log_2 p(E \cap F) + I(E) + I(F)$ donc
$-\log_2 p(E \cap F) = I(E) + I(F) - I(E, F)$ ■

EXEMPLE : $E = \{$la carte tirée est rouge$\}$
$F = \{$la carte tirée est un atout cœur$\}$
$I(E) = -\log_2(p(E)) = -\log_2(1/2) = 1$
$I(F) = -\log_2(p(F)) = -\log_2(4/32) = -\log_2(1/8) = 3$
$I(E, F) = 1$
$I(E \cap F) = -\log_2(p(E \cap F)) = -\log_2(4/32) = -\log_2(1/8) = 3$

Propriété

$$I(E \cap F) = I(E) + I(F) - I(E, F)$$

preuve : $I(E \cap F) = -\log_2 p(E \cap F)$
$I(E, F) = \log_2 p(E \cap F) - \log_2 p(E) - \log_2 p(F) =$
$\log_2 p(E \cap F) + I(E) + I(F)$ donc
$-\log_2 p(E \cap F) = I(E) + I(F) - I(E, F)$ ■

EXEMPLE : $E = \{$la carte tirée est rouge$\}$
$F = \{$la carte tirée est un atout cœur$\}$
$I(E) = -\log_2(p(E)) = -\log_2(1/2) = 1$
$I(F) = -\log_2(p(F)) = -\log_2(4/32) = -\log_2(1/8) = 3$
$I(E, F) = 1$
$I(E \cap F) = -\log_2(p(E \cap F)) = -\log_2(4/32) = -\log_2(1/8) = 3$

Propriété

$$I(E \cap F) = I(E) + I(F) - I(E, F)$$

preuve : $I(E \cap F) = -\log_2 p(E \cap F)$
$I(E, F) = \log_2 p(E \cap F) - \log_2 p(E) - \log_2 p(F) =$
$\log_2 p(E \cap F) + I(E) + I(F)$ donc
$-\log_2 p(E \cap F) = I(E) + I(F) - I(E, F)$ ■

EXEMPLE : $E = \{$la carte tirée est rouge$\}$
$F = \{$la carte tirée est un atout cœur$\}$
$I(E) = -\log_2(p(E)) = -\log_2(1/2) = 1$
$I(F) = -\log_2(p(F)) = -\log_2(4/32) = -\log_2(1/8) = 3$
$I(E, F) = 1$
$I(E \cap F) = -\log_2(p(E \cap F)) = -\log_2(4/32) = -\log_2(1/8) = 3$

Propriété

$$I(E \cap F) = I(E) + I(F) - I(E, F)$$

preuve : $I(E \cap F) = -\log_2 p(E \cap F)$
$I(E, F) = \log_2 p(E \cap F) - \log_2 p(E) - \log_2 p(F) =$
$\log_2 p(E \cap F) + I(E) + I(F)$ donc
$-\log_2 p(E \cap F) = I(E) + I(F) - I(E, F)$ ■

EXEMPLE : $E = \{$la carte tirée est rouge$\}$
$F = \{$la carte tirée est un atout cœur$\}$
$I(E) = -\log_2(p(E)) = -\log_2(1/2) = 1$
$I(F) = -\log_2(p(F)) = -\log_2(4/32) = -\log_2(1/8) = 3$
$I(E, F) = 1$
$I(E \cap F) = -\log_2(p(E \cap F)) = -\log_2(4/32) = -\log_2(1/8) = 3$

Entropie

EXEMPLE : On voudrait connaître comme contenu d'information la valeur du dé après un lancer.
Soit alors X la variable aléatoire à valeurs dans $\{1, 2, 3, 4, 5, 6\}$.
X peut prendre 6 valeurs et si le dé n'est pas truqué, les valeurs sont équiprobables. Donc à chaque valeur correspond une quantité d'information de 2,58 bits ($= -\log_2 \frac{1}{6}$).

Supposons maintenant que le dé soit truqué et que la valeur 6 sorte avec une probabilité 0,5 et que les autres valeurs soient équiprobables.
La quantité d'information pour chaque valeur n'est pas la même.
Pour avoir une vision globale on s'intéresse à déterminer l'information moyenne soit l'espérance de $I(X)$.

$$-\frac{1}{2}\log_2\frac{1}{2} - 5 \times (\frac{1}{10}\log_2\frac{1}{10}) = \frac{1}{2} + \frac{1}{2} \times \log_2 10 = 1{,}22 \text{ bits.}$$

Entropie

EXEMPLE : On voudrait connaître comme contenu d'information la valeur du dé après un lancer.
Soit alors X la variable aléatoire à valeurs dans $\{1, 2, 3, 4, 5, 6\}$.
X peut prendre 6 valeurs et si le dé n'est pas truqué, les valeurs sont équiprobables. Donc à chaque valeur correspond une quantité d'information de 2,58 bits ($= -\log_2 \frac{1}{6}$).

Supposons maintenant que le dé soit truqué et que la valeur 6 sorte avec une probabilité 0,5 et que les autres valeurs soient équiprobables.
La quantité d'information pour chaque valeur n'est pas la même.
Pour avoir une vision globale on s'intéresse à déterminer l'information moyenne soit l'espérance de $I(X)$.

$$-\frac{1}{2}\log_2\frac{1}{2} - 5 \times (\frac{1}{10}\log_2\frac{1}{10}) = \frac{1}{2} + \frac{1}{2} \times \log_2 10 = 1{,}22 \text{ bits.}$$

Entropie

EXEMPLE : On voudrait connaître comme contenu d'information la valeur du dé après un lancer.
Soit alors X la variable aléatoire à valeurs dans $\{1, 2, 3, 4, 5, 6\}$.
X peut prendre 6 valeurs et si le dé n'est pas truqué, les valeurs sont équiprobables. Donc à chaque valeur correspond une quantité d'information de 2,58 bits ($= -\log_2 \frac{1}{6}$).

Supposons maintenant que le dé soit truqué et que la valeur 6 sorte avec une probabilité 0,5 et que les autres valeurs soient équiprobables.
La quantité d'information pour chaque valeur n'est pas la même.
Pour avoir une vision globale on s'intéresse à déterminer l'information moyenne soit l'espérance de $I(X)$.

$$-\frac{1}{2}\log_2\frac{1}{2} - 5 \times (\frac{1}{10}\log_2\frac{1}{10}) = \frac{1}{2} + \frac{1}{2} \times \log_2 10 = 1{,}22 \text{ bits.}$$

Entropie

EXEMPLE : On voudrait connaître comme contenu d'information la valeur du dé après un lancer.
Soit alors X la variable aléatoire à valeurs dans $\{1, 2, 3, 4, 5, 6\}$.
X peut prendre 6 valeurs et si le dé n'est pas truqué, les valeurs sont équiprobables. Donc à chaque valeur correspond une quantité d'information de 2,58 bits ($= -\log_2 \frac{1}{6}$).

Supposons maintenant que le dé soit truqué et que la valeur 6 sorte avec une probabilité 0,5 et que les autres valeurs soient équiprobables.
La quantité d'information pour chaque valeur n'est pas la même.
Pour avoir une vision globale on s'intéresse à déterminer l'information moyenne soit l'espérance de $I(X)$.
$-\frac{1}{2}\log_2\frac{1}{2} - 5 \times (\frac{1}{10}\log_2\frac{1}{10}) = \frac{1}{2} + \frac{1}{2} \times \log_2 10 =$1,22 bits.

Entropie

Définition

On appelle entropie de X l'espérance de $I(X)$ notée $H(X)$.
$H(X) = \sum_x p(X = x) I(X = x) = -\sum_x p(X = x) \log_2 p(X = x)$

- $H(X)$ est un réel positif comme $I(X = x)$.
- $H(X)$ correspond au nombre moyen d'éléments binaires pour coder les différentes valeurs de X.
- $H(X)$ n'est fonction que de la loi de probabilité de X, pas des valeurs prises par X.

Entropie

Définition

On appelle entropie de X l'espérance de $I(X)$ notée $H(X)$.
$H(X) = \sum_x p(X = x) I(X = x) = -\sum_x p(X = x) \log_2 p(X = x)$

- $H(X)$ est un réel positif comme $I(X = x)$.
- $H(X)$ correspond au nombre moyen d'éléments binaires pour coder les différentes valeurs de X.
- $H(X)$ n'est fonction que de la loi de probabilité de X, pas des valeurs prises par X.

Entropie

Définition

On appelle entropie de X l'espérance de $I(X)$ notée $H(X)$.
$H(X) = \sum_x p(X = x) I(X = x) = -\sum_x p(X = x) \log_2 p(X = x)$

- $H(X)$ est un réel positif comme $I(X = x)$.
- $H(X)$ correspond au nombre moyen d'éléments binaires pour coder les différentes valeurs de X.
- $H(X)$ n'est fonction que de la loi de probabilité de X, pas des valeurs prises par X.

EXEMPLE : pour un jeu de 32 cartes, on définit la variable aléatoire X par $X = 0$ si la carte est rouge, $X = 1$ si la carte est un pique et $X = 2$ si la carte est un trèfle.

$$H(X) = -(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{4}\log_2\frac{1}{4} + \frac{1}{4}\log_2\frac{1}{4}) = \frac{1}{2} + 2\frac{1}{4} + 2\frac{1}{4} = 1,5 \ bit$$

EXEMPLE : pour un jeu de 32 cartes, on définit la variable aléatoire X par $X = 0$ si la carte est rouge, $X = 1$ si la carte est un pique et $X = 2$ si la carte est un trèfle.

$$H(X) = -(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{4}\log_2\frac{1}{4} + \frac{1}{4}\log_2\frac{1}{4}) = \frac{1}{2} + 2\frac{1}{4} + 2\frac{1}{4} = 1,5 \ bit$$

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Théorème

$H(X) \leq \log_2 n$ si X prend n valeurs.
$H(X) = \log_2 n$ *si et seulement si* X a une loi uniforme (c'est-à-dire $p(X = x) = 1/n$ pour tout x).

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

preuve :

$$\begin{aligned} H(X) - \log_2 n &= -\sum_x p(X=x)\log_2 p(X=x) \\ &\quad -\log_2 n(\sum_x p(X=x)) \\ &= -\sum_x p(X=x)(\log_2 p(X=x) + \log_2 n) \\ &= \sum_x p(X=x)\log_2 \frac{1}{np(X=x)} \end{aligned}$$

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

preuve :

$$\begin{aligned} H(X) - \log_2 n &= -\sum_x p(X=x)\log_2 p(X=x) \\ &\quad -\log_2 n(\sum_x p(X=x)) \\ &= -\sum_x p(X=x)(\log_2 p(X=x) + \log_2 n) \\ &= \sum_x p(X=x)\log_2 \frac{1}{np(X=x)} \end{aligned}$$

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

Or $\ln z \leq z - 1$ donc $\log_2 z \leq \dfrac{z-1}{\ln 2}$.

$$\begin{aligned} H(X) - \log_2 n &\leq \sum_x p(X = x)\left(\frac{1}{n \ln 2 p(X = x)} - \frac{1}{\ln 2}\right) \\ &\leq \sum_x \frac{1}{n \ln 2} - \frac{1}{\ln 2} \sum_x p(X = x) \\ &\leq \frac{1}{\ln 2} - \frac{1}{\ln 2} \\ &\leq 0 \end{aligned}$$

car X prend n valeurs et donc $\displaystyle\sum_x \frac{1}{n \ln 2} = n \frac{1}{n \ln 2} = \frac{1}{\ln 2}$.

Or $\ln z \leq z - 1$ donc $\log_2 z \leq \frac{z-1}{\ln 2}$.

$$\begin{aligned} H(X) - \log_2 n &\leq \sum_x p(X = x)\left(\frac{1}{n \ln 2 p(X = x)} - \frac{1}{\ln 2}\right) \\ &\leq \sum_x \frac{1}{n \ln 2} - \frac{1}{\ln 2} \sum_x p(X = x) \\ &\leq \frac{1}{\ln 2} - \frac{1}{\ln 2} \\ &\leq 0 \end{aligned}$$

car X prend n valeurs et donc $\sum_x \frac{1}{n \ln 2} = n \frac{1}{n \ln 2} = \frac{1}{\ln 2}$.

Or $\ln z \leq z - 1$ donc $\log_2 z \leq \frac{z-1}{\ln 2}$.

$$\begin{aligned} H(X) - \log_2 n &\leq \sum_x p(X = x)\left(\frac{1}{n \ln 2 p(X = x)} - \frac{1}{\ln 2}\right) \\ &\leq \sum_x \frac{1}{n \ln 2} - \frac{1}{\ln 2} \sum_x p(X = x) \\ &\leq \frac{1}{\ln 2} - \frac{1}{\ln 2} \\ &\leq 0 \end{aligned}$$

car X prend n valeurs et donc $\sum_x \frac{1}{n \ln 2} = n \frac{1}{n \ln 2} = \frac{1}{\ln 2}$.

Si $p(X = x) = \frac{1}{n}$ pour tout x alors

$$H(X) = -\sum_x \frac{1}{n} \log_2 \frac{1}{n} = \sum_x \frac{1}{n} \log_2 n = \log_2 n.$$ ■

Propriété

L'entropie augmente lorsque le nombre de valeurs possibles augmente.

preuve : Soit X une variable aléatoire prenant n valeurs $x_1, \ldots, x_n$ de probabilité respective $p_1, \ldots p_n$.
Supposons que la valeur x_n soit partagée en deux valeurs y_n, z_n de probabilité respective p'_n, p''_n telles que $p'_n + p''_n = p_n, p'_n \neq 0, p''_n \neq 0$.
On a alors une nouvelle variable aléatoire X' dont l'entropie vaut
$H(X') = H(X) + p_n \log p_n - p'_n \log p'_n - p''_n \log p''_n$ donc
$H(X') - H(X) = (p'_n + p'_n) \log p_n - p'_n \log p'_n - p''_n \log p''_n =$
$p'_n(\log p_n - \log p'_n) + p''_n(\log p_n - \log p''_n)$
or $\log p_n > \log p'_n$ et $\log p_n > \log p''_n$ donc $H(X') - H(X) > 0$. ■

La longueur moyenne d'un codage γ d'une source A est $m = \sum_{a \in A} p(a)|\gamma(a)|$.

Théorème

Soit A une source discrète sans mémoire d'entropie $H(A)$ codé en binaire par un code de longueur moyenne m. On a alors

$$H(A) \leq m$$

De plus pour toute source discrète sans mémoire A, il existe un code C codant A de longueur moyenne m tel que $H(A) \leq m < H(A) + 1$.

soit $\gamma : A \to C$ un codage de A.

$$\begin{aligned}
H(A) - m_C &= -\sum_{a \in A} p(a) \log_2 p(a) - \sum_{a \in a} p(a)|\gamma(a)| \\
&= -\sum_{a \in A} p(a)[\log_2 p(a) + |\gamma(a)|] \\
&= -\sum_{a \in A} p(a)[\log_2 p(a) + \log_2 2^{|\gamma(a)|}] \\
&= -\sum_{a \in A} p(a) \log_2 [p(a) 2^{|\gamma(a)|}] \\
&= \sum_{a \in A} p(a) \log_2 [\frac{1}{p(a) 2^{|\gamma(a)|}}]
\end{aligned}$$

or on sait que $\log_2 x \leq \frac{x-1}{\ln 2}$ donc on a

$$\begin{aligned} H(A) - m_C &\leq \frac{1}{\ln 2} \sum_{a \in A} p(a)[\frac{1}{p(a)2^{|\gamma(a)|}} - 1] \\ &\leq \frac{1}{\ln 2} \sum_{a \in A} [2^{-|\gamma(a)|} - p(a)] \\ &\leq \frac{1}{\ln 2} (\sum_{a \in A} 2^{-|\gamma(a)|} - \sum_{a \in A} p(a)) \\ &\leq \frac{1}{\ln 2} (\sum_{a \in A} 2^{-|\gamma(a)|} - 1) \\ &\leq 0 \end{aligned}$$

d'après l'inégalité de Kraft et le théorème de Mac Millan.

Pour trouver un code qui vérifie $H(A) \leq m < H(A) + 1$ il suffit de choisir dans un arbre binaire un codage de la lettre a sur le niveau n_a tel que $-\log_2 p(a) \leq n_a < -\log_2 p(a) + 1$, soit le premier entier supérieur ou égal à $-\log_2 p(a)$.

EXEMPLE : $A = \{a, b, c, d\}$ $p(a) = p(b) = 1/4$, $p(c) = 3/8$ et $p(d) = 1/8$.

$H(A) = -(1/4 \log_2 1/4 + 1/4 \log_2 1/4 + 3/8 \log_2 3/8 + 1/8 \log_2 1/8) = 1/2 + 1/2 + 0,5325 + 0,375 = 1,9075$

On calcule $n_a = n_b = 2$, $n_c = 2$ et $n_d = 3$.

On obtient par exemple le codage C suivant

a	00
b	01
c	10
d	110

$m_C = 2 \times \frac{1}{4} + 2 \times \frac{1}{4} + 2 \times \frac{3}{8} + 3 \times \frac{1}{8} = 2,125$

Pour trouver un code qui vérifie $H(A) \leq m < H(A) + 1$ il suffit de choisir dans un arbre binaire un codage de la lettre a sur le niveau n_a tel que $-\log_2 p(a) \leq n_a < -\log_2 p(a) + 1$, soit le premier entier supérieur ou égal à $-\log_2 p(a)$.

EXEMPLE : $A = \{a, b, c, d\}$ $p(a) = p(b) = 1/4$, $p(c) = 3/8$ et $p(d) = 1/8$.

$H(A) = -(1/4 \log_2 1/4 + 1/4 \log_2 1/4 + 3/8 \log_2 3/8 + 1/8 \log_2 1/8) = 1/2 + 1/2 + 0,5325 + 0,375 = 1,9075$

On calcule $n_a = n_b = 2$, $n_c = 2$ et $n_d = 3$.

On obtient par exemple le codage C suivant

a	00
b	01
c	10
d	110

$m_C = 2 \times \frac{1}{4} + 2 \times \frac{1}{4} + 2 \times \frac{3}{8} + 3 \times \frac{1}{8} = 2,125$

Pour trouver un code qui vérifie $H(A) \leq m < H(A) + 1$ il suffit de choisir dans un arbre binaire un codage de la lettre a sur le niveau n_a tel que $-\log_2 p(a) \leq n_a < -\log_2 p(a) + 1$, soit le premier entier supérieur ou égal à $-\log_2 p(a)$.

EXEMPLE : $A = \{a, b, c, d\}$ $p(a) = p(b) = 1/4$, $p(c) = 3/8$ et $p(d) = 1/8$.

$H(A) = -(1/4 \log_2 1/4 + 1/4 \log_2 1/4 + 3/8 \log_2 3/8 + 1/8 \log_2 1/8) = 1/2 + 1/2 + 0,5325 + 0,375 = 1,9075$

On calcule $n_a = n_b = 2$, $n_c = 2$ et $n_d = 3$.

On obtient par exemple le codage C suivant

a	00
b	01
c	10
d	110

$m_C = 2 \times \frac{1}{4} + 2 \times \frac{1}{4} + 2 \times \frac{3}{8} + 3 \times \frac{1}{8} = 2,125$

Pour trouver un code qui vérifie $H(A) \leq m < H(A) + 1$ il suffit de choisir dans un arbre binaire un codage de la lettre a sur le niveau n_a tel que $-\log_2 p(a) \leq n_a < -\log_2 p(a) + 1$, soit le premier entier supérieur ou égal à $-\log_2 p(a)$.

EXEMPLE : $A = \{a, b, c, d\}$ $p(a) = p(b) = 1/4$, $p(c) = 3/8$ et $p(d) = 1/8$.

$H(A) = -(1/4 \log_2 1/4 + 1/4 \log_2 1/4 + 3/8 \log_2 3/8 + 1/8 \log_2 1/8) = 1/2 + 1/2 + 0,5325 + 0,375 = 1,9075$

On calcule $n_a = n_b = 2$, $n_c = 2$ et $n_d = 3$.

On obtient par exemple le codage C suivant

a	00
b	01
c	10
d	110

$m_C = 2 \times \frac{1}{4} + 2 \times \frac{1}{4} + 2 \times \frac{3}{8} + 3 \times \frac{1}{8} = 2,125$

L'efficacité d'un codage sera alors mesurée par $E = \frac{H(A)}{m} \leq 1$

Théorème

pour toute source discrète sans mémoire il existe un codage dont l'efficacité est arbitrairement proche de 1.

l'idée est de coder A^k c'est-à-dire l'alphabet formé de tous les mots composés de lettres de A de longueur k sachant que l'on peut prouver $H(A^k) = kH(A)$.

On a alors pour le code C_k, $H(A) \leq \frac{m_{C_k}}{k} \leq H(A) + \frac{1}{k}$.

Le nombre $m = \frac{m_{C_k}}{k}$ représente alors le nombre moyen de symboles binaires pour le codage d'une lettre puisque m_{C_k} représente le nombre moyen de symboles binaires pour le codage de k lettres. Et on a $1 - \frac{1}{k} \leq \frac{H(A)}{m} \leq 1$.

L'efficacité d'un codage sera alors mesurée par $E = \frac{H(A)}{m} \leq 1$

Théorème

pour toute source discrète sans mémoire il existe un codage dont l'efficacité est arbitrairement proche de 1.

l'idée est de coder A^k c'est-à-dire l'alphabet formé de tous les mots composés de lettres de A de longueur k sachant que l'on peut prouver $H(A^k) = kH(A)$.

On a alors pour le code C_k, $H(A) \leq \frac{m_{C_k}}{k} \leq H(A) + \frac{1}{k}$.

Le nombre $m = \frac{m_{C_k}}{k}$ représente alors le nombre moyen de symboles binaires pour le codage d'une lettre puisque m_{C_k} représente le nombre moyen de symboles binaires pour le codage de k lettres. Et on a $1 - \frac{1}{k} \leq \frac{H(A)}{m} \leq 1$.

code binaire à longueur fixe

Soit une source A de cardinal n.
Pour coder A sur $\{0,1\}$ avec un nombre r fixe de symboles binaires il faut avoir 2^r supérieur au cardinal de A.
Donc on doit avoir $r \geq \log_2 n$.

Pour optimiser son efficacité on choisit alors r tel $\log_2 n \leq r \leq \log_2 n + 1$ et l'efficacité d'un tel code est alors inférieure à $\dfrac{H(A)}{\log_2 n}$.

On aura égalité si $n = 2^r$ et si la loi sur A est uniforme.

I.Exemples

II.Codes

III.Mesure de l'information

IV.Premier théorème de Shannon

V.Canaux

code binaire à longueur fixe

Soit une source A de cardinal n.
Pour coder A sur $\{0,1\}$ avec un nombre r fixe de symboles binaires il faut avoir 2^r supérieur au cardinal de A.
Donc on doit avoir $r \geq \log_2 n$.

Pour optimiser son efficacité on choisit alors r tel $\log_2 n \leq r \leq \log_2 n + 1$ et l'efficacité d'un tel code est alors inférieure à $\dfrac{H(A)}{\log_2 n}$.

On aura égalité si $n = 2^r$ et si la loi sur A est uniforme.

code binaire à longueur fixe

Soit une source A de cardinal n.
Pour coder A sur $\{0,1\}$ avec un nombre r fixe de symboles binaires il faut avoir 2^r supérieur au cardinal de A.
Donc on doit avoir $r \geq \log_2 n$.

Pour optimiser son efficacité on choisit alors r tel $\log_2 n \leq r \leq \log_2 n + 1$ et l'efficacité d'un tel code est alors inférieure à $\dfrac{H(A)}{\log_2 n}$.

On aura égalité si $n = 2^r$ et si la loi sur A est uniforme.

code binaire à longueur fixe

Soit une source A de cardinal n.
Pour coder A sur $\{0,1\}$ avec un nombre r fixe de symboles binaires il faut avoir 2^r supérieur au cardinal de A.
Donc on doit avoir $r \geq \log_2 n$.

Pour optimiser son efficacité on choisit alors r tel $\log_2 n \leq r \leq \log_2 n + 1$ et l'efficacité d'un tel code est alors inférieure à $\dfrac{H(A)}{\log_2 n}$.

On aura égalité si $n = 2^r$ et si la loi sur A est uniforme.

EXEMPLE : $A = \{a, b, c, d, e\}$ avec une loi uniforme. Il faut $[\log_2 5] + 1 = 3$ bits pour coder A : par exemple

a	010
b	011
c	000
d	100
e	101

L'efficacité est alors égale à $E = \frac{\log_2 5}{3} \approx 0,77$.

On peut améliorer cette efficacité si on choisit comme source A^2 avec toujours une loi uniforme : on a alors 25 lettres et donc $r > 2\log_2 5$, soit $r = 5$.
L'efficacité devient alors $E = \frac{2\log_2 5}{5} \approx 0,93$.

code de Shannon Fano

Cet algorithme construit un code d'une source discrète sans mémoire A en suivant les étapes suivantes :

1. on classe les lettres de A par ordre décroissant de probabilité
2. on forme 2 classes dans A de façon à ce que les probabilités de chaque classe soient les plus proches possibles puis on attribue 1 à la classe du haut et 0 à celle du bas
3. **si** chaque classe a 1 élément **alors** on arrête **sinon** on applique l'étape 2 à chaque ayant plus d'un élément

code de Shannon Fano

Cet algorithme construit un code d'une source discrète sans mémoire A en suivant les étapes suivantes :

1. on classe les lettres de A par ordre décroissant de probabilité
2. on forme 2 classes dans A de façon à ce que les probabilités de chaque classe soient les plus proches possibles puis on attribue 1 à la classe du haut et 0 à celle du bas
3. **si** chaque classe a 1 élément **alors** on arrête **sinon** on applique l'étape 2 à chaque ayant plus d'un élément

code de Shannon Fano

Cet algorithme construit un code d'une source discrète sans mémoire A en suivant les étapes suivantes :

1. on classe les lettres de A par ordre décroissant de probabilité
2. on forme 2 classes dans A de façon à ce que les probabilités de chaque classe soient les plus proches possibles puis on attribue 1 à la classe du haut et 0 à celle du bas
3. **si** chaque classe a 1 élément **alors** on arrête **sinon** on applique l'étape 2 à chaque ayant plus d'un élément

code de Shannon Fano : exemple

$A = \{a, \ldots, g\}$ avec une loi
$\{0,4; 0,2; 0,15; 0,1; 0,05; 0,05; 0,05\}$.

code de Shannon Fano : exemple

$A = \{a, \ldots, g\}$ avec une loi
$\{0,4; 0,2; 0,15; 0,1; 0,05; 0,05; 0,05\}$.

a	0,4					
b	0,2					
c	0,15					
d	0,1					
e	0,05					
f	0,05					
g	0,05					

code de Shannon Fano : exemple

$A = \{a, \ldots, g\}$ avec une loi
$\{0,4; 0,2; 0,15; 0,1; 0,05; 0,05; 0,05\}$.

a	0,4	1				
b	0,2	1				
c	0,15	0				
d	0,1	0				
e	0,05	0				
f	0,05	0				
g	0,05	0				

code de Shannon Fano : exemple

$A = \{a, \ldots, g\}$ avec une loi
$\{0,4; 0,2; 0,15; 0,1; 0,05; 0,05; 0,05\}$.

a	0,4	1	1			
b	0,2	1	0			
c	0,15	0	1			
d	0,1	0	1			
e	0,05	0	0			
f	0,05	0	0			
g	0,05	0	0			

code de Shannon Fano : exemple

$A = \{a, \ldots, g\}$ avec une loi $\{0,4; 0,2; 0,15; 0,1; 0,05; 0,05; 0,05\}$.

a	0,4	1	1			
b	0,2	1	0			
c	0,15	0	1	1		
d	0,1	0	1	0		
e	0,05	0	0	1		
f	0,05	0	0	1		
g	0,05	0	0	0		

code de Shannon Fano : exemple

$A = \{a, \ldots, g\}$ avec une loi
$\{0,4; 0,2; 0,15; 0,1; 0,05; 0,05; 0,05\}$.

a	0,4	1	1			
b	0,2	1	0			
c	0,15	0	1	1		
d	0,1	0	1	0		
e	0,05	0	0	1	1	
f	0,05	0	0	1	0	
g	0,05	0	0	0		

code de Shannon Fano : exemple

$A = \{a, \ldots, g\}$ avec une loi
$\{0,4; 0,2; 0,15; 0,1; 0,05; 0,05; 0,05\}$.

a	0,4	1	1			
b	0,2	1	0			
c	0,15	0	1	1		
d	0,1	0	1	0		
e	0,05	0	0	1	1	
f	0,05	0	0	1	0	
g	0,05	0	0	0		

code de Shannon Fano : exemple

$A = \{a, \ldots, g\}$ avec une loi $\{0,4; 0,2; 0,15; 0,1; 0,05; 0,05; 0,05\}$.

a	0,4	1	1			11
b	0,2	1	0			10
c	0,15	0	1	1		011
d	0,1	0	1	0		010
e	0,05	0	0	1	1	0011
f	0,05	0	0	1	0	0010
g	0,05	0	0	0		000

a	0,4	1	1			11
b	0,2	1	0			10
c	0,15	0	1	1		011
d	0,1	0	1	0		010
e	0,05	0	0	1	1	0011
f	0,05	0	0	1	0	0010
g	0,05	0	0	0		000

Le codage est alors le suivant

a	b	c	d	e	f	g
11	10	001	011	0011	0010	000

$$H(A) = -(0,4\log_2(0,4) + 0,2\log_2(0,2) + 0,15\log_2(0,15) + 0,1\log_2(0,1) + 0,05\log_2(0,05) + 0,05\log_2(0,05) + 0,05\log_2(0,05)) = 2,38$$

$$m_C = 0,4 \times 2 + 0,2 \times 2 + 0,15 \times 3 + 0,1 \times 3 + 0,05 \times 4 + 0,05 \times 4 + 0,05 \times 3 = 2,5$$

a	0,4	1	1			11
b	0,2	1	0			10
c	0,15	0	1	1		011
d	0,1	0	1	0		010
e	0,05	0	0	1	1	0011
f	0,05	0	0	1	0	0010
g	0,05	0	0	0		000

Le codage est alors le suivant

a	b	c	d	e	f	g
11	10	001	011	0011	0010	000

$$H(A) = -(0,4\log_2(0,4) + 0,2\log_2(0,2) + 0,15\log_2(0,15) + 0,1\log_2(0,1) + 0,05\log_2(0,05) + 0,05\log_2(0,05) + 0,05\log_2(0,05)) = 2,38$$

$$m_C = 0,4 \times 2 + 0,2 \times 2 + 0,15 \times 3 + 0,1 \times 3 + 0,05 \times 4 + 0,05 \times 4 + 0,05 \times 3 = 2,5$$

a	0,4	1	1			11
b	0,2	1	0			10
c	0,15	0	1	1		011
d	0,1	0	1	0		010
e	0,05	0	0	1	1	0011
f	0,05	0	0	1	0	0010
g	0,05	0	0	0		000

Le codage est alors le suivant

a	b	c	d	e	f	g
11	10	001	011	0011	0010	000

$$H(A) = -(0,4\log_2(0,4) + 0,2\log_2(0,2) + 0,15\log_2(0,15) + 0,1\log_2(0,1) + 0,05\log_2(0,05) + 0,05\log_2(0,05) + 0,05\log_2(0,05)) = 2,38$$

$$m_C = 0,4 \times 2 + 0,2 \times 2 + 0,15 \times 3 + 0,1 \times 3 + 0,05 \times 4 + 0,05 \times 4 + 0,05 \times 3 = 2,5$$

code de Huffman

FIGURE: David A. Huffman – 1925-1999

Cet algorithme construit un code d'une source discrète sans mémoire A en construisant un arbre

code de Huffman

1. on construit un noeud par lettre de A, affecté par la probabilité de cette lettre
2. on classe les noeuds sans parent par ordre croissant de probabilité
3. on relie 2 noeuds de plus faible probabilité par 2 arêtes à un noeud parent qui remplace ces 2 noeuds et qui est affecté de la somme des probabilités
4. on recommence l'étape 2 jusqu'à arriver à la racine de l'arbre

code de Huffman

1. on construit un noeud par lettre de A, affecté par la probabilité de cette lettre
2. on classe les noeuds sans parent par ordre croissant de probabilité
3. on relie 2 noeuds de plus faible probabilité par 2 arêtes à un noeud parent qui remplace ces 2 noeuds et qui est affecté de la somme des probabilités
4. on recommence l'étape 2 jusqu'à arriver à la racine de l'arbre

code de Huffman

1. on construit un noeud par lettre de A, affecté par la probabilité de cette lettre
2. on classe les noeuds sans parent par ordre croissant de probabilité
3. on relie 2 noeuds de plus faible probabilité par 2 arêtes à un noeud parent qui remplace ces 2 noeuds et qui est affecté de la somme des probabilités
4. on recommence l'étape 2 jusqu'à arriver à la racine de l'arbre

code de Huffman

1. on construit un noeud par lettre de A, affecté par la probabilité de cette lettre
2. on classe les noeuds sans parent par ordre croissant de probabilité
3. on relie 2 noeuds de plus faible probabilité par 2 arêtes à un noeud parent qui remplace ces 2 noeuds et qui est affecté de la somme des probabilités
4. on recommence l'étape 2 jusqu'à arriver à la racine de l'arbre

code de Huffman : exemple

g	f	e	d	c	b	a
0,05	0,05	0,05	0,1	0,15	0,2	0,4

code de Huffman : exemple

g	f	e	d	c	b	a
0,05	0,05	0,05	0,1	0,15	0,2	0,4
0,1		0,05	0,1	0,15	0,2	0,4

code de Huffman : exemple

e	g	f	d	c	b	a
0,05	0,05	0,05	0,1	0,15	0,2	0,4
0,05	0,1		0,1	0,15	0,2	0,4

code de Huffman : exemple

e	g	f	d	c	b	a
0,05	0,05	0,05	0,1	0,15	0,2	0,4
0,05	0,1		0,1	0,15	0,2	0,4
0,15			0,1	0,15	0,2	0,4

code de Huffman : exemple

d	e	g	f	c	b	a
0,1	0,05	0,05	0,05	0,15	0,2	0,4
0,1	0,05	0,1		0,15	0,2	0,4
0,1	0,15			0,15	0,2	0,4

code de Huffman : exemple

<table>
<tr><td>d</td><td>e</td><td>g</td><td>f</td><td>c</td><td>b</td><td>a</td></tr>
<tr><td>0,1</td><td>0,05</td><td>0,05</td><td>0,05</td><td>0,15</td><td>0,2</td><td>0,4</td></tr>
<tr><td>0,1</td><td>0,05</td><td colspan="2">0,1</td><td>0,15</td><td>0,2</td><td>0,4</td></tr>
<tr><td>0,1</td><td colspan="3">0,15</td><td>0,15</td><td>0,2</td><td>0,4</td></tr>
<tr><td colspan="4">0,25</td><td>0,15</td><td>0,2</td><td>0,4</td></tr>
</table>

code de Huffman : exemple

c	b	d	e	g	f	a
0,15	0,2	0,1	0,05	0,05	0,05	0,4
0,15	0,2	0,1	0,05	0,1		0,4
0,15	0,2	0,1	0,15			0,4
0,15	0,2	0,25				0,4

code de Huffman : exemple

<table>
<tr><td>c</td><td>b</td><td>d</td><td>e</td><td>g</td><td>f</td><td>a</td></tr>
<tr><td>0,15</td><td>0,2</td><td>0,1</td><td>0,05</td><td>0,05</td><td>0,05</td><td>0,4</td></tr>
<tr><td>0,15</td><td>0,2</td><td>0,1</td><td>0,05</td><td colspan="2">0,1</td><td>0,4</td></tr>
<tr><td>0,15</td><td>0,2</td><td>0,1</td><td colspan="3">0,15</td><td>0,4</td></tr>
<tr><td>0,15</td><td>0,2</td><td colspan="4">0,25</td><td>0,4</td></tr>
<tr><td colspan="2">0,35</td><td colspan="4">0,25</td><td>0,4</td></tr>
</table>

code de Huffman : exemple

d	e	g	f	c	b	a
0,1	0,05	0,05	0,05	0,15	0,2	0,4
0,1	0,05	0,1		0,15	0,2	0,4
0,1	0,15			0,15	0,2	0,4
0,25				0,15	0,2	0,4
0,25				0,35		0,4

code de Huffman : exemple

<table>
<tr><td>d</td><td>e</td><td>g</td><td>f</td><td>c</td><td>b</td><td>a</td></tr>
<tr><td>0,1</td><td>0,05</td><td>0,05</td><td>0,05</td><td>0,15</td><td>0,2</td><td>0,4</td></tr>
<tr><td>0,1</td><td>0,05</td><td colspan="2">0,1</td><td>0,15</td><td>0,2</td><td>0,4</td></tr>
<tr><td>0,1</td><td colspan="3">0,15</td><td>0,15</td><td>0,2</td><td>0,4</td></tr>
<tr><td colspan="4">0,25</td><td>0,15</td><td>0,2</td><td>0,4</td></tr>
<tr><td colspan="4">0,25</td><td colspan="2">0,35</td><td>0,4</td></tr>
<tr><td colspan="6">0,6</td><td>0,4</td></tr>
</table>

code de Huffman : exemple

a	d	e	g	f	c	b
0,4	0,1	0,05	0,05	0,05	0,15	0,2
0,4	0,1	0,05	0,1		0,15	0,2
0,4	0,1	0,15			0,15	0,2
0,4	0,25				0,15	0,2
0,4	0,25				0,35	
0,4	0,6					

code de Huffman : exemple

<table>
<tr><td>a</td><td>d</td><td>e</td><td>g</td><td>f</td><td>c</td><td>b</td></tr>
<tr><td>0,4</td><td>0,1</td><td>0,05</td><td>0,05</td><td>0,05</td><td>0,15</td><td>0,2</td></tr>
<tr><td>0,4</td><td>0,1</td><td>0,05</td><td colspan="2">0,1</td><td>0,15</td><td>0,2</td></tr>
<tr><td>0,4</td><td>0,1</td><td colspan="3">0,15</td><td>0,15</td><td>0,2</td></tr>
<tr><td>0,4</td><td colspan="4">0,25</td><td>0,15</td><td>0,2</td></tr>
<tr><td>0,4</td><td colspan="4">0,25</td><td colspan="2">0,35</td></tr>
<tr><td>0,4</td><td colspan="6">0,6</td></tr>
<tr><td colspan="7">1</td></tr>
</table>

<table>
<tr><td>a</td><td>d</td><td>e</td><td>g</td><td>f</td><td>c</td><td>b</td></tr>
<tr><td>0,4</td><td>0,1</td><td>0,05</td><td>0,05</td><td>0,05</td><td>0,15</td><td>0,2</td></tr>
<tr><td>0,4</td><td>0,1</td><td>0,05</td><td colspan="2">0,1</td><td>0,15</td><td>0,2</td></tr>
<tr><td>0,4</td><td>0,1</td><td colspan="3">0,15</td><td>0,15</td><td>0,2</td></tr>
<tr><td>0,4</td><td colspan="4">0,25</td><td>0,15</td><td>0,2</td></tr>
<tr><td>0,4</td><td colspan="4">0,25</td><td colspan="2">0,35</td></tr>
<tr><td>0,4</td><td colspan="6">0,6</td></tr>
<tr><td colspan="7">1</td></tr>
</table>

Le codage est alors le suivant

g	f	e	d	c	b	a
10110	10111	1010	100	110	111	0

$$H(A) = -(0,4\log_2(0,4) + 0,2\log_2(0,2) + 0,15\log_2(0,15) + 0,1\log_2(0,1) + 0,05\log_2(0,05) + 0,05\log_2(0,05) + 0,05\log_2(0,05)) = 2,38$$

$$m_C = 0,05 \times 5 + 0,05 \times 5 + 0,05 \times 4 + 0,1 \times 3 + 0,15 \times 3 + 0,2 \times 3 + 0,4 \times 1 = 2,45$$

a	d	e	g	f	c	b
0,4	0,1	0,05	0,05	0,05	0,15	0,2
0,4	0,1	0,05	0,1		0,15	0,2
0,4	0,1	0,15			0,15	0,2
0,4	0,25				0,15	0,2
0,4	0,25				0,35	
0,4	0,6					
1						

Le codage est alors le suivant

g	f	e	d	c	b	a
10110	10111	1010	100	110	111	0

$$H(A) = -(0,4\log_2(0,4) + 0,2\log_2(0,2) + 0,15\log_2(0,15) + 0,1\log_2(0,1) + 0,05\log_2(0,05) + 0,05\log_2(0,05) + 0,05\log_2(0,05)) = 2,38$$

$$m_C = 0,05 \times 5 + 0,05 \times 5 + 0,05 \times 4 + 0,1 \times 3 + 0,15 \times 3 + 0,2 \times 3 + 0,4 \times 1 = 2,45$$

a	d	e	g	f	c	b
0,4	0,1	0,05	0,05	0,05	0,15	0,2
0,4	0,1	0,05	0,1		0,15	0,2
0,4	0,1	0,15			0,15	0,2
0,4	0,25				0,15	0,2
0,4	0,25				0,35	
0,4	0,6					
1						

Le codage est alors le suivant

g	f	e	d	c	b	a
10110	10111	1010	100	110	111	0

$$H(A) = -(0,4\log_2(0,4) + 0,2\log_2(0,2) + 0,15\log_2(0,15) + 0,1\log_2(0,1) + 0,05\log_2(0,05) + 0,05\log_2(0,05) + 0,05\log_2(0,05)) = 2,38$$

$$m_C = 0,05 \times 5 + 0,05 \times 5 + 0,05 \times 4 + 0,1 \times 3 + 0,15 \times 3 + 0,2 \times 3 + 0,4 \times 1 = 2,45$$

De façon générale pour définir un canal de transmission on a besoin des entrées, des sorties et du bruit qui peut perturber les transmissions du canal.
Pour les canaux discrets, les entrées et les sorties seront modélisées par des alphabets finis A et B respectivement.
Quant au bruit il sera modélisé par une probabilité conditionnelle de B sachant A. En effet, la sortie doit idéalement être déterminée par l'entrée mais plus il y a de perturbation moins la sortie dépend de l'entrée.
On dira qu'un canal discret est sans mémoire si le bruit est indépendant du temps.
On définit alors un canal discret sans mémoire par la donnée de

- A un alphabet d'entrée
- B un alphabet de sortie
- M une matrice telle que $M_{i,j} = p(b_j/a_i)$

On se restreindra au canal binaire avec $A = B = \{0, 1\}$

Pourquoi doit-on coder avant le canal

EXEMPLE : $A = B = \{0, 1\}$ avec $p(A = 1) = q$ et $p(A = 0) = 1 - q$ et un canal binaire symétrique avec

$$M = \begin{pmatrix} 1-p & p \\ p & 1-p \end{pmatrix}$$

($p = p(B = 0/A = 1) = p(B = 1/A = 0) =$ probabilité d'erreur)

On va comparer une transmission sans codage préalable avant passage par le canal avec une transmission précédée d'un code à répétition.

Transmission sans codage

La probabilité d'obtenir une erreur est alors

$$\begin{aligned} P_e &= p(B=0, A=1) + p(B=1, A=0) \\ &= p(A=1)p(B=0/A=1) + p(A=0)p(B=1/A=0) \\ &= (1-q)p + qp \\ &= p \end{aligned}$$

Transmission avec codage à répétition

Codage : $0 \rightsquigarrow 000$ et $1 \rightsquigarrow 111$
(chaque bit est répété 2 fois)

Décodage :

000	001	010	011	100	101	110	111
0	0	0	1	0	1	1	1

(Le bit majoritaire l'emporte)

Transmission avec codage à répétition

Codage : $0 \rightsquigarrow 000$ et $1 \rightsquigarrow 111$

(chaque bit est répété 2 fois)

Décodage :

000	001	010	011	100	101	110	111
0	0	0	1	0	1	1	1

(Le bit majoritaire l'emporte)

On note A' l'ensemble des mots de 3 bits en entrée et B' l'ensemble des mots de 3 bits en sortie
La probabilité d'obtenir une erreur est alors

$$\begin{aligned} P_e = \; & p(B'=000, A'=111) + p(B'=100, A'=111) \\ & + p(B'=010, A'=111) + p(B'=001, A'=111) \\ & + p(B'=111, A'=000) + p(B'=011, A'=000) \\ & + p(B'=101, A'=000) + p(B'=110, A'=000) \end{aligned}$$

$$
\begin{aligned}
P_e &= p(B' = 000/A' = 111)p(A' = 111) \\
&\quad + p(B' = 100/A' = 111)p(A' = 111) \\
&\quad + p(B' = 010/A' = 111)p(A' = 111) \\
&\quad + p(B' = 001/A' = 111)p(A' = 111) \\
&\quad + p(B' = 111/A' = 000)p(A' = 000) \\
&\quad + p(B' = 011/A' = 000)p(A' = 000) \\
&\quad + p(B' = 101/A' = 000)p(A' = 000) \\
&\quad + p(B' = 110/A' = 000)p(A' = 000) \\
&= q(p^3 + 3p^2(1-p)) + (1-q)((p^3 + 3p^2(1-p))) \\
&= -2p^3 + 3p^2
\end{aligned}
$$

Comparaison

sans codage probabilité d'erreur $= p$
avec codage probabilité d'erreur $= -2p^3 + 3p^2$

- si $p < 0,5$ alors $-2p^3 + 3p^2 < p$ donc on obtient une probabilité d'erreur plus faible avec le codage
- si $p = 0,5$ alors les 2 transmissions sont équivalentes
- si $p > 0,5$ alors la transmission sans codage est alors préférable

Mais on peut remarquer que dans le dernier cas il suffit de permuter les lettres de l'alphabet de sortie ce qui changera p en $1 - p$.

avantages

- détection d'erreur : si on reçoit un mot qui contient à la fois 0 et 1 alors on est sûr qu'il y a eu une erreur de transmission puisqu'on ne devrait recevoir que 000 ou 111. On détectera 1 ou 2 erreurs mais pas 3 car alors on recevrait un mot possiblement envoyé (000 au lieu de 111 par exemple)
- correction d'erreur : dans le cas où une et une seule erreur est commise alors on pourra corriger cette erreur en examinant les 2 bits majoritaires. Mais ce n'est pas possible si 2 erreurs ont été commises

On dira que le code à répétition est 1-correcteur et 2-détecteur.

avantages

- détection d'erreur : si on reçoit un mot qui contient à la fois 0 et 1 alors on est sûr qu'il y a eu une erreur de transmission puisqu'on ne devrait recevoir que 000 ou 111. On détectera 1 ou 2 erreurs mais pas 3 car alors on recevrait un mot possiblement envoyé (000 au lieu de 111 par exemple)
- correction d'erreur : dans le cas où une et une seule erreur est commise alors on pourra corriger cette erreur en examinant les 2 bits majoritaires. Mais ce n'est pas possible si 2 erreurs ont été commises

On dira que le code à répétition est 1-correcteur et 2-détecteur.

Codages détecteur

Le plus simple des codes détecteur utilise un bit de parité.
On considère que l'on transmet des blocs de $p-1$ bits et on les code en rajoutant à la fin un p-ième bit choisi de façon que le nombre total de bits égaux à 1 dans le mot de p bits transmis soit pair.
A la reception, si le nombre de bits dans un mot de p bits est impair on aura alors détecter au moins une erreur.

EXEMPLE : p=4

000	101	010
0000	1010	0101

Codages correcteur

Codage par blocs : chaque mot de longueur m (éléments binaires) sera codé par un mot de longueur fixe $n = m + k$.
schéma de transmission :

$$i_1 i_2 \dots i_m \underbrace{\rightsquigarrow}_{codage} y_1 y_2 \dots y_n \dashrightarrow canal \dashrightarrow$$

$$y'_1 y'_2 \dots y'_n \underbrace{\rightsquigarrow}_{decodage} i'_1 i'_2 \dots i'_m$$

Parmi les 2^n mots $y'_1 y'_2 \dots y'_n$ reçus, il n'y en a que 2^m qui sont corrects il y a donc potentiellement $2^n - 2^m$ mots erronés.

Codages correcteur

Codage par blocs : chaque mot de longueur m (éléments binaires) sera codé par un mot de longueur fixe $n = m + k$.
schéma de transmission :

$$i_1 i_2 \dots i_m \underbrace{\rightsquigarrow}_{codage} y_1 y_2 \dots y_n \dashrightarrow canal \dashrightarrow$$

$$y'_1 y'_2 \dots y'_n \underbrace{\rightsquigarrow}_{decodage} i'_1 i'_2 \dots i'_m$$

Parmi les 2^n mots $y'_1 y'_2 \dots y'_n$ reçus, il n'y en a que 2^m qui sont corrects il y a donc potentiellement $2^n - 2^m$ mots erronés.

code de Hamming

FIGURE: Richard Hamming – 1915-1998

Le code de Hamming[m+k,m] est 1-correcteur.

code de Hamming[7,4]

code de Hamming[7,4] ($m = 4$, $k = 3$, $n = 7$)

Soit la matrice $H = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}$

Ici 0 et 1 appartiennent au corps $\mathbb{Z}/2\mathbb{Z}$.

$\mathbb{Z}/2\mathbb{Z} = \{0, 1\}$

0 représente les nombres pairs et 1 les nombres impairs.

$0 + 0 = 1 + 1 = 0$ et $0 + 1 = 1 + 0 = 1$

$0 \cdot 0 = 0 \cdot 1 = 1 \cdot 0 = 0$ et $1 \cdot 1 = 1$

code de Hamming[7,4]

code de Hamming[7,4] ($m = 4$, $k = 3$, $n = 7$)

Soit la matrice $H = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}$

Ici 0 et 1 appartiennent au corps $\mathbb{Z}/2\mathbb{Z}$.

$\mathbb{Z}/2\mathbb{Z} = \{0, 1\}$

0 représente les nombres pairs et 1 les nombres impairs.

$0 + 0 = 1 + 1 = 0$ et $0 + 1 = 1 + 0 = 1$

$0 \cdot 0 = 0 \cdot 1 = 1 \cdot 0 = 0$ et $1 \cdot 1 = 1$

code de Hamming[7,4]

code de Hamming[7,4] ($m = 4$, $k = 3$, $n = 7$)

Soit la matrice $H = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}$

Ici 0 et 1 appartiennent au corps $\mathbb{Z}/2\mathbb{Z}$.

$\mathbb{Z}/2\mathbb{Z} = \{0, 1\}$

0 représente les nombres pairs et 1 les nombres impairs.

$0 + 0 = 1 + 1 = 0$ et $0 + 1 = 1 + 0 = 1$

$0 \cdot 0 = 0 \cdot 1 = 1 \cdot 0 = 0$ et $1 \cdot 1 = 1$

code de Hamming[7,4]

code de Hamming[7,4] ($m = 4$, $k = 3$, $n = 7$)

Soit la matrice $H = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}$

Ici 0 et 1 appartiennent au corps $\mathbb{Z}/2\mathbb{Z}$.

$\mathbb{Z}/2\mathbb{Z} = \{0, 1\}$

0 représente les nombres pairs et 1 les nombres impairs.

$0 + 0 = 1 + 1 = 0$ et $0 + 1 = 1 + 0 = 1$

$0 \cdot 0 = 0 \cdot 1 = 1 \cdot 0 = 0$ et $1 \cdot 1 = 1$

code de Hamming[7,4]

code de Hamming[7,4] ($m = 4$, $k = 3$, $n = 7$)

Soit la matrice $H = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}$

Ici 0 et 1 appartiennent au corps $\mathbb{Z}/2\mathbb{Z}$.

$\mathbb{Z}/2\mathbb{Z} = \{0, 1\}$

0 représente les nombres pairs et 1 les nombres impairs.

$0 + 0 = 1 + 1 = 0$ et $0 + 1 = 1 + 0 = 1$

$0 \cdot 0 = 0 \cdot 1 = 1 \cdot 0 = 0$ et $1 \cdot 1 = 1$

code de Hamming[7,4]

code de Hamming[7,4] ($m = 4$, $k = 3$, $n = 7$)

Soit la matrice $H = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}$

Ici 0 et 1 appartiennent au corps $\mathbb{Z}/2\mathbb{Z}$.

$\mathbb{Z}/2\mathbb{Z} = \{0, 1\}$

0 représente les nombres pairs et 1 les nombres impairs.

$0 + 0 = 1 + 1 = 0$ et $0 + 1 = 1 + 0 = 1$

$0 \cdot 0 = 0 \cdot 1 = 1 \cdot 0 = 0$ et $1 \cdot 1 = 1$

codage

- chaque mot de 4 bits $i_1 i_2 i_3 i_4$ est tout d'abord complété par trois 0 en préfixe et on obtient un vecteur a de 7 bits $000 i_1 i_2 i_3 i_4$
- on calcule le vecteur $r = Ha$
- le mot code est alors $c = r i_1 i_2 i_3 i_4$

codage

- chaque mot de 4 bits $i_1 i_2 i_3 i_4$ est tout d'abord complété par trois 0 en préfixe et on obtient un vecteur a de 7 bits $000 i_1 i_2 i_3 i_4$
- on calcule le vecteur $r = Ha$
- le mot code est alors $c = r i_1 i_2 i_3 i_4$

codage

- chaque mot de 4 bits $i_1 i_2 i_3 i_4$ est tout d'abord complété par trois 0 en préfixe et on obtient un vecteur a de 7 bits $000 i_1 i_2 i_3 i_4$
- on calcule le vecteur $r = Ha$
- le mot code est alors $c = r i_1 i_2 i_3 i_4$

EXEMPLE : soit à coder 1010.

- $a = 0001010$
- $r = Ha = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$
- le mot code est $c = 0011010$

décodage : soit t le mot de 7 bits reçu

- on calcule $s = Ht$
- si $s = 0$ alors on garde les 4 derniers bits de t comme résultat du décodage de la transmission
- sinon s correspond à une colonne j de H et on change alors le bit j de t et on en garde les 4 derniers bits comme résultat du décodage de la transmission.

décodage : soit t le mot de 7 bits reçu

- on calcule $s = Ht$
- si $s = 0$ alors on garde les 4 derniers bits de t comme résultat du décodage de la transmission
- sinon s correspond à une colonne j de H et on change alors le bit j de t et on en garde les 4 derniers bits comme résultat du décodage de la transmission.

décodage : soit t le mot de 7 bits reçu

- on calcule $s = Ht$
- si $s = 0$ alors on garde les 4 derniers bits de t comme résultat du décodage de la transmission
- sinon s correspond à une colonne j de H et on change alors le bit j de t et on en garde les 4 derniers bits comme résultat du décodage de la transmission.

EXEMPLE : soit $t = 0010010$ le mot reçu.

- $s = Ht = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$
- s correspond à la colonne 4 de H, on change le bit 4 de $t = 0010010$
- le mot décodé est 1010.

Ce code de Hamming ne fonctionne que si une erreur au plus s'est produite.
Pour le cas général, une matrice $k \times n$ est utilisée sur le même principe.
Il est appliqué au Minitel avec $m = 120$, $k = 7$ en rajoutant un bit de parité et un octet formé de 8 zéros qui permet de détecter les pannes techniques importantes (foudre ...).

I.Calcul propositionnel

II.Logique des prédicats

III.Sémantique

IV.Exercices

Chapitre II – Logique

- I.Calcul propositionnel
- II.Logique des prédicats
- III.Sémantique
- IV.Exercices

I.Calcul propositionnel

II.Logique des prédicats

III.Sémantique

IV.Exercices

Chapitre II – Logique

- I.Calcul propositionnel
- II.Logique des prédicats
- III.Sémantique
- IV.Exercices

I.Calcul propositionnel

II.Logique des prédicats

III.Sémantique

IV.Exercices

Chapitre II – Logique

- I.Calcul propositionnel
- II.Logique des prédicats
- III.Sémantique
- IV.Exercices

I.Calcul propositionnel

II.Logique des prédicats

III.Sémantique

IV.Exercices

Chapitre II – Logique

- I.Calcul propositionnel
- II.Logique des prédicats
- III.Sémantique
- IV.Exercices

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Le calcul propositionnel permet une modélisation du raisonnement mathématique simple : il traite des problèmes où les assertions ne peuvent prendre que deux valeurs possibles.
Le calcul propositionnel a une propriété fondamentale : il est complet c'est-à-dire que tout ce qui est vrai est démontrable.
Mais ce cadre ne suffit pas à décrire certaines situations mathématiques courantes comme l'existence d'un objet satisfaisant à une propriété donnée.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

De façon générale, la logique fait apparaître la différence entre

- la *syntaxe* – les règles formelles de manipulation des symboles utilisés
- la *sémantique* – l'interprétation des formules.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

De façon générale, la logique fait apparaître la différence entre

- la *syntaxe* – les règles formelles de manipulation des symboles utilisés
- la *sémantique* – l'interprétation des formules.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Définition des formules propositionnelles

Soit $\mathcal{A}$ un ensemble de symboles constitué de la façon suivante :

- $\vee, \wedge, \rightarrow, \neg \ \in \mathcal{A}$
- $T, F \in \mathcal{A}$
- $\mathcal{V} \subset \mathcal{A}$ où $\mathcal{V}$ est un ensemble dont les éléments sont appelés variables propositionnelles ou atomes
- $(,)$.

$\mathcal{A} = \{\vee, \wedge, \rightarrow, \neg, (,), T, F\} \cup \mathcal{V}$.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Définition des formules propositionnelles

Soit $\mathcal{A}$ un ensemble de symboles constitué de la façon suivante :

- $\vee, \wedge, \rightarrow, \neg \ \in \mathcal{A}$
- $T, F \in \mathcal{A}$
- $\mathcal{V} \subset \mathcal{A}$ où $\mathcal{V}$ est un ensemble dont les éléments sont appelés variables propositionnelles ou atomes
- $(,)$.

$\mathcal{A} = \{\vee, \wedge, \rightarrow, \neg, (,), T, F\} \cup \mathcal{V}$.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Définition des formules propositionnelles

Soit $\mathcal{A}$ un ensemble de symboles constitué de la façon suivante :

- $\vee, \wedge, \rightarrow, \neg \ \in \mathcal{A}$
- $T, F \in \mathcal{A}$
- $\mathcal{V} \subset \mathcal{A}$ où $\mathcal{V}$ est un ensemble dont les éléments sont appelés variables propositionnelles ou atomes
- $(,)$.

$\mathcal{A} = \{\vee, \wedge, \rightarrow, \neg, (,), T, F\} \cup \mathcal{V}$.

Définition des formules propositionnelles

Soit $\mathcal{A}$ un ensemble de symboles constitué de la façon suivante :

- $\vee, \wedge, \rightarrow, \neg \ \in \mathcal{A}$
- $T, F \in \mathcal{A}$
- $\mathcal{V} \subset \mathcal{A}$ où $\mathcal{V}$ est un ensemble dont les éléments sont appelés variables propositionnelles ou atomes
- $(,)$.

$\mathcal{A} = \{\vee, \wedge, \rightarrow, \neg, (,), T, F\} \cup \mathcal{V}$.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Définition

L'ensemble $\mathcal{P}$ des formules propositionnelles construites sur $\mathcal{V}$ est défini inductivement par

- $T \in \mathcal{P}$, $F \in \mathcal{P}$, $\mathcal{V} \subset \mathcal{P}$ (les éléments de $\mathcal{V}$ sont alors appelés formules atomiques)
- si $\varphi \in \mathcal{P}$ alors $\neg\varphi \in \mathcal{P}$
- si $\varphi, \psi \in \mathcal{P}$ alors $(\varphi \vee \psi), (\varphi \wedge \psi), (\varphi \to \psi) \in \mathcal{P}$

$\vee, \wedge, \to, \neg$ sont des symboles logiques.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Définition

L'ensemble $\mathcal{P}$ des formules propositionnelles construites sur $\mathcal{V}$ est défini inductivement par

- $T \in \mathcal{P}$, $F \in \mathcal{P}$, $\mathcal{V} \subset \mathcal{P}$ (les éléments de $\mathcal{V}$ sont alors appelés formules atomiques)
- si $\varphi \in \mathcal{P}$ alors $\neg\varphi \in \mathcal{P}$
- si $\varphi, \psi \in \mathcal{P}$ alors $(\varphi \vee \psi), (\varphi \wedge \psi), (\varphi \to \psi) \in \mathcal{P}$

$\vee, \wedge, \to, \neg$ sont des symboles logiques.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Définition

L'ensemble $\mathcal{P}$ des formules propositionnelles construites sur $\mathcal{V}$ est défini inductivement par

- $T \in \mathcal{P}$, $F \in \mathcal{P}$, $\mathcal{V} \subset \mathcal{P}$ (les éléments de $\mathcal{V}$ sont alors appelés formules atomiques)
- si $\varphi \in \mathcal{P}$ alors $\neg\varphi \in \mathcal{P}$
- si $\varphi, \psi \in \mathcal{P}$ alors $(\varphi \vee \psi), (\varphi \wedge \psi), (\varphi \to \psi) \in \mathcal{P}$

$\vee, \wedge, \to, \neg$ sont des symboles logiques.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Définition

L'ensemble $\mathcal{P}$ des formules propositionnelles construites sur $\mathcal{V}$ est défini inductivement par

- $T \in \mathcal{P}$, $F \in \mathcal{P}$, $\mathcal{V} \subset \mathcal{P}$ (les éléments de $\mathcal{V}$ sont alors appelés formules atomiques)
- si $\varphi \in \mathcal{P}$ alors $\neg\varphi \in \mathcal{P}$
- si $\varphi, \psi \in \mathcal{P}$ alors $(\varphi \vee \psi), (\varphi \wedge \psi), (\varphi \to \psi) \in \mathcal{P}$

$\vee, \wedge, \to, \neg$ sont des symboles logiques.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Soit $\mathcal{V} = \{A, B, C\}$.

$\varphi = (A \to (B \to \neg C)) \in \mathcal{P}$

$\psi = (\neg(A \to C) \to B) \in \mathcal{P}$.

$A \to B \to \neg C \notin \mathcal{P}$

REMARQUE : On pourra omettre les parenthèses extérieures

$(A \to B) \to \neg C$ et $A \to (B \to \neg C)$ sont deux formules différentes.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Soit $\mathcal{V} = \{A, B, C\}$.

$\varphi = (A \to (B \to \neg C)) \in \mathcal{P}$

$\psi = (\neg(A \to C) \to B) \in \mathcal{P}$.

$A \to B \to \neg C \notin \mathcal{P}$

REMARQUE : On pourra omettre les parenthèses extérieures

$(A \to B) \to \neg C$ et $A \to (B \to \neg C)$ sont deux formules différentes.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Soit $\mathcal{V} = \{A, B, C\}$.

$\varphi = (A \to (B \to \neg C)) \in \mathcal{P}$

$\psi = (\neg(A \to C) \to B) \in \mathcal{P}$.

$A \to B \to \neg C \notin \mathcal{P}$

REMARQUE : On pourra omettre les parenthèses extérieures

$(A \to B) \to \neg C$ et $A \to (B \to \neg C)$ sont deux formules différentes.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Soit $\mathcal{V} = \{A, B, C\}$.

$\varphi = (A \to (B \to \neg C)) \in \mathcal{P}$

$\psi = (\neg(A \to C) \to B) \in \mathcal{P}$.

$A \to B \to \neg C \notin \mathcal{P}$

REMARQUE : On pourra omettre les parenthèses extérieures

$(A \to B) \to \neg C$ et $A \to (B \to \neg C)$ sont deux formules différentes.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Soit $\mathcal{V} = \{A, B, C\}$.

$\varphi = (A \to (B \to \neg C)) \in \mathcal{P}$

$\psi = (\neg(A \to C) \to B) \in \mathcal{P}$.

$A \to B \to \neg C \notin \mathcal{P}$

REMARQUE : On pourra omettre les parenthèses extérieures

$(A \to B) \to \neg C$ et $A \to (B \to \neg C)$ sont deux formules différentes.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Soit $\mathcal{V} = \{A, B, C\}$.

$\varphi = (A \to (B \to \neg C)) \in \mathcal{P}$

$\psi = (\neg(A \to C) \to B) \in \mathcal{P}$.

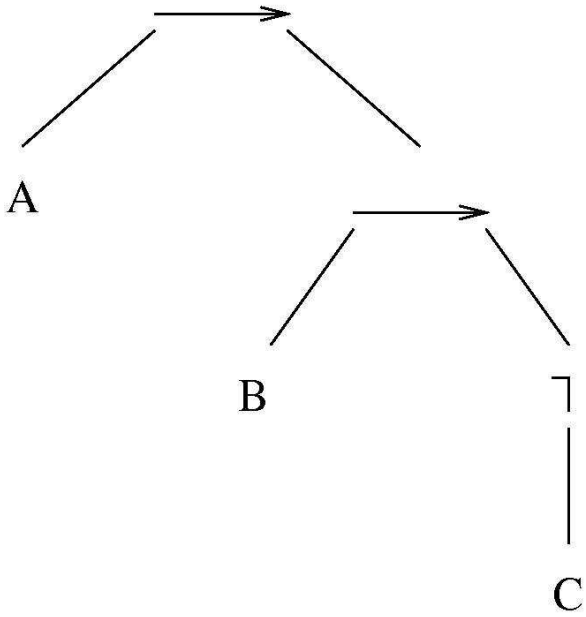
$A \to B \to \neg C \notin \mathcal{P}$

REMARQUE : On pourra omettre les parenthèses extérieures

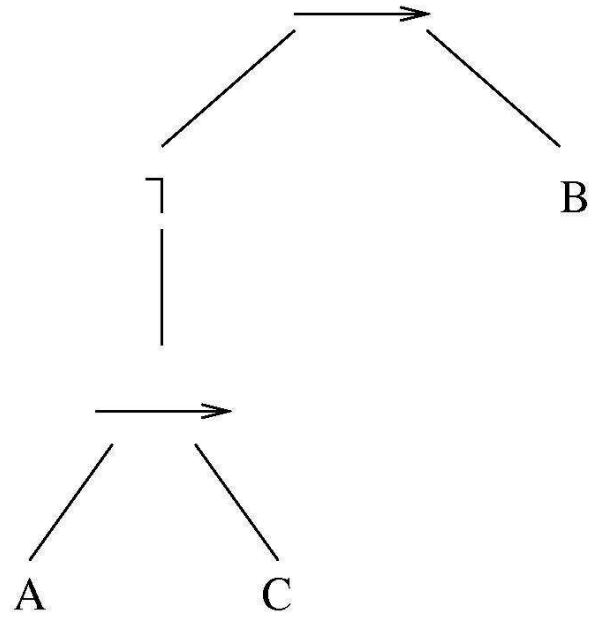
$(A \to B) \to \neg C$ et $A \to (B \to \neg C)$ sont deux formules différentes.

On peut considérer que la structure des éléments de $\mathcal{P}$ est arborescente :

- les feuilles sont les formules atomiques
- les noeuds internes sont des symboles logiques.



φ



ψ

Cela permet de définir facilement la notion de *sous-formule*

Définition

l'ensemble des sous-formules d'une formule propositionnelle φ est l'ensemble des sous-arbres au sens large de φ.

EXEMPLE : pour $\varphi = A \to (B \to \neg C)$ les sous-formules sont

- φ
- A (sous-arbre gauche)
- $B \to \neg C$ (sous-arbre droit)
- B
- $\neg C$
- C

Substitution

En remplaçant les feuilles par d'autres arbres on obtient un nouvel arbre.

Définition

Soient $A_1, \cdots A_n$ n variables propositionnelles distinctes deux à deux. Soient $\varphi_1, \cdots, \varphi_n$ n formules propositionnelles et ψ une formule propositionnelle. On définit $\psi' = \psi_{\varphi_1/A_1, \cdots, \varphi_n/A_n}$ par

- si $\psi \in \mathcal{V}$ et si $\psi \notin \{A_1, \cdots, A_n\}$ alors $\psi' = \psi$
- si $\psi = A_k$ alors $\psi' = \varphi_k$
- si $\psi = \neg\phi$ alors $\psi' = \neg\phi_{\varphi_1/A_1, \cdots, \varphi_n/A_n}$
- si $\psi = (\alpha \diamond \beta)$ alors $\psi' = \alpha_{\varphi_1/A_1, \cdots, \varphi_n/A_n} \diamond \beta_{\varphi_1/A_1, \cdots, \varphi_n/A_n}$ pour $\diamond = \vee, \wedge, \rightarrow$

Théorème

$\psi_{\varphi_1/A_1, \cdots, \varphi_n/A_n}$ est une formule propositionnelle.

Substitution

En remplaçant les feuilles par d'autres arbres on obtient un nouvel arbre.

Définition

Soient $A_1, \cdots A_n$ n variables propositionnelles distinctes deux à deux. Soient $\varphi_1, \cdots, \varphi_n$ n formules propositionnelles et ψ une formule propositionnelle. On définit $\psi' = \psi_{\varphi_1/A_1, \cdots, \varphi_n/A_n}$ par

- si $\psi \in \mathcal{V}$ et si $\psi \notin \{A_1, \cdots, A_n\}$ alors $\psi' = \psi$
- si $\psi = A_k$ alors $\psi' = \varphi_k$
- si $\psi = \neg\phi$ alors $\psi' = \neg\phi_{\varphi_1/A_1, \cdots, \varphi_n/A_n}$
- si $\psi = (\alpha \diamond \beta)$ alors $\psi' = \alpha_{\varphi_1/A_1, \cdots, \varphi_n/A_n} \diamond \beta_{\varphi_1/A_1, \cdots, \varphi_n/A_n}$ pour $\diamond = \vee, \wedge, \rightarrow$

Théorème

$\psi_{\varphi_1/A_1, \cdots, \varphi_n/A_n}$ est une formule propositionnelle.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Substitution

En remplaçant les feuilles par d'autres arbres on obtient un nouvel arbre.

Définition

Soient $A_1, \cdots A_n$ n variables propositionnelles distinctes deux à deux. Soient $\varphi_1, \cdots, \varphi_n$ n formules propositionnelles et ψ une formule propositionnelle. On définit $\psi' = \psi_{\varphi_1/A_1,\cdots,\varphi_n/A_n}$ par

- si $\psi \in \mathcal{V}$ et si $\psi \notin \{A_1, \cdots, A_n\}$ alors $\psi' = \psi$
- si $\psi = A_k$ alors $\psi' = \varphi_k$
- si $\psi = \neg\phi$ alors $\psi' = \neg\phi_{\varphi_1/A_1,\cdots,\varphi_n/A_n}$
- si $\psi = (\alpha \diamond \beta)$ alors $\psi' = \alpha_{\varphi_1/A_1,\cdots,\varphi_n/A_n} \diamond \beta_{\varphi_1/A_1,\cdots,\varphi_n/A_n}$ pour $\diamond = \vee, \wedge, \rightarrow$

Théorème

$\psi_{\varphi_1/A_1,\cdots,\varphi_n/A_n}$ est une formule propositionnelle.

Substitution

En remplaçant les feuilles par d'autres arbres on obtient un nouvel arbre.

Définition

Soient $A_1, \cdots A_n$ n variables propositionnelles distinctes deux à deux. Soient $\varphi_1, \cdots, \varphi_n$ n formules propositionnelles et ψ une formule propositionnelle. On définit $\psi' = \psi_{\varphi_1/A_1, \cdots, \varphi_n/A_n}$ par

- si $\psi \in \mathcal{V}$ et si $\psi \notin \{A_1, \cdots, A_n\}$ alors $\psi' = \psi$
- si $\psi = A_k$ alors $\psi' = \varphi_k$
- si $\psi = \neg\phi$ alors $\psi' = \neg\phi_{\varphi_1/A_1, \cdots, \varphi_n/A_n}$
- si $\psi = (\alpha \diamond \beta)$ alors $\psi' = \alpha_{\varphi_1/A_1, \cdots, \varphi_n/A_n} \diamond \beta_{\varphi_1/A_1, \cdots, \varphi_n/A_n}$ pour $\diamond = \vee, \wedge, \rightarrow$

Théorème

$\psi_{\varphi_1/A_1, \cdots, \varphi_n/A_n}$ est une formule propositionnelle.

Substitution

En remplaçant les feuilles par d'autres arbres on obtient un nouvel arbre.

Définition

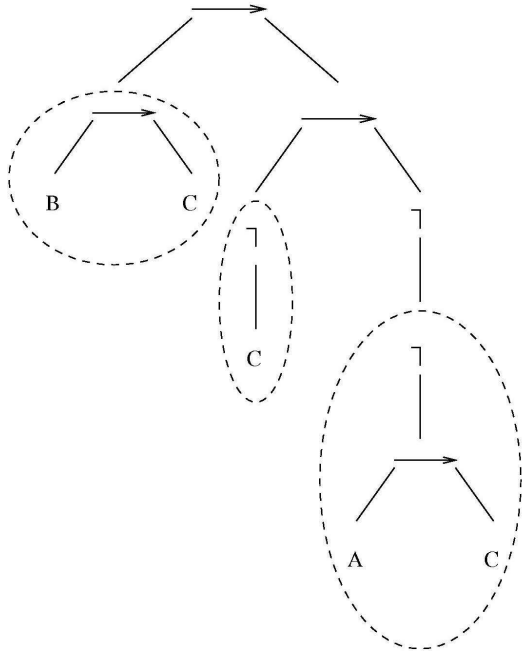
Soient $A_1, \cdots A_n$ n variables propositionnelles distinctes deux à deux. Soient $\varphi_1, \cdots, \varphi_n$ n formules propositionnelles et ψ une formule propositionnelle. On définit $\psi' = \psi_{\varphi_1/A_1,\cdots,\varphi_n/A_n}$ par

- si $\psi \in \mathcal{V}$ et si $\psi \notin \{A_1, \cdots, A_n\}$ alors $\psi' = \psi$
- si $\psi = A_k$ alors $\psi' = \varphi_k$
- si $\psi = \neg\phi$ alors $\psi' = \neg\phi_{\varphi_1/A_1,\cdots,\varphi_n/A_n}$
- si $\psi = (\alpha \diamond \beta)$ alors $\psi' = \alpha_{\varphi_1/A_1,\cdots,\varphi_n/A_n} \diamond \beta_{\varphi_1/A_1,\cdots,\varphi_n/A_n}$ pour $\diamond = \vee, \wedge, \rightarrow$

Théorème

$\psi_{\varphi_1/A_1,\cdots,\varphi_n/A_n}$ est une formule propositionnelle.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices



Algèbre de Boole minimale

L'algèbre de Boole minimale, notée $\mathbb{B}$est l'ensemble $\{0, 1\}$ muni de deux opérations binaires $+$, $\cdot$ décrites par leur table :

$+$	0	1
0	0	1
1	1	1

$\cdot$	0	1
0	0	0
1	0	1

et d'une opération unaire $\bar{}$ telle que $\overline{0} = 1$ et $\overline{1} = 0$.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Propriétés

Théorème

pour tous $x, y \in \mathbb{B}$, on a

- $0 + x = x + 0 = x$ (0 est neutre pour $+$)
- $1 + x = x + 1 = 1$ (1 est absorbant pour $+$)
- $1 \cdot x = x \cdot 1 = x$ (1 est neutre pour $\cdot$)
- $0 \cdot x = x \cdot 0 = 0$ (0 est absorbant pour $\cdot$)
- $\overline{\overline{x}} = x$, $\overline{x} + x = 1$, $\overline{x} \cdot x = 0$
- $x + x = x$, $x \cdot x = x$
- $+$ est commutatif, associatif et distributif par rapport à $\cdot$
- $\cdot$ est commutatif, associatif et distributif par rapport à $+$
- $\overline{x + y} = \overline{x} \cdot \overline{y}$ et $\overline{x \cdot y} = \overline{x} + \overline{y}$ sont les lois de De Morgan.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Propriétés

Théorème

pour tous $x, y \in \mathbb{B}$, on a

- $0 + x = x + 0 = x$ (0 est neutre pour $+$)
- $1 + x = x + 1 = 1$ (1 est absorbant pour $+$)
- $1 \cdot x = x \cdot 1 = x$ (1 est neutre pour $\cdot$)
- $0 \cdot x = x \cdot 0 = 0$ (0 est absorbant pour $\cdot$)
- $\overline{\overline{x}} = x$, $\overline{x} + x = 1$, $\overline{x} \cdot x = 0$
- $x + x = x$, $x \cdot x = x$
- $+$ est commutatif, associatif et distributif par rapport à $\cdot$
- $\cdot$ est commutatif, associatif et distributif par rapport à $+$
- $\overline{x + y} = \overline{x} \cdot \overline{y}$ et $\overline{x \cdot y} = \overline{x} + \overline{y}$ sont les lois de De Morgan.

Propriétés

Théorème

pour tous $x, y \in \mathbb{B}$, on a

- $0 + x = x + 0 = x$ (0 est neutre pour $+$)
- $1 + x = x + 1 = 1$ (1 est absorbant pour $+$)
- $1 \cdot x = x \cdot 1 = x$ (1 est neutre pour $\cdot$)
- $0 \cdot x = x \cdot 0 = 0$ (0 est absorbant pour $\cdot$)
- $\overline{\overline{x}} = x$, $\overline{x} + x = 1$, $\overline{x} \cdot x = 0$
- $x + x = x$, $x \cdot x = x$
- $+$ est commutatif, associatif et distributif par rapport à $\cdot$
- $\cdot$ est commutatif, associatif et distributif par rapport à $+$
- $\overline{x + y} = \overline{x} \cdot \overline{y}$ et $\overline{x \cdot y} = \overline{x} + \overline{y}$ sont les lois de De Morgan.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Propriétés

Théorème

pour tous $x, y \in \mathbb{B}$, on a

- $0 + x = x + 0 = x$ (0 est neutre pour $+$)
- $1 + x = x + 1 = 1$ (1 est absorbant pour $+$)
- $1 \cdot x = x \cdot 1 = x$ (1 est neutre pour $\cdot$)
- $0 \cdot x = x \cdot 0 = 0$ (0 est absorbant pour $\cdot$)
- $\overline{\overline{x}} = x$, $\overline{x} + x = 1$, $\overline{x} \cdot x = 0$
- $x + x = x$, $x \cdot x = x$
- $+$ est commutatif, associatif et distributif par rapport à $\cdot$
- $\cdot$ est commutatif, associatif et distributif par rapport à $+$
- $\overline{x + y} = \overline{x} \cdot \overline{y}$ et $\overline{x \cdot y} = \overline{x} + \overline{y}$ sont les lois de De Morgan.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Propriétés

Théorème

pour tous $x, y \in \mathbb{B}$, on a

- $0 + x = x + 0 = x$ (0 est neutre pour $+$)
- $1 + x = x + 1 = 1$ (1 est absorbant pour $+$)
- $1 \cdot x = x \cdot 1 = x$ (1 est neutre pour $\cdot$)
- $0 \cdot x = x \cdot 0 = 0$ (0 est absorbant pour $\cdot$)
- $\overline{\overline{x}} = x$, $\overline{x} + x = 1$, $\overline{x} \cdot x = 0$
- $x + x = x$, $x \cdot x = x$
- $+$ est commutatif, associatif et distributif par rapport à $\cdot$
- $\cdot$ est commutatif, associatif et distributif par rapport à $+$
- $\overline{x + y} = \overline{x} \cdot \overline{y}$ et $\overline{x \cdot y} = \overline{x} + \overline{y}$ sont les lois de De Morgan.

Propriétés

Théorème

pour tous $x, y \in \mathbb{B}$, on a

- $0 + x = x + 0 = x$ (0 est neutre pour $+$)
- $1 + x = x + 1 = 1$ (1 est absorbant pour $+$)
- $1 \cdot x = x \cdot 1 = x$ (1 est neutre pour $\cdot$)
- $0 \cdot x = x \cdot 0 = 0$ (0 est absorbant pour $\cdot$)
- $\overline{\overline{x}} = x$, $\overline{x} + x = 1$, $\overline{x} \cdot x = 0$
- $x + x = x$, $x \cdot x = x$
- $+$ est commutatif, associatif et distributif par rapport à $\cdot$
- $\cdot$ est commutatif, associatif et distributif par rapport à $+$
- $\overline{x + y} = \overline{x} \cdot \overline{y}$ et $\overline{x \cdot y} = \overline{x} + \overline{y}$ sont les lois de De Morgan.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Propriétés

Théorème

pour tous $x, y \in \mathbb{B}$, on a

- $0 + x = x + 0 = x$ (0 est neutre pour $+$)
- $1 + x = x + 1 = 1$ (1 est absorbant pour $+$)
- $1 \cdot x = x \cdot 1 = x$ (1 est neutre pour $\cdot$)
- $0 \cdot x = x \cdot 0 = 0$ (0 est absorbant pour $\cdot$)
- $\overline{\overline{x}} = x$, $\overline{x} + x = 1$, $\overline{x} \cdot x = 0$
- $x + x = x$, $x \cdot x = x$
- $+$ est commutatif, associatif et distributif par rapport à $\cdot$
- $\cdot$ est commutatif, associatif et distributif par rapport à $+$
- $\overline{x + y} = \overline{x} \cdot \overline{y}$ et $\overline{x \cdot y} = \overline{x} + \overline{y}$ sont les lois de De Morgan.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Propriétés

Théorème

pour tous $x, y \in \mathbb{B}$, on a

- $0 + x = x + 0 = x$ (0 est neutre pour $+$)
- $1 + x = x + 1 = 1$ (1 est absorbant pour $+$)
- $1 \cdot x = x \cdot 1 = x$ (1 est neutre pour $\cdot$)
- $0 \cdot x = x \cdot 0 = 0$ (0 est absorbant pour $\cdot$)
- $\overline{\overline{x}} = x$, $\overline{x} + x = 1$, $\overline{x} \cdot x = 0$
- $x + x = x$, $x \cdot x = x$
- $+$ est commutatif, associatif et distributif par rapport à $\cdot$
- $\cdot$ est commutatif, associatif et distributif par rapport à $+$
- $\overline{x + y} = \overline{x} \cdot \overline{y}$ et $\overline{x \cdot y} = \overline{x} + \overline{y}$ sont les lois de De Morgan.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Propriétés

Théorème

pour tous $x, y \in \mathbb{B}$, on a

- $0 + x = x + 0 = x$ (0 est neutre pour $+$)
- $1 + x = x + 1 = 1$ (1 est absorbant pour $+$)
- $1 \cdot x = x \cdot 1 = x$ (1 est neutre pour $\cdot$)
- $0 \cdot x = x \cdot 0 = 0$ (0 est absorbant pour $\cdot$)
- $\overline{\overline{x}} = x$, $\overline{x} + x = 1$, $\overline{x} \cdot x = 0$
- $x + x = x$, $x \cdot x = x$
- $+$ est commutatif, associatif et distributif par rapport à $\cdot$
- $\cdot$ est commutatif, associatif et distributif par rapport à $+$
- $\overline{x + y} = \overline{x} \cdot \overline{y}$ et $\overline{x \cdot y} = \overline{x} + \overline{y}$ sont les lois de De Morgan.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Interprétation – valeurs de vérité

Définition

une interprétation est une application $\nu_{\mathcal{V}}$ de $\mathcal{V}$ dans $\mathbb{B}$ (à chaque formule atomique est associé un élément de $\mathbb{B}$).

Le résultat suivant permet d'étendre la définition de $\nu_{\mathcal{V}}$ à $\mathcal{P}$.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Interprétation – valeurs de vérité

Définition

une interprétation est une application $v_{\mathcal{V}}$ de $\mathcal{V}$ dans $\mathbb{B}$ (à chaque formule atomique est associé un élément de $\mathbb{B}$).

Le résultat suivant permet d'étendre la définition de $v_{\mathcal{V}}$ à $\mathcal{P}$.

Théorème

Il existe une unique application v de $\mathcal{P}$ dans $\mathbb{B}$ prolongeant $v_{\mathcal{V}}$ telle que

- $v(A) = v_{\mathcal{V}}(A)$ pour tout $A \in \mathcal{V}$,
- $v(T) = 1$ et $v(F) = 0$,
- $v(\neg\varphi) = \overline{v(\varphi)}$,
- $v(\varphi \vee \psi) = v(\varphi) + v(\psi)$,
- $v(\varphi \wedge \psi) = v(\varphi) \cdot v(\psi)$,
- $v(\varphi \to \psi) = \overline{v(\varphi)} + v(\psi)$.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\varphi = (A \to (B \to \neg C))$

$\overline{v(\varphi)} = \overline{\overline{v(A)} + \overline{v((B \to \neg C)}} = \overline{v(A)} + \overline{v(B)} + v(\neg C) = \overline{v(A)} + \overline{v(B)} + \overline{v(C)}$

Supposons que $v_{\mathcal{V}}(A) = 1, v_{\mathcal{V}}(B) = 1, v_{\mathcal{V}}(C) = 1$, ($A, B, C$ sont tous les trois vrais) alors
$v(\varphi) = \overline{1} + \overline{1} + \overline{1} = 0 + 0 + 0 = 0$ donc φ est fausse.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Table de vérité

Combien y-a-t il d'interprétations possibles de $\mathcal{V}$ dans $\mathbb{B}$?

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Table de vérité

Combien y-a-t il d'interprétations possibles de $\mathcal{V}$ dans $\mathbb{B}$?
C'est égal au nombre d'applications de $\mathcal{V}$ dans $\mathbb{B}$, donc il y en a $|\mathbb{B}|^{|\mathcal{V}|} = 2^n$ (si n est le nombre de formules atomiques)

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Table de vérité

Combien y-a-t il d'interprétations possibles de $\mathcal{V}$ dans $\mathbb{B}$?
C'est égal au nombre d'applications de $\mathcal{V}$ dans $\mathbb{B}$, donc il y en a $|\mathbb{B}|^{|\mathcal{V}|} = 2^n$ (si n est le nombre de formules atomiques)

Définition

On appelle table de vérité d'une formule φ le tableau dans lequel figurent sur chaque ligne une interprétation $v_{\mathcal{V}}$ et $v(\varphi)$.

Il y a donc 2^n lignes pour n formules atomiques

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$$\varphi = (A \to (B \to \neg C))$$

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\varphi = (A \to (B \to \neg C))$

A	B	C	φ
0	0	0	

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\varphi = (A \to (B \to \neg C))$

A	B	C	φ
0	0	0	
0	0	1	

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\varphi = (A \to (B \to \neg C))$

A	B	C	φ
0	0	0	
0	0	1	
0	1	0	

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\varphi = (A \to (B \to \neg C))$

A	B	C	φ
0	0	0	
0	0	1	
0	1	0	
0	1	1	

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\varphi = (A \to (B \to \neg C))$

A	B	C	φ
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\varphi = (A \to (B \to \neg C))$

A	B	C	φ
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\varphi = (A \to (B \to \neg C))$

A	B	C	φ
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\varphi = (A \to (B \to \neg C))$

A	B	C	φ
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\varphi = (A \to (B \to \neg C))$

A	B	C	φ
0	0	0	1
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\varphi = (A \to (B \to \neg C))$

A	B	C	φ
0	0	0	1
0	0	1	1
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Exemple

$\varphi = (A \to (B \to \neg C))$

A	B	C	φ
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\varphi = (A \to (B \to \neg C))$

A	B	C	φ
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	
1	0	1	
1	1	0	
1	1	1	

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\varphi = (A \to (B \to \neg C))$

A	B	C	φ
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	
1	1	0	
1	1	1	

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\varphi = (A \to (B \to \neg C))$

A	B	C	φ
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	
1	1	1	

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\varphi = (A \to (B \to \neg C))$

A	B	C	φ
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\varphi = (A \to (B \to \neg C))$

A	B	C	φ
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Cas particuliers

Il est important de noter pour les connecteurs $\vee, \wedge, \rightarrow$ que leurs tables de vérité comportent toujours un cas particulier qui correspond à *Faux* ou *Vrai* selon le connecteur :

Propriétés

Soient φ et ψ deux formules et v une interprétation quelconque.

- $v(\varphi \rightarrow \psi) = 0$ *si et seulement si* $v(\varphi) = 1$ et $v(\psi) = 0$
- $v(\varphi \vee \psi) = 0$ *si et seulement si* $v(\varphi) = 0$ et $v(\psi) = 0$
- $v(\varphi \wedge \psi) = 1$ *si et seulement si* $v(\varphi) = 1$ et $v(\psi) = 1$

REMARQUE : en particulier, si $v(\varphi) = 0$ alors $v(\varphi \rightarrow \psi) = 1$ quelque soit la valeur de $v(\psi)$.

Cas particuliers

Il est important de noter pour les connecteurs $\vee, \wedge, \rightarrow$ que leurs tables de vérité comportent toujours un cas particulier qui correspond à *Faux* ou *Vrai* selon le connecteur :

Propriétés

Soient φ et ψ deux formules et v une interprétation quelconque.

- $v(\varphi \rightarrow \psi) = 0$ *si et seulement si* $v(\varphi) = 1$ et $v(\psi) = 0$
- $v(\varphi \vee \psi) = 0$ *si et seulement si* $v(\varphi) = 0$ et $v(\psi) = 0$
- $v(\varphi \wedge \psi) = 1$ *si et seulement si* $v(\varphi) = 1$ et $v(\psi) = 1$

REMARQUE : en particulier, si $v(\varphi) = 0$ alors $v(\varphi \rightarrow \psi) = 1$ quelque soit la valeur de $v(\psi)$.

Cas particuliers

Il est important de noter pour les connecteurs $\vee, \wedge, \rightarrow$ que leurs tables de vérité comportent toujours un cas particulier qui correspond à *Faux* ou *Vrai* selon le connecteur :

Propriétés

Soient φ et ψ deux formules et v une interprétation quelconque.

- $v(\varphi \rightarrow \psi) = 0$ *si et seulement si* $v(\varphi) = 1$ et $v(\psi) = 0$
- $v(\varphi \vee \psi) = 0$ *si et seulement si* $v(\varphi) = 0$ et $v(\psi) = 0$
- $v(\varphi \wedge \psi) = 1$ *si et seulement si* $v(\varphi) = 1$ et $v(\psi) = 1$

REMARQUE : en particulier, si $v(\varphi) = 0$ alors $v(\varphi \rightarrow \psi) = 1$ quelque soit la valeur de $v(\psi)$.

Cas particuliers

Il est important de noter pour les connecteurs $\vee, \wedge, \rightarrow$ que leurs tables de vérité comportent toujours un cas particulier qui correspond à *Faux* ou *Vrai* selon le connecteur :

Propriétés

Soient φ et ψ deux formules et v une interprétation quelconque.

- $v(\varphi \rightarrow \psi) = 0$ *si et seulement si* $v(\varphi) = 1$ et $v(\psi) = 0$
- $v(\varphi \vee \psi) = 0$ *si et seulement si* $v(\varphi) = 0$ et $v(\psi) = 0$
- $v(\varphi \wedge \psi) = 1$ *si et seulement si* $v(\varphi) = 1$ et $v(\psi) = 1$

REMARQUE : en particulier, si $v(\varphi) = 0$ alors $v(\varphi \rightarrow \psi) = 1$ quelque soit la valeur de $v(\psi)$.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$$\alpha = ((A \to B) \to (A \to C))$$

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\alpha = ((A \to B) \to (A \to C))$

A	B	C	α
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\alpha = ((A \to B) \to (A \to C))$

A	B	C	α
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	**0**	
1	0	1	
1	1	**0**	
1	1	1	

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\alpha = ((A \to B) \to (A \to C))$

A	B	C	α
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	**1**	**0**	0
1	1	1	

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

$\alpha = ((A \to B) \to (A \to C))$

A	B	C	α
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	**1**	**0**	0
1	1	1	1

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Equivalence

Définition

Deux formules φ et ψ sont équivalentes si pour toute interprétation v on a $v(\varphi) = v(\psi)$.
On note $\varphi \equiv \psi$

EXEMPLE : $A \to B \equiv \neg A \vee B$

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Equivalence

Définition

Deux formules φ et ψ sont équivalentes si pour toute interprétation v on a $v(\varphi) = v(\psi)$.
On note $\varphi \equiv \psi$

EXEMPLE : $A \to B \equiv \neg A \vee B$

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

« Calcul par morceaux »

Propriété

Soit φ une formule et α une sous-formule de φ. Si β est équivalente à α alors la formule φ' obtenue en substituant β à α dans φ est équivalente à φ.

On pourra donc calculer les valeurs de vérité des sous-formules de φ sur chaque ligne de la table de vérité puis utiliser ces valeurs pour achever le calcul pour φ.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

on veut calculer la table de vérité de la formule $\varphi = ((A \to B) \to C) \to (A \to C)$.

Pour toute interprétation v, $v(\varphi) = v'(K \to L)$ où $v'(K) = v((A \to B) \to C)$ et $v'(L) = v(A \to C)$.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Exemple

on veut calculer la table de vérité de la formule $\varphi = ((A \to B) \to C) \to (A \to C)$.
Pour toute interprétation v, $v(\varphi) = v'(K \to L)$ où $v'(K) = v((A \to B) \to C)$ et $v'(L) = v(A \to C)$.

A	B	C	$(A \to B) \to C$	$A \to C$	φ
0	0	0			
0	0	1			
0	1	0			
0	1	1			
1	0	0			
1	0	1			
1	1	0			
1	1	1			

Exemple

on veut calculer la table de vérité de la formule $\varphi = ((A \to B) \to C) \to (A \to C)$.
Pour toute interprétation v, $v(\varphi) = v'(K \to L)$ où $v'(K) = v((A \to B) \to C)$ et $v'(L) = v(A \to C)$.

A	B	C	$(A \to B) \to C$	$A \to C$	φ
0	0	0	0		
0	0	1	1		
0	1	0	0		
0	1	1	1		
1	0	0	1		
1	0	1	1		
1	1	0	0		
1	1	1	1		

Exemple

on veut calculer la table de vérité de la formule $\varphi = ((A \to B) \to C) \to (A \to C)$.
Pour toute interprétation v, $v(\varphi) = v'(K \to L)$ où $v'(K) = v((A \to B) \to C)$ et $v'(L) = v(A \to C)$.

A	B	C	$(A \to B) \to C$	$A \to C$	φ
0	0	0	0	1	
0	0	1	1	1	
0	1	0	0	1	
0	1	1	1	1	
1	0	0	1	0	
1	0	1	1	1	
1	1	0	0	0	
1	1	1	1	1	

Exemple

on veut calculer la table de vérité de la formule $\varphi = ((A \to B) \to C) \to (A \to C)$.
Pour toute interprétation v, $v(\varphi) = v'(K \to L)$ où $v'(K) = v((A \to B) \to C)$ et $v'(L) = v(A \to C)$.

A	B	C	$(A \to B) \to C$	$A \to C$	φ
0	0	0	0	1	1
0	0	1	1	1	1
0	1	0	0	1	1
0	1	1	1	1	1
1	0	0	1	0	0
1	0	1	1	1	1
1	1	0	0	0	1
1	1	1	1	1	1

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Lien avec le langage courant

Le connecteur logique $\vee$ correspond à la conjonction *ou* prise au sens large.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Lien avec le langage courant

Le connecteur logique $\vee$ correspond à la conjonction *ou* prise au sens large.

EXEMPLE : « je mangerai des pâtes ou une salade ».

A quelle condition cette affirmation se révèle fausse ?

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Lien avec le langage courant

Le connecteur logique $\vee$ correspond à la conjonction *ou* prise au sens large.

EXEMPLE : « je mangerai des pâtes ou une salade ».

A quelle condition cette affirmation se révèle fausse ?
un seul cas : **ni** pâtes **ni** salade

Lien avec le langage courant

Le connecteur logique $\vee$ correspond à la conjonction *ou* prise au sens large.

EXEMPLE : « je mangerai des pâtes ou une salade ».

A quelle condition cette affirmation se révèle fausse ?
un seul cas : **ni** pâtes **ni** salade
Cela correspond bien à la propriété

$$v(\varphi \vee \psi) = 0 \textit{ si et seulement si } v(\varphi) = 0 \text{ et } v(\psi) = 0$$

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Lien avec le langage courant

Le connecteur logique $\wedge$ correspond à la conjonction *et*.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Lien avec le langage courant

Le connecteur logique $\wedge$ correspond à la conjonction *et*.
EXEMPLE : « j'irai à la piscine et au cinéma ».

A quelle condition cette affirmation se révèle vraie ?

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Lien avec le langage courant

Le connecteur logique $\wedge$ correspond à la conjonction *et*.
EXEMPLE : « j'irai à la piscine et au cinéma ».

A quelle condition cette affirmation se révèle vraie ?
un seul cas : piscine **et** cinéma

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Lien avec le langage courant

Le connecteur logique $\wedge$ correspond à la conjonction *et*.
EXEMPLE : « j'irai à la piscine et au cinéma ».

A quelle condition cette affirmation se révèle vraie ?
un seul cas : piscine **et** cinéma
Cela correspond bien à la propriété

$$v(\varphi \wedge \psi) = 1 \textit{ si et seulement si } v(\varphi) = 1 \text{ et } v(\psi) = 1$$

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Lien avec le langage courant

Le connecteur logique $\rightarrow$ correspond à la construction de phrase *si* . . .*alors* . . ..

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Lien avec le langage courant

Le connecteur logique $\rightarrow$ correspond à la construction de phrase *si* . . .*alors* . . ..

EXEMPLE : ≪ si j'ai le temps alors je ferai mon devoir d'anglais ≫.

A quelle condition cette affirmation se révèle fausse ?

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Lien avec le langage courant

Le connecteur logique $\rightarrow$ correspond à la construction de phrase *si* . . .*alors* . . ..

EXEMPLE : ≪ si j'ai le temps alors je ferai mon devoir d'anglais ≫.

A quelle condition cette affirmation se révèle fausse ?
un seul cas : j'ai le temps **et** je ne fais pas mon devoir d'anglais

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Lien avec le langage courant

Le connecteur logique $\rightarrow$ correspond à la construction de phrase *si* . . .*alors* . . ..
EXEMPLE : ≪ si j'ai le temps alors je ferai mon devoir d'anglais ≫.

A quelle condition cette affirmation se révèle fausse ?
un seul cas : j'ai le temps **et** je ne fais pas mon devoir d'anglais
Cela correspond bien à la propriété

$$v(\varphi \rightarrow \psi) = 0 \text{ \textit{si et seulement si} } v(\varphi) = 1 \text{ et } v(\psi) = 0$$

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

« S'il neige alors il fait froid » est vraie car on n'a jamais de neige sans froid.

« S'il fait froid alors il neige » est fausse car il peut faire froid sans neige.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

« S'il neige alors il fait froid » est vraie car on n'a jamais de neige sans froid.

« S'il fait froid alors il neige » est fausse car il peut faire froid sans neige.

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Propriété de $\equiv$

La relation $\equiv$ est une relation d'équivalence.
Deux formules équivalentes signifient la même chose.
Si on donne n formules atomiques, le nombre de formules propositionnelles est infini.

Par exemple $A \to B$, $(A \to B) \to A$, $((A \to B) \to A) \to B$, ...

Mais le nombre de significations est fini : en effet le nombre de tables de vérité à n colonnes est égal à

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Propriété de $\equiv$

La relation $\equiv$ est une relation d'équivalence.
Deux formules équivalentes signifient la même chose.
Si on donne n formules atomiques, le nombre de formules propositionnelles est infini.

Par exemple $A \to B$, $(A \to B) \to A$, $((A \to B) \to A) \to B$, ...

Mais le nombre de significations est fini : en effet le nombre de tables de vérité à n colonnes est égal à

$$2^{2^n}$$

Propriété de $\equiv$

La relation $\equiv$ est une relation d'équivalence.
Deux formules équivalentes signifient la même chose.
Si on donne n formules atomiques, le nombre de formules propositionnelles est infini.

Par exemple $A \to B$, $(A \to B) \to A$, $((A \to B) \to A) \to B$, ...

Mais le nombre de significations est fini : en effet le nombre de tables de vérité à n colonnes est égal à

$$2^{2^n}$$

A une table de vérité correspond une infinité de formules propositionnelles équivalentes entre elles (une classe d'équivalence de $\equiv$).
Peut-on dégager une formule « type » pour représenter toutes celles qui sont dans une même classe ?

Forme normale disjonctive

Définition

une formule φ est sous *forme normale disjonctive* (FND) *si et seulement si*

$$\varphi = \bigvee (L_1 \wedge \cdots \wedge L_k)$$

où L_i est une formule atomique ou sa négation

Forme normale disjonctive

Définition

une formule φ est sous *forme normale disjonctive* (FND) *si et seulement si*

$$\varphi = \bigvee (L_1 \wedge \cdots \wedge L_k)$$

où L_i est une formule atomique ou sa négation

EXEMPLE :

$$(A \wedge \neg C) \vee (B \wedge C) \vee (\neg B \wedge \neg C)$$

$$A \vee B$$

2 termes

$$A \wedge C$$

1 terme

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

Forme normale conjonctive

Définition

une formule φ est sous *forme normale conjonctive* (FNC) *si et seulement si*

$$\varphi = \bigwedge (L_1 \vee \cdots \vee L_k)$$

où L_i est une formule atomique ou sa négation

Forme normale conjonctive

Définition

une formule φ est sous *forme normale conjonctive* (FNC) *si et seulement si*

$$\varphi = \bigwedge (L_1 \vee \cdots \vee L_k)$$

où L_i est une formule atomique ou sa négation

EXEMPLE :

$$(\neg A \vee C) \wedge (\neg B \vee C) \wedge (B \vee \neg C)$$

$$A \vee B$$

1 terme

$$A \wedge C$$

2 termes

De la table de vérité à la forme normale

$\varphi = (A \to (((B \wedge \neg A) \vee (\neg C \wedge A)) \longleftrightarrow (A \vee (A \to B)))).$
table de vérité de φ :

A	B	C	φ
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

φ est satisfait par les interprétations suivantes :

0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	1	0

$\neg\varphi$ est satisfait par les interprétations suivantes :

1	0	1
1	1	1

On peut alors écrire

$\varphi \equiv (\neg A \wedge \neg B \wedge \neg C) \vee (\neg A \wedge \neg B \wedge C) \vee (\neg A \wedge B \wedge \neg C) \vee (\neg A \wedge B \wedge C) \vee (A \wedge \neg B \wedge \neg C) \vee (A \wedge B \wedge \neg C)$ (FND)

$\neg\varphi \equiv (A \wedge \neg B \wedge C) \vee (A \wedge B \wedge C)$.

Par les lois de De Morgan

$\varphi \equiv (\neg A \vee B \vee \neg C) \wedge (\neg A \vee \neg B \vee \neg C)$ (FNC)

De la formule à la forme normale par des équivalences

$\varphi \wedge \varphi \equiv \varphi$ (idempotence)
$\varphi \vee \varphi \equiv \varphi$

$\varphi \wedge \psi \equiv \psi \wedge \varphi$ (commutativité)
$\varphi \vee \psi \equiv \psi \vee \varphi$

$(\varphi \wedge \psi) \wedge \phi \equiv \varphi \wedge (\psi \wedge \phi)$ (associativité)
$(\varphi \vee \psi) \vee \phi \equiv \varphi \vee (\psi \vee \phi)$

$\varphi \vee (\psi \wedge \phi) \equiv (\varphi \vee \psi) \wedge (\varphi \vee \phi)$ (distributivité)
$\varphi \wedge (\psi \vee \phi) \equiv (\varphi \wedge \psi) \vee (\varphi \wedge \phi)$

$$\varphi \wedge (\varphi \vee \psi) \equiv \varphi \qquad \text{(absorption)}$$
$$\varphi \vee (\varphi \wedge \psi) \equiv \varphi$$

$$\neg(\varphi \vee \psi) \equiv (\neg\varphi \wedge \neg\psi) \qquad \text{(lois de de Morgan)}$$
$$\neg(\varphi \wedge \psi) \equiv (\neg\varphi \vee \neg\psi)$$

$$\varphi \to \psi \equiv \neg\varphi \vee \psi$$
$$\varphi \to \psi \equiv \neg\psi \to \neg\varphi \qquad \text{(contraposée)}$$
$$\varphi \to \psi \equiv \neg(\varphi \wedge \neg\psi) \qquad \text{(par l'absurde)}$$

$$\varphi \vee \psi \equiv \neg\varphi \to \psi$$

$$\varphi \wedge \psi \equiv \neg(\varphi \to \neg\psi)$$

I.Calcul propositionnel
1.Syntaxe
2.Sémantique
3.Formes normales
II.Logique des prédicats
III.Sémantique
IV.Exercices

$$\varphi \wedge \neg\varphi \equiv F$$
$$\varphi \vee \neg\varphi \equiv T$$

$$\varphi \wedge F \equiv F \qquad \varphi \wedge T \equiv \varphi$$
$$\varphi \vee T \equiv T \qquad \varphi \vee F \equiv \varphi$$

$$\neg T \equiv F \qquad \neg F \equiv T$$

I.Calcul propositionnel

II.Logique des prédicats

1.Syntaxe

III.Sémantique

IV.Exercices

On veut maintenant exprimer des assertions qui vont dépendre d'un certain environnement.

« Il y a des entiers qui ne sont pas des carrés parfaits ».

« Tout nombre réel peut être approché par un nombre rationnel à n'importe quelle précision fixée ».

Le lien avec cet environnement se fera par l'intermédiaire de la notion de *variable* qui sera quantifiée par un *quantificateur universel* ou par un *quantificateur existentiel*.

I.Calcul propositionnel

II.Logique des prédicats

1.Syntaxe

III.Sémantique

IV.Exercices

On veut maintenant exprimer des assertions qui vont dépendre d'un certain environnement.

≪ Il y a des entiers qui ne sont pas des carrés parfaits ≫.

≪ Tout nombre réel peut être approché par un nombre rationnel à n'importe quelle précision fixée ≫.

Le lien avec cet environnement se fera par l'intermédiaire de la notion de *variable* qui sera quantifiée par un *quantificateur universel* ou par un *quantificateur existentiel*.

I.Calcul propositionnel

II.Logique des prédicats

1.Syntaxe

III.Sémantique

IV.Exercices

On veut maintenant exprimer des assertions qui vont dépendre d'un certain environnement.
« Il y a des entiers qui ne sont pas des carrés parfaits ».
« Tout nombre réel peut être approché par un nombre rationnel à n'importe quelle précision fixée ».
Le lien avec cet environnement se fera par l'intermédiaire de la notion de *variable* qui sera quantifiée par un *quantificateur universel* ou par un *quantificateur existentiel*.

Langage

Définition

Un *langage* des prédicats $\mathcal{L}$ est défini par son alphabet :

- $\neg, \wedge, \vee, \rightarrow$ (connecteurs logiques)
- T, F (constantes logiques)
- $($ $)$, (parenthèses et virgule)
- $\forall$ (quantificateur universel)
- $\exists$ (quantificateur existentiel)
- $\mathcal{X}$ un ensemble dénombrable de symboles de *variables*
- $\mathcal{C}$ un ensemble dénombrable de symboles de *constantes*
- $\mathcal{F}$ un ensemble dénombrable de symboles de *fonctions*, si $f \in \mathcal{F}$ alors le nombre d'arguments de f sera fixé
- $\mathcal{R}$ un ensemble dénombrable de symboles de *relations*, si $R \in \mathcal{R}$ alors le nombre d'arguments de R sera fixé

$\mathcal{C}, \mathcal{F}, \mathcal{R}$ forment la *signature* de $\mathcal{L}$.

Langage

Définition

Un *langage* des prédicats $\mathcal{L}$ est défini par son alphabet :

- $\neg, \wedge, \vee, \rightarrow$ (connecteurs logiques)
- T, F (constantes logiques)
- $($ $)$, (parenthèses et virgule)
- $\forall$ (quantificateur universel)
- $\exists$ (quantificateur existentiel)
- $\mathcal{X}$ un ensemble dénombrable de symboles de *variables*
- $\mathcal{C}$ un ensemble dénombrable de symboles de *constantes*
- $\mathcal{F}$ un ensemble dénombrable de symboles de *fonctions*, si $f \in \mathcal{F}$ alors le nombre d'arguments de f sera fixé
- $\mathcal{R}$ un ensemble dénombrable de symboles de *relations*, si $R \in \mathcal{R}$ alors le nombre d'arguments de R sera fixé

$\mathcal{C}, \mathcal{F}, \mathcal{R}$ forment la *signature* de $\mathcal{L}$.

Langage

Définition

Un *langage* des prédicats $\mathcal{L}$ est défini par son alphabet :

- $\neg, \wedge, \vee, \rightarrow$ (connecteurs logiques)
- T, F (constantes logiques)
- ( ) , (parenthèses et virgule)
- $\forall$ (quantificateur universel)
- $\exists$ (quantificateur existentiel)
- $\mathcal{X}$ un ensemble dénombrable de symboles de *variables*
- $\mathcal{C}$ un ensemble dénombrable de symboles de *constantes*
- $\mathcal{F}$ un ensemble dénombrable de symboles de *fonctions*, si $f \in \mathcal{F}$ alors le nombre d'arguments de f sera fixé
- $\mathcal{R}$ un ensemble dénombrable de symboles de *relations*, si $R \in \mathcal{R}$ alors le nombre d'arguments de R sera fixé

$\mathcal{C}, \mathcal{F}, \mathcal{R}$ forment la *signature* de $\mathcal{L}$.

Langage

Définition

Un *langage* des prédicats $\mathcal{L}$ est défini par son alphabet :

- $\neg, \wedge, \vee, \rightarrow$ (connecteurs logiques)
- T, F (constantes logiques)
- ( ) , (parenthèses et virgule)
- $\forall$ (quantificateur universel)
- $\exists$ (quantificateur existentiel)
- $\mathcal{X}$ un ensemble dénombrable de symboles de *variables*
- $\mathcal{C}$ un ensemble dénombrable de symboles de *constantes*
- $\mathcal{F}$ un ensemble dénombrable de symboles de *fonctions*, si $f \in \mathcal{F}$ alors le nombre d'arguments de f sera fixé
- $\mathcal{R}$ un ensemble dénombrable de symboles de *relations*, si $R \in \mathcal{R}$ alors le nombre d'arguments de R sera fixé

$\mathcal{C}, \mathcal{F}, \mathcal{R}$ forment la *signature* de $\mathcal{L}$.

Langage

Définition

Un *langage* des prédicats $\mathcal{L}$ est défini par son alphabet :

- $\neg, \wedge, \vee, \rightarrow$ (connecteurs logiques)
- T, F (constantes logiques)
- $(\ )\ ,$ (parenthèses et virgule)
- $\forall$ (quantificateur universel)
- $\exists$ (quantificateur existentiel)
- $\mathcal{X}$ un ensemble dénombrable de symboles de *variables*
- $\mathcal{C}$ un ensemble dénombrable de symboles de *constantes*
- $\mathcal{F}$ un ensemble dénombrable de symboles de *fonctions*, si $f \in \mathcal{F}$ alors le nombre d'arguments de f sera fixé
- $\mathcal{R}$ un ensemble dénombrable de symboles de *relations*, si $R \in \mathcal{R}$ alors le nombre d'arguments de R sera fixé

$\mathcal{C}, \mathcal{F}, \mathcal{R}$ forment la *signature* de $\mathcal{L}$.

Langage

Définition

Un *langage* des prédicats $\mathcal{L}$ est défini par son alphabet :

- $\neg, \wedge, \vee, \rightarrow$ (connecteurs logiques)
- T, F (constantes logiques)
- ( ) , (parenthèses et virgule)
- $\forall$ (quantificateur universel)
- $\exists$ (quantificateur existentiel)
- $\mathcal{X}$ un ensemble dénombrable de symboles de *variables*
- $\mathcal{C}$ un ensemble dénombrable de symboles de *constantes*
- $\mathcal{F}$ un ensemble dénombrable de symboles de *fonctions*, si $f \in \mathcal{F}$ alors le nombre d'arguments de f sera fixé
- $\mathcal{R}$ un ensemble dénombrable de symboles de *relations*, si $R \in \mathcal{R}$ alors le nombre d'arguments de R sera fixé

$\mathcal{C}, \mathcal{F}, \mathcal{R}$ forment la *signature* de $\mathcal{L}$.

Langage

Définition

Un *langage* des prédicats $\mathcal{L}$ est défini par son alphabet :

- $\neg, \wedge, \vee, \rightarrow$ (connecteurs logiques)
- T, F (constantes logiques)
- $(\ )\ ,$ (parenthèses et virgule)
- $\forall$ (quantificateur universel)
- $\exists$ (quantificateur existentiel)
- $\mathcal{X}$ un ensemble dénombrable de symboles de *variables*
- $\mathcal{C}$ un ensemble dénombrable de symboles de *constantes*
- $\mathcal{F}$ un ensemble dénombrable de symboles de *fonctions*, si $f \in \mathcal{F}$ alors le nombre d'arguments de f sera fixé
- $\mathcal{R}$ un ensemble dénombrable de symboles de *relations*, si $R \in \mathcal{R}$ alors le nombre d'arguments de R sera fixé

$\mathcal{C}, \mathcal{F}, \mathcal{R}$ forment la *signature* de $\mathcal{L}$.

Langage

Définition

Un *langage* des prédicats $\mathcal{L}$ est défini par son alphabet :

- $\neg, \wedge, \vee, \rightarrow$ (connecteurs logiques)
- T, F (constantes logiques)
- $(\ )\ ,$ (parenthèses et virgule)
- $\forall$ (quantificateur universel)
- $\exists$ (quantificateur existentiel)
- $\mathcal{X}$ un ensemble dénombrable de symboles de *variables*
- $\mathcal{C}$ un ensemble dénombrable de symboles de *constantes*
- $\mathcal{F}$ un ensemble dénombrable de symboles de *fonctions*, si $f \in \mathcal{F}$ alors le nombre d'arguments de f sera fixé
- $\mathcal{R}$ un ensemble dénombrable de symboles de *relations*, si $R \in \mathcal{R}$ alors le nombre d'arguments de R sera fixé

$\mathcal{C}, \mathcal{F}, \mathcal{R}$ forment la *signature* de $\mathcal{L}$.

Langage

Définition

Un *langage* des prédicats $\mathcal{L}$ est défini par son alphabet :

- $\neg, \wedge, \vee, \rightarrow$ (connecteurs logiques)
- T, F (constantes logiques)
- $(\ )\ ,$ (parenthèses et virgule)
- $\forall$ (quantificateur universel)
- $\exists$ (quantificateur existentiel)
- $\mathcal{X}$ un ensemble dénombrable de symboles de *variables*
- $\mathcal{C}$ un ensemble dénombrable de symboles de *constantes*
- $\mathcal{F}$ un ensemble dénombrable de symboles de *fonctions*, si $f \in \mathcal{F}$ alors le nombre d'arguments de f sera fixé
- $\mathcal{R}$ un ensemble dénombrable de symboles de *relations*, si $R \in \mathcal{R}$ alors le nombre d'arguments de R sera fixé

$\mathcal{C}, \mathcal{F}, \mathcal{R}$ forment la *signature* de $\mathcal{L}$.

EXEMPLE : $\mathcal{F} = \{f, g, h\}$ avec $a(f) = a(g) = 2$, $a(h) = 1$.
$\mathcal{R} = \{R, P\}$ avec $a(R) = 1$ et $a(P) = 2$.
$\mathcal{C} = \{a, b\}$.

Termes

Définition

L'ensemble des termes définis sur le langage $\mathcal{L}$ est donné récursivement par

- si $x \in \mathcal{X}$ alors x est un terme
- si $c \in \mathcal{C}$ alors c est un terme
- si $t_1, \cdots, t_k$ sont des termes et si $f \in \mathcal{F}$ a k arguments alors $f(t_1 \cdots t_k)$ est un terme

EXEMPLE : $\mathcal{F} = \{f, g, h\}$ avec $a(f) = a(g) = 2$, $a(h) = 1$.
$\mathcal{R} = \{R, P\}$ avec $a(R) = 1$ et $a(P) = 2$.
$\mathcal{C} = \{a, b\}$.
$f(h(a), x)$, $h(g(x, y))$ sont des termes.

Termes

Définition

L'ensemble des termes définis sur le langage $\mathcal{L}$ est donné récursivement par

- si $x \in \mathcal{X}$ alors x est un terme
- si $c \in \mathcal{C}$ alors c est un terme
- si $t_1, \cdots, t_k$ sont des termes et si $f \in \mathcal{F}$ a k arguments alors $f(t_1 \cdots t_k)$ est un terme

EXEMPLE : $\mathcal{F} = \{f, g, h\}$ avec $a(f) = a(g) = 2$, $a(h) = 1$.
$\mathcal{R} = \{R, P\}$ avec $a(R) = 1$ et $a(P) = 2$.
$\mathcal{C} = \{a, b\}$.
$f(h(a), x)$, $h(g(x, y))$ sont des termes.

Termes

Définition

L'ensemble des termes définis sur le langage $\mathcal{L}$ est donné récursivement par

- si $x \in \mathcal{X}$ alors x est un terme
- si $c \in \mathcal{C}$ alors c est un terme
- si $t_1, \cdots, t_k$ sont des termes et si $f \in \mathcal{F}$ a k arguments alors $f(t_1 \cdots t_k)$ est un terme

EXEMPLE : $\mathcal{F} = \{f, g, h\}$ avec $a(f) = a(g) = 2$, $a(h) = 1$.
$\mathcal{R} = \{R, P\}$ avec $a(R) = 1$ et $a(P) = 2$.
$\mathcal{C} = \{a, b\}$.
$f(h(a), x)$, $h(g(x, y))$ sont des termes.

Termes

Définition

L'ensemble des termes définis sur le langage $\mathcal{L}$ est donné récursivement par

- si $x \in \mathcal{X}$ alors x est un terme
- si $c \in \mathcal{C}$ alors c est un terme
- si $t_1, \cdots, t_k$ sont des termes et si $f \in \mathcal{F}$ a k arguments alors $f(t_1 \cdots t_k)$ est un terme

EXEMPLE : $\mathcal{F} = \{f, g, h\}$ avec $a(f) = a(g) = 2$, $a(h) = 1$.
$\mathcal{R} = \{R, P\}$ avec $a(R) = 1$ et $a(P) = 2$.
$\mathcal{C} = \{a, b\}$.
$f(h(a), x)$, $h(g(x, y))$ sont des termes.

Termes

Définition

L'ensemble des termes définis sur le langage $\mathcal{L}$ est donné récursivement par

- si $x \in \mathcal{X}$ alors x est un terme
- si $c \in \mathcal{C}$ alors c est un terme
- si $t_1, \cdots, t_k$ sont des termes et si $f \in \mathcal{F}$ a k arguments alors $f(t_1 \cdots t_k)$ est un terme

EXEMPLE : $\mathcal{F} = \{f, g, h\}$ avec $a(f) = a(g) = 2$, $a(h) = 1$.
$\mathcal{R} = \{R, P\}$ avec $a(R) = 1$ et $a(P) = 2$.
$\mathcal{C} = \{a, b\}$.
$f(h(a), x)$, $h(g(x, y))$ sont des termes.

Formules atomiques

Définition

L'ensemble des formules atomiques est défini par
si $t_1, \cdots, t_n$ sont des termes et si $R \in \mathcal{R}$ a n arguments alors $Rt_1 \cdots t_n$ est une formule atomique.

EXEMPLE : $\mathcal{F} = \{f, g, h\}$ avec $a(f) = a(g) = 2$, $a(h) = 1$.
$\mathcal{R} = \{R, P\}$ avec $a(R) = 1$ et $a(P) = 2$.
$\mathcal{C} = \{a, b\}$.
$R(f(h(a), x))$, $P(h(g(x, y)), f(y, b))$ sont des formules atomiques.

Formules atomiques

Définition

L'ensemble des formules atomiques est défini par
si $t_1, \cdots, t_n$ sont des termes et si $R \in \mathcal{R}$ a n arguments alors $Rt_1 \cdots t_n$ est une formule atomique.

EXEMPLE : $\mathcal{F} = \{f, g, h\}$ avec $a(f) = a(g) = 2$, $a(h) = 1$.
$\mathcal{R} = \{R, P\}$ avec $a(R) = 1$ et $a(P) = 2$.
$\mathcal{C} = \{a, b\}$.
$R(f(h(a), x))$, $P(h(g(x, y)), f(y, b))$ sont des formules atomiques.

Formules

Définition

L'ensemble des formules sur le langage $\mathcal{L}$ est défini récursivement par

- toute formule atomique est une formule
- T, F sont des formules
- si φ et ψ sont des formules alors $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \to \psi$, $\varphi \longleftrightarrow \psi$ sont des formules.
- si φ est une formule, si x est un symbole de variables alors $\exists x(\varphi)$ et $\forall x(\varphi)$ sont des formules et on dit que φ est la portée du quantificateur.

Formules

Définition

L'ensemble des formules sur le langage $\mathcal{L}$ est défini récursivement par

- toute formule atomique est une formule
- T, F sont des formules
- si φ et ψ sont des formules alors $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \to \psi$, $\varphi \longleftrightarrow \psi$ sont des formules.
- si φ est une formule, si x est un symbole de variables alors $\exists x(\varphi)$ et $\forall x(\varphi)$ sont des formules et on dit que φ est la portée du quantificateur.

Formules

Définition

L'ensemble des formules sur le langage $\mathcal{L}$ est défini récursivement par

- toute formule atomique est une formule
- T, F sont des formules
- si φ et ψ sont des formules alors $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \to \psi$, $\varphi \longleftrightarrow \psi$ sont des formules.
- si φ est une formule, si x est un symbole de variables alors $\exists x(\varphi)$ et $\forall x(\varphi)$ sont des formules et on dit que φ est la portée du quantificateur.

Formules

Définition

L'ensemble des formules sur le langage $\mathcal{L}$ est défini récursivement par

- toute formule atomique est une formule
- T, F sont des formules
- si φ et ψ sont des formules alors $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \to \psi$, $\varphi \longleftrightarrow \psi$ sont des formules.
- si φ est une formule, si x est un symbole de variables alors $\exists x(\varphi)$ et $\forall x(\varphi)$ sont des formules et on dit que φ est la portée du quantificateur.

EXEMPLE : $\mathcal{F} = \{f, g, h\}$ avec $a(f) = a(g) = 2$, $a(h) = 1$.
$\mathcal{R} = \{R, P\}$ avec $a(R) = 1$ et $a(P) = 2$.
$\mathcal{C} = \{a, b\}$.
$\exists x(R(h(x)))$, $\forall y(R(g(b, y)) \rightarrow \exists x P(x, y))$ sont des formules

REMARQUE : on ne verra que des formules dans lesquelles tous les symboles de variables sont dans la portée d'un quantificateur.

Pas de formules du genre $\exists y P(x, y)$

EXEMPLE : la formule $\exists y \forall x P(x, y)$ est-elle vraie ou fausse ? Tout dépend de la signification donnée au symbole P et du domaine sur lequel cette interprétation sera valable.

Si P est la relation $\leq$ sur l'ensemble $\mathbb{N}$, alors la formule est vraie, on peut trouver un entier naturel – 0 – tel qu'il soit inférieur à tout entier naturel.

Mais si P est la relation $\leq$ sur l'ensemble $\mathbb{Z}$, alors la formule est fausse car aucun entier relatif ne peut être inférieur à tout entier relatif.

EXEMPLE : la formule $\exists y \forall x P(x, y)$ est-elle vraie ou fausse ? Tout dépend de la signification donnée au symbole P et du domaine sur lequel cette interprétation sera valable.

Si P est la relation $\leq$ sur l'ensemble $\mathbb{N}$, alors la formule est vraie, on peut trouver un entier naturel – 0 – tel qu'il soit inférieur à tout entier naturel.

Mais si P est la relation $\leq$ sur l'ensemble $\mathbb{Z}$, alors la formule est fausse car aucun entier relatif ne peut être inférieur à tout entier relatif.

EXEMPLE : la formule $\exists y \forall x P(x, y)$ est-elle vraie ou fausse ? Tout dépend de la signification donnée au symbole P et du domaine sur lequel cette interprétation sera valable.

Si P est la relation $\leq$ sur l'ensemble $\mathbb{N}$, alors la formule est vraie, on peut trouver un entier naturel – 0 – tel qu'il soit inférieur à tout entier naturel.

Mais si P est la relation $\leq$ sur l'ensemble $\mathbb{Z}$, alors la formule est fausse car aucun entier relatif ne peut être inférieur à tout entier relatif.

Structure et interprétation d'un langage

Définition

Si $\mathcal{C}, \mathcal{F}, \mathcal{R}$ forment la signature du langage $\mathcal{L}$, une interprétation de $\mathcal{L}$ est réalisée par une structure $\mathcal{U}$ composée de

- un ensemble non vide $\mathcal{D}$ appelé *domaine*
- une application de $\mathcal{C}$ dans $\mathcal{D}$ qui associe à chaque symbole de constante c son interprétation $c^{\mathcal{U}}$
- pour chaque symbole de fonction f de $\mathcal{F}$, une application de $\mathcal{D}^k$ dans $\mathcal{D}$ noté $f^{\mathcal{U}}$ où $a(f) = k$
- pour chaque symbole de relation R de $\mathcal{R}$, une relation sur $\mathcal{D}^n$ noté $R^{\mathcal{U}}$ où $a(R) = n$

Structure et interprétation d'un langage

Définition

Si $\mathcal{C}, \mathcal{F}, \mathcal{R}$ forment la signature du langage $\mathcal{L}$, une interprétation de $\mathcal{L}$ est réalisée par une structure $\mathcal{U}$ composée de

- un ensemble non vide $\mathcal{D}$ appelé *domaine*
- une application de $\mathcal{C}$ dans $\mathcal{D}$ qui associe à chaque symbole de constante c son interprétation $c^{\mathcal{U}}$
- pour chaque symbole de fonction f de $\mathcal{F}$, une application de $\mathcal{D}^k$ dans $\mathcal{D}$ noté $f^{\mathcal{U}}$ où $a(f) = k$
- pour chaque symbole de relation R de $\mathcal{R}$, une relation sur $\mathcal{D}^n$ noté $R^{\mathcal{U}}$ où $a(R) = n$

Structure et interprétation d'un langage

Définition

Si $\mathcal{C}, \mathcal{F}, \mathcal{R}$ forment la signature du langage $\mathcal{L}$, une interprétation de $\mathcal{L}$ est réalisée par une structure $\mathcal{U}$ composée de

- un ensemble non vide $\mathcal{D}$ appelé *domaine*
- une application de $\mathcal{C}$ dans $\mathcal{D}$ qui associe à chaque symbole de constante c son interprétation $c^{\mathcal{U}}$
- pour chaque symbole de fonction f de $\mathcal{F}$, une application de $\mathcal{D}^k$ dans $\mathcal{D}$ noté $f^{\mathcal{U}}$ où $a(f) = k$
- pour chaque symbole de relation R de $\mathcal{R}$, une relation sur $\mathcal{D}^n$ noté $R^{\mathcal{U}}$ où $a(R) = n$

Structure et interprétation d'un langage

Définition

Si $\mathcal{C}, \mathcal{F}, \mathcal{R}$ forment la signature du langage $\mathcal{L}$, une interprétation de $\mathcal{L}$ est réalisée par une structure $\mathcal{U}$ composée de

- un ensemble non vide $\mathcal{D}$ appelé *domaine*
- une application de $\mathcal{C}$ dans $\mathcal{D}$ qui associe à chaque symbole de constante c son interprétation $c^{\mathcal{U}}$
- pour chaque symbole de fonction f de $\mathcal{F}$, une application de $\mathcal{D}^k$ dans $\mathcal{D}$ noté $f^{\mathcal{U}}$ où $a(f) = k$
- pour chaque symbole de relation R de $\mathcal{R}$, une relation sur $\mathcal{D}^n$ noté $R^{\mathcal{U}}$ où $a(R) = n$

Valeur d'une formule

Définition

Soit $\mathcal{U}$ une interprétation de $\mathcal{L}$. Les symboles de constantes, fonctions et relations sont interprétés selon $\mathcal{U}$.

- $\exists x(\varphi)$ est satisfaite par $\mathcal{U}$ si il existe une valeur $d \in \mathcal{D}$ telle $\varphi_{[x \leftarrow d]}$ est vrai
- $\forall x(\varphi)$ est satisfaite par $\mathcal{U}$ si pour toute valeur $d \in \mathcal{D}$, $\varphi_{[x \leftarrow d]}$ est vraie

Valeur d'une formule

Définition

Soit $\mathcal{U}$ une interprétation de $\mathcal{L}$. Les symboles de constantes, fonctions et relations sont interprétés selon $\mathcal{U}$.

- $\exists x(\varphi)$ est satisfaite par $\mathcal{U}$ si il existe une valeur $d \in \mathcal{D}$ telle $\varphi_{[x \leftarrow d]}$ est vrai
- $\forall x(\varphi)$ est satisfaite par $\mathcal{U}$ si pour toute valeur $d \in \mathcal{D}$, $\varphi_{[x \leftarrow d]}$ est vraie

Exemple

$\mathcal{F} = \{f, g, h\}$ avec $a(f) = a(g) = 2$, $a(h) = 1$. $\mathcal{R} = \{R, P\}$ avec $a(R) = 1$ et $a(P) = 2$. $\mathcal{C} = \{a, b\}$.
$\mathcal{U} : \mathcal{D} = \mathbb{N}$, $f = +, g = \times, h =$ *successeur*, $R =\ll$ être premier $\gg$, $P = \leq$, $a = 0, b = 1$.

$\exists x(R(h(x)))$ est vrai : $x \leftarrow 2$ convient puisque *successeur*$(2) = 3$ est premier.

$\forall x(R(h(x)))$ est faux : $x \leftarrow 3$ ne satisfait pas *successeur*(3) est premier.

Exemple

$\mathcal{F} = \{f, g, h\}$ avec $a(f) = a(g) = 2$, $a(h) = 1$. $\mathcal{R} = \{R, P\}$ avec $a(R) = 1$ et $a(P) = 2$. $\mathcal{C} = \{a, b\}$.
$\mathcal{U} : \mathcal{D} = \mathbb{N}$, $f = +, g = \times, h =$ *successeur*, $R =\ll$ être premier $\gg$, $P = \leq$, $a = 0, b = 1$.

$\exists x(R(h(x)))$ est vrai : $x \leftarrow 2$ convient puisque *successeur*$(2) = 3$ est premier.

$\forall x(R(h(x)))$ est faux : $x \leftarrow 3$ ne satisfait pas *successeur*(3) est premier.

Exemple

$\mathcal{F} = \{f, g, h\}$ avec $a(f) = a(g) = 2$, $a(h) = 1$. $\mathcal{R} = \{R, P\}$ avec $a(R) = 1$ et $a(P) = 2$. $\mathcal{C} = \{a, b\}$.
$\mathcal{U} : \mathcal{D} = \mathbb{N}$, $f = +, g = \times, h =$ *successeur*, $R =\ll$ être premier $\gg$, $P = \leq$, $a = 0, b = 1$.

$\exists x(R(h(x)))$ est vrai : $x \leftarrow 2$ convient puisque *successeur*$(2) = 3$ est premier.

$\forall x(R(h(x)))$ est faux : $x \leftarrow 3$ ne satisfait pas *successeur*(3) est premier.

Exemple

$\mathcal{F} = \{f, g, h\}$ avec $a(f) = a(g) = 2$, $a(h) = 1$. $\mathcal{R} = \{R, P\}$ avec $a(R) = 1$ et $a(P) = 2$. $\mathcal{C} = \{a, b\}$.
$\mathcal{U} : \mathcal{D} = \mathbb{N}$, $f = +, g = \times, h =$ *successeur*, $R =\ll$ être premier $\gg$, $P = \leq$, $a = 0, b = 1$.

$\exists x(R(h(x)))$ est vrai : $x \leftarrow 2$ convient puisque *successeur*$(2) = 3$ est premier.

$\forall x(R(h(x)))$ est faux : $x \leftarrow 3$ ne satisfait pas *successeur*(3) est premier.

Exemple

$\mathcal{F} = \{f, g, h\}$ avec $a(f) = a(g) = 2$, $a(h) = 1$. $\mathcal{R} = \{R, P\}$ avec $a(R) = 1$ et $a(P) = 2$. $\mathcal{C} = \{a, b\}$.
$\mathcal{U} : \mathcal{D} = \mathbb{N}$, $f = +, g = \times, h =$ *successeur*, $R =\ll$ être premier $\gg$, $P = \leq$, $a = 0, b = 1$.

$\exists x(R(h(x)))$ est vrai : $x \leftarrow 2$ convient puisque *successeur*$(2) = 3$ est premier.

$\forall x(R(h(x)))$ est faux : $x \leftarrow 3$ ne satisfait pas *successeur*(3) est premier.

Equivalence

Définition

Deux formules φ et ψ sont équivalentes si pour toute structure $\mathcal{U}$ si $\mathcal{U}$ satisfait φ alors $\mathcal{U}$ satisfait ψ et si $\mathcal{U}$ satisfait ψ alors $\mathcal{U}$ satisfait φ.

EXEMPLE : $\forall x \forall y P(x, y) \equiv \forall y \forall x P(x, y)$

Quelques équivalences

$\neg\forall x\varphi \equiv \exists x\neg\varphi$ $\qquad$ $\neg\exists x\varphi \equiv \forall x\neg\varphi$

$\forall x(\varphi \wedge \psi) \equiv \forall x\varphi \wedge \forall x\psi$
$\exists x(\varphi \vee \psi) \equiv \exists x\varphi \vee \exists x\psi$
$\exists x(\varphi \to \psi) \equiv \forall x\varphi \to \exists x\psi$

$\forall x\forall y\varphi \equiv \forall y\forall x\varphi$ $\qquad$ $\exists x\exists y\varphi \equiv \exists y\exists x\varphi$

Les trois formules suivantes sont universellement valides (c'est-à-dire elles sont vraies pour toute interprétation)

$\exists x(\varphi \wedge \psi) \to (\exists x\varphi \wedge \exists x\psi)$
$\forall x\varphi \vee \forall x\psi \to \forall x(\varphi \vee \psi)$
$\exists x\forall y\varphi \to \forall y\exists x\varphi$

Attention

Les réciproques sont fausses.

Pour chaque formule trouvez un langage et une structure qui **ne satisfait pas**

$(\exists x\varphi \wedge \exists x\psi) \to \exists x(\varphi \wedge \psi)$

$\forall x(\varphi \vee \psi) \to \forall x\varphi \vee \forall x\psi$

$\forall y\exists x\varphi \to \exists x\forall y\varphi$

Attention

Les réciproques sont fausses.
Pour chaque formule trouvez un langage et une structure qui **ne satisfait pas**

$(\exists x\varphi \wedge \exists x\psi) \rightarrow \exists x(\varphi \wedge \psi)$

$\forall x(\varphi \vee \psi) \rightarrow \forall x\varphi \vee \forall x\psi$

$\forall y\exists x\varphi \rightarrow \exists x\forall y\varphi$

Attention

Les réciproques sont fausses.
Pour chaque formule trouvez un langage et une structure qui **ne satisfait pas**

$(\exists x\varphi \wedge \exists x\psi) \rightarrow \exists x(\varphi \wedge \psi)$

$\forall x(\varphi \vee \psi) \rightarrow \forall x\varphi \vee \forall x\psi$

$\forall y\exists x\varphi \rightarrow \exists x\forall y\varphi$

Attention

Les réciproques sont fausses.
Pour chaque formule trouvez un langage et une structure qui **ne satisfait pas**
$(\exists x\varphi \wedge \exists x\psi) \rightarrow \exists x(\varphi \wedge \psi)$
$\forall x(\varphi \vee \psi) \rightarrow \forall x\varphi \vee \forall x\psi$
$\forall y\exists x\varphi \rightarrow \exists x\forall y\varphi$

I.Calcul propositionnel

II.Logique des prédicats

III.Sémantique

IV.Exercices

I.Calcul propositionnel

II.Logique des prédicats

III.Sémantique

IV.Exercices

Un coffre-fort est muni de n serrures et peut être ouvert uniquement lorsque ses n serrures sont simultanément ouvertes. Cinq personnes a, b, c, d, e doivent recevoir des clefs correspondant à certaines de ces serrures. chaque clef peut être disponible en autant d'exemplaires que l'on souhaite. On demande de choisir pour l'entier n la plus petite valeur possible et de lui associer une répartition des clefs parmi les 5 personnes de telle manière que le coffre puisse être ouvert si et seulement si on se trouve dans au moins une des situations suivantes :

- présence simultanée de a et b,
- présence simultanée de a, c et d,
- présence simultanée de b, d et e.

Mai 2009

Devant vous se trouvent un soldat gardant trois coffres. Vous savez que **au moins un des trois coffres** contient un diamant, que les autres contiennent des cailloux et que le soldat dit la vérité et affirme :

- « si le coffre rouge contient un diamant alors le coffre vert contient des cailloux »
- « si le coffre rouge contient des cailloux alors le coffre bleu contient un diamant »
- « si le coffre rouge contient un diamant alors le coffre vert ou le coffre bleu aussi »

Dans quel coffre est-on sûr de trouver un diamant ?

Pour chaque formule trouvez un langage et une structure qui prouvent que les paires de formules suivantes ne sont pas équivalentes

$\alpha = \forall x(\varphi \vee \psi), \beta = \forall x\varphi \vee \forall x\psi$

$\alpha = \exists x(\varphi \wedge \psi), \beta = \exists x\varphi \wedge \exists x\psi$

Pour chaque formule trouvez un langage et une structure qui prouvent que les paires de formules suivantes ne sont pas équivalentes

$\alpha = \forall x(\varphi \vee \psi), \beta = \forall x\varphi \vee \forall x\psi$

$\alpha = \exists x(\varphi \wedge \psi), \beta = \exists x\varphi \wedge \exists x\psi$

Soit $\mathcal{D} = \{n \in \mathbb{N} \, ; n \geq 2\}$ et M la relation "multiple" c'est-à-dire $M(k, n)$ est vrai ssi n est un multiple de k. Cette structure satisfait-elle chacune des trois formules suivantes :

- $\phi_1 : \forall x \forall y \ (M(y, x) \rightarrow x = y)$
- $\phi_2 : \exists x \forall y \ (M(y, x) \rightarrow x = y)$
- $\phi_3 : \forall x \forall y \forall z \ ((M(y, x) \wedge M(z, x)) \rightarrow (M(y, z) \vee M(z, y)))$

Soient f, g deux symboles de fonctions à un argument et $=$ un symbole de relation binaire qui sera interprété comme l'égalité.

- $\psi_1 : \forall x\ f(x) = g(x)$
- $\psi_2 : \forall x \exists y\ f(y) = g(x)$
- $\psi_3 : \forall x \exists y\ f(x) = g(y)$
- $\psi_4 : \exists x \forall y\ f(y) = g(x)$
- $\psi_5 : \exists x \forall y\ f(x) = g(y)$
- $\psi_6 : \exists x \exists y\ f(x) = g(x)$

On fixe le domaine à $\mathbb{N}$

1. Pour chaque formule, trouver une structure qui la satisfasse.
2. Trouver une structure qui satisfait $\psi_4 \wedge \psi_5$
3. Trouver une structure qui satisfait $\neg\psi_1 \wedge \psi_2 \wedge \neg\psi_3$

Trouver un langage pour exprimer les propriétés suivantes et écrire les formules les traduisant :

- il y a des Anglais qui n'ont pas le pied marin
- tout être humain a un père et une mère
- chaque homme ne peut avoir qu'une épouse

Trouver un langage pour exprimer la phrase suivante en une formule :
You can fool some of the people all of the time, and all of the people some of the time, but you can not fool all of the people all of the time.

Mai 2009

On fixe comme domaine l'ensemble des plantes.
Soient

- F un symbole de relation à un argument qui sera interprété ainsi :
  $F(x)$ signifie « x est une fleur »,
- A un symbole de relation à un argument qui sera interprété ainsi :
  $A(x)$ signifie « x est un arbre »,
- E un symbole de relation à deux arguments qui sera interprété ainsi :
  $E(x, y)$ signifie « x pousse sur y »

1. Ecrire une formule de ce langage qui exprime que toute fleur ne pousse pas nécessairement sur un arbre.
2. Que signifie la formule suivante : $\exists x(\neg F(x) \wedge \neg A(x))$

Mai 2009 (suite)

Trouver le langage et les formules pour exprimer

- tout lion est carnivore (c'est-à-dire mange de la viande)
- il existe des poissons qui ne sont pas carnivores
- deux animaux carnivores ne sont pas nécessairement de la même espèce

I.Calcul propositionnel

II.Logique des prédicats

III.Sémantique

IV.Exercices

I.Calcul propositionnel

II.Logique des prédicats

III.Sémantique

IV.Exercices