Codificação por entropia - Códigos de Huffman

Departamento de Eletrônica e Computação

Centro de Tecnologia

ELC1120 – TELECOMUNICAÇÕES II

Profa. Candice Müller   Prof. Fernando DeCastro

Entropia – uma possível medida de informação

variável aleatória
x
variável determinística

- **A observação da ocorrência de um evento do espaço amostral de uma variável aleatória nos dá <u>informação:</u>**

- **Eventos raros contêm mais informação do que eventos comuns:**

– "O sol nasceu à leste hoje pela manhã":
evento comum → pouca informação;

– "Porto Alegre foi atingida por um terremoto hoje pela manhã":
evento raro → maior conteúdo de informação.

Era da Informação
x
Era dos dados

A Entropia (proposta por Hartley, em 1928) é uma medida logarítmica de informação que reflete este raciocínio intuitivo.

Entropia – uma possível medida de informação

- Ao registrarmos o valor das amostras na saída do quantizador de um codificador que apresente M níveis de quantização, após o registro de um número suficiente de amostras, podemos fazer um estudo estatístico da probabilidade de ocorrência de cada uma das M possíveis amostras (mensagens de $N = log_2 M$ bits);

- A saída do quantizador pode ser considerada uma variável aleatória discreta X, com espaço de amostras definido pelo conjunto $\Omega_X = \{m_k\} = \{m_0, m_1, \dots, m_{M-1}\}$ de M mensagens (M palavras binárias) m_k com probabilidade de ocorrência p_k, $k = 0,1, \dots, M-1$.

- Segundo Hartley, a auto-informação $h(m_k)$ implícita na ocorrência de uma mensagem m_k, com probabilidade de ocorrência p_k, e' definida por:

$$h(m_k) = -log_2(p_k)\ [bits]$$

Entropia – uma possível medida de informação

$$h(m_k) = -log_2(p_k)\ [bits]$$

$$log_2(y) = \frac{log_{10} y}{log_{10} 2}$$

$$log_2(y) = \frac{\ln y}{\ln 2}$$

- A partir da equação da auto-informação, pode-se concluir que:
  - Como $0 \leq p_k \leq 1$, $h(m_k)$ é sempre um número positivo;
  - $h(m_k)$ é medida em bits, devido a função logarítmica de base 2;
  - Como $log_2(u)$ é uma função monotonicamente crescente com u, a auto-informação $h(m_k) = -log_2(p_k)$ de uma mensagem rara é maior do que a de uma mensagem comum.

Auto-informação

	p_k	0.2	0.4	0.6	0.8	1
	$log_2(p_k)$	-2.32	-1.32	-0.74	-0.32	0
$h(m_k) =$	$-log_2(p_k)$	2.32	1.32	0.74	0.32	0

Mensagem rara ← → *Mensagem comum*

Entropia – uma possível medida de informação

A média da Auto-Informação das M mensagens m_k do conjunto $\boldsymbol{\Omega}_X = \{m_0, m_1, \dots, m_{M-1}\}$ é denominada ENTROPIA da variável aleatória X.

(ENTROPIA da variável aleatória $X \equiv$ Entropia do conjunto $\boldsymbol{\Omega}_X$ de mensagens).

Assim, a entropia $H(X)$ da variável aleatória X, cujo espaço de amostras é o conjunto Ω_X de M mensagens, é dada por:

$$H(X) = E\{h(m_k)\} = E\{-\log_2(p_k)\} = -\sum_{k=0}^{M-1} p_k log_2(p_k) \ \ [bits],$$

onde $E\{.\}$ é o operador estatístico que retorna o valor esperado do argumento [Carlson].

Note que, se as M mensagens apresentam probabilidade de ocorrência iguais (mensagens equiprovaveis), então $p_k = 1/M$ para $k = 0,1, \dots, M-1$ e

$$H(X) = -\frac{1}{M}\sum_{k=0}^{M-1} log_2\left(\frac{1}{M}\right) = log_2(M) \ \ [bits]$$

$$Para\ M = 4; \quad \boldsymbol{\Omega}_X = \{m_0, m_1, m_2, m_3\}; \quad p_0 = p_1 = p_2 = p_3 = 0.25;$$

$$H(X) = -\frac{1}{4}\left\{log_2\left(\frac{1}{4}\right)4\right\} = 2\ (= \ log_2 4)\ [bits]$$

Exemplo 1:

Seja um sistema para transmissão digital que utilize no codificador de fonte um conjunto $\Omega_X = \{m_0, m_1\}$ com $M = 2$ possíveis mensagens (ou $M = 2$ possíveis níveis de quantização).

Seja q a probabilidade de ocorrência que a saída X do quantizador assuma valor m_0, isto e', $q = P(X = m_0)$. Determine o gráfico da Entropia de X em função de *q.*

Solução: Para determinar o gráfico da Entropia de X em função de *q*, consideremos que, se

$$q = P(X = m_0) \rightarrow P(X = m_1) = 1 - q.$$

Portanto,

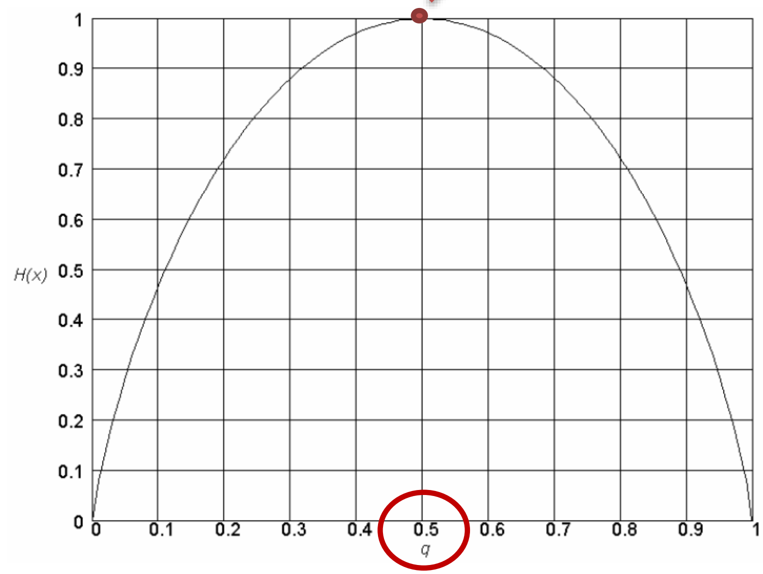
$$H(X) = -\sum_{k=0}^{M-1} p_k log_2(p_k) = -p_0 log_2(p_0) - p_1 log_2(p_1) =$$
$$= -q log_2(q) - (1-q) log_2(1-q) [bits]$$

Entropia – uma possível medida de informação

$$H(X) = -q log_2(q) - (1-q) log_2(1-q)$$

Máxima $H(X)$

$H(X)\ x\ q$



MENSAGENS EQUIPROVÁVEIS
≡
MÁXIMA INCERTEZA!!

Note, pelo gráfico, que $H(X)$ é máxima quando as mensagens m_0 e m_1 têm a mesma probabilidade de ocorrência, ou seja, quando $q = (1-q) = 0.5$.

$$máx\ H(X) = -0.5 log_2(0.5) - (0.5) log_2(0.5) = -0.5(-1) - (0.5)(-1) = 1$$

- Este comportamento acontece não só para um espaço de amostras $\boldsymbol{\Omega}_X$ com apenas $M = 2$ mensagens de probabilidades iguais, mas ocorre também para qualquer quantidade M de mensagens de mesma probabilidade.

- O valor máximo da entropia de uma variável aleatória X é

$$H(X) = log_2\,(M),$$

valor que ocorre quando as probabilidades de ocorrência dos M elementos do espaço de amostras $\boldsymbol{\Omega}_X$ são todas iguais à $1/M$ (i. e., os M elementos de $\boldsymbol{\Omega}_X$ são equiprováveis).

Taxa de transporte da informação

- Seja uma fonte de informação A aplicada à entrada de um codificador.

- Suponhamos que estamos registrando a saída X do quantizador e calculando a entropia $H(X)$.

- Se a fonte é amostrada a uma taxa tal que o quantizador gera $r[mensagens/segundo]$ com uma entropia $H[bits/mensagem]$, então a **Taxa de Informação** R é definida como

$$R = rH\ [bits/s] \qquad \qquad \frac{mensagens}{s} * \frac{bits}{mensagens} = bits/s$$

- A **Taxa de Informação** é uma medida do número médio de bits que precisa ser transportado por segundo através do sistema.

Exemplo 2:

Seja um sistema para transmissão digital que utilize no codificador de fonte um conjunto $\Omega_X = \{m_0, m_1, m_2, m_3\}$ com $M = 4$ possíveis mensagens (ou $M = 4$ níveis de quantização).

As amostras na saída X do quantizador são tais que a ocorrência de uma não altera a probabilidade de ocorrência da outra (i. é, as mensagens são estatisticamente independentes).

As probabilidades de ocorrência são:

$P(X = m_0) = P(X = m_3) = 1/8$ e $P(X = m_1) = P(X = m_2) = 3/8$

O intervalo de amostragem de $m(t)$ é $T_s = \frac{1}{2f_M} = 50\mu s$.

Determine a taxa de informação gerada pelo sinal $m(t)$ na saida X do quantizador.

Solução:

$$H(X) = -\sum_{k=0}^{M-1} p_k log_2(p_k) \left[\frac{bits}{mensagem}\right] \qquad R = rH\ [bits/s]$$

Taxa de transporte da informação

$P(X = m_0) = P(X = m_3) = 1/8$ e $P(X = m_1) = P(X = m_2) = 3/8$

$T_s = \frac{1}{2f_M} = 50\mu s; \quad R = rH\ [bits/s]$

- A informação média gerada pelo sinal fonte $m(t)\ em\ X$ (Entropia) é:

$$H(X) = -\frac{1}{8}log_2\left(\frac{1}{8}\right) - \frac{3}{8}log_2\left(\frac{3}{8}\right) - \frac{3}{8}log_2\left(\frac{3}{8}\right) - \frac{1}{8}log_2\left(\frac{1}{8}\right)=1.8\ [bits/mensagem]$$

- Como o intervalo de amostragem de $m(t)$ é $T_s = \frac{1}{2f_M} = 50\mu s$, são geradas

$$r = \frac{1}{T_s} = 20000\left[\frac{mensagens}{segundo}\right].$$

- Assim, a taxa de informação R será:

$$R = rH = 20000 * 1.8 = 36000\left[\frac{bits}{s}\right]$$

Portanto, este sinal fonte demandará 36kbps para que possa ser transmitido.

- Consideremos que o quantizador de um codificador apresente M níveis de quantização e codifique o sinal $m(t)$ quantizado com sequências de $N = log_2\,(M)$ bits.

- **O código para compressão de dados considera cada uma das M possíveis sequências de N bits como uma mensagem de N bits e associa a cada uma delas uma palavra-código cujo número de bits depende da probabilidade de ocorrência da mensagem.**

probabilidade ↑ bits ↓

- Este critério baseado na probabilidade de ocorrência da mensagem é crucial para a eficiência da compressão. Um código que segue este critério faz com que mensagens que ocorrem frequentemente necessitem de menos bits para serem transmitidas e, portanto, **o efeito global é o de permitir que mais informação possa ser transmitida no mesmo intervalo de tempo.**

- Quando um sistema digital é projetado, é feito um estudo estatístico da probabilidade de ocorrência de cada uma das possíveis mensagens para que o código compressor possa ser especificado. O conjunto de *M* valores obtidos, cuja soma forçosamente tende para 1.0, é uma boa aproximação das probabilidades de ocorrência de cada uma das *M* possíveis mensagens.

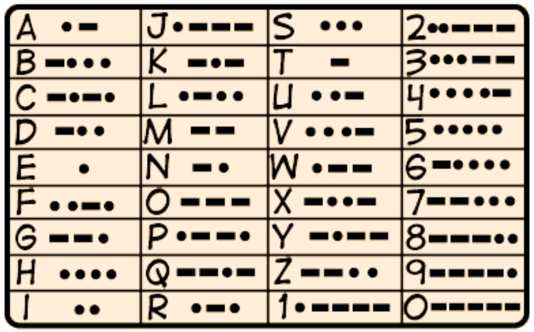
Códigos para compressão com base no princípio

probabilidade ↑ bits ↓

são denominados de processos para **Codificação por Entropia**.

Codificação por entropia

Conforme discutido no CapI das notas de aula, o veterano **Código Morse**, utilizado para enviar informação por telegrafia desde a I Guerra Mundial, é um exemplo histórico desta classe de códigos.



probabilidade ↑ código ↓

A letra “E” é a letra mais frequente na escrita em inglês e é representada por um único “ponto”.

Cada letra do alfabeto A – Z é uma mensagem do Código Morse;

O conjunto de caracteres utilizado para compor as palavras-código do Código Morse é o conjunto $\{\bullet, -\} \Leftrightarrow \{0,1\}$.

A cada mensagem é atribuída uma sequência de “pontos” e/ou “traços” representados em telegrafia por tons audíveis curtos e/ou longos.

O mapeamento é tal que letras mais prováveis na escrita inglesa são associadas a palavras-código curtas e letras menos prováveis são associadas a palavras-código longas.

A Entropia é uma medida do conteúdo de informação associado a uma variável aleatória discreta $\boldsymbol{X}$, com espaço de amostras definido pelo conjunto $\Omega_{\boldsymbol{X}} = \{\boldsymbol{x_i}\} = \{\boldsymbol{x_0}, \boldsymbol{x_1}, \dots, \boldsymbol{x_{M-1}}\}$ de $\boldsymbol{M}$ eventos $\boldsymbol{x_i}$ com probabilidades de ocorrência $\{p_i\},\ i = 0,1, \dots, M-1$.

Quando $\boldsymbol{X}$ é a saída de uma fonte de informação discreta, a entropia $\boldsymbol{H}(\boldsymbol{X})$ da fonte representa a quantidade média de informação emitida pela fonte.

Podemos considerar um código para compressão por entropia como um operador $\theta\{\,.\,\}$, tal que $S = \theta\{\Omega\}$, onde

$\Omega = \{x_i\} = \{x_0, x_1, \dots, x_{M-1}\}$ é o conjunto de M possíveis **mensagens** a serem codificadas e

$S = \{s_i\} = \{s_0, s_1, \dots, s_{M-1}\}$ é o conjunto de M possíveis **palavras-código** ou **símbolos** resultantes da codificação.

O operador $\theta\{\,.\,\}$ efetua um **mapeamento unívoco** entre cada mensagem e respectiva palavra-código, tal que mensagens com maior probabilidade de ocorrência são mapeadas em palavras-código de menor tamanho, e vice-versa.

- O **conjunto de caracteres do código** ou **alfabeto do código** é o conjunto $A = \{a_0, a_1, \ldots, a_{D-1}\}$ composto por D elementos, de cuja composição são formadas cada uma das palavra-código.

- As palavras-código formadas do alfabeto A, as quais constituem o conjunto imagem do mapeamento $\theta\{.\}$, são assumidas serem distintas entre si, caso contrário $\theta\{.\}$ não seria unívoco.

Exemplo 3: Seja o alfabeto $A = \{a_0, a_1, a_2\}$ e o conjunto de mensagens $\Omega = \{x_0, x_1, x_2, x_3\}$. Um possível código $\theta\{.\}$ seria conforme tabela abaixo.

Mensagem	Palavra-código s_i associada a x_i por $s_i = \theta\{x_i\}$
x_0	$a_0 a_1$
x_1	$a_0 a_1 a_2$
x_2	a_0
x_3	a_1

Exemplo 4: Seja o alfabeto $A = \{a_0, a_1, a_2\}$ e o conjunto de mensagens $\Omega = \{x_0, x_1, x_2, x_3\} = \{00,01,10,11\}$ resultante da codificação da saída de um quantizador com 4 níveis de quantização. Um possível código $\theta\{\,.\,\}$ seria

Mensagem	Sequência	Palavra-código s_i associada a x_i por $s_i = \theta\{x_i\}$
x_0	00	$a_0 a_1$
x_1	01	$a_0 a_1 a_2$
x_2	10	a_0
x_3	11	a_1

Obs:

As palavras-código usualmente originam-se de um alfabeto binário $A = \{0,1\} \rightarrow bits$. Para um alfabeto ternário, $A = \{0,1,2\} \rightarrow trits$, etc.

Exemplo 5: Seja o alfabeto $A = \{0,1\}$ e o conjunto de mensagens $\Omega = \{x_0, x_1, x_2, x_3\} = \{00,01,10,11\}$. Um possível código $\theta\{\,.\,\}$ seria:

Mensagem	Sequência	Palavra-código s_i associada a x_i por $s_i = \theta\{x_i\}$
x_0	00	0
x_1	01	010
x_2	10	01
x_3	11	10

O tamanho l_i de uma palavra-código ou símbolo s é definido pelo número de caracteres do alfabeto A utilizado na construção da palavra-código.

Exemplo 6: Seja o código binário ($A = \{0,1\}$) do Exemplo 5. O tamanho l_i de cada palavra-código ou símbolo s_i é

Mensagem	Sequência	Palavra-Código s_i associada a m_i por $s_i = \theta\{m_i\}$	l_i
x_0	00	0	1
x_1	01	010	3
x_2	10	01	2
x_3	11	10	2

O objetivo da **Codificação por Entropia** é encontrar um código $\theta\{\,.\,\}$ que minimize o tamanho médio $\bar{L}$ dos símbolos emitidos pela fonte, a partir do conjunto de M possíveis símbolos $S = \{s_i\} = \{s_0, s_1, \ldots, s_{M-1}\}$, sendo $\bar{L}$ dado por

$$\bar{L} = \sum_{i=0}^{M-1} p_i l_i$$

onde p_i *é* a probabilidade de ocorrência da mensagem x_i e l_i é o tamanho do símbolo s_i, associado à mensagem x_i através do código $\theta\{\,.\,\}$.

- A Codificação por Entropia assume que a fonte é **sem memória.**
- Uma fonte é considerada sem memória quando as mensagens emitidas pela fonte são estatisticamente independentes, i.e., a ocorrência de uma determinada mensagem x_i não afeta a probabilidade de ocorrência da mensagem x_j , com $i, j = 0,1,\cdots, M-1$.
- Esta condição é necessária pois, caso contrário, a função $\overline{L} = f(p_i, l_i)$ a ser minimizada, dependeria do desenrolar temporal da sequência de mensagens emitidas pela fonte, o que resultaria em um código $\theta\{\,.\,\}$ variável no tempo.
- Embora poucas fontes físicas sigam exatamente o modelo de uma fonte sem memória, códigos $\theta\{\,.\,\}$ constantes no tempo (resultantes da suposição de independência estatística) são amplamente utilizados como códigos compressores, mesmo quando a dependência estatística da fonte resulta na impossibilidade de minimização de $\overline{L}$ durante a totalidade do tempo de codificação.

Exemplo 7: Seja um sistema para transmissão digital que utilize no Codificador de Fonte um conjunto $\Omega = \{x_0, x_1, x_2, x_3\} = \{00{,}01{,}10{,}11\}$ com M= 4 possíveis mensagens (ou M=4 níveis de quantização sob o ponto de vista do quantizador).

As amostras na saída X do quantizador são tais que a ocorrência de uma não altera a probabilidade de ocorrência da outra (i.e., as mensagens são estatisticamente independentes). As probabilidades são

$$P(X = x_0) = \frac{1}{2},\ P(X = x_1) = \frac{1}{4}, P(X = x_2) =, P(X = x_3) = \frac{1}{8}$$

0 código compressor $\theta\{\,.\,\}$ é conforme tabela abaixo.

Mensagem	Sequência	Palavra-Código s_i associada a m_i por $s_i = \theta\{m_i\}$
x_0	00	0
x_1	01	10
x_2	10	110
x_3	11	111

Determine a Entropia da Fonte $H(X)$, medida na saída do quantizador, e o comprimento médio $\bar{L}\ (\theta)$ do código $\theta\{\,.\,\}$.

Solução:

$$H(X) = -\sum_{k=0}^{M-1} p_k log_2(p_k) \qquad \bar{L} = \sum_{i=0}^{M-1} p_i l_i$$

x_i	p_i	Símbolo s_i associado a x_i por $s_i = \theta\{x_i\}$	l_i
x_0	1/2	0	1
x_1	1/4	10	2
x_2	1/8	110	3
x_3	1/8	111	3

Entropia da fonte $(H(X))$

$$H(X) = -\frac{1}{2} log_2\left(\frac{1}{2}\right) - \frac{1}{4} log_2\left(\frac{1}{4}\right) - \frac{1}{8} log_2\left(\frac{1}{8}\right) - \frac{1}{8} log_2\left(\frac{1}{8}\right) = 1.75\ [bits/mensagem]$$

Comprimento médio do código $\theta\{\,.\,\}(\bar{L}\ (\theta))$

$$\bar{L}(\theta) = \frac{1}{2} * 1 + \frac{1}{4} * 2 + \frac{1}{8} * 3 + \frac{1}{8} * 3 = 1.75\ [bits/símbolo]$$

Exemplo 8: Seja o código compressor $\theta\{\,.\,\}$ conforme definido abaixo:

x_i	p_i	Símbolo s_i associado a x_i por $s_i = \theta\{x_i\}$	l_i
x_0	1/3	0	1
x_1	1/3	10	2
x_2	1/3	11	2

Determine a Entropia da Fonte $H(X)$, medida na saída do quantizador, e o comprimento médio $\bar{L}\,(\theta)$ do código $\theta\{\,.\,\}$.

Solução:

$$H(X) = -\sum_{k=0}^{M-1} p_k log_2(p_k) \qquad \bar{L} = \sum_{i=0}^{M-1} p_i l_i$$

$$H(X) = -\frac{1}{3} log_2\left(\frac{1}{3}\right) - \frac{1}{3} log_2\left(\frac{1}{3}\right) - \frac{1}{3} log_2\left(\frac{1}{3}\right) = 1.58\ [bits/mensagem]$$

$$\bar{L}(\theta) = \frac{1}{3} * 1 + \frac{1}{3} * 2 + \frac{1}{3} * 2 = 1.67\ [bits/símbolo]$$

- Um código que pretenda ser útil deve pertencer à classe de códigos Univocamente Decodificáveis, caso contrário é impossível efetuar a decodificação sem que ocorra ambiguidade.

- Um código é Univocamente Decodificável (UD) quando qualquer sequência de caracteres do alfabeto A passível de ser formada a partir da justaposição de um número qualquer de símbolos pertencentes $S = \{s_i\} = \{s_0, s_1, \dots, s_{M-1}\}$ puder ser associada, ao ser decodificada, a uma única mensagem em $\Omega = \{x_i\} = \{x_0, x_1, \dots, x_{M-1}\}$.

- Conceito de justaposição: A justaposição de N símbolos (ou palavras-código) $\{s_i, s_{i+1}, \dots, s_{i+N-1}\}$ é a sequência α formada pela transmissão do símbolo s_i, seguido da transmissão do símbolo s_{i+1}, e assim sucessivamente até a transmissão do símbolo s_{i+N-1}, cuja representação é $\alpha = S_i S_{i+1} \dots S_{i+N-1}$.

Exemplo 9: Verifique se o código $\theta\{\,.\,\}$ abaixo é UD:

Mensagem	Palavra-código s_i associada a x_i por $s_i = \theta\{x_i\}$
x_0	0
x_1	010
x_2	01
x_3	10

Como decodificar a sequência 010?

- x_1?
- $x_2 x_0$?
- $x_0 x_3$?

A sequência 010 poderia corresponder a qualquer uma das três sequências de mensagens. Portanto $\theta\{\,.\,\}$ não é UD.

Códigos instantâneos

- A ambiguidade do código do Exemplo 9 talvez pudesse ser resolvida se aguardássemos a recepção de bits adicionais para resolver a incerteza, mas tal tempo de espera é indesejável, dada a constante busca por velocidade de decodificação (é desejável que o receptor seja capaz de decodificar os dados à medida que os mesmos são recebidos).

- Uma maneira de assegurar que um código seja UD e que nenhum tempo de espera seja necessário para a correta decodificação é utilizar códigos denominados **prefixos** ou **instantâneos**.

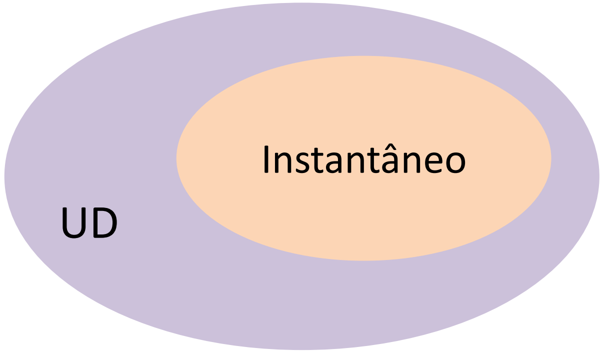
- A denominação "instantâneo" decorre de não haver necessidade, para tais códigos, de aguardar a recepção de bits adicionais para que se resolva ambiguidades.

- Um **código instantâneo** ou **prefixo** pode ser decodificado sem referência a palavras-código futuras, porque o final de uma palavra-código é imediatamente reconhecido no decodificador.

Códigos instantâneos

- Todos os códigos instantâneos são UD, mas nem todos os códigos UD são instantâneos. Ou seja, o conjunto dos códigos instantâneos é um sub-conjunto do conjunto dos códigos UD.

- Um código é chamado Instantâneo se nenhuma palavra-código é prefixo de nenhuma outra palavra-código pertencente ao código.



Conceito de prefixo:

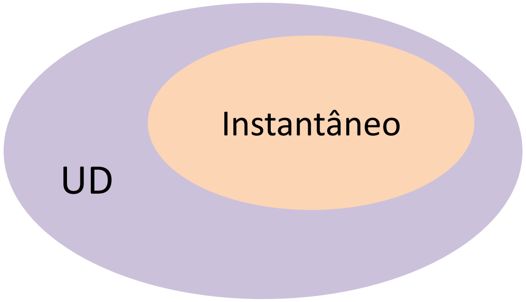
Sejam as sequências α_a, α_b e α_c formadas pela justaposição de, respectivamente, N_a, N_b e N_c palavras-código s_i, pertencentes ao código $\theta\{\,.\,\}$, sendo $N_a = N_b + N_c$ um número qualquer de palavras-código. Dizemos que α_b é prefixo de α_a, se α_a puder ser representada por $\alpha_b \alpha_c$, para alguma sequência α_c denominada sufixo.

Exemplo 10: Verifique se o código $\theta\{\,.\,\}$ abaixo é instantâneo:

Mensagem	Palavra-Código s_i associada a x_i por $s_i = \theta\{x_i\}$
x_0	10
x_1	00
x_2	11
x_3	110

Como 11 é prefixo de 110, $\theta\{\,.\,\}$ não é instantâneo.

No entanto, não podemos afirmar que não seja UD, pelo fato de não ser instantâneo.



Teste para código UD

Seja um código $\theta\{\,.\,\}$ com alfabeto $A = \{\alpha_0, \alpha_1, \dots, \alpha_{D-1}\}$ e conjunto imagem $S = \{s_i\} = \{s_0, s_1, \dots, s_{M-1}\}$. Para testar se $\theta\{\,.\,\}$ é UD, constrói-se a sequência de conjunto $S_0, S_1, \dots$ da seguinte maneira:

1. S_0 é o próprio conjunto imagem $S = \{s_i\} = \{s_0, s_1, \dots, s_{M-1}\}$.

2. Para definir S_1, forma-se a partir de S_0 o conjunto **P** de todos os pares $s_i s_j$ de palavras-código, $s_i \neq s_j$ possíveis de serem formados por justaposição de duas palavras-código distintas pertencentes ao conjunto S_0:

	s_0	s_1	…	s_{M-1}
s_0	-	$s_0 s_1$	…	$s_0 s_{M-1}$
s_1	$s_1 s_0$	-	…	$s_1 s_{M-1}$
…	…	…	-	
s_{M-1}	$s_{M-1} s_0$	$s_{M-1} s_1$	…	-

Teste para código UD

3. Se a palavra-código $s_i \in S_0$ é prefixo da palavra-código $s_j \in S_0$, i.e. $s_j = s_i\sigma$, então o sufixo σ é um elemento do conjunto S_1, i.e. $\sigma \in S_1$.

Executa-se a verificação $s_j = s_i\sigma$ para todos os elementos de **P** até que todos os sufixos sejam atribuídos ao conjunto $S_1 = \{\alpha_0, \alpha_1, \ldots\}$, onde cada sequência α_k de caracteres de A é um sufixo originado pelo resultado positivo do teste $s_j = s_i\sigma$.

4. Para definir S_n, $n > 1$, compara-se S_0 e S_{n-1}de modo bidirecional:

I) Se uma palavra-código $s_i \in S_0$ é prefixo de uma sequência $\alpha_j \in S_{n-1}$, tal que $\alpha_j = s_i\sigma$, então o sufixo $\sigma \in S_n$.

II) Se uma sequência $\alpha'_j \in S_{n-1}$ é prefixo de uma palavra-código $s'_i \in S_0$ tal que $s'_i = \alpha'_j\sigma'$, então o sufixo $\sigma' \in S_n$.

5. Define-se tantos conjuntos S_n até um valor de n tal que $S_n = \{\emptyset\}$ ou até um valor de n tal que $S_n = S_{n-1}$.

6. O código $\theta\{\,.\,\}$ é UD se e somente se **nenhum** dos conjuntos da sequência

de conjuntos S_1, S_2,... contenha uma palavra-código que pertença ao conjunto S_0 .

Exemplo 11: Verifique se o código $\theta\{.\}$ abaixo, com alfabeto $A = \{a, b, c, d, e\}$ é instantâneo e/ou UD.

Mensagem	Palavra-Código s_i associada a m_i por $s_i = \theta\{m_i\}$
x_0	a
x_1	c
x_2	ad
x_3	abb
x_4	bad
x_5	deb
x_6	$bbcde$

Teste para código UD

Solução:

S_0	S_1							
a	d							
c	bb							
ad								
abb								
bad								
deb								
$bbcde$								

- Construir S_1.
- S_1contém todos os sufixos encontrados ao mapear palavras de S_0 que são prefixo de outras palavras de S_0.
- Alguma palavra de $S_1 \in S_0$?
  - Se sim, $\theta\{\,.\,\}$ não é UD.
  - Se não, construir S_2, a partir de S_0 e S_1.

Teste para código UD

Solução:

S_0	S_1	S_2						
a	d	eb						
c	bb	cde						
ad								
abb								
bad								
deb								
$bbcde$								

- S_2 contém todos os sufixos encontrados ao mapear palavras de S_0 que são prefixo de palavras de S_1 e
- S_2 contém todos os sufixos encontrados ao mapear palavras de S_1 que são prefixo de palavras de S_0.
- Alguma palavra de $S_2 \in S_0$?
  - Se sim, $\theta\{.\}$ não é UD.
  - Se não, construir S_3, a partir de $S_0 \; e \; S_2$.

Teste para código UD

Solução:

S_0	S_1	S_2	S_3					
a	d	eb	de					
c	bb	cde						
ad								
abb								
bad								
deb								
$bbcde$								

- S_3 contém todos os sufixos encontrados ao mapear palavras de S_0 que são prefixo de palavras de S_2 e
- S_3 contém todos os sufixos encontrados ao mapear palavras de S_2 que são prefixo de palavras de S_0.
- Alguma palavra de $S_3 \in S_0$?
  - Se sim, $\theta\{\,.\,\}$ não é UD.
  - Se não, construir S_4, a partir de $S_0\ e\ S_3$.

Solução:

S_0	S_1	S_2	S_3	S_4				
a	d	eb	de	b				
c	bb	cde						
ad								
abb								
bad								
deb								
$bbcde$								

- S_4 contém todos os sufixos encontrados ao mapear palavras de S_0 que são prefixo de palavras de S_3 e
- S_4 contém todos os sufixos encontrados ao mapear palavras de S_3 que são prefixo de palavras de S_0.
- Alguma palavra de $S_4 \in S_0$?
  - Se sim, $\theta\{.\}$ não é UD.
  - Se não, construir S_5, a partir de S_0 e S_4.

Solução:

S_0	S_1	S_2	S_3	S_4	S_5			
a	d	eb	de	b	ad			
c	bb	cde			$bcde$			
ad								
abb								
bad								
deb								
$bbcde$								

- S_5 contém todos os sufixos encontrados ao mapear palavras de S_0 que são prefixo de palavras de S_4 e
- S_5 contém todos os sufixos encontrados ao mapear palavras de S_4 que são prefixo de palavras de S_0.
- Alguma palavra de $S_5 \in S_0$?

Solução:

S_0	S_1	S_2	S_3	S_4	S_5			
a	d	eb	de	b	ad			
c	bb	cde			$bcde$			
ad								
abb								
bad								
deb								
$bbcde$								

- S_5 contém todos os sufixos encontrados ao mapear palavras de S_0 que são prefixo de palavras de S_4 e
- S_5 contém todos os sufixos encontrados ao mapear palavras de S_4 que são prefixo de palavras de S_0.
- Alguma palavra de $S_5 \in S_0$?

Teste para código UD

Solução:

S_0	S_1	S_2	S_3	S_4	S_5			
a	d	eb	de	b	ad			
c	bb	cde			$bcde$			
ad								
abb								
bad								
deb								
$bbcde$								

- S_5 contém todos os sufixos encontrados ao mapear palavras de S_0 que são prefixo de palavras de S_4 e
- S_5 contém todos os sufixos encontrados ao mapear palavras de S_4 que são prefixo de palavras de S_0.
- Alguma palavra de $S_5 \in S_0$?

  Visto que $ad \in S_5$ e $ad \in S_0$, logo $\theta\{\,.\,\}$ não é UD.

Teste para código UD

Solução:

S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8
a	d	eb	de	b	ad	d	eb	$\{\emptyset\}$
c	bb	cde			$bcde$			
ad								
abb								
bad								
deb								
$bbcde$								

- Note que poderíamos ter encerrado o procedimento ao obter S_5 quando, então, já tínhamos elementos suficientes para decidir que $\theta\{\,.\,\}$ não é UD.
- No entanto, o procedimento foi continuado até não encontrarmos mais elementos na coluna S construída.
- Este é o procedimento que deve obrigatoriamente ser seguido caso não se encontre uma palavra pertencente a S_0 em alguma das colunas construídas. Somente ao encontrar $\{\emptyset\}$ é possível afirmar que o código é UD.

Exemplo 12: Verifique se os códigos $\theta_I\{.\}, \theta_{II}\{.\}$ e $\theta_{III}\{.\}$ abaixo, com alfabeto $A = \{0,1\}$ são instantâneos e/ou UDs.

$\theta_I\{.\}$ Não é instantâneo, nem UD.

S_0	S_1	S_2
1	0	0
00		1
01		
10		

$\theta_{II}\{.\}$ Não é instantâneo, mas é UD.

S_0	S_1	S_2
0	1	11
01	11	1
011		
111		

$\theta_{III}\{.\}$ Instantâneo.

S_0	S_1
0	$\{\emptyset\}$
10	
110	
111	

“ Seja uma variável aleatória discreta X com espaço de amostras definido pelo conjunto $\Omega = \{x_i\} = \{x_0, x_1, \dots, x_{M-1}\}$ de M eventos estatisticamente independentes m_i, com probabilidade de ocorrência $p_i, i = 0,1, \dots, M-1$.

Então é possível construir um código Instantâneo $\theta\{.\}$ com um conjunto de palavras-código $S = \{s_i\} = \{s_0, s_1, \dots, s_{M-1}\}$ formadas a partir do alfabeto $A = \{a_0, a_1, \dots, a_{D-1}\}$, tal que o conjunto $L = \{l_i\} = \{l_0, l_1, \dots, l_{M-1}\}$ dos tamanhos das palavras-código respectivas em S satisfaça à desigualdade

$$\frac{H(X)}{log_2 D} \leq \bar{L} \leq \frac{H(X)}{log_2 D} + 1$$

onde:

$H(X)$ é a Entropia X da fonte e

$\bar{L}$ é o tamanho médio das palavras-códigos, dado por $\bar{L} = \sum_{i=0}^{M-1} p_i l_i$ ”

O Teorema da Codificação de Fonte (TCF) garante a viabilidade teórica de implementação de códigos instantâneos *D*-ários, cujo tamanho médio dos símbolos pode ser reduzido a um valor tão pequeno quanto o valor da Entropia $H(X)$ da fonte, ou, se impossível, pelo menos a um valor menor que $H(X) + 1$.

Uma decorrência do TCF é a definição da Eficiência de Codificação η dada por

$$\eta = \frac{H(X)}{\bar{L} log_2 D}$$

- Um código é **Absolutamente Ótimo** *(matched to the source* - casado com a fonte) quando $\eta = 1.0$, isto é, quando

$$\bar{L} = \frac{H(X)}{log_2 D}$$

- Um código é **Quase Absolutamente Ótimo** quando

$$\frac{H(X)}{log_2 D} < \bar{L} \leq \frac{H(X)}{log_2 D} + 1$$

Teorema da Codificação de Fonte (*Noiseless Coding Theorem*)

Para a variável aleatória discreta X com espaço de amostras definido pelo conjunto $\Omega = \{x_i\} = \{x_0, x_1, \ldots, x_{M-1}\}$ de M eventos estatisticamente independentes m_i, com probabilidade de ocorrência $p_i, i = 0,1,\ldots,M-1$, será possível construir um código Instantâneo $\theta\{.\}$ com um conjunto de palavras-código $S = \{s_i\} = \{s_0, s_1, \ldots, s_{M-1}\}$ formadas a partir do alfabeto $A = \{0,1\}$, tal que o conjunto $L = \{l_i\} = \{l_0, l_1, \ldots, l_{M-1}\}$ dos tamanhos das palavras-código respectivas em S satisfaça à desigualdade

$$H(X) \leq \bar{L} \leq H(X) + 1$$

onde:

$H(X)$ é a Entropia X da fonte e

$\bar{L}$ é o tamanho médio das palavras-códigos, dado por $\bar{L} = \sum_{i=0}^{M-1} p_i l_i$"

O Teorema da Codificação de Fonte garante a viabilidade teórica de implementação de códigos instantâneos binários, cujo tamanho médio dos símbolos pode ser reduzido a um valor tão pequeno quanto o valor da Entropia $H(X)$ da fonte, ou, se impossível, pelo menos a um valor menor que $H(X)+1$.

Uma decorrência do TCF é a definição da Eficiência de Codificação η dada por

$$\eta = \frac{H(X)}{\bar{L}}$$

- Um código é **Absolutamente Ótimo** *(matched to the source* - casado com a fonte) quando $\eta = 1.0$, isto é, quando

$$\bar{L} = H(X)$$

- Um código é **Quase Absolutamente Ótimo** quando

$$H(X) < \bar{L} \leq H(X) \ + 1$$

Teorema da Codificação de Fonte (*Noiseless Coding Theorem*)

- Tomemos como exemplo o código binário (*D = 2*) estudado no Exemplo 7 e reproduzido ao lado, em que:

x_i	p_i	$s_i = \theta\{x_i\}$	l_i
x_0	1/2	0	1
x_1	1/4	10	2
x_2	1/8	110	3
x_3	1/8	111	3

$$H(X) = 1.75 \left[\frac{bits}{mensagem}\right], \qquad \bar{L}(\theta) = 1.75 \left[\frac{bits}{símbolo}\right] \; e \;\; log_2 D = log_2 2 = 1$$

Para este código

$$\bar{L} = \frac{H(X)}{log_2 D} = H(X)$$

Portanto, <u>o código é Absolutamente Ótimo.</u>

- Embora o TCF nos garanta que é possível obter códigos instantâneos com $\bar{L}$. tão pequeno quanto a própria Entropia *H(X)* da fonte, nenhuma informação é dada sobre como construir tais códigos.

- A construção de códigos ótimos baseia-se na minimização de $\bar{L} = \sum_{i=0}^{M-1} p_i l_i$.
- Um código instantâneo que minimize $\bar{L}$ é denominado de **Código Ótimo**.
- Existe um teorema que prova que se um código ótimo $\theta^*\{\,.\,\}$ resulta em $\overline{L^*}$, é impossível existir um outro código instantâneo $\theta\{\,.\,\}$ com tamanho médio $\bar{L}$ tal que $\bar{L} < \overline{L^*}$.
- Um Código Ótimo *D*-ário cujas palavras-código $S = \{s_i\} = \{s_0, s_1, \dots, s_{M-1}\}$ são formadas a partir do alfabeto $A = \{a_0, a_1, \dots, a_{D-1}\}$ satisfaz as seguintes propriedades (se o código for binário cada dígito *D*-ário é um bit) :

1. Palavras-código com maior probabilidade possuem menor tamanho.
2. As *D* palavras-código menos prováveis possuem o mesmo tamanho.
3. As *D* palavras-código menos prováveis diferem somente no último dígito *D-ário*.

Exemplo 13: Verifique se o código $\theta\{\,.\,\}$ abaixo é Ótimo.

Mensagem	p_i	Palavra-Código s_i associada a x_i por $s_i = \theta\{x_i\}$
x_0	0.6	0
x_1	0.2	100
x_2	0.1	101
x_3	0.04	1101
x_4	0.06	1110

Solução:

Propriedades de códigos ótimos sob o ponto de vista da entropia:

1. Palavras-código com maior probabilidade possuem menor tamanho.
2. As *2* palavras-código menos prováveis possuem o mesmo tamanho.
3. As *2* palavras-código menos prováveis diferem somente no último dígito binário.

- As propriedades 1 e 2 são satisfeitas.
- A propriedade 3 não é satisfeita: x_3 e x_4 não diferem somente no último bit.
- Portanto, $\theta\{\,.\,\}$ não é ótimo.

Para a construção de $\theta\{\,.\,\}$ efetua-se:

- Seja, inicialmente, $k = j = 0$.

1. Organizar as probabilidades p_i de alto a baixo em uma coluna em ordem decrescente de valor, denominada Coluna k.

2. Somar as *D* menores probabilidades na Coluna k e transferi-las para a próxima coluna (à direita), denominada Coluna $k+1$, obedecendo a ordem decrescente. As demais probabilidades da Coluna k são transferidas inalteradas para a Coluna $k+1$.

3. Incrementar k de 1 e repetir 1 a 3 até restarem somente *D* probabilidades na Coluna $k+1$, então denominada Coluna j.

4. Na Coluna j, atribuir a palavra-código representada pelo caractere a_0 à maior probabilidade, atribuir a palavra-código representada pelo caractere a_1, à segunda maior probabilidade, e assim sucessivamente até atribuir a palavra-código representada pelo caractere a_{D-1} à menor probabilidade.

Método para construção de códigos ótimos

5. Localizar na Coluna $j+1$ **,** imediatamente à esquerda da Coluna j, quais as D probabilidades geradoras que, ao serem somadas, resultaram na probabilidade gerada na Coluna j. Atribuir às D probabilidades geradoras na Coluna $j+1$ a palavra-código já atribuída à probabilidade gerada na Coluna j. As probabilidades não-geradoras na Coluna $j+1$ são atribuídas as palavras-código já atribuídas respectivas probabilidades não-geradas por soma na Coluna j .

6. Na Coluna $j+1$**,** as palavras-códigos já atribuídas em *5* as D probabilidades geradoras, justapor a palavra-código representada pelo caractere a_0 aquela geradora de maior probabilidade, justapor a palavra-código representada pelo caractere a_1, aquela geradora de segunda maior probabilidade, e assim sucessivamente até justapor a palavra-código representada pelo caractere a_{D-1} a palavra-código geradora de menor probabilidade.

7. Incrementar *j* de 1 e repetir 5 a 7 até que todas as colunas tenham palavras-código associadas as probabilidades nelas contidas.

8. Após a execução de 7, o Código de Huffman estará definido na coluna mais a esquerda.

Exemplo 14: Seja uma fonte de informação representada pela variável aleatória discreta X com espaço de amostras definido pelo conjunto $\Omega = \{x_i\} = \{x_0, x_1, \dots, x_{M-1}\}$ de $M = 6$ eventos estatisticamente independentes m_i com probabilidades de ocorrência p_i, $i = 0,1, \dots, M-1$, conforme tabela abaixo.

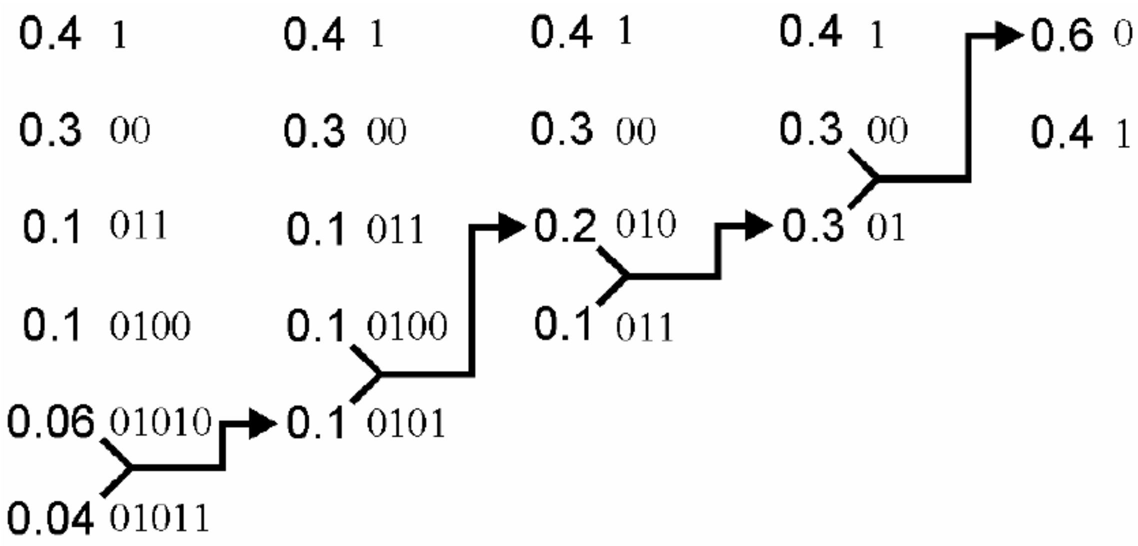
a) Determine um código ótimo $\theta\{.\}$ cujo conjunto de palavras-código $S = \{s_i\} = \{s_0, s_1, \dots, s_{M-1}\}$ é formado a partir do alfabeto $A = \{0,1\}$.

b) Determine a eficiência de $\theta\{.\}$.

c) Determine se $\theta\{.\}$ é absolutamente ótimo ou quase absolutamente ótimo.

Mensagem	p_i
x_0	0.4
x_1	0.3
x_2	0.1
x_3	0.1
x_4	0.06
x_5	0.04

Códigos Ótimos – Códigos de Huffman



Mensagem	p_i	s_i
x_0	0.4	1
x_1	0.3	00
x_2	0.1	011
x_3	0.1	0100
x_4	0.06	01010
x_5	0.04	01011

$$H(X) = -\sum_{i=0}^{M-1} p_i log_2(p_i) = 2.14 \; [bits/mensagem]$$

$$\bar{L} = \sum_{i=0}^{M-1} p_i l_i = 2.20 \; [bits/símbolo]$$

$$\eta = \frac{H(X)}{\bar{L}} = \frac{2.14}{2.20} = 97.3\%$$

Visto que $\frac{H(X)}{log_2 D} < \bar{L} \leq \frac{H(X)}{log_2 D} + 1, \theta\{\,.\,\}$ é quase absolutamente ótimo.

Códigos Ótimos – Códigos de Huffman

0.4

0.3

0.1

0.1

0.06

0.04

Códigos Ótimos – Códigos de Huffman

0.4	0.4
0.3	0.3
0.1	0.1
0.1	0.1
0.06	0.1
0.04	

Códigos Ótimos – Códigos de Huffman

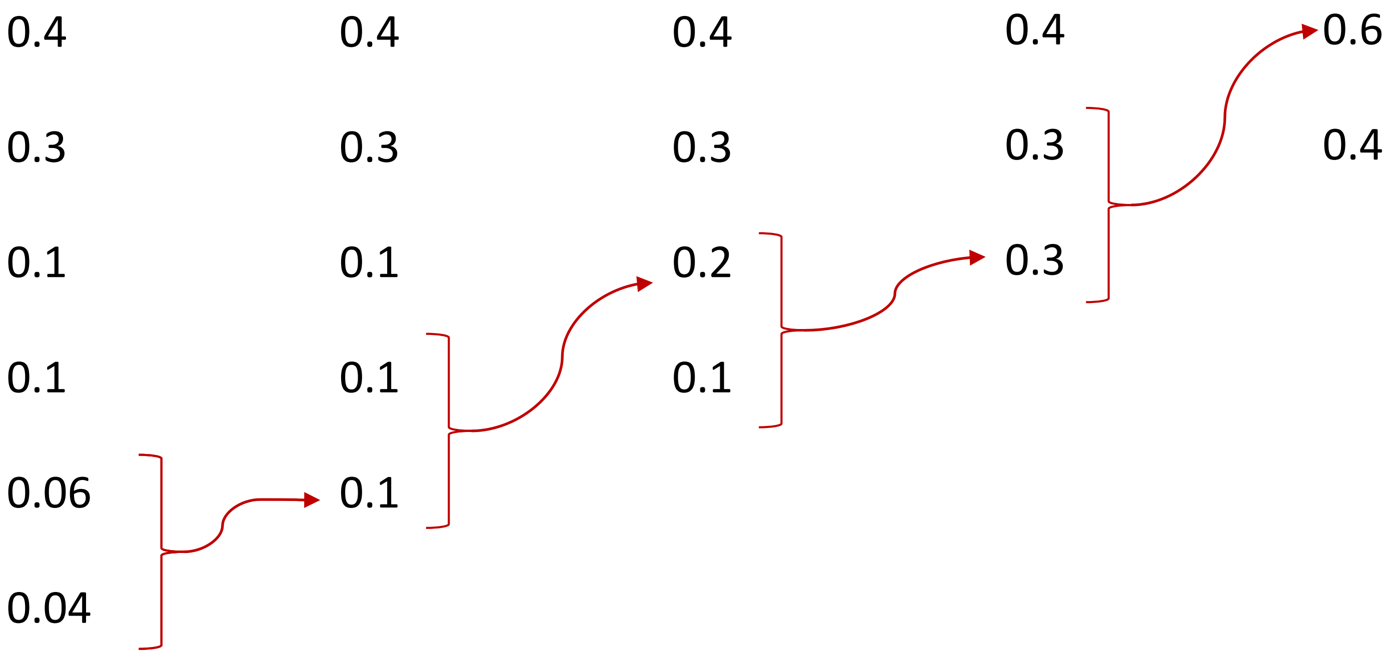
0.4	0.4	0.4
0.3	0.3	0.3
0.1	0.1	0.2
0.1	0.1	0.1
0.06	0.1	
0.04		

Códigos Ótimos – Códigos de Huffman

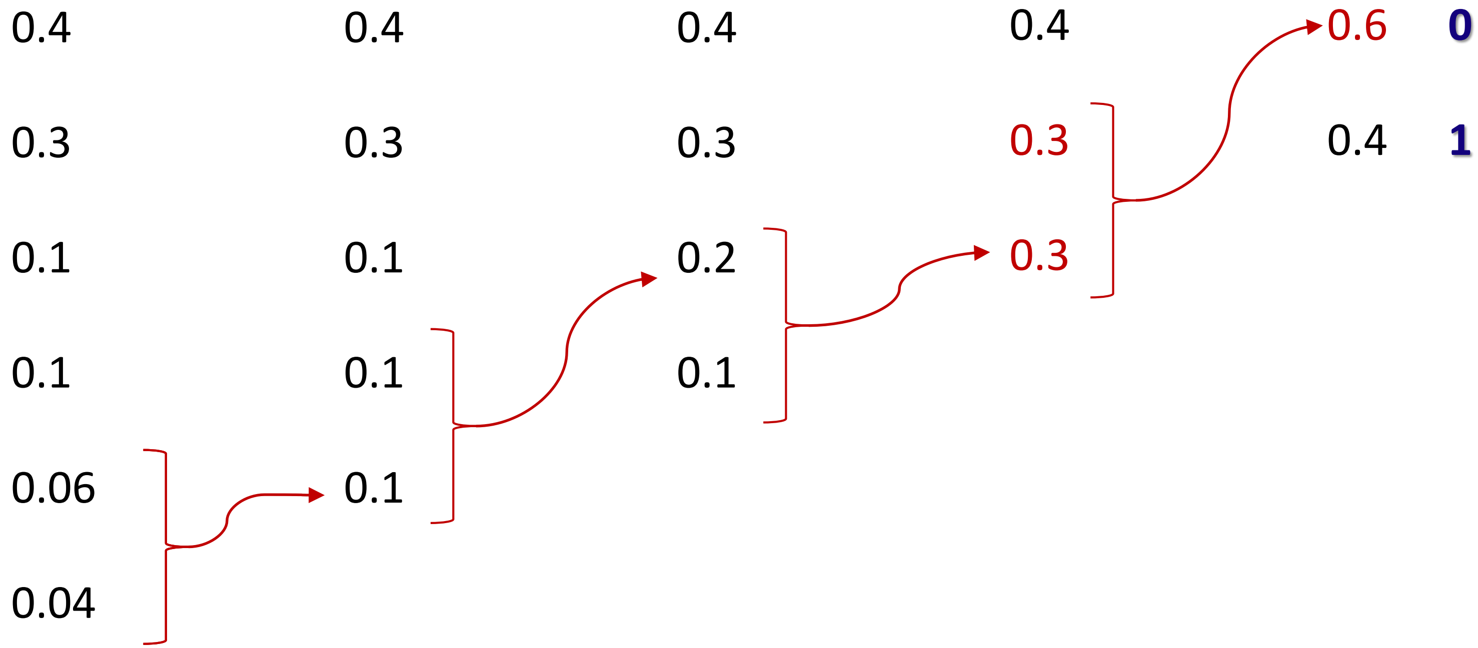
0.4	0.4	0.4	0.4
0.3	0.3	0.3	0.3
0.1	0.1	0.2	0.3
0.1	0.1	0.1	
0.06	0.1		
0.04			

Códigos Ótimos – Códigos de Huffman

0.4	0.4	0.4	0.4	0.6
0.3	0.3	0.3	0.3	0.4
0.1	0.1	0.2	0.3	
0.1	0.1	0.1		
0.06	0.1			
0.04				


0.4
0.3
0.1
0.1
0.06
0.04
0.4
0.3
0.1
0.1
0.1
0.4
0.3
0.2
0.1
0.4
0.3
0.3
0.6
0.4

Códigos Ótimos – Códigos de Huffman



Códigos Ótimos – Códigos de Huffman

0.4	0.4	0.4	0.4	**1**	0.6	**0**
0.3	0.3	0.3	0.3	**0**	0.4	**1**
0.1	0.1	0.2	0.3	**0**		
0.1	0.1	0.1				
0.06	0.1					
0.04						

Códigos Ótimos – Códigos de Huffman

0.4	0.4	0.4	0.4	**1**	0.6	**0**
0.3	0.3	0.3	0.3	**00**	0.4	**1**
0.1	0.1	0.2	0.3	**01**		
0.1	0.1	0.1				
0.06	0.1					
0.04						

Códigos Ótimos – Códigos de Huffman

0.4	0.4	0.4	0.4	**1**	0.6	**0**
0.3	0.3	0.3	0.3	**00**	0.4	**1**
0.1	0.1	0.2	0.3	**01**		
0.1	0.1	0.1				
0.06	0.1					
0.04						

Códigos Ótimos – Códigos de Huffman

0.4	0.4	0.4	0.4	**1**	0.6	**0**
0.3	0.3	0.3	0.3	**00**	0.4	**1**
0.1	0.1	0.2	0.3	**01**		
0.1	0.1	0.1				
0.06	0.1					
0.04						

Códigos Ótimos – Códigos de Huffman

0.4	0.4	0.4	1	0.4	1	0.6	0
0.3	0.3	0.3	00	0.3	00	0.4	1
0.1	0.1	0.2	010	0.3	01		
0.1	0.1	0.1	011				
0.06	0.1						
0.04							

Códigos Ótimos – Códigos de Huffman

0.4	0.4	0.4	**1**	0.4	**1**	0.6	**0**
0.3	0.3	0.3	**00**	0.3	**00**	0.4	**1**
0.1	0.1	0.2	**010**	0.3	**01**		
0.1	0.1	0.1	**011**				
0.06	0.1						
0.04							

Códigos Ótimos – Códigos de Huffman

0.4	0.4	0.4	**1**	0.4	**1**	0.6	**0**
0.3	0.3	0.3	**00**	0.3	**00**	0.4	**1**
0.1	0.1	0.2	**010**	0.3	**01**		
0.1	0.1	0.1	**011**				
0.06	0.1						
0.04							

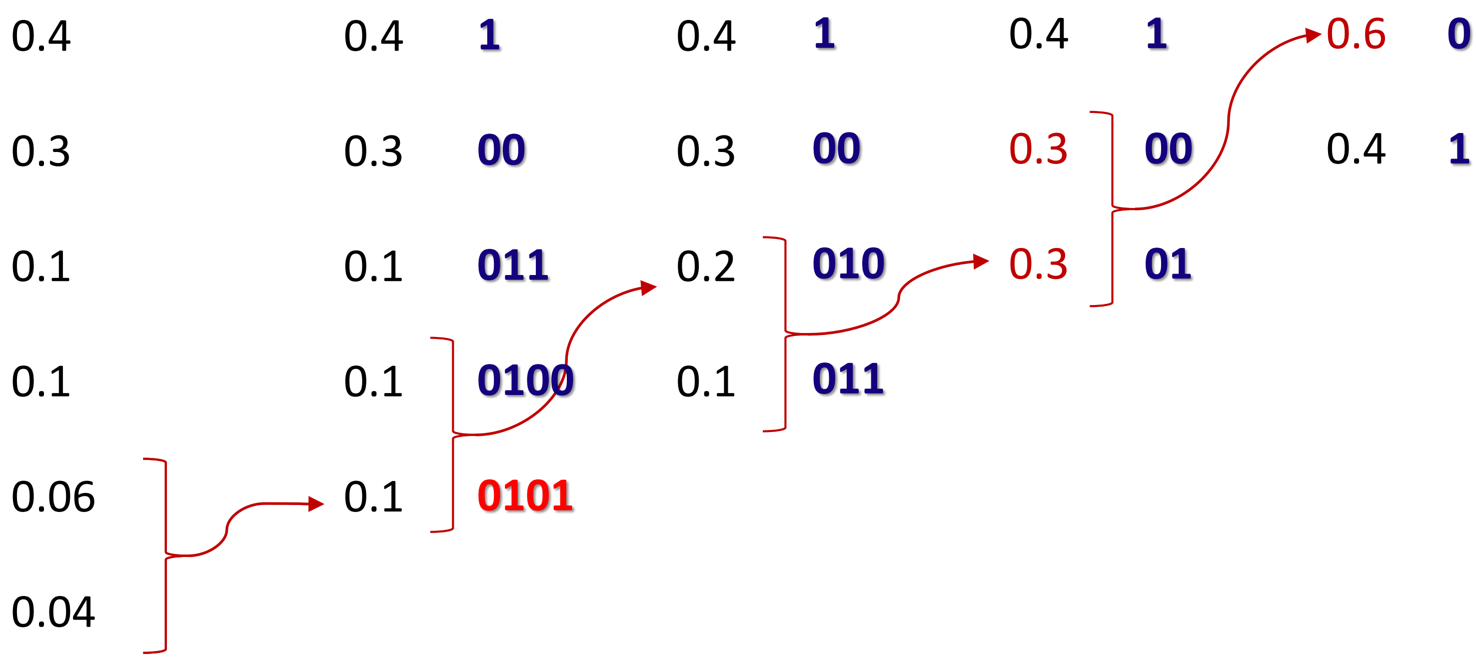
Códigos Ótimos – Códigos de Huffman

0.4	0.4	1	0.4	1	0.4	1	0.6	0
0.3	0.3	00	0.3	00	0.3	00	0.4	1
0.1	0.1	011	0.2	010	0.3	01		
0.1	0.1	0100	0.1	011				
0.06	0.1	0101						
0.04								

Códigos Ótimos – Códigos de Huffman

0.4	0.4	**1**	0.4	**1**	0.4	**1**	0.6	**0**
0.3	0.3	**00**	0.3	**00**	0.3	**00**	0.4	**1**
0.1	0.1	**011**	0.2	**010**	0.3	**01**		
0.1	0.1	**0100**	0.1	**011**				
0.06	0.1	**0101**						
0.04								

Códigos Ótimos – Códigos de Huffman



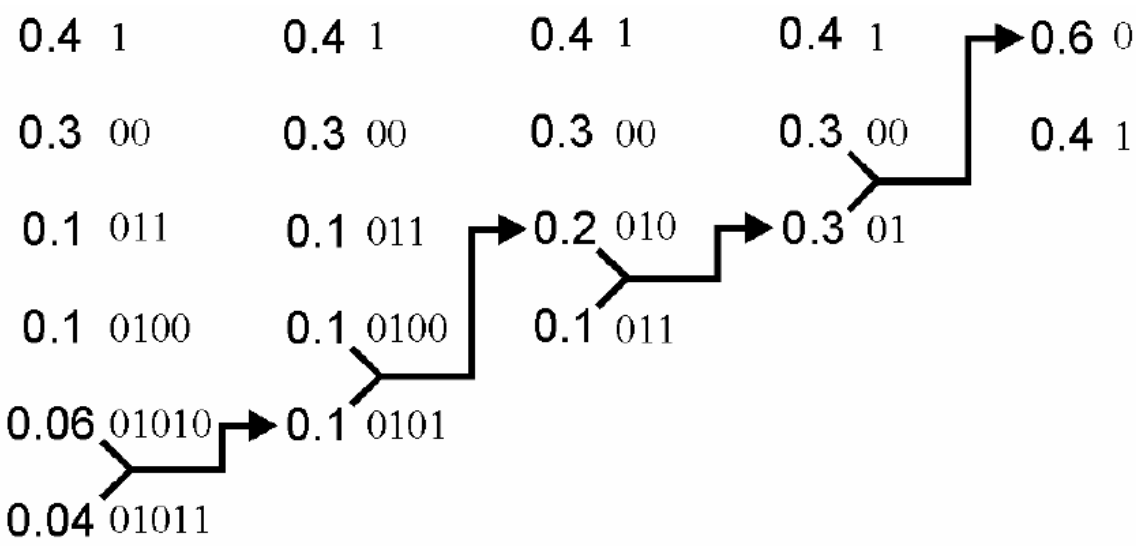
Códigos Ótimos – Códigos de Huffman

0.4	**1**	0.4	**1**	0.4	**1**	0.4	**1**	0.6	**0**
0.3	**00**	0.3	**00**	0.3	**00**	0.3	**00**	0.4	**1**
0.1	**011**	0.1	**011**	0.2	**010**	0.3	**01**		
0.1	**0100**	0.1	**0100**	0.1	**011**				
0.06	**01010**	0.1	**0101**						
0.04	**01011**								

Códigos Ótimos – Códigos de Huffman

0.4	**1**	0.4	**1**	0.4	**1**	0.4	**1**	0.6	**0**
0.3	**00**	0.3	**00**	0.3	**00**	0.3	**00**	0.4	**1**
0.1	**011**	0.1	**011**	0.2	**010**	0.3	**01**		
0.1	**0100**	0.1	**0100**	0.1	**011**				
0.06	**01010**	0.1	**0101**						
0.04	**01011**								

Códigos Ótimos – Códigos de Huffman



Mensagem	p_i	s_i
x_0	0.4	1
x_1	0.3	00
x_2	0.1	011
x_3	0.1	0100
x_4	0.06	01010
x_5	0.04	01011

$$H(X) = -\sum_{i=0}^{M-1} p_i log_2(p_i) = 2.14\ [bits/mensagem]$$

$$\bar{L} = \sum_{i=0}^{M-1} p_i l_i = 2.20\ [bits/símbolo] \qquad \eta = \frac{H(X)}{\bar{L}} = \frac{2.14}{2.20} = 97.3\%$$

Visto que $\frac{H(X)}{log_2 D} < \bar{L} \leq \frac{H(X)}{log_2 D} + 1, \theta\{\,.\,\}$ é quase absolutamente ótimo.