

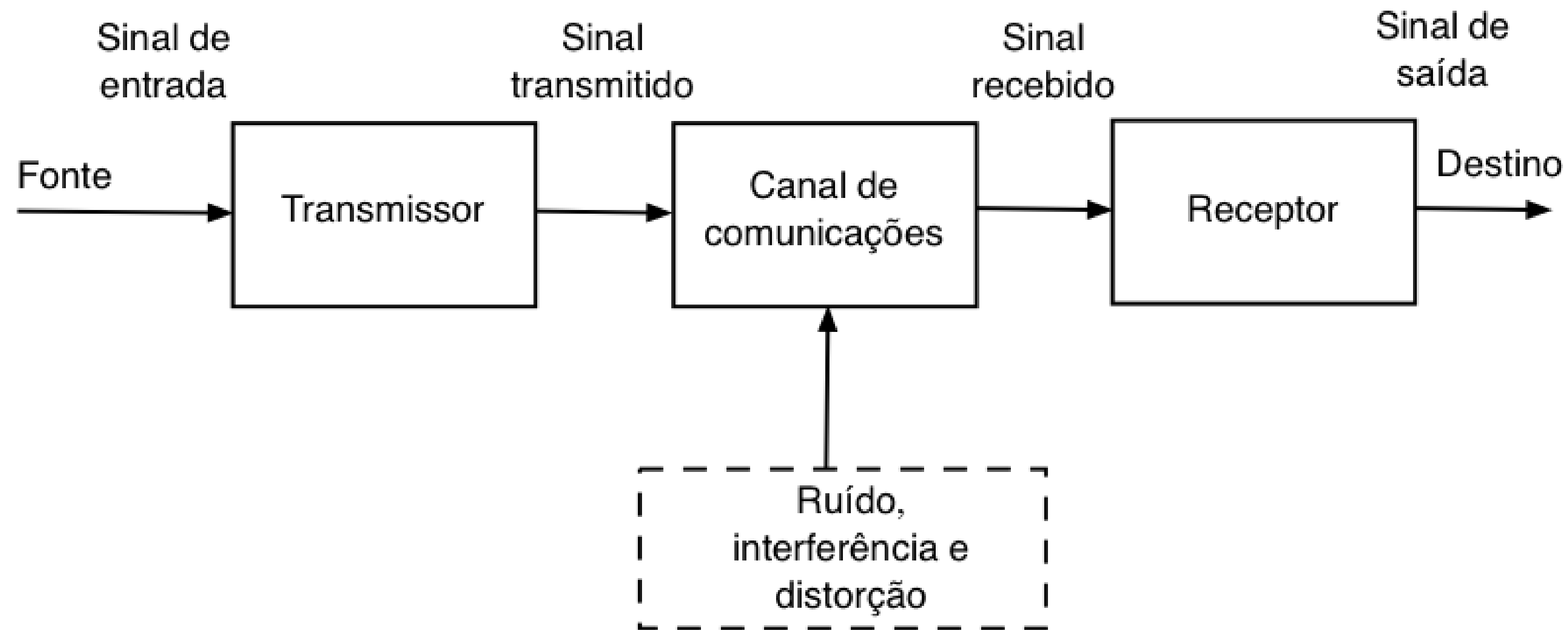
Curso: Engenharia de Computação

Sistemas de Comunicações Móveis

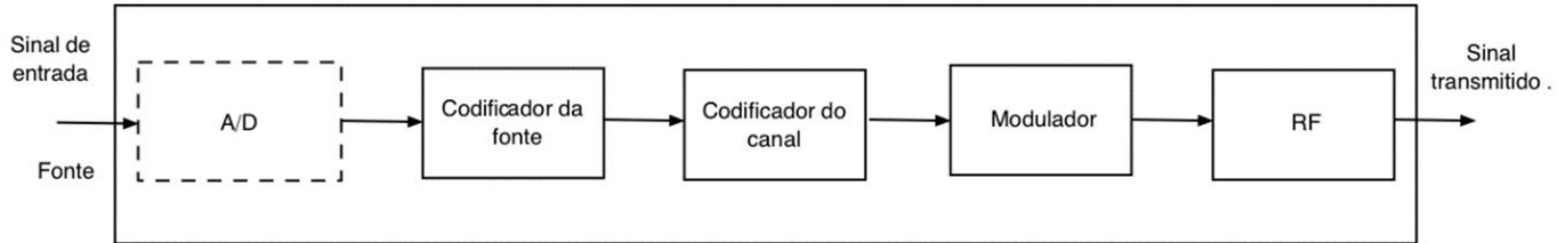
Prof. Clayton J A Silva, MSc
clayton.silva@professores.ibmec.edu.br



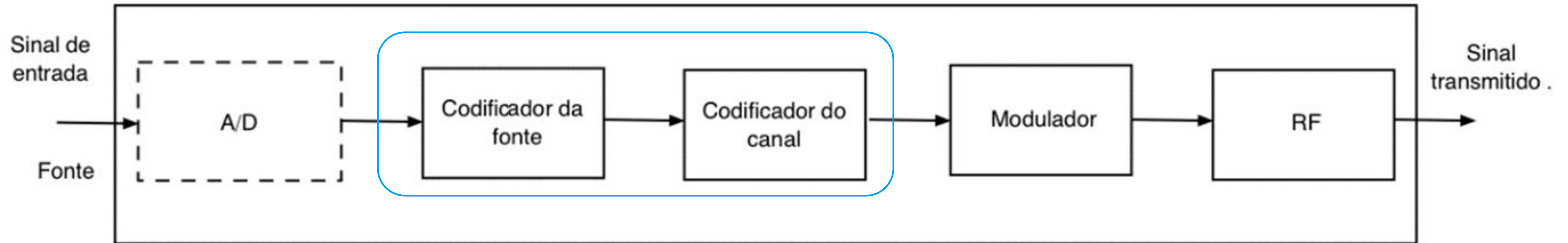
Modelo do Sistema de Comunicações



Transmissor



Transmissor



Codificador da fonte

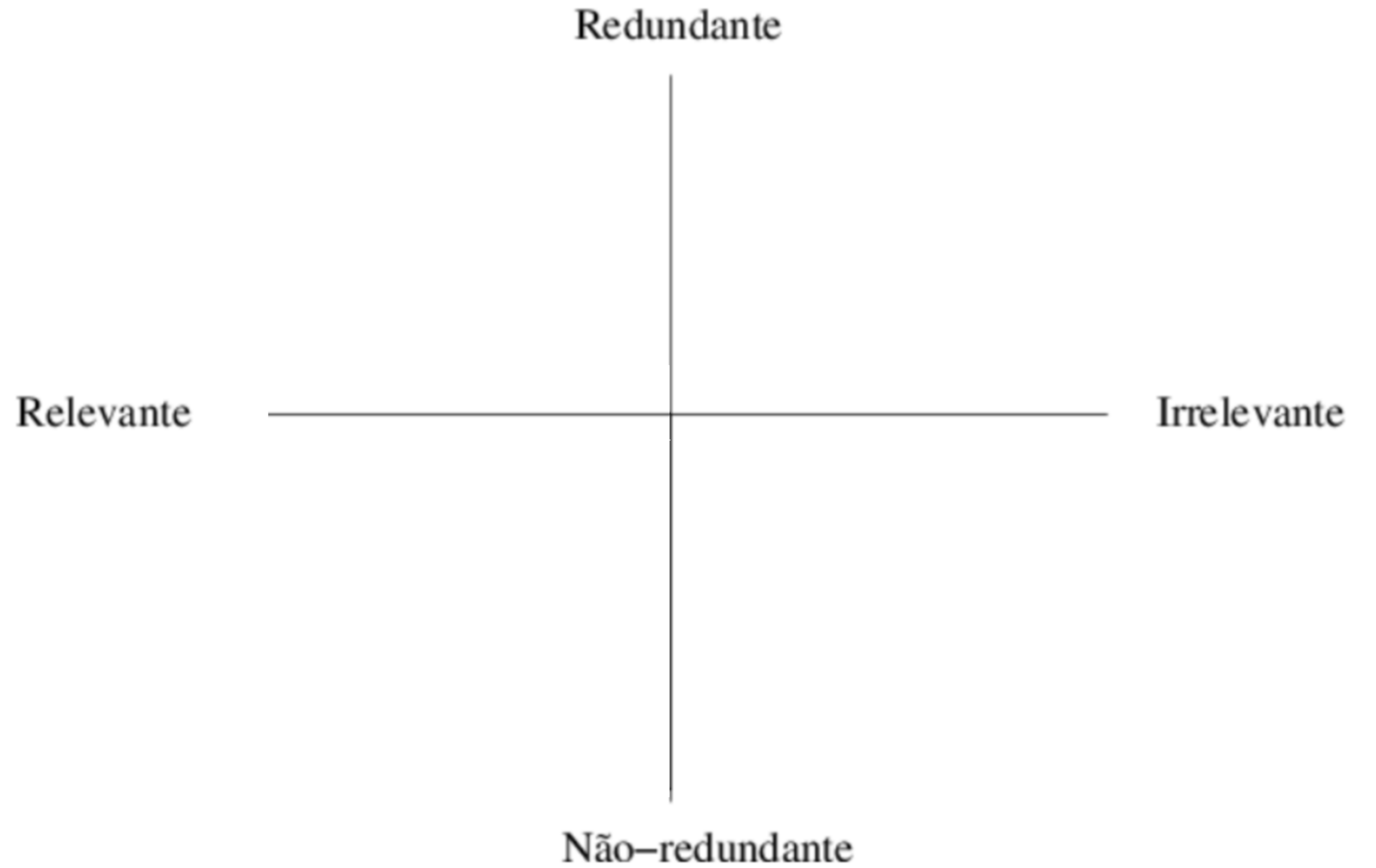
Codificador da fonte

- Processamento de símbolos para melhorar a comunicação quando a informação é digital ou pode ser aproximada na forma de símbolos discretos.
- Transforma uma mensagem digital em uma nova sequência de símbolos.

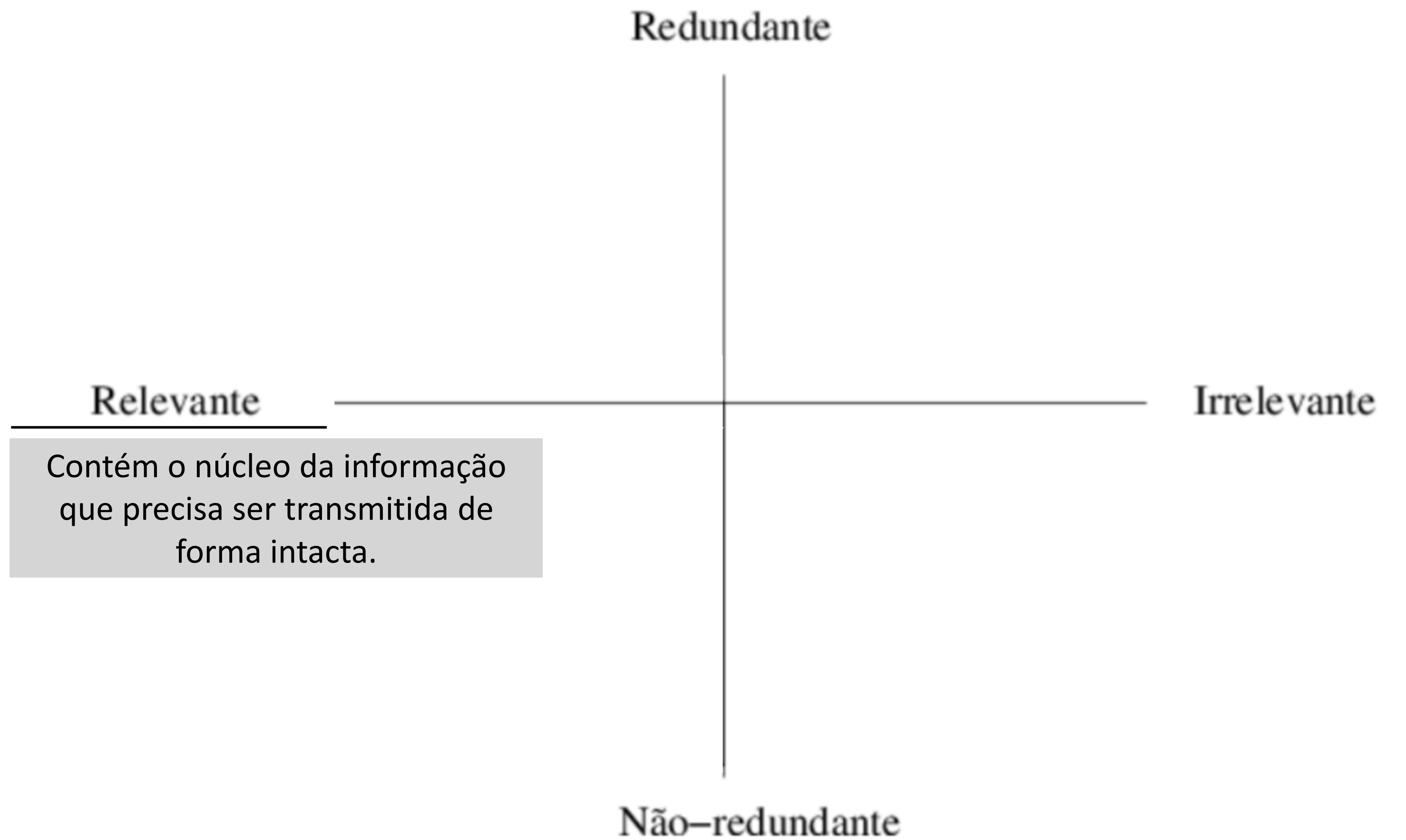
Compressão de sinais

Na codificação da fonte se realiza a compressão de sinais, cujo objetivo fundamental é reduzir o número de bits necessários para representar adequadamente os sinais a serem transmitidos

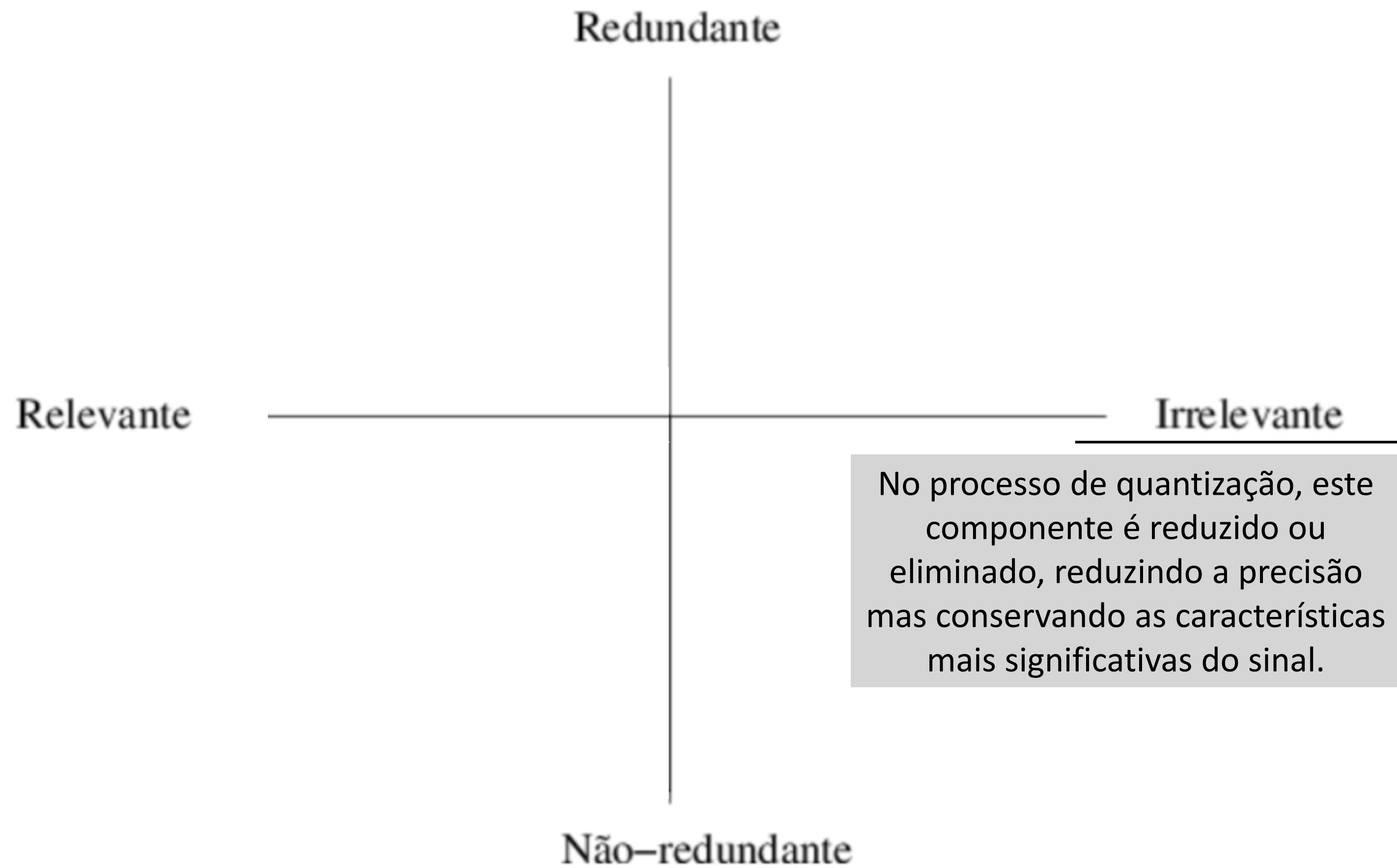
Informação dos sinais: componentes



Informação dos sinais: componentes



Informação dos sinais: componentes

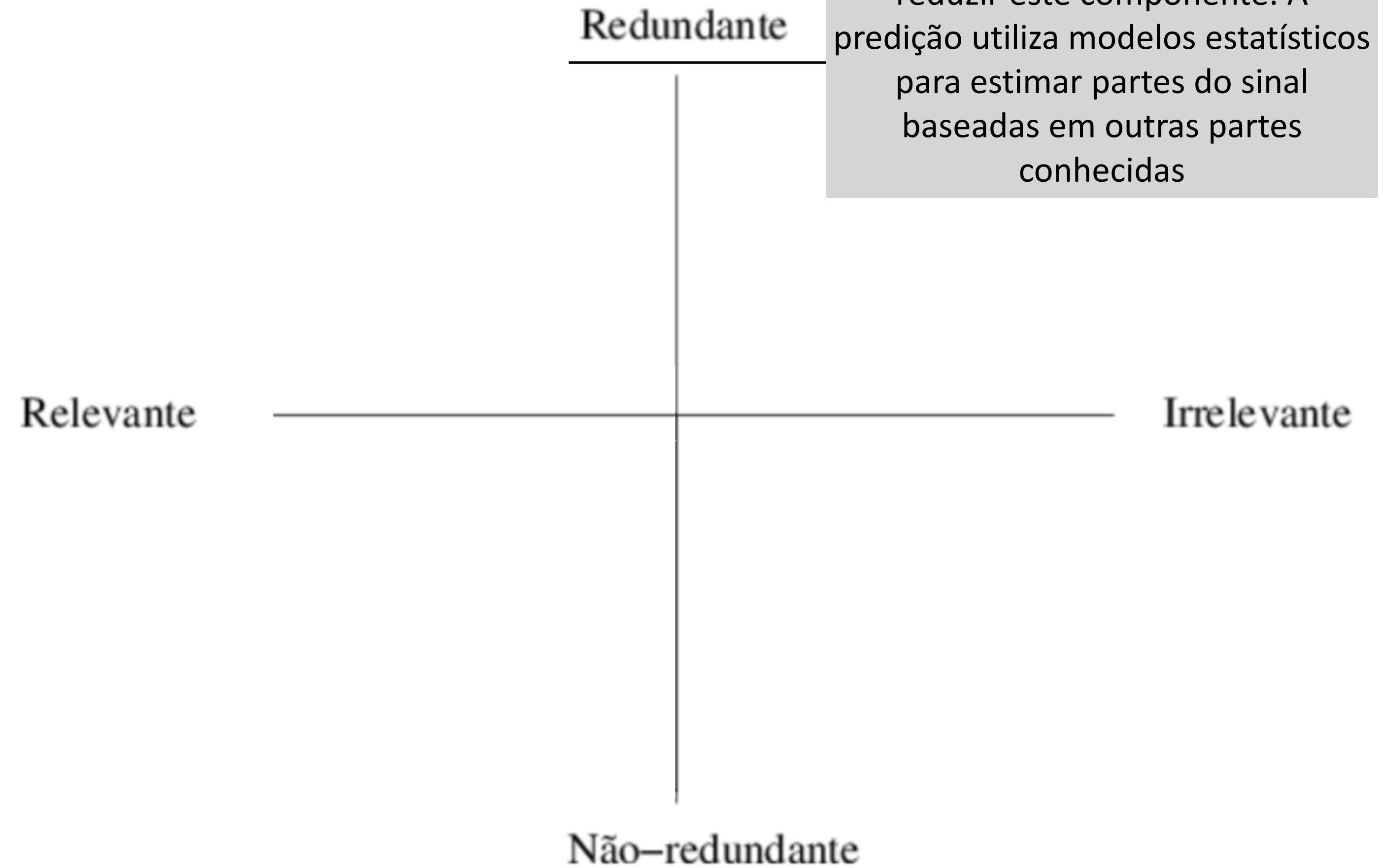


Informação dos sinais

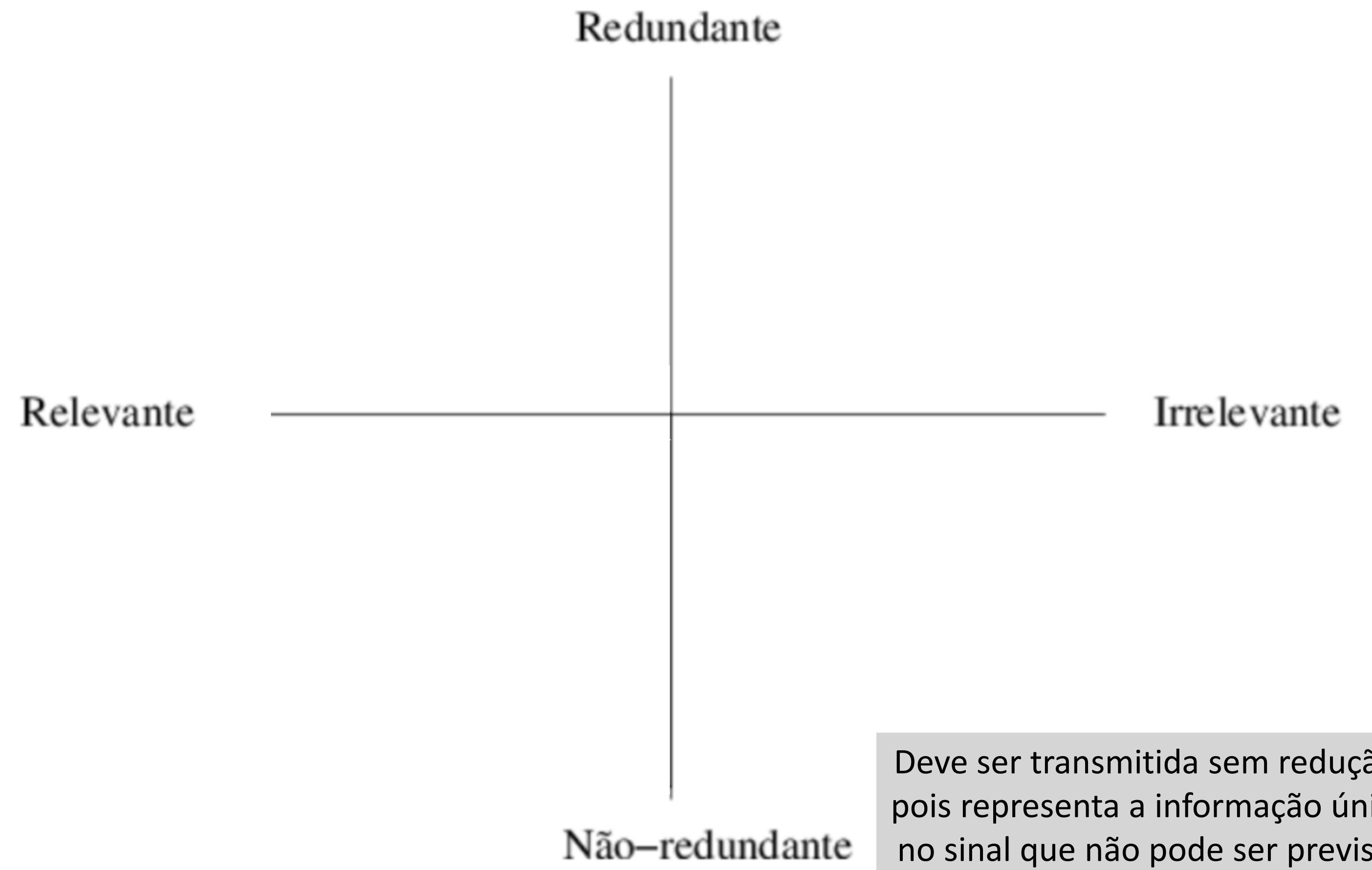
componentes relevantes x irrelevantes

- Os **componentes relevantes** carregam informações essenciais e têm maior impacto para transmitir o significado; os **componentes irrelevantes** carregam pouca ou nenhuma informação que seja realmente necessária para interpretar a mensagem.
- **Exemplo: No texto "HELLO",**
 - as consoantes "H", "L", e "O" podem ser consideradas **componentes relevantes**
 - a vogal "E" e o segundo "L" poderiam ser considerados irrelevante, pois não contribuem de forma decisiva para a compreensão
 - o codificador poderá transmitir "H" "L" "O"

Informação dos sinais: componentes



Informação dos sinais: componentes



Deve ser transmitida sem redução, pois representa a informação única no sinal que não pode ser prevista ou inferida a partir de outras partes da mensagem

Informação dos sinais

componentes redundantes x não redundantes

- Os componentes redundantes carregam informações que possuem menos incerteza de ocorrer na detecção (baixa entropia); ao contrário dos componentes não redundantes.
- Exemplo: No texto "HELLO",
 - as vogais "E" e "O" podem ser consideradas componentes redundantes
 - e as demais não redundantes
 - o esquema de predição ou transformada levará em consideração esse atributo

Compressão no codificador da fonte

- Componente relevante – transmitida
- Componente irrelevante – reduzida por meio da **quantização**
- Componente não redundante – **transmitida**
- Componente redundante, reduzida por meio de **técnicas de predição e transformação do sinal**

Desempenho da compressão

- O problema geral da compressão é **minimizar a taxa de bits na representação digital do sinal**, mantendo os níveis requeridos de:
 1. Qualidade do sinal reconstruído
 2. Complexidade da implementação
 3. Retardo da comunicação

Medidas de desempenho do codificador de fonte

1. Qualidade do sinal reconstruído

- Medidas de qualidade: subjetiva (qualitativas) ou objetiva (definidas matematicamente e estatisticamente)
- Medidas de qualidade objetiva:
 - Erro instantâneo, $e(n)$;
 - Erro médio (ME);
 - Erro médio quadrático (MSE); e
 - razão sinal-ruído de erro ($SENR$)

Erro instantâneo

O erro instantâneo é a diferença entre o valor original do sinal x_i e o valor aproximado ou reconstruído \hat{x}_i após a compressão, para um ponto específico no tempo.

$$e_i = x_i - \hat{x}_i$$

Onde:

- x_i é o valor original do sinal no instante i ,
- \hat{x}_i é o valor estimado ou reconstruído após a compressão.

Erro instantâneo

- Lembrar que em sistemas digitais, transmite-se o símbolo:

O símbolo transmitido $s(t)$ em modulação digital pode ser representado por uma **onda senoidal** com uma **amplitude A** e uma **fase θ** . Em um canal ideal, sem ruído, o símbolo transmitido seria:

$$s(t) = A \cdot \cos(\omega_c t + \theta)$$

Onde:

- A é a amplitude do sinal, ω_c é a frequência angular da portadora ($\omega_c = 2\pi f_c$), t é o tempo, θ é a fase

Erro instantâneo

Na prática, devido ao ruído e distorções do canal, o símbolo recebido $r(t)$ pode ter sua **amplitude e fase alteradas**. O sinal recebido seria:

$$r(t) = A_r \cdot \cos(\omega_c t + \theta_r)$$

Onde:

- A_r é a **amplitude recebida**, que pode ser diferente da amplitude A originalmente transmitida devido a fatores como atenuação no canal.
- θ_r é a **fase recebida**, que pode ser diferente da fase θ transmitida devido a distorções no canal, como deslocamento de fase ou interferência.

Outras medidas

- Erro Médio: $\bar{e} = \frac{1}{N} \sum_{i=1}^N e_i$
- Erro Médio Quadrático (MSE): $MSE = \frac{1}{N} \sum_{i=1}^N (x_i - \hat{x}_i)^2$
- Razão Sinal-Ruído de Erro (SNR): $SNR = 10 \log_{10} \left(\frac{\sum_{i=1}^N x_i^2}{\sum_{i=1}^N (x_i - \hat{x}_i)^2} \right) \text{ dB}$

Medida de taxa de erro de bit (BER – *Bit Error Rate*)

Se você quiser medir o erro **por bit** (e não por símbolo), é possível calcular a **Taxa de Erro de Bit (BER)**. Para isso, conta-se o número total de **bits errados** em todos os símbolos transmitidos. A BER é dada por:

$$\text{BER} = \frac{N_{\text{erro bit}}}{N_{\text{total bits}}}$$

Onde:

- $N_{\text{erro bit}}$ é o número de **bits** recebidos incorretamente,
- $N_{\text{total bits}}$ é o número total de **bits** transmitidos (que é o número de símbolos multiplicado pelo número de bits por símbolo).

Medidas de desempenho do codificador de fonte

2. Complexidade do algoritmo de compressão

- Número de instruções na unidade de tempo, normalmente medida em MIPS, e requisitos de **espaço de armazenamento** em memória requeridos para processamento do algoritmo.
- Tamanho físico, **custo** e **consumo** de potência do codificador, decodificador ou **codec** (codificador+decodificador)

Medidas de desempenho do codificador de fonte

3. Retardo

- O retardo ou atraso decorrente do processamento pelo *codec*
- O impacto do atraso sobre a comunicação depende da aplicação. Algumas aplicações admitem **limites** mais rigorosos de atraso.

Codificador do canal

Codificador de canal

- A codificação de canal é o processo através do qual o transmissor adiciona **redundância controlada** à informação de modo a permitir a **detecção** e a **correção** de erros.
- Dependendo do número de bits adicionados, os códigos de canal podem permitir a **correção de erros** na transmissão ou somente a **detecção dos erros** ocorridos.
- Existem duas grandes famílias de códigos detetores e corretores de erros: os **códigos de bloco** e os **convolucionais**.

Codificador de canal

- Frequentemente, no contexto de comunicações digitais, ocorrem erros de detecção acarretados por problemas de transmissão, implicando a necessidade de correção de erros
- Tipos de erros:
 - Erros **aleatórios**. Ocorrem de forma **independente** e não correlacionada em cada bit do bloco. Erros são isolados — cada erro acontece em um bit sem uma relação com o erro em outro bit. Ocorrem em canais com ruído elevado.
 - Erros **em surto**. Ocorrem de maneira **correlacionada** e geralmente **afetam uma sequência contínua de bits**. Em um erro de surto, vários bits consecutivos podem ser corrompidos. Comum em canais que experimentam interferências temporárias ou interrupções rápidas e intensas.

Codificador de canal

- A transmissão confiável de dados a **altas taxas** tem representado um desafio cada vez maior aos engenheiros
- Os **códigos corretores** têm contribuído de modo significativo para o avanço na área
- A solução está na **teoria da codificação**

Todo símbolo produzido por uma fonte de informação possui uma **probabilidade de ocorrência** em uma mensagem, logo pode ser modelado como uma **variável aleatória**

Entropia é a medida da incerteza associada a uma variável aleatória que representa o comportamento da transmissão dos símbolos

Taxa de informação é a média ponderada da informação associada a cada símbolo, medida em bits.

Codificador de canal

- A solução está na **teoria da codificação**

Existem códigos que podem tornar a probabilidade de erro na decodificação muito pequena

A probabilidade decresce exponencialmente quando n é aumentado, no entanto o aumento da complexidade do sistema também ocorre

Objetivos da teoria da codificação:

1. Encontrar **códigos longos** e eficientes
2. Encontrar **métodos práticos** de codificação/decodificação eficientes

Codificador de canal

- Codificadores (n, k, d)
- Codificadores de repetição
- Codificadores com 1 bit de paridade
- Códigos de *Hamming*

Processo de codificação (n, k, d)

1. Segmenta-se a mensagem em blocos de k dígitos
2. Acrescenta-se a cada bloco $n-k$ dígitos redundantes
3. Cada bloco gera uma palavra de n dígitos
4. Os dígitos redundantes são definidos a partir dos k dígitos da mensagens e destinam-se à **detecção** simples ou **detecção e correção** de erros
5. A **eficiência** do código é definida pela razão entre o número de bits de informação e o número total de bits, k/n
6. A distância de Hamming, ou seja, o número de posições nas quais os símbolos correspondentes são diferentes, é d

Processo de codificação (n, k, d)

- Um código (n, k, d) é definido como sendo um conjunto de 2^k *n -uplas* chamadas de **palavras código**, que
 - possuem K bits de informação,
 - possuem $n-k$ bits de redundância,
 - diferem entre si pelo menos d posições

Exemplo: *código (7,3,4)*

$k1$	$k2$	$k3$
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

$$c_1 = k1 \otimes k2$$

$$c_2 = k2 \otimes k3$$

$$c_3 = k1 \otimes k3$$

$$c_4 = k1 \otimes k2 \otimes k3$$

k			$n-k$			
$k1$	$k2$	$k3$	$c1$	$c2$	$c3$	$c4$
0	0	0	0	0	0	0
0	0	1	0	1	1	1
0	1	0	1	1	0	1
0	1	1	1	0	1	0
1	0	0	1	0	1	1
1	0	1	1	1	0	0
1	1	0	0	1	1	0
1	1	1	0	0	0	1

n

Processo de codificação (n, k, d)

- Detecção de erros

Em um código com distância mínima d , o menor número de mudanças necessárias para converter uma palavra do código em outra é pelo menos d , logo é possível detectar

$d - 1$ erros

Processo de codificação (n, k, d)

- Correção de erros

Em um código com distância mínima d , supondo palavras com mesma probabilidade, é possível identificar e corrigir até t erros e decidir com acerto na detecção, se

$$2t + 1 \leq d$$

Processo de codificação de repetição

- Em códigos de repetição, os parâmetros são:
 $k=1$, $d \geq 1$ e $n-1$ bits redundantes
- Como $k=1$, o código tem apenas duas palavras, uma delas é uma sequência de n 0s; a outra uma sequência de n 1s.
- Os dígitos de paridade são todos uma repetição do dígito c_i de informação $n-1$ vezes.

Processo de codificação de repetição

- Uma regra simples de decodificação é decidir pelo bit que aparece mais vezes na recepção.
- A distância de *Hamming* do código é *igual a n* e a eficiência é $1/n$.

Processo de codificação com 1 bit de paridade

- Em códigos com 1 bit de paridade, os parâmetros são:

$$k \geq 1, d=2 \text{ e } n=k+1$$

- O código tem apenas um bit c redundante, que é definido na transmissão para tornar o número de bits 1 par (ou ímpar).

Processo de codificação com 1 bit de paridade

- A regra de decodificação é contar o número de bits recebidos. Se a paridade não estiver correta significa que houve um erro na transmissão.
- A distância de *Hamming* do código é *igual a 2* e a eficiência é $k/(k+1)$.
- O código permite a detecção de 1 erro, mas não corrige. Pode ser eficaz quando se dispõe de canal de retorno para retransmissão da palavra.

Códigos de *Hamming*

- Nos códigos de *Hamming*, o número de bits de paridade pode ser dado pela quantidade de divisões inteiras sucessivas de n por 2 até quociente igual a 0.

Por exemplo, se $n=7$ bits, $7 \text{ div } 2 = 3$, $3 \text{ div } 2 = 1$, $1 \text{ div } 2 = 0$, logo $d=3$,
 $k=n-3=7-3=4$

- Ou seja, em códigos de *Hamming*, os parâmetros são

$$n \leq 2^d - 1$$

Códigos de *Hamming*

- No algoritmo de *Hamming*,
 1. Os bits c de redundância e os bits k de informação são organizados em uma **sequência de n bits a partir da posição 1...**

$$b_1 b_2 b_3 b_4 \dots b_n$$

2. ..., de modo que os bits de redundância ocuparão as **posições que são potência de 2, logo**

$$c_1 c_2 k_3 c_4 \dots$$

Códigos de *Hamming*

- No algoritmo de *Hamming*,
 3. Cada bit de redundância (bit de paridade) ajusta a **paridade par** (ou ímpar) dos bits **específicos** de informação...
 4. Cada bit de informação k_j , ou seja, o bit de informação que ocupa a posição j , é verificado pelos bits de redundância cuja soma das suas posições seja j .

Por exemplo, o bit k_3 é verificado pelos bits c_1 e c_2 . O bit k_5 é verificado pelos bits c_1 e c_4 . O bit k_6 é verificado pelos bits c_2 e c_4 .

Códigos de *Hamming*

- No algoritmo de *Hamming*,
 5. Cada bit de paridade será definido pelo codificador como 0 ou 1 de modo que o número de bits verificados seja par (ou ímpar)
 6. No receptor, a determinação do bit incorreto pode ser obtida **pela soma das posições de todos os bits de paridade incorretos**

Por exemplo, se na recepção os bits c_1 e c_2 estão incorretos, significa que o erro identificado foi no bit k_3 .



IBMEC.BR

 /IBMEC

 IBMEC

 @IBMEC_OFICIAL

 @IBMEC

 **ibmec**