UNIVERSIDADE FEDERAL DO CARIRI
CENTRO DE CIÊNCIAS E TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM MATEMÁTICA
EM REDE NACIONAL

JOSÉ AUGUSTO PEREIRA NOGUEIRA

APLICAÇÕES MATEMÁTICAS EM CÓDIGOS CORRETORES DE ERROS

JUAZEIRO DO NORTE
2019

JOSÉ AUGUSTO PEREIRA NOGUEIRA

APLICAÇÕES MATEMÁTICAS EM CÓDIGOS CORRETORES DE ERROS

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Matemática em Rede Nacional, do Centro de Ciências e Tecnologia da Universidade Federal do Cariri, como requisito parcial para obtenção do Título de Mestre em Matemática. Área de concentração: Ensino de Matemática.

Orientadora:
Prof. Dra. Clarice Dias de Albuquerque.

JUAZEIRO DO NORTE
2019

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Cariri
Sistema de Bibliotecas

N71a Nogueira, José Augusto Pereira.
Aplicações Matemáticas em Códigos Corretores de Erro / José Augusto Pereira Nogueira. – 2019.
81 f.: il.; color.; enc. ; 30 cm.
(Inclui bibliografia p.11-79).

Dissertação (Mestrado) – Universidade Federal do Cariri, Centro de Ciências e Tecnologia –Programa de Pós-graduação em Matemática em Rede Nacional, Juazeiro do Norte, 2019.

Área de Concentração: Ensino de Matemática.

Orientação: Profª Dra. Clarice Dias de Albuquerque.

1. Códigos Corretores de Erro. 2. Código de Hamming. 3. Matrizes.. I. Título.

CDD 512.5

Bibliotecário: João Bosco Dumont do Nascimento – CRB 3/1355

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO CARIRI
CENTRO DE CIÊNCIAS E TECNOLOGIA
MESTRADO PROFISSIONAL EM MATEMÁTICA EM REDE NACIONAL - PROFMAT

APLICAÇÕES MATEMÁTICAS EM CÓDIGOS CORRETORES DE ERROS

JOSÉ AUGUSTO PEREIRA NOGUEIRA

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Matemática em Rede Nacional – PROFMAT do Centro de Ciências e Tecnologia da Universidade Federal do Cariri, como requisito parcial para obtenção do Título de Mestre em Matemática. Área de concentração: Ensino de Matemática.

Aprovada em 25 de junho de 2019.

Banca Examinadora

Clarice Dias de Albuquerque

Prof.ª Dr.ª Clarice Dias de Albuquerque
Orientadora

Paulo César Cavalcante de Oliveira

Prof. Dr. Paulo César Cavalcante de Oliveira
Coorientador URCA

Plácido Andrade

Prof. Dr. Plácido Francisco de Assis Andrade
UFCA

Cátia R. de O. Quilles Queiroz

Prof.ª Dr.ª Cátia Regina de Oliveira Quilles Queiroz
UNIFAL

Dedico à minha família, pela paciência, compreensão e incentivo.

AGRADECIMENTOS

A Deus pelo dom da vida, saúde e coragem para enfrentar os obstáculos.

À minha mãe Maria Auxiliadora e ao meu pai Luiz, *in Memorian*, por todo o esforço para criar seus filhos com dignidade e honestidade e também pelo apoio e incentivo aos estudos para que buscássemos um futuro melhor. E aos meus irmãos que contribuíram para a minha conquista.

À minha orientadora Dra. Clarice por todo o suporte durante os estudos e escrita da dissertação. E agradeço também ao meu coorientador Dr. Paulo César, por todo o incentivo e apoio desde a graduação.

Aos meus amigos de sala por todos os momentos de interação e aprendizado, em especial a João Paulo, Renan e Andrea, por compartilhar seus conhecimentos e à Aline pelo companheirismo.

Agradeço a todos os professores da UFCA que participaram do PROFMAT.

Aos meus colegas de trabalho pelo suporte, em especial ao meu coordenador Cicefran por toda disponibilidade em ajudar sempre que eu necessitava de tempo para estudar e me dedicar à dissertação.

Aos meus professores de graduação que contribuíram em meu aprendizado e formação.

Enfim, a todos que contribuíram para que eu chegasse até aqui.

"Quando os problemas se tornam absurdos,
os desafios se tornam apaixonantes."
(Dom Hélder Câmara)

RESUMO

No mundo moderno estamos frequentemente recebendo informação, seja pelo celular, pelo computador ou pela televisão. Diante desse turbilhão de mensagens sendo enviadas ao mesmo tempo, os meios pelo qual esses dados trafegam podem sofrer interferências e estas podem causar alterações nas informações transmitidas. Para garantir que recebamos as mensagens conforme foram enviadas, os canais de transmissão utilizam-se de processos matemáticos chamados Códigos Corretores de Erros. Tais códigos, desenvolvidos através de conteúdos básicos como matrizes e álgebra, são responsáveis por detectar e corrigir possíveis erros encontrados durante a transmissão de informação. Neste trabalho faremos uma introdução à Teoria dos Códigos Corretores de Erros, enfatizando os Códigos Lineares e o Código de Hamming. Também mostraremos alguns códigos observados no nosso cotidiano, como os que estão presentes no CPF e no Cartão de Crédito, estes por sua vez utilizam em sua composição as operações matemáticas básicas, adição, multiplicação, subtração e divisão. Pretendemos que ao se depararem com o material aqui abordado, professores e alunos do Ensino Básico e Superior, possam compreender que a Matemática está presente diariamente em nossas vidas e que o conteúdo visto em sala de aula possui mais aplicações do que possamos imaginar.

Palavras-chave: Códigos Corretores de Erro. Código de Hamming. Matrizes.

ABSTRACT

In the modern world we are often receiving information, whether by cellphone, computer or television. Faced with this whirlwind of messages being sent at the same time, the means by which this data travels may suffer interference, these can cause changes in the information transmitted. To ensure that we receive the messages, the broadcast channels use mathematical processes called Error Correcting Codes. These codes, developed through basic contents such as arrays and algebra, are responsible for detecting and correcting possible errors encountered during the transmission of information. In this work we will introduce the Error Correcting Codes Theory, emphasizing the Linear Codes and the Hamming Code. We will also show some codes observed in our daily life, such as those that are present in the CPF and in the credit card, which in turn use in their composition the basic mathematical operations, addition, multiplication, subtraction and division. We pretend that when they come across the material discussed here, teachers and students of Basic and Higher Education, can understand that Mathematics is present daily in our lives and that the content seen in the classroom has more applications than we can imagine.

Keywords: Error Correcting Codes. Hamming Code. Matrices.

Lista de Figuras

Sumário

1 INTRODUÇÃO

A Matemática é estudada na escola desde os anos iniciais, e por se tratar de uma ciência exata, necessita de um pensamento um pouco mais abstrato e ao mesmo tempo preciso. Com isso, a maioria dos alunos costuma ter aversão a esta matéria e muitas vezes questiona onde os conteúdos estudados são realmente aplicados.

Atualmente vivemos na era digital, onde a propagação de conhecimento é constante. Estamos frenquentemente enviando e recebendo informação, seja ela por meio de texto, imagem ou vídeo. Essa transmissão incessante de mensagens está propensa a interrupções que podem alterar a informação enviada e para que recebamos a mensagem original sem erros, utilizamo-nos, mesmo que inconscientemente, de artifícios matemáticos, os quais são chamados Códigos Corretores de Erros.

Neste trabalho faremos um estudo inicial sobre a Teoria dos Códigos Corretores de Erros, que por sua vez são desenvolvidos através de conteúdos matemáticos estudados no Ensino Médio e Superior. Tais códigos são essenciais para as comunicações modernas, pois são capazes de detectar e corrigir eventuais erros que podem ocorrer durante a transmissão de informações.

No segundo capítulo veremos alguns conceitos algébricos que servirão de base para o desenvolvimento do tema proposto, mostrando que a Matemática vista em sala de aula, considerada muitas vezes como desnecessária, é utilizada diariamente, mesmo sem termos conhecimento.

O terceiro capítulo será voltado para o nosso tema principal, os Códigos Corretores de Erros. Faremos uma breve explanação histórica sobre o surgimento de tais códigos e mostraremos sua necessidade e importância na transmissão de mensagens. Além disso, veremos como funcionam estes códigos e quais suas principais propriedades e características.

No quarto capítulo, daremos ênfase aos Códigos Lineares, os quais são os mais utilizados na prática. E veremos que o processo de codificação e decodificação de mensagens, que se faz por meio do produto de matrizes, é realizado dentro de um canal de transmissão, que está sujeito à interferências.

Já no Capítulo 5 veremos mais exemplos de códigos lineares. Abordaremos o Código de Hamming, que foi um dos primeiros códigos a ser criado e que possui capacidade de corrigir um erro, desde que seja único. Veremos também alguns códigos que estão presentes no

nosso cotidiano, mas que não percebemos, são os códigos que possuem dígitos de controle (ou de verificação), como o CPF, por exemplo.

2 CONCEITOS ALGÉBRICOS

Para quem é professor de matemática, seja do Ensino Fundamental, Médio ou Superior, ao ministrar um conteúdo em sala de aula, provavelmente já deve ter ouvido a frase: "professor, onde eu vou usar isso em minha vida?" Esse é um questionamento constante entre os estudantes, pois eles não conseguem enxergar uma aplicação prática do que está sendo estudado, e isso muitas vezes acarreta no desinteresse pela matemática.

Veremos em nossos estudos que a matemática é utilizada diariamente, sem ao menos percebermos. Por exemplo, ao assistir televisão, enviar e receber mensagens por e-mail ou pelo celular, estamos usando a matemática através dos Códigos Corretores de Erros. Este assunto, que é o tema deste trabalho, é uma aplicação direta de conteúdos vistos no Ensino Básico (adição, subtração, multiplicação, divisão e operações com matrizes) e no Ensino Superior (álgebra linear, vetorial e abstrata).

Neste capítulo estudaremos alguns temas que servirão de base para introduzirmos os conceitos de Códigos Corretores de Erros e nos capítulos seguintes veremos como estes assuntos são aplicados.

2.1 Matrizes

2.1.1 Um pouco de história

Em 1841 o matemático Arthur Cayley, durante seus trabalhos sobre a teoria dos invariantes, começou o estudo da teoria das matrizes. Seu primeiro trabalho no qual introduziu a teoria básica das matrizes foi publicado em francês e intitulado por *Remarques sur la notation des fonctions algébriques* que traduzido significa *Observações sobre a notação das funções algébricas*. Este trabalho foi publicado em 1855 na revista de Crelle, e já se usava o conceito de matriz $m \times n$. A notação que Cayley usava para matrizes 2×2 era

$$\begin{array}{|cc|} \alpha, & \beta \\ \alpha', & \beta' \end{array}.$$

O trabalho mais importante de Cayley sobre a teoria das matrizes foi pulicado em 1858 na Philosophical Transactions of de Royal Society of London, de título *Memoir on the theory of matrices* (Memória sobre a teoria das matrizes). Nestes trabalhos foram

introduzidas as matrizes nula e identidade e foram definidas adição de matrizes e enunciada sua associatividade e comutatividade, produto de escalar por uma matriz e o produto de duas matrizes, sendo enunciado que essa multiplicação é associativa mas não é comutativa. Ainda foi definida a potência n-ésima de uma matriz.

Usando a notação atual, Cayley definiu o produto de matrizes 2×2 como se segue:

$$\begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} . \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = \begin{pmatrix} b_{11}a_{11} + b_{12}a_{21} & b_{11}a_{12} + b_{12}a_{22} \\ b_{21}a_{11} + b_{22}a_{21} & b_{21}a_{12} + b_{22}a_{22} \end{pmatrix}.$$

Neste último trabalho Cayley estabeleceu a inversa de uma matriz e também define a transposta de uma matriz, que é dada por:

$$tr \begin{pmatrix} \alpha, & \beta \\ \alpha', & \beta' \end{pmatrix} = \begin{pmatrix} \alpha, & \alpha' \\ \beta, & \beta' \end{pmatrix}.$$

Dada uma matriz M, enunciou que se $trM = M$ diz-se que M é uma matriz simétrica e se $trM = -M$, diz-se que M é uma matriz anti-simétrica.

Além de Cayley outros matemáticos deram contribuições no estudo da teoria das matrizes, onde podemos citar F. Georg Frobenius, H. J. S. Smith, Arthur Buchheim, Camille Jordan, William H. Metzler, entre outros.

2.1.2 Definição e Tipos Especiais de Matrizes

Definição 2.1. *Dados dois números inteiros positivos m e n, chama-se matriz $m \times n$ à tabela retangular de números reais dispostos ordenadamente em m linhas e n colunas. Os números que formam uma matriz são chamados de termos da matriz.*

Notação: Denotamos uma matriz A, $m \times n$, da seguinte forma:

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn} \end{pmatrix}.$$

Também podemos denotar a matriz acima por $A = (a_{ij})$, com $1 \leq i \leq m$ e $1 \leq j \leq n$. Observamos que os índices i e j indicam a posição de um termo na matriz, sendo i a linha e j a coluna que este ocupa. Por exemplo, o termo a_{23} está disposto na segunda linha e terceira coluna.

1. **Igualdade de Matrizes**
   Se $A = (a_{ij})$ e $B = (b_{ij})$ são matrizes $m \times n$, dizemos que A e B são iguais se, e somente se, $a_{ij} = b_{ij}$, para quaisquer valores de i e j.

2. **Tipos Especiais de Matrizes**

Matriz Linha: É toda matriz composta de apenas uma linha, ou seja, da forma $1 \times n$.
Exemplos: $A = \begin{pmatrix} 2 \end{pmatrix}$; $B = \begin{pmatrix} 1 & -2 & 0 & 7 \end{pmatrix}$.

Matriz Coluna: É toda matriz composta de uma única coluna, isto é, da forma $m \times 1$.
Exemplos: $A = \begin{pmatrix} 2 \end{pmatrix}$; $C = \begin{pmatrix} 5 \\ -3 \\ 0 \end{pmatrix}$.

Matriz Nula: É a matriz cujos termos são todos iguais a zero. Denotamos a matriz nula $m \times n$ por $O_{m \times n}$, ou quando se conhece o número de linhas e colunas da matriz indicamos apenas por O.
Exemplos: $O_{3\times 2} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$; $O_{2\times 4} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$.

Matriz Quadrada: É uma matriz da forma $n \times n$.
Considere uma matriz quadrada $n \times n$ dada por $A = (a_{ij})$. Chama-se *diagonal principal* da matriz A, a lista formada pelos elementos $(a_{11}, a_{22}, ..., a_{nn})$, ou seja, são os termos (a_{ij}) tais que $i = j$. Chama-se *diagonal secundária* a lista dos termos $(a_{1n}, a_{2(n-1)}, ..., a_{(n-1)2}), a_{n1})$. Observamos ainda que a soma dos índices dos elementos da diagonal secundária é igual a $n + 1$.
Exemplos: $A = \begin{pmatrix} 2 \end{pmatrix}$; $D = \begin{pmatrix} 5 & -1 & 0 \\ -3 & 2 & 2 \\ 0 & 1 & -1 \end{pmatrix}$.
Na matriz D os elementos da diagonal principal são $\{5, 2, -1\}$ e os elementos da diagonal secundária são $\{0, 2, 0\}$.

Matriz Triangular Superior: É uma matriz quadrada tal que $a_{ij} = 0$ para todo $i > j$, ou seja, todos os termos abaixo da diagonal principal são nulos.
Exemplos: $E = \begin{pmatrix} 2 & 1 \\ 0 & -2 \end{pmatrix}$; $F = \begin{pmatrix} 5 & 1 & 0 \\ 0 & 1 & -2 \\ 0 & 0 & -1 \end{pmatrix}$.

Matriz Triangular Inferior: É uma matriz quadrada tal que $a_{ij} = 0$ para todo $i < j$, ou seja, os elementos acima da diagonal principal são todos nulos.
Exemplos: $G = \begin{pmatrix} 2 & 0 \\ 3 & 2 \end{pmatrix}$; $H = \begin{pmatrix} 2 & 0 & 0 \\ 3 & 1 & 0 \\ -5 & 4 & 1 \end{pmatrix}$.

Matriz Diagonal: Refere-se a uma matriz quadrada tal que $a_{ij} = 0$ para todo $i \neq j$, isto é, todos os termos que estão fora da diagonal principal são nulos.

Exemplos: $J = \begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 3 \end{pmatrix}$; $K = \begin{pmatrix} -2 & 0 \\ 0 & 1 \end{pmatrix}$.

Matriz Identidade: Trata-se de uma matriz diagonal tal que todos os elementos da diagonal principal são iguais a 1, ou seja, $a_{ij} = 0$ para todo $i \neq j$ e $a_{ij} = 1$ sempre que $i = j$. Denotamos po I_n a matriz identidade $n \times n$.

Exemplos: $I_4 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$; $I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$.

Matriz simétrica: Se refere a uma matriz quadrada tal que os termos da matriz são simétricos em relação à diagonal principal, isto é, dada uma matriz $A = (a_{ij})$ tem-se $a_{ij} = a_{ji}$.

Exemplos: $L = \begin{pmatrix} 2 & 3 & 0 \\ 3 & -1 & -2 \\ 0 & -2 & 1 \end{pmatrix}$; $M = \begin{pmatrix} 0 & 2 \\ 2 & 1 \end{pmatrix}$; toda matriz diagonal; I_n.

Matriz Anti-Simétrica: Trata-se de uma matriz quadrada tal que os elementos da diagonal principal são nulos e os termos posicionados simetricamente em relação a diagonal principal são números reais simétricos, isto é, dada uma matriz $A = (a_{ij})$ tem-se $a_{ij} = -a_{ji}$.

Exemplos: $N = \begin{pmatrix} 0 & -3 & 5 \\ 3 & 0 & 2 \\ -5 & -2 & 0 \end{pmatrix}$; $P = \begin{pmatrix} 0 & 2 \\ -2 & 0 \end{pmatrix}$; toda matriz quadrada nula.

2.1.3 Operações com Matrizes

1. **Adição**

   Dadas duas matrizes $A = (a_{ij})$ e $B = (b_{ij})$, $m \times n$, definimos a soma das matrizes A e B como sendo $A + B = (a_{ij} + b_{ij})$, ou seja, somamos os termos correspondentes das duas matrizes.

   Exemplos: Tomando as matrizes E, G, H e L, dos exemplos anteriores, temos

   $$E + G = \begin{pmatrix} 2 & 1 \\ 0 & -2 \end{pmatrix} + \begin{pmatrix} 2 & 0 \\ 3 & 2 \end{pmatrix} = \begin{pmatrix} 2+2 & 1+0 \\ 0+3 & -2+2 \end{pmatrix} = \begin{pmatrix} 4 & 1 \\ 3 & 0 \end{pmatrix}.$$

$$H + L = \begin{pmatrix} 2 & 0 & 0 \\ 3 & 1 & 0 \\ -5 & 4 & 1 \end{pmatrix} + \begin{pmatrix} 2 & 3 & 0 \\ 3 & -1 & -2 \\ 0 & -2 & 1 \end{pmatrix} = \begin{pmatrix} 4 & 3 & 0 \\ 6 & 0 & -2 \\ -5 & 2 & 2 \end{pmatrix}.$$

Propriedades da Adição:

Sejam $A = (a_{ij})$, $B = (b_{ij})$ e $C = (c_{ij})$ matrizes $m \times n$, então valem as seguintes propriedades:

(a) Associatividade: $(A + B) + C = A + (B + C)$
Demonstração: Temos que os termos das matrizes são números reais, portanto vale a associatividade desses números, assim

$$(A + B) + C = ((a_{ij} + b_{ij}) + c_{ij}) = (a_{ij} + (b_{ij} + c_{ij})) = A + (B + C).$$

(b) Comutatividade: $A + B = B + A$
Demonstração: Como os termos das matrizes são números reais, então vale a comutatividade, dessa forma

$$A + B = (a_{ij} + b_{ij}) = (b_{ij} + a_{ij}) = B + A.$$

(c) Elemento Neutro: O elemento neutro para a adição de matrizes é a matriz nula O, $m \times n$, pois $A + O = A$.

(d) Matriz Oposta: Seja $A = (a_{ij})$ uma matriz $m \times n$. A matriz oposta de A é definida por $-A = (-a_{ij})$, onde seus elementos são simétricos aos termos correspondentes da matriz A. Além disso verificamos que $A + (-A) = O$.

□

2. **Subtração**
Sejam $A = (a_{ij})$ e $B = (b_{ij})$ duas matrizes $m \times n$. Define-se a subtração B de A como a soma de A com a matriz oposta de B, isto é,

$$A - B = A + (-B) = (a_{ij} - b_{ij}).$$

Exemplo: Sejam $A = \begin{pmatrix} 1 & 3 \\ -3 & 1 \\ 0 & -2 \end{pmatrix}$ e $B = \begin{pmatrix} 0 & -3 \\ 2 & 1 \\ 5 & -4 \end{pmatrix}$. Assim

$$A - B = \begin{pmatrix} 1 & 3 \\ -3 & 1 \\ 0 & -2 \end{pmatrix} - \begin{pmatrix} 0 & -3 \\ 2 & 1 \\ 5 & -4 \end{pmatrix} = \begin{pmatrix} 1-0 & 3-(-3) \\ -3-2 & 1-1 \\ 0-5 & -2-(-4) \end{pmatrix} = \begin{pmatrix} 1 & 6 \\ -5 & 0 \\ -5 & 2 \end{pmatrix}.$$

3. **Multiplicação de um Número por uma Matriz**
Dada a matriz $A = (a_{ij})$ e um número real x, definimos o produto de x por A, denotado por xA, como sendo a multiplicação de x por todos os termos de A, ou seja, $xA = (xa_{ij})$.

Exemplos: Dadas as matrizes $A = \begin{pmatrix} -1 & 0 \\ 3 & 1 \end{pmatrix}$ e $B = \begin{pmatrix} 0 & -3 & 4 \\ 2 & 1 & -1 \end{pmatrix}$ e os números reais -3 e 4, temos

$$4A = 4 \cdot \begin{pmatrix} -1 & 0 \\ 3 & 1 \end{pmatrix} = \begin{pmatrix} 4 \cdot (-1) & 4 \cdot 0 \\ 4 \cdot 3 & 4 \cdot 1 \end{pmatrix} = \begin{pmatrix} -4 & 0 \\ 12 & 4 \end{pmatrix}.$$

$$-3B = -3 \cdot \begin{pmatrix} 0 & -3 & 4 \\ 2 & 1 & -1 \end{pmatrix} = \begin{pmatrix} -3 \cdot 0 & -3 \cdot (-3) & -3 \cdot 4 \\ -3 \cdot 2 & -3 \cdot 1 & -3 \cdot (-1) \end{pmatrix} = \begin{pmatrix} 0 & 9 & -12 \\ -6 & -3 & 3 \end{pmatrix}.$$

Propriedades da Multiplicação por um número real
Sejam $A = (a_{ij})$ e $B = (b_{ij})$ matrizes $m \times n$, e x e y números reais. Valem as seguintes afirmações:

(a) $x(A + B) = xA + xB$.
Demonstração: Como os termos das matrizes são números reais e x também o é, então vale a distributividade, logo

$$x(A + B) = (x(a_{ij} + b_{ij})) = (xa_{iij} + xb_{ij}) = xA + xB.$$

(b) $(x + y)A = xA + yA$.
Demonstração: Sabendo que $(x + y)a_{ij} = xa_{ij} + ya_{ij}$, então

$$(x + y)A = ((x + y)a_{ij}) = xa_{ij} + ya_{ij} = xA + yA.$$

(c) $x(yA) = (xy)A$.
Demonstração: Uma vez que vale a associatividade do produto de números reais, segue que

$$x(yA) = (x(ya_{ij})) = ((xy)a_{ij}) = (xy)A.$$

(d) $1 \cdot A = A$.
Demonstração: Como o número 1 é o elemento neutro multiplicativo no conjunto dos números reais e todos os termos da matriz A são reais, segue o resultado.

□

4. **Multiplicação de Matrizes**
Consideremos a matriz $A = (a_{ij})$, $m \times n$. Indicamos a i-ésima linha de A por A_i e a j-ésima coluna de A por A^j, ou seja,

$$A_i = \begin{pmatrix} a_{i1} & a_{i2} & \cdots & a_{in} \end{pmatrix} \text{ e } A^j = \begin{pmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{mj} \end{pmatrix}.$$

Por exemplo, para a matriz $A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \\ a_{41} & a_{42} & a_{43} \end{pmatrix}$, temos

$$A_3 = \begin{pmatrix} a_{31} & a_{32} & a_{33} \end{pmatrix} \text{ e } A^1 = \begin{pmatrix} a_{11} \\ a_{21} \\ a_{31} \\ a_{41} \end{pmatrix}.$$

Definição 2.2. *Consideremos as matrizes $A = (a_{ij})$, $m \times n$ e $B = (b_{jk})$, $n \times p$. Define-se o produto escalar da i-ésima linha da matriz A pela k-ésima coluna da matriz B como*

$$A_i \cdot B^k = \sum_{j=1}^{n} a_{ij} b_{jk} = a_{i1}b_{1k} + a_{i2}b_{2k} + \cdots + a_{in}b_{nk}.$$

Definição 2.3. *Sejam A e B as matrizes da definição anterior, temos que o produto $A \cdot B$ é por definição*

$$A \cdot B = \begin{pmatrix} A_1 \cdot B^1 & A_1 \cdot B^2 & \cdots & A_1 \cdot B^p \\ A_2 \cdot B^1 & A_2 \cdot B^2 & \cdots & A_2 \cdot B^p \\ \vdots & \vdots & \ddots & \vdots \\ A_m \cdot B^1 & A_m \cdot B^2 & \cdots & A_m \cdot B^p \end{pmatrix}.$$

Observamos que o produto resultante é uma matriz $m \times p$.

Propriedades da Multiplicação de Matrizes

(a) Se $A = (a_{ik})$ é uma matriz $m \times n$, então $I_m \cdot A = A = A \cdot I_n$.
Demonstração: Mostraremos inicialmente que $I_m \cdot A = A$. Seja $I_m = (b_{ij})$, assim pela definição de matriz identidade, $b_{ij} = 1$ se $i = j$ e $b_{ij} = 0$ se $i \neq j$. Temos que $(I_m)_i \cdot A^k$ é o termo geral do produto $I_m \cdot A$ e a_{ik} é o termo geral de A. Como duas matrizes são iguais se, e somente se, os seus termos correspondentes forem iguais, então basta mostrar que $(I_m)_i \cdot A^k = a_{ik}$. De fato, pela Definição 2 temos $(I_m)_i \cdot A^k = \sum_{j=1}^{n} b_{ij}a_{jk}$, mas $b_{ij} = 0$ se $i \neq j$, logo $\sum_{j=1}^{n} b_{ij}a_{jk} = b_{ii} \cdot a_{ik} = a_{ik}$, pois $b_{ii} = 1$. Portanto, $I_m \cdot A = A$.

Agora mostraremos que $A \cdot I_n = A$. Seja $I_m = (b_{jk})$, assim $b_{jk} = 1$ se $j = k$ e $b_{jk} = 0$ se $j \neq k$. Logo devemos ter $A_i \cdot (I_n)^k = a_{ik}$. De fato, $A_i \cdot (I_n)^k = \sum_{j=1}^{n} a_{ij} b_{jk} = a_{ik} \cdot b_{kk} = a_{ik}$, pois $b_{kk} = 1$ e $b_{jk} = 0$ sempre que $j \neq k$. Portanto, $I_m \cdot A = A$.

(b) Se B e C são matrizes $m \times n$ e A é uma matriz $n \times p$, então $(B+C)A = BA+CA$.
Demonstração: Sejam $B = (b_{ij})$, $C = (c_{ij})$ e $A = (a_{jk})$. Temos que o termo geral de $(B+C)A = (B+C)_i \cdot A^k$ e o termo geral de $BA+CA = B_i \cdot A^k + C_i \cdot A^k$. Assim, para provar a igualdade basta mostrar que $(B+C)_i \cdot A^k = B_i \cdot A^k + C_i \cdot A^k$. De fato, temos que

$$(B + C)_i \cdot A^k = \sum_{j=1}^{n} (b_{ij} + c_{ij}) a_{jk} = \sum_{j=1}^{n} (b_{ij} a_{jk} + c_{ij} a_{jk}) = \sum_{j=1}^{n} b_{ij} a_{jk} + \sum_{j=1}^{n} c_{ij} a_{jk} = B_i \cdot A^k + C_i \cdot A^k.$$

(c) Se A é uma matriz $m \times n$ e B e C são matrizes $n \times p$, então $A(B+C) = AB+AC$.
Demonstração: Análoga ao item anterior.

(d) Sejam $A = (a_{ij})$ uma matriz $m \times n$, $B = (b_{jk})$ uma matriz $n \times p$ e $x \in \mathbb{R}$. Então $(xA)B = A(xB) = x(AB)$.
Demonstração: Tomando os termos gerais das matrizes, devemos mostrar que $(xA)_i \cdot B^k = x(A_i \cdot B^k) = A_i (xB)^k$. De fato, temos que

$$(xA)_i \cdot B^k = \sum_{j=1}^{n} (x a_{ij}) b_{jk} = x(\sum_{j=1}^{n} a_{ij} b_{jk}) = x(A_i \cdot B^k) \text{ e}$$
$$A_i (xB)^k = \sum_{j=1}^{n} a_{ij} (x b_{jk}) = x(\sum_{j=1}^{n} a_{ij} b_{jk}) = x(A_i \cdot B^k).$$

Portanto, a igualdade é verificada.

(e) Sejam A, B e C matrizes $m \times n$, $n \times p$ e $p \times q$, respectivamente. Então $(AB)C = A(BC)$.
Demonstração: Ver [4]. □

Definição 2.4. *Seja n um número inteiro não negativo e A uma matriz quadrada. Definimos a n-ésima potência da matriz A, e denotamos por A^n, à recorrência*

$$\begin{cases} A^0 = I, \\ A^n = A^{n-1} \cdot A, \textit{para} \;\; n > 0 \end{cases}.$$

Observamos ainda que, para $n > 0$, $A^n = A.A...A$, onde A se repete n vezes.

2.1.4 Transposição de Matrizes

Seja A uma matriz $m \times n$. Chamamos *transposta* da matriz A, à matriz $n \times m$ definida por $A^t = (b_{ji})$, onde $b_{ji} = a_{ij}$.

Tomando a i-ésima coluna de A^t, temos $\begin{pmatrix} b_{1i} \\ b_{2i} \\ \vdots \\ b_{ni} \end{pmatrix} = \begin{pmatrix} a_{i1} \\ a_{i2} \\ \vdots \\ a_{in} \end{pmatrix}$. Isto é, a i-ésima coluna da matriz transposta corresponde a i-ésima linha da matriz dada.

Exemplo: Dadas as matrizes $A = \begin{pmatrix} 3 \\ 0 \\ -2 \\ 1 \end{pmatrix}$, $B = \begin{pmatrix} 1 & -5 \\ 2 & -2 \\ 0 & 4 \end{pmatrix}$ e $C = \begin{pmatrix} 6 & -9 \\ -1 & 5 \end{pmatrix}$.

Temos

$A^t = \begin{pmatrix} 3 & 0 & -2 & 1 \end{pmatrix}$, $B^t = \begin{pmatrix} 1 & 2 & 0 \\ -5 & -2 & 4 \end{pmatrix}$ e $C^t = \begin{pmatrix} 6 & -1 \\ -9 & 5 \end{pmatrix}$.

Propriedades:

Se A e B são matrizes $m \times n$, C é uma matriz $n \times p$ e $x \in \mathbb{R}$, então as propriedades a seguir são válidas.

1. $(A^t)^t = A$.
2. $(A + B)^t = A^t + B^t$.
3. $(xA)^t = x(A^t)$.
4. $(AC)^t = C^t \cdot A^t$.

2.1.5 Inversão de Matrizes

Sejam A e B matrizes $n \times n$ tais que $AB = I_n$, então dizemos que B é inversa à direita de A e que A é inversa à esquerda de B.

Quando uma matriz A, $n \times n$, assume inversa à direita e à esquerda, essas inversas são iguais, isto é, se $AB = I_n$ e $CA = I_n$, então $C = B$. Com efeito,

$$C = C \cdot I_n = C(AB) = (CA)B = I_n \cdot B = B.$$

Definição 2.5. *Uma matriz quadrada A é dita invertível se possuir inversa à direita e à esquerda. Como vimos essa matriz é única e é chamada matriz inversa de A, denotada por A^{-1}.*

Se A é invertível, então A^{-1} satisfaz $A \cdot A^{-1} = I$ e $A^{-1} \cdot A = I$.

Propriedades:

1. I_n é invertível e $(I_n)^{-1} = I_n$.
2. Se A é invertível, então $(A^{-1})^{-1} = A$.
3. Sejam A e B duas matrizes invertíveis, $n \times n$. O produto AB é invertível e $(AB)^{-1} = B^{-1} \cdot A^{-1}$.

Demonstração: Para a igualdade ser verdadeira devemos mostrar que

$$(AB) \cdot (B^{-1}.A^{-1}) = I_n \text{ e } (B^{-1}.A^{-1}) \cdot (AB) = I_n.$$

De fato,

$$(AB) \cdot (B^{-1}.A^{-1}) = A((B.B^{-1})A^{-1}) = A(I_n \cdot A^{-1}) = A \cdot A^{-1} = I_n.$$
$$(B^{-1}.A^{-1}) \cdot (AB) = B^{-1}((A^{-1}.A)B) = B^{-1}(I_n \cdot B) = B^{-1}.B = I_n.$$

□

2.2 Espaços Vetoriais

Seja V um conjunto não vazio no qual estão definidas duas operações, a saber *adição* e *multiplicação por escalar*, isto é, para todo u, $v \in V$ e $\alpha \in \mathbb{R}$, tem-se

$$u + v \in V \quad \text{e} \quad \alpha \cdot u \in V.$$

O conjunto V definido com as operações acima será chamado *Espaço Vetorial Real* se forem satisfeitas as propriedades a seguir.
Sejam u, v, $w \in V$ e α, $\beta \in \mathbb{R}$, então

1. $(u + v) + w = u + (v + w)$.
2. $u + v = v + u$.
3. Existe $0 \in V$ tal que $u + 0 = u$, para todo $u \in V$.
4. Para todo $u \in V$, existe $-u \in V$ tal que $u + (-u) = 0$.
5. $(\alpha\beta)u = \alpha(\beta u)$.
6. $(\alpha + \beta)u = \alpha u + \beta u$.
7. $\alpha(u + v) = \alpha u + \alpha v$.
8. $1.u = u$ para todo $u \in V$.

Observação: os elementos de um espaço vetorial V são chamados *vetores*.

Exemplos:

1. O conjunto $V = \mathbb{R}^n$ é um espaço vetorial com as operações de adição e multiplicação usuais. Em particular, os conjuntos $\mathbb{R}^2 = \{(x, y); x, y \in \mathbb{R}\}$ e $\mathbb{R}^3 = \{(x, y, z); x, y, z \in \mathbb{R}\}$ são espaços vetoriais.

2. O conjunto $M_{2\times 2}$ constituído pelas matrizes quadradas de ordem 2 (duas linhas e duas colunas) é um espaço vetorial.

3. O conjunto $P_n(x) = \{a_0 + a_1x + \cdots + a_nx^n; a_0, a_1, ..., a_n \in \mathbb{R}\}$ constituído pelos polinômios de grau menor ou igual a n, é um espaço vetorial.

2.2.1 Subespaços Vetoriais

Sejam V um espaço vetorial e W um subconjunto de V, não-vazio. O subconjunto W será um subespaço vetorial de V, se for um espaço vetorial com as operações induzidas de V.

Teorema 2.1. *Um subconjunto, não-vazio, W de um espaço vetorial V, será um subespaço vetorial de V, se forem satisfeitas as seguintes condições.*

1. *Para quaisquer $u, v \in W$ tem-se $u + v \in W$.*

2. *Para quaisquer $v \in W$ e $\alpha \in \mathbb{R}$ tem-se $\alpha v \in W$.*

Demonstração: Podemos observar que W é um subconjunto de V, assim herda as propriedades 1, 2, 5, 6, 7, e 8 da definição de espaço vetorial. Dessa forma, basta mostrar que valem as propriedades 3 e 4. De fato, da condição 2 do teorema, segue que para quaisquer $v \in W$ e $\alpha \in \mathbb{R}$ tem-se $\alpha v \in W$. Tomando $\alpha = 0$, temos $0 = 0.u \in$ W, o que mostra a propriedade 3, isto é, existe $0 \in W$ tal que $u + 0 = u$ para todo $u \in$ W. Tomando $\alpha = -1$, temos $-u = -1.u \in$ W. Logo, existe $-u \in$ W tal que $u + (-u) = 0$, o que mostra a propriedade 4. Portanto, são satisfeitas todas as propriedades de espaço vetorial. E com isso, segue que W é um subespaço do espaço vetorial V. □

Exemplos:
1. Considere o espaço vetorial $V = \mathbb{R}^3$ e $W = \{(x, y, z); y = 2x, z = -x\}$ um subconjunto de V. Temos que W é um subespaço vetorial de V. De fato, sejam $u, v \in W$ e $\alpha \in \mathbb{R}$, então $u = (x_1, 2x_1, -x_1)$ e $w = (x_2, 2x_2, -x_2)$, dessa forma

$$\begin{aligned} u + w &= (x_1, 2x_1, -x_1) + (x_2, 2x_2, -x_2) \\ &= (x_1 + x_2, 2x_1 + 2x_2, -x_1 - x_2) \\ &= (x_1 + x_2, 2(x_1 + x_2), -(x_1 + x_2)) \in W. \end{aligned}$$

$$\begin{aligned} \alpha u &= \alpha(x_1, 2x_1, -x_1) \\ &= (\alpha x_1, \alpha 2x_1, \alpha(-x_1)) \\ &= (\alpha x_1, 2(\alpha x_1), -(\alpha x_1)) \in W. \end{aligned}$$

2. Considere o espaço vetorial $M_{2\times 2} = \left\{ \begin{bmatrix} a & b \\ c & d \end{bmatrix}; a, b, c, d \in \mathbb{R} \right\}$ e o subconjunto $U = \left\{ \begin{bmatrix} a & b \\ c & d \end{bmatrix}; a = 2c, d = -b \right\}$. Temos que U é um subespaço vetorial de V, pois dados $u, v \in U$ e $\alpha \in \mathbb{R}$, segue que $u = \begin{bmatrix} 2c_1 & b_1 \\ c_1 & -b_1 \end{bmatrix}$ e $v = \begin{bmatrix} 2c_2 & b_2 \\ c_2 & -b_2 \end{bmatrix}$, logo

$$u + v = \begin{bmatrix} 2c_1 & b_1 \\ c_1 & -b_1 \end{bmatrix} + \begin{bmatrix} 2c_2 & b_2 \\ c_2 & -b_2 \end{bmatrix} = \begin{bmatrix} 2(c_1 + c_2) & b_1 + b_2 \\ c_1 + c_2 & -(b_1 + b_2) \end{bmatrix} \in U.$$

$$\alpha u = \alpha \begin{bmatrix} 2c_1 & b_1 \\ c_1 & -b_1 \end{bmatrix} = \begin{bmatrix} 2(\alpha c_1) & \alpha b_1 \\ \alpha c_1 & -(\alpha b_1) \end{bmatrix} \in U.$$

Interseção de Subespaços

Definição 2.6. *Sejam V um espaço vetorial e U e W subespaços de V. A interseção de U e W, representada por $U \cap W$, é o conjunto de todos os vetores $v \in V$ tais que $v \in U$ e $v \in W$.*

Teorema 2.2. *Se U e W são subespaços de um espaço vetorial V, então $U \cap W$ também é um subespaço vetorial de V.*

Demonstração: Sejam $u, v \in U \cap W$ e $\alpha \in \mathbb{R}$, devemos mostrar que $u + v \in U \cap W$ e $\alpha u \in U \cap W$.
(i) De $u, v \in U \cap W$, temos que $u, v \in U$ e $u, v \in W$. Como U e W são subespaços vetoriais, então $u + v \in U$ e $u + v \in W$, logo, segue que $u + v \in U \cap W$.
(ii) Dado $u \in U \cap W$, temos que $u \in U$ e $u \in W$. Como U e W são subespaços vetoriais, então existe $\alpha \in \mathbb{R}$ tal que $\alpha u \in U$ e $\alpha u \in W$, ou seja, $\alpha u \in U \cap W$. □

Soma de Subespaços

Definição 2.7. *Sejam V um espaço vetorial e U e W subespaços de V. Definimos a soma de U e V por*

$$U + W = \{u + w; u \in U \text{ e } w \in W\}.$$

Teorema 2.3. *Sejam U e W subespaços vetoriais de V, então $U + W$ também é um subespaço vetorial de V.*

Demonstração: Sejam $x, y \in U + W$ e $\alpha \in \mathbb{R}$. Devemos mostrar que $x + y \in U + W$ e $\alpha x \in U + W$.
(i) Como $x \in U + W$, então $x = u_1 + w_1$, onde $u_1 \in U$ e $w_1 \in W$. De $y \in U + W$, segue que $y = u_2 + w_2$, com $u_2 \in U$ e $w_2 \in W$. Assim, $x + y = (u_1 + w_1) + (u_2 + w_2) = (u_1 + u_2) + (w_1 + w_2) \in U + W$.
(ii) Dado $\alpha \in \mathbb{R}$, temos $\alpha x = \alpha(u_1 + w_1) = \alpha u_1 + \alpha w_1$. Como U e W são subespaços vetoriais, então $\alpha u_1 \in U$ e $\alpha w_1 \in W$, portanto, $\alpha x \in U + W$. □

2.2.2 Combinação Linear

Definição 2.8. *Seja V um espaço vetorial e $v_1, v_2, ..., v_n$ vetores pertencentes a V. Dado $v \in V$, se existirem números reais $a_1, a_2, ..., a_n$, tais que*

$$v = a_1 v_1 + a_2 v_2 + \cdots + a_n v_n$$

então dizemos que v é uma combinação linear dos vetores $v_1, v_2, ..., v_n$.

Exemplo: Considere em $\mathbb{R}^2$ o vetor $v = (1, 10)$. Temos que v pode ser escrito como uma combinação linear dos vetores $v_1 = (1, 2)$ e $v_2 = (-1, 2)$. De fato, pois fazendo $v = av_1 + bv_2$, temos

$$(1, 10) = a(1, 2) + b(-1, 2) \Rightarrow (1, 10) = (a - b, 2a + 2b).$$

Esta equação é equivalente ao sistema

$$\begin{cases} a - b = 1 \\ 2a + 2b = 10. \end{cases}$$

Resolvendo esse sistema encontramos a sua única solução, a qual é $a = 3$ e $b = 2$. Logo, $v = 3v_1 + 2v_2$.

Subespaços Gerados

Seja V um espaço vetorial e $A = \{v_1, v_2, ..., v_n\}$ um subconjunto de V. O conjunto S, de todas as combinações lineares dos vetores de A é um subespaço vetorial de V.

Dizemos que S é gerado pelos vetores $v_1, v_2, ..., v_n$ ou que é gerado pelo conjunto A. Denotamos por $S = [v_1, v_2, ..., v_n]$ ou $S = G(A)$. Os vetores $v_1, v_2, ..., v_n$ são chamados geradores de S, enquanto A é chamado conjunto gerador de S.

2.2.3 Dependência e Independência Linear

Definição 2.9. *Seja $A = \{v_1, v_2, ..., v_n\}$ um subconjunto de um espaço vetorial V. Dizemos que o conjunto A é linearmente independente (L.I.), ou que os vetores $v_1, v_2, ..., v_n$ são L.I. se*

$$a_1v_1 + a_2v_2 + \cdots + a_nv_n = 0$$

implica que $a_1 = a_2 = \cdots = a_n = 0$.

Definição 2.10. *Dado A como na definição anterior, se na equação*

$$a_1v_1 + a_2v_2 + \cdots + a_nv_n = 0$$

houver $a_i \neq 0$, para algum $i = 1, 2, ..., n$, dizemos que os vetores são linearmente dependentes (L.D.).

A partir da Definição 2.10, podemos observar que se em um conjunto $A = \{v_1, v_2, ..., v_n\}$ pudermos escrever um dos vetores como combinação linear dos demais, então o conjunto é linearmente dependente.

2.2.4 Base e Dimensão

Definição 2.11. *Sejam V um espaço vetorial e $B = \{v_1, v_2, ..., v_n\} \subset V$. Dizemos que B é uma base de V se:*

1. *B é linearmente independente;*
2. *B gera V (qualquer vetor de V pode ser escrito como combinação linear dos vetores de B).*

Exemplo: O conjunto $B = \{(1,0,0), (0,1,0), (0,0,1)\}$ é uma base de $\mathbb{R}^3$. De fato,
(i) Sejam $a, b, c \in \mathbb{R}$. Assim,

$$a(1,0,0) + b(0,1,0) + c(0,0,1) = (0,0,0) \Rightarrow (a,b,c) = (0,0,0) \Rightarrow a = b = c = 0$$

logo B é L.I.
(ii) Tomando $(x, y, z) \in \mathbb{R}^3$, temos

$$(x,y,z) = (x,0,0) + (0,y,0) + (0,0,z) = x(1,0,0) + y(0,1,0) + z(0,0,1),$$

ou seja, qualquer vetor de $\mathbb{R}^3$ pode ser escrito como combinação linear dos vetores de B. Assim, B gera $\mathbb{R}^3$.

Teorema 2.4. *Dada uma base B de V, todo vetor de V é escrito de maneira única, como combinação linear dos vetores de B.*

Demonstração: Ver [14], página 75. □

Teorema 2.5. *Se $B = \{v_1, ..., v_n\}$ é uma base de um espaço vetorial V, então qualquer conjunto com mais de n vetores será linearmente dependente.*

Demonstração: Ver [14], página 71. □

Teorema 2.6. *Duas bases quaisquer de um espaço vetorial V possuem o mesmo número de vetores.*

Demonstração: Sejam $B_1 = \{u_1, ..., u_m\}$ e $B_2 = \{v_1, ..., v_n\}$ duas bases de um espaço vetorial V. Mostraremos que $m = n$. De fato, como B_1 é base de V, então se $m < n$ teríamos que B_2 seria L.D, o que contraria o fato de ser base, logo $m \leq n$. Por outro lado, como B_2 é base de V, se $n < m$, então B_1 seria L.D. o que não pode ocorrer, pois B_1 é base de V, assim temos que $n \leq m$. Portanto, de $m \leq n$ e $n \leq m$, segue que $m = n$. □

Definição 2.12. *Seja V um espaço vetorial finito. Se V possui um base com n vetores, então dizemos que a dimensão de V é n, e denotamos por $dimV = n$.*

Teorema 2.7. *Se $dimV = n$, qualquer conjunto de vetores L.I. formará uma base de V.*

Demonstração: Ver [4], página 171. □

Proposição 2.8. *Seja V um espaço vetorial de dimensão n. Qualquer conjunto de vetores L.I. em V pode ser completado até formar uma base.*

Demonstração: Ver [14], página 74. □

2.3 Transformações Lineares

Estudaremos agora um tipo especial de função, em que o domínio e o contradomínio são espaços vetoriais reais.

$$\begin{array}{ccc} V & \longrightarrow & W \\ v & \mapsto & T(v) \end{array}$$

Definição 2.13. *Sejam V e W espaços vetoriais reais. Uma função $T : V \to W$ é chamada transformação linear de V em W se para todo $u, v \in V$ e $\alpha \in \mathbb{R}$ tem-se*

1. $T(u + v) = T(u) + T(v)$;
2. $T(\alpha u) = \alpha T(u)$.

Exemplo: A função $T : \mathbb{R}^2 \to \mathbb{R}^3; T(x, y) = (3x, 2y, x + y)$ é uma transformação linear. De fato, dados $u, v \in \mathbb{R}^2$, temos $u = (x_1, y_1)$ e $v = (x_2, y_2)$. Assim,

1.

$$\begin{aligned} T(u+v) &= T((x_1,y_1)+(x_2,y_2)) \\ &= T(x_1+x_2, y_1+y_2) \\ &= (3(x_1+x_2), 2(y_1+y_2), (x_1+x_2)+(y_1+y_2)) \\ &= (3x_1, 2y_1, x_1+y_1)+(3x_2, 2y_2, x_2+y_2) \\ &= T(u)+T(v). \end{aligned}$$

2.

$$\begin{aligned} T(\alpha u) &= T(\alpha(x_1,y_1)) \\ &= T(\alpha x_1, \alpha y_1) \\ &= (3(\alpha x_1), 2(\alpha y_1), \alpha(x_1+\alpha y_1) \\ &= \alpha(3x_1, 2y_1, x_1+y_1) \\ &= \alpha T(u). \end{aligned}$$

2.3.1 Propriedades

Seja $T: V \to W$ uma transformação linear, então valem as seguintes propriedades

1. A imagem do vetor $0 \in V$ é o vetor $\mathbf{0} \in W$, ou seja, $T(0) = \mathbf{0}$.
2. Para todo $u, v \in V$

(a) $T(-v) = -T(v)$

(b) $T(u-v) = T(u) - T(v)$.

3. Para todos $v_1, v_2, ..., v_n \in V$ e $a_1, a_2, ..., a_n \in \mathbb{R}$

$$T(a_1v_1 + a_2v_2 + \cdots a_nv_n) = a_1T(v_1) + a_2T(v_2) + \cdots + a_nT(v_n).$$

Em palavras, a imagem da combinação linear dos vetores é a combinação linear das imagens dos vetores. □

2.3.2 Núcleo e Imagem de uma Transformação Linear

Definição 2.14. *Chama-se núcleo de uma Transformação Linear ao conjunto de todos os vetores $v \in V$ que são transformados em $0 \in W$. Indicamos o Núcleo da transformação por $Ker(T)$, logo*

$$Ker(T) = \{v \in V; T(v) = 0\}.$$

Observamos que $Ker(T) \neq \emptyset$, pois $0 \in Ker(T)$, uma vez que $T(0) = 0$.

Proposição 2.9. *O núcleo de uma Transformação Linear $T: V \to W$ é um subespaço vetorial de V.*

Demonstração: Ver [14], página 170. □

Definição 2.15. *Chama-se imagem de uma transformação linear $T : V \to W$ ao conjunto de vetores $w \in W$ que são imagem de pelo menos um $v \in V$. Indicamos o conjunto imagem por $Im(T)$. Assim,*

$$Im(T) = \{w \in W; w = T(v) \text{ para algum } v \in V\}.$$

Podemos ver que $Im(T) \subset W$ e $Im(T) \neq \emptyset$, pois $0 = T(0) \in Im(T)$. Alem disso, se $Im(T) = W$, então T é sobrejetora, isto é, para todo $w \in W$ existe pelo menos um $v \in V$ tal que $T(v) = w$.

Proposição 2.10. *A imagem de uma transformação linear $T : V \to W$ é um subespaço vetorial de W.*

Demonstração: Ver [14], página 173. □

Teorema 2.11. *(**Teorema do Núcleo e da Imagem**) Sejam V e W espaços vetoriais de dimensão finita e $T : V \to W$ uma transformação linear. Então*

$$dimV = dimKer(T) + dimIm(T).$$

Demonstração: Seja $B_1 = \{u_1, ..., u_m\}$ uma base para $Ker(T)$, logo $dimKer(T) = m$. Como $Ker(T) \subset V$ é um subespaço vetorial de V, então podemos completar B_1 até formarmos uma base para V, digamos

$$B_2 = \{u_1, ..., u_m, v_1, ..., v_n\}.$$

Observemos que $dimV = m + n$, dessa forma, basta mostrar que $dimIm(T) = n$. Para isso, mostremos que $T(v_1), ..., T(v_n)$ é uma base de $Im(T)$. Isto é,
(i) $T(v_1), ..., T(v_n)$ são L.I.
Temos que

$$a_1T(v_1) + \cdots + a_nT(v_n) = 0 \Rightarrow T(a_1v_1 + \cdots + a_nv_n) = 0 \Rightarrow a_1v_1 + \cdots + a_nv_n \in Ker(T).$$

Assim, podemos escrever $a_1v_1 + \cdots + a_nv_n$ como combinação linear dos elementos de B_1, que formam uma base para $Ker(T)$. Logo,

$$a_1v_1 + \cdots + a_nv_n = b_1u_1 + \cdots + b_mu_m \Rightarrow a_1v_1 + \cdots + a_nv_n - b_1u_1 - \cdots - b_mu_m = 0.$$

Mas $B_2 = \{u_1, ..., u_m, v_1, ..., v_n\}$ é uma base para V, e com isso segue que

$$a_1 = \cdots = a_n = b_1 = \cdots = b_m = 0.$$

Portanto, $T(v_1), ..., T(v_n)$ são L.I.
(ii) $T(v_1), ..., T(v_n)$ geram $Im(T)$.
Seja $w \in Im(T)$, então existe $v \in V$ tal que $T(v) = w$. Assim,

$$v = a_1u_1 + \cdots + a_mu_m + b_1v_1 + \cdots + b_nv_n.$$

Logo,

$$w = T(v) = T(a_1u_1 + \cdots + a_mu_m + b_1v_1 + \cdots + b_nv_n) = a_1T(u_1) + \cdots + a_mT(u_m) + b_1T(v_1) + \cdots + b_nT(v_n).$$

No entanto, como $u_1, ..., u_m \in Ker(T)$, então $T(u_1) = \cdots = T(u_m) = 0$. Assim,

$$w = b_1T(v_1) + \cdots + b_nT(v_n).$$

Portanto, $T(v_1), ..., T(v_n)$ geram $Im(T)$.
Segue então que $\{T(v_1), ..., T(v_n)\}$ são uma base de $Im(T)$, daí $dimIm(T) = n$. E

$$dimV = m + n \Rightarrow dimV = dimKer(T) + dimIm(T).$$

□

Corolário 2.12. *Sejam V e W espaços vetoriais de diminesão finita e $T : V \to W$ uma transformação linear. Se $dimV = dimW$ então T é injetora se, e somente se, é sobrejetora.*

Demonstração: Ver [14], página 179. □

2.4 Anéis e Corpos

Seja A um conjunto não vazio em que estão definidas duas operações, adição ($+$) e multiplicação ($\cdot$).

2.4.1 Definições

Definição 2.16. *A terna $(A, +, \cdot)$ será chamada anel se forem satisfeitas as condições a seguir:*

1. *(associatividade da adição): $a + (b + c) = (a + b) + c$, para todo $a, b, c \in A$.*
2. *(existência do elemento neutro aditivo): existe $0 \in A$ tal que $0 + a = a + 0 = a$ para todo $a \in A$.*
3. *(existência do elemento simétrico): para todo $a \in A$, existe $b \in A$ tal que $a + b = b + a = 0$.*
4. *(comutatividade da adição): $a + b = b + a$, para todo $a, b \in A$.*
5. *(associatividade da multiplicação): $a \cdot (b \cdot c) = (a \cdot b) \cdot c$, para todo $a, b, c \in A$.*

6. *(distributividade da multiplicação em relação a adição):* $a \cdot (b + c) = a \cdot b + a \cdot c$, *para todo* $a, b, c \in A$.

Observações:

1. Na maioria das vezes indicamos o anel $(A, +, \cdot)$ como apenas A, sendo que ficam subentendidas as operações de adição e multiplicação.

2. Se existe $1 \in A$ tal que $a \cdot 1 = 1 \cdot a = a$, para todo $a \in A$, dizemos que A é um anel com unidade 1.

3. Se $a \cdot b = b \cdot a$, para todo $a, b \in A$, dizemos que A é um anel comutativo.

4. O elemento simétrico é único.

5. Em um anel vale a lei do corte, isto é, dados $x, y, z \in A$, se $x \cdot y = x \cdot z \Rightarrow y = z$, para todo $x \neq 0$.

6. Para todo $x \in A$ tem-se $x \cdot 0 = 0$.

Denotaremos por A^* o conjunto $A \setminus \{0\}$.

Definição 2.17. *Um anel D é chamado domínio de integridade quando para todo $x, y \in D^*$, se $x \cdot y = 0$, então $x = 0$ ou $y = 0$.*

A definição acima nos faz pensar que este é um resultado óbvio, no entanto, nem sempre ele é verdadeiro. Tomando, por exemplo, no anel das matrizes quadradas de ordem 2, as matrizes $A = \begin{pmatrix} 1 & 2 \\ 1 & 2 \end{pmatrix}$ e $B = \begin{pmatrix} 2 & 2 \\ -1 & -1 \end{pmatrix}$, temos que

$$A \cdot B = \begin{pmatrix} 1 & 2 \\ 1 & 2 \end{pmatrix} \cdot \begin{pmatrix} 2 & 2 \\ -1 & -1 \end{pmatrix} = \begin{pmatrix} 2-2 & 2-2 \\ 2-2 & 2-2 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

Ou seja, tomamos dois elementos diferentes do elemento neutro das matrizes, que é a matriz nula (O), mas que o produto entre eles resulta neste elemento. Isto é, $A \neq O$ e $B \neq O$, porém $A \cdot B = O$.

Um exemplo de *domínio de integridade* é o anel $\mathbb{Z}$ (conjunto dos números inteiros), com as operações de adição e multiplicação usuais.

Definição 2.18. *Seja K um anel comutativo com unidade. K é chamado de corpo se para todo $x \in K^*$, existe $y \in K$ tal que $x \cdot y = 1$.*

Da definição de corpo, segue que todo $x \in K$ possui um *inverso multiplicativo*, o qual será denotado por x^{-1}.

São exemplos de *corpos* os conjuntos $\mathbb{Q}$ (conjunto dos números racionais), $\mathbb{R}$ (conjunto dos números reais) e $\mathbb{C}$ (conjunto dos números complexos), com as operações de adição e multiplicação.

Proposição 2.13. *O inverso multiplicativo de um elemento $x \neq 0$ em um corpo é único.*

Demonstração: Sejam y e z ambos inversos multiplicativos de $x \neq 0$, isto é, $y \cdot x = 1$ e $x \cdot z = 1$. Devemos mostrar que $y = z$. De fato,

$$y = y \cdot 1 = y \cdot (x \cdot z) = (y \cdot x) \cdot z = 1 \cdot z = z.$$

Logo $y = z$. □

Note que na segunda igualdade usamos o fato que $1 = x \cdot z$, na terceira igualdade usamos a associatividade do produto, e na penúltima igualdade usamos que $y \cdot x = 1$.

Se $x \in K$ possui inverso multiplicativo, então dizemos que x é *invertível*.

Definição 2.19. *Seja A um anel e S um subconjunto não vazio de A. Dizemos que S é um subanel de A se valem as seguintes condições:*

1. *S é fechado para as operações de adição e multiplicação de A. Isto é, dados $a, b \in S$ tem-se que $a + b \in S$ e $a \cdot b \in S$.*
2. *S é também um anel.*

Podemos ver, por exemplo, que o conjunto $\mathbb{Q}$ é um subanel de $\mathbb{R}$.

Proposição 2.14. *Seja A um anel e S um subconjunto não vazio de A. S será um subanel de A se, e somente se, para todo $a, b \in S$ valem as seguintes condições:*

1. *$0 \in S$;*
2. *$a - b \in S$;*
3. *$a \cdot b \in S$.*

Demonstração: ($\Rightarrow$) Se S é um subanel de A então as operações induzidas de A são válidas em S e assim, para todo $a, b \in S$ tem-se

$$a - b = a + (-b) \in S \quad \text{e} \quad a \cdot b \in S,$$

e ainda $0 = a + (-a) \in S$.
($\Leftarrow$) Seja S um subconjunto de A tal que para todo $a, b \in S$, tem-se $0 \in S$, $a - b \in S$ e $a \cdot b \in S$. Pela primeira condição, temos que S possui elemento neutro aditivo. Dado $b \in S$, da segunda condição temos $-b = 0 - b \in S$, isto é, todo elemento de S possui simétrico. Temos ainda que $a + b = a - (-b) \in S$, ou seja, S é fechado para a operação de adição, valendo as propriedades associativa e comutativa da adição e a distributividade. De $a \cdot b \in S$, segue que S é fechado para o produto, herdando assim, a propriedade associativa da multiplicação. Dessa forma, vemos que as seis condições de anel são satisfeitas e portanto S é um subanel de A. □

2.4.2 O corpo $\mathbb{F}_2$

Vimos na seção anterior alguns exemplos de corpos como os conjuntos dos números racionais ($\mathbb{Q}$) e dos reais ($\mathbb{R}$). Esses corpos possuem infinitos elementos, porém existem outros que contém uma quantidade finita de elementos, chamados de corpos finitos.

Nessa seção estudaremos o corpo finito $\mathbb{F}_2 = \{0, 1\}$, o qual é formado pelos possíveis restos na divisão de um número inteiro qualquer por 2. Este corpo, que possui apenas dois elementos, também é chamado de conjunto binário e será utilizado com frequência nos próximos capítulos.

Por se tratar de um corpo, no conjunto $\mathbb{F}_2$ estão definidas duas operações, adição e multiplicação, em que podemos ver suas tabelas a seguir:

$+$	0	1
0	0	1
1	1	0

$\times$	0	1
0	0	0
1	0	1

.

Ao observamos a tabela da soma, é normal nos questionar: por que 1+1=0? Então, como foi dito anteriormente, o corpo $\mathbb{F}_2$ é formado pelos restos possíveis na divisão de qualquer número inteiro por 2. E ao realizarmos uma operação dentro deste conjunto, o resultado obtido deve pertencer ao mesmo. Dessa forma, ao fazermos a operação de soma temos

$$1 + 1 = 2 = 2.1 + 0$$

ou seja, ao dividirmos 2 por 2, temos que o resto é zero (0) e portanto será o resultado desejado.

Observamos que as propriedades de corpo são satisfeitas no conjunto $\mathbb{F}_2$, isto é, valem as propriedades comutativa da adição e multiplicação, associativa da adição e multiplicação, distributiva, elementos neutros aditivo e multiplicativo(0 para adição e 1 para multiplicação) e elementos simétrico e inverso.
Os elementos simétrico e inverso são ambos iguais a 1, pois

$$1 + 1 = 0 \text{ (elemento neutro aditivo)}$$

e

$$1 \times 1 = 1 \text{ (elemento neutro multiplicativo).}$$

Uma outra característica do corpo $\mathbb{F}_2$ é que podemos definir espaços vetoriais da forma $\mathbb{F}_2^n = \{(a_1, a_2, ..., a_n); a_1, a_2, ..., a_n \in \mathbb{F}_2\}$, assim como fazemos em $\mathbb{R}^n$. Por exemplo, para $n = 2$ e $n = 3$, temos, respectivamente,

$$\mathbb{F}_2^2 = \{(0, 0).(0, 1), (1, 0), (1, 1)\}$$

e

$$\mathbb{F}_2^3 = \{(0,0,0),(0,0,1),(0,1,0),(1,0,0),(0,1,1),(1,0,1),(1,1,0),(1,1,1)\}.$$

Podemos notar que o número de elementos de $\mathbb{F}_2^2$ é igual a $4 = 2^2$ e da mesma forma, o número de elementos de $\mathbb{F}_2^3$ é igual a $8 = 2^3$. Diante disso, somos levados a pensar: será que em $\mathbb{F}_2^4$ o número de elementos é $2^4 = 16$? E em $\mathbb{F}_2^7$, será $2^7 = 128$? A resposta para essas perguntas é sim, e veremos que tal fato é comprovado nos capítulos subsequentes.

Observação Existem muitos outros corpos finitos. Pode-se demonstrar que todo conjunto da forma $\mathbb{F}_p = \{1, 2, ..., p-1\}$, sendo p um número primo, é um corpo. Porém este assunto requer um estudo mais aprofundado em Álgebra Abstrata. Para o leitor que tiver interesse no assunto, as referências [5], [6], [7] e [16] contém um conteúdo mais aprofundado.

3 CÓDIGOS CORRETORES DE ERROS

A partir de agora nos debruçaremos sobre o estudos dos Códigos Corretores de Erros, cuja teoria é uma aplicação dos conteúdos matemáticos vistos no capítulo anterior. Porém, isso nos traz algumas indagações como: o que é Código Corretor de Erros? Para que servem?

Veremos neste capítulo o que é um Código Corretor de Erros e como funciona, e também mostraremos sua importância no mundo moderno das comunicações. Inicialmente vejamos como surgiu essa teoria.

3.1 Aspectos Históricos

A Teoria dos Códigos Corretores de Erros teve início na década de 40, quando em 1947, o matemático Richard W. Hamming, começou seus estudos sobre códigos no Laboratório Bell de Tecnologia, em Nova Jersey, EUA. Hamming tinha o Laboratório à sua disposição somente no fim de semana, onde usava os computadores ali disponíveis para realizar suas pesquisas. Porém, quando as máquinas, ao executarem um programa, encontravam um erro, a leitura do programa era interrompida impossibilitando a continuidade da pesquisa.

Ao se deparar com esses erros por dois finais de semana consecutivos, Hamming não conseguia dar andamento em seus estudos diante da paralisação dos computadores ao encontrar um erro durante a leitura do programa. A partir dessas dificuldades Hamming desenvolveu um código que tinha a capacidade de detectar até dois erros e corrigir um.

Conforme sua pesquisa avançava, internamente no Laboratório Bell, Hamming publicou alguns artigos sobre os códigos criados por ele. No entanto, diante desses trabalhos Hamming estudava a possibilidade da criação de códigos que tivessem maior eficiência que o primeiro por ele desenvolvido. A resposta para essas perguntas foi dada em 1948, por Claude E. Shannon, na publicação do artigo intitulado "*A Mathematical Theory of Communication*", que inclusive contava com partes das pesquisas desenvolvidas pelo próprio Hamming. Shannon também trabalhava no Laboratório Bell e a partir de seu trabalho desevolveu-se dois campos de pesquisas matemáticas, os quais foram a Teoria dos Códigos e a Teoria da Informação. O trabalho de Hamming sobre códigos, devido a alguns

problemas de patente com o Laboratório Bell, só foi publicado em 1950.

Em face do chamado (7,4)-código de Hamming, descrito no trabalho de Shannon, em 1949, Marcel J. E. Golay, que trabalhava no *Signal Corps Enginneering Laboratories at Fort Monmouth*, publicou no *Proceedings of the I. R. E. (I.E.E.E.)*, um artigo intitulado "*Notes on Digital Coding*", que apesar de ter apenas uma página é considerado até hoje um dos mais importantes trabalhos da Teoria dos Códigos.

Com base neste trabalho Golay desenvolveu o que chamamos atualmente de (23,12) e (11,6) códigos de Golay. Para se ter uma noção da dimensão da importância deste trabalho, em 1979, a nave espacial *Voyager* conseguiu transmitir fotografias coloridas de Júpter e Saturno, usando o chamado (21,4096)-código de Golay.

Podemos dizer então que Hamming, Shannon e Golay foram os pioneiros no estudo dos Códigos Corretores de Erros, desenvolvendo pesquisas que são aplicadas até a atualidade.

3.2 O que é um Código Corretor de Erros?

Antes de darmos a definição formal de Códigos Corretores de Erros, vejamos onde tal teoria se aplica e façamos um exemplo simples para entender como funciona esse processo de correção de erros.

No atual mundo das comunicações, estamos a todo momento enviando e recebendo informações, seja através da internet, televisão, rádio, etc. Ao enviarmos uma mensagem no celular, por exemplo, o destinatário recebe exatamente como foi enviada. Ou quando um programa de TV é transmitido, recebemos a imagem em nossa televisão igual a imagem que foi produzida.

No entanto, ao recebermos uma mensagem, ou imagem, exatamente como esta foi enviada, não significa que a transmissão deu-se de forma perfeita, mas que se houve algum erro durante o processo, este foi detectado e corrigido para que recebêssemos a mensagem corretamente. Para garantir que isso aconteça é que existe a Teoria dos Códigos Corretores de Erros.

As transmissões de informação são realizadas através do que chamamos de *canal*, que pode ser um cabo, radiofrequência, circuito integrado digital, entre outros. Durante esse processo de envio da mensagem o canal pode sofrer interferências, chamados de *ruídos*, causando erros que alteram o conteúdo enviado. Vejamos um exemplo.

Exemplo 3.1 Considere, em um jogo de computador, um tabuleiro de xadrez e sobre ele a peça denominada *Torre*, a qual só pode realizar movimentos horizontais e verticais. Digamos que tais ações podem ser enxergadas como direções, isto é, Norte, Sul, Leste e Oeste. Vamos *codificar* os movimentos da Torre como elementos de $\mathbb{F}_2 \times \mathbb{F}_2$, ou seja,

$$\begin{array}{ll} \text{Norte} \longmapsto 00 & \text{Leste} \longmapsto 10 \\ \text{Sul} \longmapsto 11 & \text{Oeste} \longmapsto 01 \end{array}$$

As informações iniciais, no caso as direções, são chamadas de *fonte*, enquanto os símbolos (números) que as representam são chamados *código da fonte*. Supondo que o *código do canal* seja o mesmo da fonte, imaginemos que ao darmos o comando 00, por um erro no canal, a mensagem recebida seja 01, então ao invés da Torre ir para Norte, que é a mensagem original, ela irá para Leste, uma vez que não podemos identificar que houve erro, assim, a peça fará o movimento incorreto.

Para melhorar o código de canal, vamos acrescentar dígitos aos códigos da fonte de modo a triplicar os pares, isto é,

$$\begin{aligned} \text{Norte} &\longmapsto 000000 \\ \text{Sul} &\longmapsto 111111 \\ \text{Leste} &\longmapsto 101010 \\ \text{Oeste} &\longmapsto 010101. \end{aligned}$$

A esse acréscimo de informação chamamos *redundância*. Agora suponhamos que a mensagem recebida tenha sido 101011. Logo observamos que ocorreu um erro, pois essa mensagem não corresponde a nenhuma das direções. Porém, notamos que a palavra 101010 difere da mensagem recebida apenas de um dígito, enquanto as demais palavras tem muitos dígitos diferentes, dessa forma podemos detectar e corrigir o erro e assim a Torre irá para a direção correta, ou seja, Leste.

Este processo de detecção e correção de erros em um canal de transmissão pode ser resumido como se segue

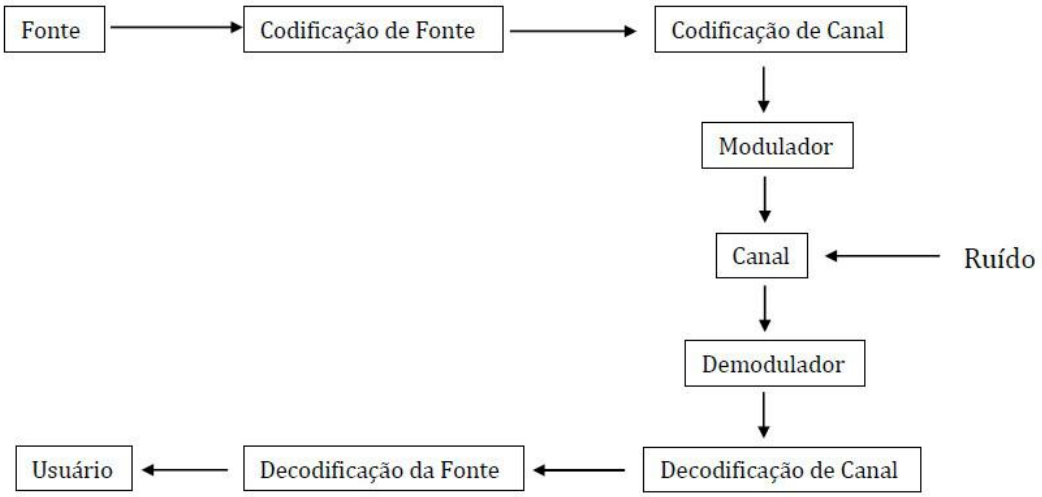

Figura 1: Sistema de Comunicação.

Assim, o objetivo base da Teoria dos Códigos Corretores de Erros é codificar a mensagem inicial, para assim detectar e corrigir possíveis erros que podem ocorrer durante a transmissão de uma mensagem, e dessa forma, o destinatário receber a informação conforme foi enviada originalmente.

Para este trabalho vamos considerar apenas canais simétricos, ou seja, onde todos os símbolos a serem transmitidos possuem a mesma probabilidade de serem recebidos errados e além disso, se um símbolo é recebido de forma errada, a probabilidade de ele ser qualquer um dos outros é a mesma.

Vejamos agora a definição formal de Código Corretor de Erros.

3.2.1 Códigos de Bloco

Consideremos inicialmente um conjunto finito A, o qual será chamado *alfabeto*. Um *Código Corretor de Erros* é um subconjunto próprio de A^n, onde n é um número natural qualquer.

Denotamos por $|A| = q$ o número de elementos de A, e dizemos assim que A é um código q-ário. As sequências finitas formadas pelos símbolos de A são chamadas de *palavras*, enquanto n representa o comprimento do código, isto é, o número de símbolos das palavras. No Exemplo 3.1, da peça de xadrez, tínhamos $A = \mathbb{F}_2 = \{0, 1\}$, ou seja, o alfabeto tinha apenas dois elementos. Este código é chamado de *binário*. Além disso, as palavras do código possuem seis dígitos, portanto o comprimento do código é 6.

Os Códigos Corretores de Erros C de comprimento n estão agrupados em palavras de igual comprimento, por isso dizemos que C é um *código de bloco*. Porém, todos os códigos que estudaremos neste trabalho são de bloco, assim omitiremos esse termo.

Um exemplo bem simples de Código Corretor de Erros é um idioma. Por exemplo, seja A um alfabeto composto pelas 26 letras do alfabeto da língua portuguesa, o c cedilha (ç), as vogais acentuadas (á,à,â,ã,é,ê,í,ó,ô,õ,ú) e o espaço em branco, totalizando assim 39 termos. Qualquer palavra da língua portuguesa pode ser vista como um elemento de A^{46}, onde 46 é o comprimento de sua maior palavra (consideramos aqui a palavra pneumoultramicroscopicossilicovulcanoconiótico). Para que não se tenham repetições desnecessárias, em palavras de menor comprimento são adicionados espaços em branco ao lado direito de cada palavra, estes espaços são omitidos durante a escrita.

Por ser, a língua portuguesa, um subconjunto próprio C de A^{46} então podemos considerá-lo de certa forma um código detector e corretor de erros. Com efeito, ao enviarmos a palavra *caderno* e por um erro de transmissão recebermos a palavra *cadermo*, vemos que a palavra está errada, pois não é um elemento de C. Vemos assim, que houve um erro, o qual pode ser detectado e corigido, uma vez que a palavra de C que mais se aproxima é *caderno*, justamente a palavra transmitida. Porém, este código não traz muita eficiência na correção pois, se ao transmitirmos a palavra *mala* e, por um erro na transmissão, recebermos a palavra *fala* ou *maca*, não podemos detectar o erro, pois todas essas palavras pertencem a língua portuguesa, C. Como existem muitas palavras na língua portuguesa que são parecidas, então este código não é eficiente para detectar e corrigir erros.

Existem outros tipos de códigos, um dos mais utilizados no nosso cotidiano são os que possuem dígitos de verificação, responsáveis por detectar e corrigir eventuais erros. Vemos a utilização desse tipo de códigos no CPF, em números de cartões de créditos, códigos de barras, ISBN (*International Standard Book Number*), que é o número de "identidade" de um livro, entre outros. Veremos alguns exemplos desses códigos no último capítulo.

3.3 Métrica de Hamming

Para indicar a proximidade entre palavras de um código, usamos uma função chamada *distância*, a qual possui algumas características próprias.

Uma função $d : A^n \times A^n$ é dita uma função distância se, e somente se, satisfaz as propriedades a seguir:

i) Positividade: $d(\mathbf{u}, \mathbf{v}) \geq 0$, valendo a igualdade se, e somente se, $\mathbf{u} = \mathbf{v}$;

ii) Simetria: $d(\mathbf{u}, \mathbf{v}) = d(\mathbf{v}, \mathbf{u})$;

iii) Desigualdade Triangular: $d(\mathbf{u}, \mathbf{v}) \leq d(\mathbf{u}, \mathbf{w}) + d(\mathbf{w}, \mathbf{v})$,

sendo $\mathbf{u}$, $\mathbf{v}$ e $\mathbf{w}$ elementos de A^n.

A função distância caracteriza o que chamamos, em matemática, de métrica. Veremos a seguir uma função distância muito importante na Teoria dos Códigos.

Definição 3.20. *Dados dois elementos* $\boldsymbol{u}= (u_1, u_2, ..., u_n)$, $\boldsymbol{v}= (v_1, v_2, ..., v_n) \in A^n$, *a distância de Hamming entre* $\boldsymbol{u}$ *e* $\boldsymbol{v}$ *é definida como*

$$d(u, v) = |\{i; u_i \neq v_i, 1 \leq i \leq n\}|.$$

Relembrando que $|A|$ é a quantidade de elementos do conjunto.

Por exemplo, em $\mathbb{F}_2^4$, temos

$$d(0011, 1111) = 2$$
$$d(1001, 1101) = 1$$
$$d(1110, 0001) = 4.$$

A distância de Hamming entre elementos de A^n satisfaz as três propriedades de distância e por isso também é chamada de *métrica de Hamming*.

A distância de Hamming pode ser interpretada geometricamente. Sendo $A = \mathbb{F}_2$, temos que $\mathbb{F}_2^n$ representa os vértices de um hipercubo em $\mathbb{R}^n$, onde os códigos binários são considerados subconjuntos desse conjunto de vértices. A distância de Hamming entre dois elementos de $\mathbb{F}_2^n$, será o número de arestas que interligam esses vértices. Na Figura 2 temos a representação geométrica do caso $\mathbb{F}_2^3$.

Sendo $C = \{110, 001, 111\} \subset \mathbb{F}_2^3$, podemos determinar a distância de Hamming entre as palavras de C contando a quantidade mínima de arestas necessárias para ir de um vértice a outro. Assim

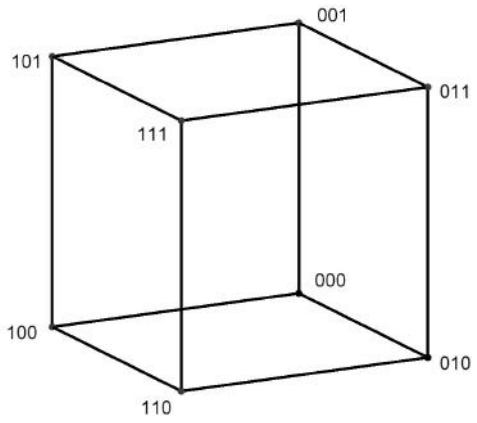

Figura 2: Distância de Hamming em $\mathbb{F}_2^3$.

$$\begin{cases} d(110,001)=3 \\ d(110,111)=1 \\ d(001,111)=2 \end{cases}.$$

Dados um elemento $\mathbf{a} \in A^n$ e um número real $t \geq 0$, definimos o *disco* e a *esfera* de centro em $\mathbf{a}$ e raio t como sendo, respectivamente, os conjuntos

$$\begin{cases} D(a,t)=\{\mathbf{u} \in A^n; d(\mathbf{u},\mathbf{a}) \leq t\} \\ S(a,t)=\{\mathbf{u} \in A^n; d(\mathbf{u},\mathbf{a}) = t\} \end{cases}.$$

Esses conjuntos são finitos e o próximo lema nos fornecerá as suas cardinalidades.

Lema 3.15. *Para todo $\boldsymbol{a} \in A^n$ e todo número natural $r > 0$ temos*

$$|D(\boldsymbol{a},r)| = \sum_{i=0}^{r} \binom{n}{i} (q-1)^i.$$

Demonstração: Ver [15], página 5.

Definição 3.21. *Seja C um código. A distância mínima de C é o número*

$$d = min\{d(\boldsymbol{u},\boldsymbol{v}); \boldsymbol{u},\boldsymbol{v} \in C \text{ e } \boldsymbol{u} \neq \boldsymbol{v}\}.$$

Seja C um código com distância mínima d, definimos

$$\kappa = \left\lfloor \frac{d-1}{2} \right\rfloor,$$

onde $\lfloor t \rfloor$ representa a parte inteira de um número real t. Por exemplo, $\lfloor 5,7 \rfloor = 5$ e $\lfloor \sqrt{8} \rfloor = 2$.

Lema 3.16. *Seja C um código com distância mínima d. Se $\boldsymbol{x}$ e $\boldsymbol{y}$ são palavras distintas de C, então*

$$D(\boldsymbol{x},\kappa) \cap D(\boldsymbol{y},\kappa) = \emptyset.$$

Demonstração: Ver [15], página 6.

Teorema 3.17. *Seja C um código com distância mínima d. Então C pode corrigir até $\kappa = \left\lfloor \frac{d-1}{2} \right\rfloor$ erros e detectar até $d-1$ erros.*

Demonstração Se na transmissão de uma palavra $\mathbf{x}$ do código ocorrerem t erros com $t \leq \kappa$, receberemos uma palavra $\mathbf{r}$, então $d(\mathbf{r}, \mathbf{x}) = t \leq \kappa$, assim temos $\mathbf{r} \in D(\mathbf{x}, \kappa)$ e ainda pelo Lema 3.16, $\mathbf{r} \notin D(\mathbf{y}, \kappa)$, com $\mathbf{x} \neq \mathbf{y}$. Daí podemos concluir que $d(\mathbf{r}, \mathbf{x})$ é menor que a distância de $\mathbf{r}$ a qualquer outra palavra do código. Isso determina $\mathbf{x}$ univocamente a partir de $\mathbf{r}$.

Por outro lado, sendo a distância mínima do código igual a d, podemos introduzir até $d-1$ erros sem encontrar outra palavra do código, pois assim, $d(\mathbf{r}, \mathbf{x}) \leq d-1$ e, portanto, $\mathbf{r}$ não irá pertencer ao código. □

O Teorema 3.17 nos propõe que quanto maior for a distância mínima de um código C, maior será sua capacidade de correção. Assim, é imprescindível para a Teoria dos Códigos determinar o valor de d ou ao menos um valor mínimo para ele.

Definição 3.22. *Seja $C \subset A^n$ um código com distância mínima d e seja $\kappa = \left\lfloor \frac{d-1}{2} \right\rfloor$. O código C será dito perfeito se*

$$\bigcup_{\boldsymbol{c} \in C} D(\boldsymbol{c}, \kappa) = A^n.$$

Note também que o Teorema 3.17 nos apresenta um método para detecção e correção de erros. Seja C um código com distância mínima d e capacidade de correção $\kappa = \left\lfloor \frac{d-1}{2} \right\rfloor$, quando enviamos uma mensagem, se o destinatário recebe uma palavra $\mathbf{r}$, pode ocorrer uma das seguintes situações:

(i) A palavra recebida$\mathbf{r}$ está em um disco de raio κ em torno da palavra $\mathbf{x}$ do código. Assim, substituímos $\mathbf{r}$ por $\mathbf{x}$.

(ii) A palavra $\mathbf{r}$ pode não estar em nenhum disco de raio κ em volta de uma palavra $\mathbf{x}$ do código, dessa forma, não podemos decodificar $\mathbf{r}$ com boa margem de segurança.

Veja que o item (i) não garante que a palavra $\mathbf{x}$ tenha sido a mesma que foi transmitida, uma vez que poderiam ter ocorrido mais que κ erros e assim a palavra recebida se afastaria da palavra enviada. Temos ainda que se o código for perfeito, então o item (ii) não pode ocorrer.

3.4 Equivalência de Códigos

Ao determinarmos uma classe de elementos matemáticos é sempre conveniente associarmos tais classes através do que chamamos de *equivalência*. Veremos a seguir uma noção de equivalência entre códigos, a qual se faz por meio de *isometrias*.

Definição 3.23. *Sejam A um alfabeto e n um número natural. Diremos que uma função $F : A^n \to A^n$ é uma isometria de A^n se ela preserva distâncias de Hamming. Em símbolos,*

$$d(F(\boldsymbol{x}), F(\boldsymbol{y})) = d(\boldsymbol{x}, \boldsymbol{y}); \quad \forall \boldsymbol{x}, \boldsymbol{y} \in A^n.$$

Proposição 3.18. *Toda isometria de A^n é uma bijeção de A^n.*

Demonstração: Ver [15], página 9. □

Proposição 3.19. *i) A função identidade de A^n é uma isometria.*

ii) Se F é uma isometria de A^n, então F^{-1} é uma isometria de A^n.

iii) Se F e G são isometrias de A^n, então $F \circ G$ é uma isometria de A^n.

Demonstração: Ver [15], página 9. □

Definição 3.24. *Dados dois códigos C e C' em A^n, diremos que C' é equivalente a C se existir uma isometria F de A^n tal que $F(C) = C'$.*

Da Proposição 3.19, segue que a equivalência de códigos é uma relação de equivalência, isto é, possui as seguintes propriedades:

i) É reflexiva: todo código é equivalente a si próprio.

Pelo item (i) da Proposição 3.19, todo código C é equivalente a si mesmo, pois $I(C) = C$, onde I é a função identidade.

ii) É simétrica: se C' é equivalente a C, então C é equivalente a C'.

Pela Proposição 3.19 (ii), temos que se F é uma isometria, então F^{-1} é uma isometria. Dessa forma, sejam C e C' códigos de A^n tais que C é equivalente a C'. Assim, existe uma isometria F tal que $F(C) = C'$, de modo que $F^{-1}(C') = F^{-1}(F(C)) = C$, isto é, C' é equivalente a C.

iii) É transitiva: se C é equivalente a C' e C' é equivalente a C'', então C é equivalente C''.

Utilizando o item (iii) da Proposição 3.19, temos que dados C, C' e C'' códigos de A^n tais que C é equivalente a C' e C' é equivalente a C'', então existem isometrias F e G, tais que $F(C) = C'$ e $G(C') = C''$, onde $G(F(C)) = G(C') = C''$, isto é, C é equivalente a C''.

Pela definição, temos que se dois códigos são equivalentes, então terão os mesmos parâmetros.

Vejamos a seguir algumas isometrias de suma importância.

Exemplo 3.2 Se $f : A \to A$ é uma bijeção, e i é um número inteiro tal que $1 \leq i \leq n$, a função

$$\begin{array}{rccc} T_f^i : & A^n & \longrightarrow & A^n \\ & (a_1, \ldots, a_n) & \mapsto & (a_1, \ldots, f(a_i), \ldots, a_n) \end{array}$$

é uma isometria. De fato, sejam $\mathbf{x}$, $\mathbf{y} \in A^n$ tais que $\mathbf{x} = (x_1, \ldots, x_n)$ e $\mathbf{y} = (y_1, \ldots, y_n)$. Então,

$$d(T_f^i(\mathbf{x}), T_f^i(\mathbf{y})) = d((x_1, \ldots, f(x_i), \ldots, x_n), (y_1, \ldots, f(y_i), \ldots, y_n)).$$

Como f é uma bijeção, se $f(x_i) = f(y_i)$ então $x_i = y_i$. E como f é uma função, se $f(x_i) \neq f(y_i)$ então $x_i \neq y_i$. Assim a contribuição de $f(x_i)$ e $f(y_i)$ para $d(T_f^i(x), T_f^i(y))$ é a mesma obtida se substituirmos $f(x_i)$ por x_i e $f(y_i)$ por y_i. Logo,

$$\begin{gathered} d(T_f^i(\mathbf{x}), T_f^i(\mathbf{y})) = d((x_1, \ldots, f(x_i), \ldots, x_n), (y_1, \ldots, f(y_i), \ldots, y_n)) = \\ d((x_1, \ldots, x_i, \ldots, x_n), (y_1, \ldots, y_i, \ldots, y_n)) = d(\mathbf{x}, \mathbf{y}). \end{gathered}$$

Portanto, T_f^i é uma isometria. □

Exemplo 3.3 Se π é uma bijeção do conjunto $\{1, \ldots, n\}$ nele próprio, também chamada de *permutação* de $\{1, \ldots, n\}$, a aplicação permutação de coordenadas

$$\begin{array}{rccc} T_\pi : & A^n & \longrightarrow & A^n \\ & (a_1, \ldots, a_n) & \mapsto & (a_{\pi(1)}, \ldots, a_{\pi(n)}) \end{array}$$

é uma isometria. Com efeito, sejam $\mathbf{x}$, $\mathbf{y} \in A^n$ tais que $\mathbf{x} = (x_1, \ldots, x_n)$ e $\mathbf{y} = (y_1, \ldots, y_n)$. Então,

$$d(T_\pi(x), T_\pi(y)) = d((x_{\pi(1)}, \ldots, x_{\pi(n)}), (y_{\pi(1)}, \ldots, y_{\pi(n)})) = |\{\pi(i); x_{\pi(i)} \neq y_{\pi(i)}, 1 \leq i \leq n\}|.$$

Note que π é uma permutação dos elementos de n. Assim,

$$d((x_{\pi(1)}, \ldots, x_{\pi(n)}), (y_{\pi(1)}, \ldots, y_{\pi(n)})) = d(\mathbf{x}, \mathbf{y}),$$

pois nas coordenadas de $\mathbf{x}$ e $\mathbf{y}$ atua a mesma permutação, e dessa forma a contribuição de $\mathbf{x}$ e $\mathbf{y}$ para $d(\mathbf{x}, \mathbf{y})$ é a mesma que $d((x_{\pi(1)}, \ldots, x_{\pi(n)}), (y_{\pi(1)}, \ldots, y_{\pi(n)}))$. Logo,

$$d(T_\pi(x), T_\pi(y)) = d(\mathbf{x}, \mathbf{y}).$$

Portanto, T_π é uma isometria. □

Teorema 3.20. *Seja $F : A^n \to A^n$ uma isometria, então existem uma permutação π de $\{1, \ldots, n\}$ e bijeções f_i de A, $i = 1, \ldots, n$, tais que*

$$F = T_\pi \circ T_{f_1}^1 \circ \cdots \circ T_{f_n}^n.$$

Demonstração: Ver [15], página 209. □

Corolário 3.21. *Sejam C e C' dois códigos em A^n. Temos que C e C' são equivalentes se, e somente se, existem uma permutação π de $\{1, \ldots, n\}$ e bijeções $f_1, \ldots, f_n$ de A tais que*

$$C' = \{(f_{\pi(1)}(x_{\pi(1)}), \ldots, f_{\pi(n)}(x_{\pi(n)}));\ (x_1, \ldots, x_n) \in C\}.$$

Demonstração: Ver [15], página 10. □

Em alguns textos sobre códigos, a definição de códigos equivalentes pode ser vista como a seguir.

Dois códigos de comprimento n, sobre um alfabeto A onde os elementos são chamados de letras, são equivalentes se, e somente se, um pode ser obtido do outro mediante operações tais como:

1. Através de uma bijeção sobre A, substitui-se as letras numa posição fixa em todas as palavras do código.

2. Por meio de uma permutação fixa de $\{1, 2, \ldots, n\}$, faz-se uma permutação nas posições das letras de todas as palavras de C.

3.5 Distância de Lee

Como foi dito na seção 2.5, além do corpo $\mathbb{F}_2$, existem outros corpos finitos $\mathbb{F}_p = \{0, 1, 2, ..., p-1\}$, com p um número primo. São exemplos de corpos finitos

$$\mathbb{F}_5 = \{0, 1, 2, 3, 4\} \text{ e } \mathbb{F}_{13} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\}.$$

Observamos ainda que os elementos de $\mathbb{F}_p$ são formados pelos possíveis restos na divisão de qualquer número inteiro n por p.

Da mesma forma como definimos as operações de adição e multiplicação em $\mathbb{F}_2$, as definimos em $\mathbb{F}_p$. Por exemplo, tomando os elementos 2 e 4 em $\mathbb{F}_5$, temos

$$2 + 4 = 6 = 5.1 + 1 \Rightarrow 2 + 4 = 1,$$

$$2.4 = 8 = 5.1 + 3 \Rightarrow 2.4 = 3.$$

Para conjuntos da forma $\mathbb{F}_p$ podemos definir uma função distância diferente da métrica de Hamming, é a chamada *distância (ou métrica) de Lee*, a qual veremos a seguir. Sejam $a, b \in \mathbb{F}_p$, então

$$d_{Lee}(a,b) = min\{|a-b|, p-|a-b|\}.$$

Por exemplo, seja $\mathbb{F}_7 = \{0,1,2,3,4,5,6\}$, então

$$d_{Lee}(5,2) = min\{|5-2|, 7-|5-2|\} = min\{3, 7-3\} = min\{3,4\} = 3$$
$$d_{Lee}(1,6) = min\{|1-6|, 7-|1-6|\} = min\{5, 7-5\} = min\{5,2\} = 2.$$

A distância de Lee pode ser interpretada geometricamente, onde podemos distribuir, ordenadamente, os elementos de $\mathbb{F}_p$ sobres os vértices de um polígono de p lados. A distância de Lee entre dois elementos será o menor número de arestas necessárias para interligar os vértices em que estão inseridos. Dado o conjunto $\mathbb{F}_7$ visto anteriormente, na Figura 3, podemos observar que o menor número de arestas que interligam os elementos 2 e 5 é exatamente igual a 3. Enquanto, para 1 e 6, o menor o número de arestas é 2.

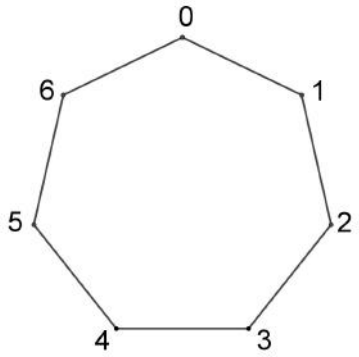

Figura 3: Distância de Lee em $\mathbb{F}_7$.

Em $\mathbb{F}_p^n$, definimos a distância de Lee como sendo a soma das distância entre as coordenadas, isto é,

$$d_{Lee} = ((a_1, a_2, ..., a_n), (b_1, b_2, ..., b_n)) = \sum_{i=1}^{n} d_{Lee}(a_i, b_i)$$

Por exemplo, em $\mathbb{F}_5^3$, dados $x = (1,0,2)$ e $y = (2,2,3)$, temos

$$d_{Lee}(x,y) = d_{Lee}((1,0,2),(2,2,3)) =$$
$$\sum_{i=1}^{3} d_{Lee}(a_i, b_i) = d_{Lee}(1,2) + d_{Lee}(0,2) + d_{Lee}(2,3) =$$
$$min\{|1-2|, 5-|1-2|\} + min\{|0-2|, 5-|0-2|\} + min\{|2-3|, 5-|2-3|\} =$$
$$min\{1,4\} + min\{2,3\} + min\{1,4\} = 1+2+1 = 4.$$

Definição 3.25. *Dado um código linear $C \subset \mathbb{F}_p^n$, definimos a distância de Lee mínima de C por*

$$d_{Lee}(C) = min\{d_{Lee}(a,b); a,b \in C, a \neq b\}.$$

Uma observação acerca da distância de Lee, em comparação com a distância de Hamming, é que dado $C \in \mathbb{F}_p^n$, para $p = 2$ e $p = 3$, essas distâncias são iguais. De fato, sejam $a, b \in \mathbb{F}_p^n$, temos que $a = (a_1, a_2, ..., a_n)$ e $b = (b_1, b_2, ..., b_n)$. Assim,

$$d_{Lee}(a, b) = \sum_{i=1}^{n} d_{Lee}(a_i, b_i) = \sum_{i=1}^{n} min\{|a - b|, p - |a - b|\}.$$

Para $p = 2$, temos

$$min\{|a - b|, 2 - |a - b|\} = \begin{cases} 0, & se\ a_i = b_i \\ 1, & se\ a_i \neq b_i \end{cases}.$$

E, para $p = 3$

$$min\{|a - b|, 3 - |a - b|\} = \begin{cases} 0, & se\ a_i = b_i \\ 1, & se\ a_i \neq b_i \end{cases}.$$

Dessa forma

$$\sum_{i=1}^{n} min\{|a - b|, p - |a - b|\} = |\{i, a_i \neq b_i,\ 1 \leq i \leq n\}| = d(a, b).$$

Portanto,

$$d_{Lee}(a, b) = d(a, b)$$

isto é, as distâncias de Lee e Hamming são equivalentes.

Para $p > 3$, a distância de Lee é superior à de Hamming.

3.6 Características Fundamentais de um Código

3.6.1 Parâmetros

Seja A um alfabeto e C um código corretor de erros sobre A. Dizemos que o código C possui três **parâmetros** principais, a saber $[n, M, d]$, onde n é o comprimentos do código, isto é, o comprimento de cada palavra de C, M é o número de elementos do código, denotamos por $M = |C|$ e d é a distância de Hamming mínima do código C.

Pode ocorrer de não existir um código com parâmetros $[n, M, d]$, uma vez que há uma relação de interdependência entre esses números. Um dos principais objetivos da Teoria dos Códigos é estudar essa interdependência entre n, M e d. Veremos a seguir dois resultados essenciais para os parâmetros de um código.

Teorema 3.22. *(Limitante de Hamming) Seja C um código com parâmetros $[n, M, d]$, então*

$$M \leq \frac{q^n}{\sum_{i=1}^{r} \binom{n}{i} (q-1)^i}.$$

Valendo a igualdade se, e somente se, C é um código perfeito.

Demonstração: Ver [15], página 189. □

Limitante de Singleton. Seja C um código com parâmetros $[n, k, d]$, sendo k a dimensão de C. Então vale a seguinte desigualdade:

$$d \leq n - k + 1.$$

Quando a igualdade ocorre, dizemos que C é um código MDS (*Maximum Distance Separable*), isto é, código de máxima distância separável.

Veremos uma demonstração do Limitante de Singleton na seção 4.3.

3.6.2 Taxa de Informação

Considerando o alfabeto A como um conjunto finito com q elementos, definimos a **taxa de informação** de um código C por

$$R(C) = \frac{log_q M}{n}.$$

No Exemplo 3.1, temos que o código do canal é

$$C = \{000000, 101010, 010101, 111111\}.$$

Assim, temos que o comprimento de cada palavra é $n = 6$, o número de elementos do código é $M = 4$, o número de elementos de $\mathbb{F}_2$, alfabeto sobre o qual está C, é $q = 2$. Dessa forma, a taxa de informação de C é

$$R(C) = \frac{log_q M}{n} = \frac{log_2 4}{6} = \frac{log_2 2^2}{6} = \frac{2}{6} = \frac{1}{3}.$$

Quanto mais a taxa de informação de um código se aproxima de 1, melhor sua eficiência na detecção e correção de erros.

4 CÓDIGOS LINEARES

O conteúdo desenvolvido neste capítulo tem por base as referências [12], de própria autoria, e [15].

4.1 O que é um Código Linear?

Na Teoria dos Códigos, a classe de códigos mais utilizada é a dos Códigos Lineares ´pois apresentam uma estrutura de Espaço Vetorial, a qual traz propriedades que facilitam as operações a serem realizadas..

Seja K um corpo finito com q elementos considerado como alfabeto. Assim, para cada natural n, temos um espaço vetorial K^n, sobre K, cuja dimensão é n.

Definição 4.26. *Um código $C \subset K^n$ será chamado Código Linear se for um subespaço vetorial de K^n.*

Por definição, segue que um Código Linear qualquer é um espaço vetorial de dimensão finita. Consideremos o código C, de dimensão k e seja $v_1, v_2, \ldots, v_k$ uma de suas bases, assim, qualquer elemento $v \in C$ pode ser escrito de maneira única como combinação linear dos elementos de sua base, isto é,

$$v = \alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_k v_k,$$

onde os $\alpha_1, \alpha_2, ..., \alpha_n \in K$. Dessa forma, o número de elementos de C é

$$M = |C| = q^k.$$

Definição 4.27. *Seja $\boldsymbol{x} \in K^n$. Definimos o peso de $\boldsymbol{x}$ como sendo o número inteiro*

$$\omega(\boldsymbol{x}) := |\{i; x_i \neq 0\}|.$$

Em outros termos, o peso é o número de coordenadas não nulas de $\mathbf{x}$. E ainda

$$\omega(\mathbf{x}) = d(\mathbf{x}, \mathbf{0}),$$

onde d é a métrica de Hamming. Isto é, o peso de um elemento é igual a sua distância até o vetor nulo

Definição 4.28. *O peso de um Código Linear C é o inteiro*

$$\omega(C) := min\{\omega(\boldsymbol{x}); \boldsymbol{x} \in C \setminus \{\boldsymbol{0}\}\}.$$

Proposição 4.23. *Seja $C \subset K^n$ um Código Linear com distância mínima d. Temos que*

i) $\forall \boldsymbol{x}, \boldsymbol{y} \in K^n, \quad d(\boldsymbol{x}, \boldsymbol{y}) = \omega(\boldsymbol{x} - \boldsymbol{y})$.

ii) $d = \omega(C)$.

Demonstração: i) Dados $\mathbf{x}$, $\mathbf{y} \in K^n$, temos que

$$d(\mathbf{x}, \mathbf{y}) = |\{i;\ x_i \neq y_i, 1 \leq i \leq n\}| = |\{i;\ x_i - y_i \neq 0, 1 \leq i \leq n\}| = \omega(\mathbf{x} - \mathbf{y}).$$

ii) Dados $\mathbf{x}$, $\mathbf{y} \in C$, tal que $\mathbf{x} \neq \mathbf{y}$, temos que $\mathbf{z} = \mathbf{x} - \mathbf{y} \in C \setminus \{\mathbf{0}\}$, de modo que $d(\mathbf{x}, \mathbf{y}) = \omega(\mathbf{x} - \mathbf{y}) = \omega(\mathbf{z})$. Assim,

$$d = \min\{d(\mathbf{x}, \mathbf{y});\ \mathbf{x}, \mathbf{y} \in C \text{ e } \mathbf{x} \neq \mathbf{y}\} = \min\{\omega(\mathbf{z});\ \mathbf{z} \in C \setminus \{\mathbf{0}\}\} = \omega(C).$$

□

Diante do item (ii) da Proposição 4.23, vemos que a distância mínima de um Código Linear C também pode ser considerada como peso do código C. Assim, ao calcularmos a distância mínima de um código C, ao invés de compararmos os vetores dois a dois, basta verificar qual dos vetores de C possui menor peso e este valor será a distância mínima deste código.

No estudo da Álgebra Linear podemos descrever um subespaço vetorial C de um espaço vetorial K^n, como imagem ou como núcleo de Transformações Lineares.

Vejamos como representar C através da imagem. Seja $\mathbf{v}_1$, $\mathbf{v}_2$, ..., $\mathbf{v}_k$ uma base de C e tomemos a transformação linear

$$\begin{array}{rccc} T: & K^k & \longrightarrow & K^n \\ & \mathbf{x} = (x_1, \ldots, x_k) & \mapsto & x_1\mathbf{v}_1 + x_2\mathbf{v}_2 + \cdots + x_k\mathbf{v}_k \end{array}.$$

Note que T é uma transformação linear injetora, pois o único vetor $x \in K^k$ tal que $T(x) = 0$, é o vetor nulo. Além disso, a imagem de T é C, isto é,

$$Im(T) = C.$$

Assim, um código $C \subset K^n$ de dimensão k é equivalente a uma transformação linear injetora

$$T : K^k \longrightarrow K^n$$

tal que $C = Im(T)$.

Agora veremos como descrever um código C como núcleo de uma transformação linear. Assim, seja C' um subespaço de K^n, complementar de C, ou seja,

$$C \oplus C' = K^n,$$

e considere a transformação linear

$$\begin{array}{rccc} H: & C \oplus C' & \longrightarrow & K^{n-k} \\ & \mathbf{u} \oplus \mathbf{v} & \mapsto & \mathbf{v} \end{array}$$

em que o núcleo é exatamente C.

Para verificar se um elemento $\mathbf{v} \in K^n$ pertence ou não a C, basta observar se $H(\mathbf{v})$ é o vetor nulo de K^{n-k}.

Exemplo 4.1: Consideremos o corpo finito $\mathbb{F}_2 = \{0, 1\}$, o qual possui dois elementos e seja $C \subset \mathbb{F}_2^4$ o código gerado pelos vetores $\mathbf{v}_1 = 1011$ e $\mathbf{v}_2 = 1100$. Esse código possui $4(q^k = 2^2)$ elementos, pois tem dimensão 2 sobre um corpo de 2 elementos. Assim, C pode ser representado por

$$x_1\mathbf{v}_1 + x_2\mathbf{v}_2$$

onde x_1, $x_2 \in \mathbb{F}_2$. O código C pode ser visto como núcleo da transformação linear

$$\begin{array}{rccc} H: & \mathbb{F}_2^4 & \longrightarrow & \mathbb{F}_2^2 \\ & (x_1, \ldots, x_4) & \mapsto & (x_1 + x_2 + x_3, x_1 + x_2 + x_4) \end{array}$$

De fato, dados $v_1 = (1, 0, 1, 1)$ e $v_2 = (1, 1, 0, 0)$, temos

$$H(v_1) = (1 + 0 + 1, 1 + 0 + 1) = (2, 2) = (0, 0),$$

logo, $v_1 \in \mathrm{Ker}H$, e

$$H(v_2) = (1 + 1 + 0, 1 + 1 + 0) = (2, 2) = (0, 0),$$

ou seja, $v_2 \in \mathrm{Ker}H$.

Assim, $C \subset \mathrm{Ker}H$.

Temos ainda

$$(x_1 + x_2 + x_3, x_1 + x_2 + x_4) = (x_1, x_1) + (x_2, x_2) + (x_3, 0) + (0, x_4) = x_1(1, 1) + x_2(1, 1) + x_3(1, 0) + x_4(0, 1).$$

Mas, (1,1) pode ser escrito como combinação linear de (1,0) e (0,1). Logo,

$$\mathrm{Im}H = \{(1, 0), (0, 1)\}$$

isto é, $\dim \mathrm{Im}H = 2$.

Dessa forma, pelo Teorema do Núcleo e da Imagem, Teorema 2.11, página 31, temos

$$\dim H = \dim \mathrm{Ker}H + \dim \mathrm{Im}H \Rightarrow 4 = \dim \mathrm{Ker}H + 2 \Rightarrow \dim \mathrm{Ker}H = 2.$$

Portanto, $C = \mathrm{Ker}H$. □

Definição 4.29. *Seja K um corpo finito. Dois Códigos Lineares C e C' são linearmente equivalentes se existir uma isometria linear $T : K^n \to K^n$ tal que $T(C) = C'$.*

Em alguns textos sobre Teoria dos Códigos a definição de códigos linearmente equivalentes é usualmente vista, como a seguir: dois códigos lineares são linearmente equivalentes se, e somente se, um pode ser obtido do outro atrés de operações como

(1) Multiplicação, em todas as palavras, dos elementos numa posição fixa dada por um escalar diferente de zero;

(2) Permutação das posições de todas as palavras do código, através de uma permutação fixa de $\{1, 2, ..., n\}$.

4.2 Matrizes Geradoras e Teste de Paridade

Consideremos o corpo finito K com q elementos e um código linear $C \subset K^n$. Os *parâmetros do código* C são a terna de inteiros (n, k, d), onde n é o comprimento do código, k é a dimensão de C sobre K e d a distância mínima de C, que por sua vez também é igual ao peso do código, $\omega(C)$. Note que o número de elementos de C é $M = q^k$.

Sejam $\mathcal{B} = \{\mathbf{v}_1, ..., \mathbf{v}_k\}$ uma base de C e G a matriz

$$G = \begin{pmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_k \end{pmatrix} = \begin{pmatrix} v_{11} & v_{12} & \cdots & v_{1n} \\ \vdots & \vdots & & \vdots \\ v_{k1} & v_{k2} & \cdots & v_{kn} \end{pmatrix}.$$

Veja que as linhas de G são os vetores $\mathbf{v}_i = (v_{i1}, ..., v_{in})$, $i = 1, ..., k$. Chamamos a matriz G de *matriz geradora* de C associada à base $\mathcal{B}$.

Seja T a transformação linear dada por

$$\begin{array}{rccc} T: & K^k & \longrightarrow & K^n \\ & \mathbf{x} & \mapsto & \mathbf{x}G \end{array}.$$

Dado $\mathbf{x} = (x_1, ..., x_k)$, temos

$$T(\mathbf{x}) = \mathbf{x}G = x_1\mathbf{v}_1 + \cdots + x_k\mathbf{v}_k,$$

assim, $T(K^k) = C$. Dessa forma, podemos admitir K^k como sendo o código da fonte, C, o código de canal e a transformação T, uma codificação.

Observe que a matriz G é determinada a partir de uma base de C, portanto essa matriz não é única, uma vez que ao mudar a base, teremos uma outra matriz geradora.

Além disso, dada uma matriz geradora G de um código C, podemos encontrar uma outra matriz geradora de C através de operações como:

(l_1) permutação de duas linhas;

(l_2) multiplicação de uma linha por um escalar não nulo;

(l_3) adição de um múltiplo escalar de uma linha a outra.

Também é possível construir um código C a partir de uma matriz geradora. Para isso, tomamos uma matriz G tal que suas linhas sejam linearmente independentes e adotamos C de modo que seja a imagem da seguinte transformação linear

$$\begin{array}{rccc} T: & K^k & \to & K^n \\ & \mathbf{x} & \mapsto & \mathbf{x}G \end{array}.$$

Exemplo 4.2: Seja $K = \mathbb{F}_2$ e tome $G = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}$. Dada a transformação linear

$$\begin{array}{rccc} T: & \mathbb{F}_2^3 & \to & \mathbb{F}_2^4 \\ & \mathbf{x} & \mapsto & \mathbf{x}G \end{array},$$

encontramos em $\mathbb{F}_2^4$ um código C, imagem da transformação T.

Tomando a palavra $\mathbf{x} = 111 \in \mathbb{F}_2^3$, temos que

$$\mathbf{x}G = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 1+0+1 & 0+1+1 & 1+1+0 & 1+0+0 \end{pmatrix} = \begin{pmatrix} 2 & 2 & 2 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 1 \end{pmatrix}$$

Logo, a palavra 111 do código da fonte é codificada como 0001.

Para decodificar a palavra 0110 do código, devemos encontrar a palavra $\mathbf{x} = (x_1\ x_2\ x_3)$ de $\mathbb{F}_2^3$ que se origina a partir de T. Assim, temos

$$\begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix} G = \begin{pmatrix} 0 & 1 & 1 & 0 \end{pmatrix},$$

$$\begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix} \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 \end{pmatrix}$$

$$\begin{pmatrix} x_1+x_3 & x_2+x_3 & x_1+x_2 & x_1 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 \end{pmatrix}$$

isto é,

$$\begin{cases} x_1 + x_3 &= 0 \\ x_2 + x_3 &= 1 \\ x_1 + x_2 &= 1 \\ x_1 &= 0 \end{cases}$$

cuja solução é $x_1 = 0$, $x_2 = 1$ e $x_3 = 0$. Dessa forma, a palavra 0110 é decodificada como 010. □

Definição 4.30. *Uma matriz geradora G de um código C está na forma padrão se*

$$G = (Id_k \mid A),$$

onde Id_k é a matriz identidade $k \times k$ e A, uma matriz $k \times (n-k)$.

Observemos que dado um código C, nem sempre é possível encontrar uma matriz geradora de C na forma padrão. Por exemplo, dado um código C em $\mathbb{F}_2^6$ cuja matriz geradora é

$$\begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}.$$

Temos que essa matriz não poderá ser colocada na forma padrão através de operações sobre suas linhas. Porém, ao realizar permutações das colunas de G, obtemos a matriz

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix},$$

que é uma matriz geradora na forma padrão de um código C' equivalente a C.

De maneira geral, também podemos efetuar operações sobre as colunas de uma matriz geradora G de um código linear C, tais como:

(c1) permutação de duas colunas;

(c2) multiplicação de uma coluna por um escalar diferente de zero,

e obter, assim, uma matriz G' de um código C' equivalente a C.

Ao utilizar-se de operações do tipo (c1), temos o resultado a seguir:

Teorema 4.24. *Dado um código C, existe um código equivalente C' com matriz geradora na forma padrão.*

Demonstração: Ver [15], página 92. □

4.3 Códigos Duais

Sejam $\mathbf{u} = (u_1, ..., u_n)$ e $\mathbf{v} = (v_1, ..., v_n)$ elementos de K^n. Definimos o *produto interno* de $\mathbf{u}$ por $\mathbf{v}$ como

$$\langle \mathbf{u}, \mathbf{v} \rangle = u_1 v_1 + \cdots + u_n v_n.$$

Dado $C \subset K^n$ um código linear, definimos

$$C^{\perp} = \{\mathbf{v} \in K^n;\ \langle \mathbf{v}, \mathbf{u} \rangle = 0, \quad \forall \mathbf{u} \in C\}.$$

Lema 4.25. *Se $C \subset K^n$ é um Código Linear, com matriz geradora G, então*

i) $C^{\perp}$ é um subespaço vetorial de K^n;

ii) $\boldsymbol{x} \in C^{\perp} \Leftrightarrow G\boldsymbol{x}^t = 0$.

Demonstração: i) Sejam $\mathbf{u}, \mathbf{v} \in C^{\perp}$ e $\alpha \in K$. Devemos mostrar que $\mathbf{u}+\mathbf{v} \in C^{\perp}$ e $\alpha\mathbf{u} \in C^{\perp}$. De fato, como $\mathbf{u}, \mathbf{v} \in C^{\perp}$, para todo $\mathbf{x} \in C$, temos $\langle \mathbf{u}, \mathbf{x} \rangle = 0$ e $\langle \mathbf{v}, \mathbf{x} \rangle = 0$, assim

$$\langle \mathbf{u} + \mathbf{v}, \mathbf{x} \rangle = \langle \mathbf{u}, \mathbf{x} \rangle + \langle \mathbf{v}, \mathbf{x} \rangle = 0 + 0 = 0,$$

e

$$\langle \alpha\mathbf{u}, \mathbf{x} \rangle = \alpha \langle \mathbf{u}, \mathbf{x} \rangle = \alpha.0 = 0.$$

portanto $C^{\perp}$ é um subespaço vetorial de K^n.
ii) Temos $\mathbf{x} \in C^{\perp} \Leftrightarrow \langle \mathbf{x}, \mathbf{v} \rangle = 0,\ \forall \mathbf{v} \in C$. Temos ainda que as linhas de G são uma base para o código C, e tomando $v_1, v_2, ..., v_k$ uma base de C podemos construir G da seguinte maneira

$$G = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_k \end{pmatrix}.$$

Dessa forma, tomando $\mathbf{x} = (x_1, x_2, ..., x_n) \in C^{\perp}$ temos que

$$G\mathbf{x}^t = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_k \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} v_1x_1 + \cdots + v_1x_n \\ v_2x_1 + \cdots + v_2x_n \\ \vdots \\ v_kx_1 + \cdots + v_kx_n \end{pmatrix} = \begin{pmatrix} \langle v_1, \mathbf{x} \rangle \\ \langle v_2, \mathbf{x} \rangle \\ \vdots \\ \langle v_k, \mathbf{x} \rangle \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Logo $G\mathbf{x}^t = 0$.
Por outro lado, se $G\mathbf{x}^t = 0$, onde $\mathbf{x} \in K^n$, então

$$\begin{pmatrix} \langle v_1, \mathbf{x}\rangle \\ \langle v_2, \mathbf{x}\rangle \\ \vdots \\ \langle v_k, \mathbf{x}\rangle \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Seja $\mathbf{v}$ um elemento qualquer de C, logo $\mathbf{v} = \alpha_1 v_1 + \cdots + \alpha_k v_k$. Tomando o produto vetorial $\langle \mathbf{v}, \mathbf{x}\rangle$, temos

$$\langle \mathbf{v}, \mathbf{x}\rangle = \langle \alpha_1 v_1 + \cdots + \alpha_k v_k, \mathbf{x}\rangle = \alpha_1\langle v_1, \mathbf{x}\rangle + \cdots + \alpha_k\langle v_k, \mathbf{x}\rangle = 0.$$

Portanto, $\mathbf{x} \in C^{\perp}$. □

O subespaço vetorial $C^{\perp} \subset K^n$ é também um Código Linear, chamado *Código Dual.*

Proposição 4.26. *Se $C \subset K^n$ um código de dimensão k com matriz geradora $G = (Id_k \mid A)$, na forma padrão, então*

i) $dim C^{\perp} = n - k$;

ii) $H = (-A^t \mid Id_{n-k})$ é uma matriz geradora de $C^{\perp}$.

Demonstração:
i) Sabemos que $\mathbf{x} = (x_1, ..., x_n) \in C^{\perp} \Leftrightarrow G\mathbf{x}^t = 0$. Assim

$$G\mathbf{x}^t = (Id_k \mid A)\mathbf{x}^t = 0 \Rightarrow$$

$$\begin{pmatrix} 1 & 0 & \cdots & 0 & a_{(k+1)1} & a_{(k+2)1} & \cdots & a_{n1} \\ 0 & 1 & \cdots & 0 & a_{(k+1)2} & a_{(k+2)2} & \cdots & a_{n2} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & a_{(k+1)k} & a_{(k+2)k} & \cdots & a_{nk} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \Rightarrow$$

$$\begin{pmatrix} x_1 + a_{(k+1)1}x_{k+1} + \cdots + a_{n1}x_n \\ x_2 + a_{(k+1)2}x_{k+1} + \cdots + a_{n2}x_n \\ \vdots \\ x_k + a_{(k+1)k}x_{k+1} + \cdots + a_{nk}x_n \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \Rightarrow$$

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \end{pmatrix} = \begin{pmatrix} -a_{(k+1)1}x_{k+1} - \cdots - a_{n1}x_n \\ -a_{(k+1)2}x_{k+1} - \cdots - a_{n2}x_n \\ \vdots \\ -a_{(k+1)k}x_{k+1} - \cdots - a_{nk}x_n \end{pmatrix} \Rightarrow$$

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \end{pmatrix} = - \begin{pmatrix} a_{(k+1)1} + \cdots + a_{n1} \\ a_{(k+1)2} + \cdots + a_{n2} \\ \vdots \\ a_{(k+1)k} + \cdots + a_{nk} \end{pmatrix} \begin{pmatrix} x_{k+1} \\ x_{k+2} \\ \vdots \\ x_n \end{pmatrix}.$$

Assim, os $n-k$ elementos $x_{k+1}, ..., x_n$ podem ser escolhidos aleatoriamente. Portanto, temos que $\dim C^{\perp} = n-k$.

ii) Observe que $\mathbf{x}_i = -a_{(k+1)i}x_{k+1} - \cdots - a_{ni}x_n,\ i = 1, ..., k$. Logo, dado $\mathbf{x} \in C^{\perp}$, temos

$$\mathbf{x} = (-a_{(k+1)1}x_{k+1} - \cdots - a_{n1}x_n, ..., -a_{(k+1)k}x_{k+1} - \cdots - a_nx_n, x_{k+1}, ..., x_n).$$

Daí, uma base para $C^{\perp}$ será

$$\{(-a_{(k+1)1}, ..., -a_{(k+1)k}, 1, 0, ..., 0), (-a_{(k+2)1}, ..., -a_{(k+2)k}, 0, 1, ..., 0), ..., (-a_{n1}, ..., -a_{nk}, 0, 0, ..., 1)\}.$$

Com isso, temos que

$$H = \begin{pmatrix} -a_{(k+1)1} & -a_{(k+1)2} & \cdots & -a_{(k+1)k} & 1 & 0 & \cdots & 0 \\ -a_{(k+2)1} & -a_{(k+2)2} & \cdots & -a_{(k+2)k} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ -a_{n1} & -a_{n2} & \cdots & -a_{nk} & 0 & 0 & \cdots & 1 \end{pmatrix} = (-A^t \mid Id_{n-k}).$$

é uma matriz geradora de $C^{\perp}$. □

Recordando as isometrias lineares básicas de K^n definidas na Seção 3.4, página 45, temos:

$$\begin{array}{rccc} T_{\pi}: & K^n & \longrightarrow & K^n \\ & (x_1, ..., x_n) & \mapsto & (x_{\pi(1)}, ..., x_{\pi(n)}) \end{array},$$

onde π é uma permutação de $\{1, ..., n\}$, e

$$\begin{array}{rccc} T_k^i: & K^n & \longrightarrow & K^n \\ & (x_1, ..., x_i, ..., x_n) & \mapsto & (x_1, ..., kx_i, ..., x_n) \end{array},$$

com $k \in K^*$ e $i = 1, ..., n$.

O Lema a seguir relaciona Códigos Lineares equivalentes e Códigos Duais.

Lema 4.27. *Seja C um Código Linear em K^n. Para toda permutação π de $\{1, ..., n\}$, para todo $k \in K^*$ e para todo $i = 1, ..., n$ temos*
i) $(T_{\pi}(C))^{\perp} = T_{\pi}(C^{\perp})$
ii) $(T_k^i(C))^{\perp} = T_{k^{-1}}^i(C^{\perp})$.

Demonstração: Ver [15], página 95. □

Proposição 4.28. *Sejam C e D dois Códigos Lineares em K^n. Se C e D são linearmente equivalentes, então $C^{\perp}$ e $D^{\perp}$ também são linearmente equivalentes.*

Demonstração: Se C e D são linearmente equivalentes, pela definição de códigos linearmente equivalentes, considere que existem uma permutação π de $\{1, ..., n\}$ e $k_1, ..., k_n \in K^*$ de modo que

$$D = T_\pi \circ T^1_{k_1} \circ \cdots \circ T^n_{k_n}(C).$$

E, pelo Lema 4.27, segue o resultado, pois

$$D^\perp = (T_\pi \circ T^1_{k_1} \circ \cdots \circ T^n_{k_n}(C))^\perp = T_\pi \circ T^1_{k_1^{-1}} \circ \cdots \circ T^n_{k_n^{-1}}(C^\perp).$$

□

Corolário 4.29. *Se D é um Código Linear em K^n de dimensão k, então $D^\perp$ é um código de dimensão $n-k$.*

Demonstração: Pelo Teorema 4.24, o código D é equivalente a um código C, cuja dimensão é k, e possui matriz geradora na forma padrão. Dessa forma, pela Proposição 4.26 (ii), temos que $\dim C^\perp = n-k$. Porém, da Proposição anterior, segue que $D^\perp$ é equivalente a $C^\perp$ e, consequentemente, tem também dimensão $n-k$. □

Lema 4.30. *Suponha que C seja um código de dimensão k em K^n com matriz geradora G. Uma matriz H de ordem $(n-k)\times n$, com coeficientes em K e com linhas linearmente independentes, é uma matriz geradora de $C^\perp$ se, e somente, se,*

$$G \cdot H^t = 0.$$

Demonstração: Ver [15], página 96. □

Corolário 4.31. $(C^\perp)^\perp = C$.

Demonstração: Sejam G e H, matrizes geradoras de C e $C^\perp$, respectivamente. Então, $G \cdot H^t = 0$. Usando a transposição nessa última igualdade, temos

$$G \cdot H^t = 0 \;\Rightarrow\; (G \cdot H^t)^t = 0 \;\Rightarrow\; (H^t)^t \cdot G^t = 0 \;\Rightarrow\; H \cdot G^t = 0,$$

assim, G é matriz geradora de $(C^\perp)^\perp$, e portanto $(C^\perp)^\perp = C$. □

Proposição 4.32. *Se C é um Código Linear e H uma matriz geradora de $C^\perp$,então*

$$\boldsymbol{v} \in C \;\Leftrightarrow\; H\boldsymbol{v}^t = 0.$$

Demonstração: Pelo Corolário anterior, temos que $\mathbf{v} \in C \;\Leftrightarrow\; \mathbf{v} \in (C^\perp)^\perp$. Mas, pelo Lema 4.25 (ii), $\mathbf{v} \in (C^\perp)^\perp \;\Leftrightarrow\; H\mathbf{v}^t = 0$. Portanto, $\mathbf{v} \in C \;\Leftrightarrow\; H\mathbf{v}^t = 0$. □

A matriz geradora H de $C^\perp$ é chamada *matriz teste de paridade* de C.

Em resumo, seja C um código linear com matriz geradora $G = (Id_k|A)$ na forma padrão, onde A é uma matriz $k \times (n-k)$, sua matriz teste de paridade é $H = (-A^t|Id_{n-k})$. Para verificar se um vetor $\mathbf{v} \in C$, basta verificar se a equação $H\mathbf{v}^t = 0$ é satisfeita. O vetor $H\mathbf{v}^t$ é chamado *síndrome* de $\mathbf{v}$.

Exemplo 4.3 Seja C um código sobre $\mathbb{F}_2$ com matriz geradora

$$G = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{pmatrix}.$$

Determinaremos, o comprimento, a dimensão, o número de elementos e a matriz teste de paridade de C. Inicialmente, verifiquemos se os vetores $v_1 = 11101$ e $v_2 = 11011$ pertencem a C. Observemos que G está na forma padrão, onde notamos que $k = 3$, pois temos a matriz identidade 3×3, e também a matriz A é 3×2, isto é, $n - k = 2 \Rightarrow n = 5$. Logo, o comprimento do código C é $n = 5$ e sua dimensão é $k = 3$. Além disso, como $q = 2$, pois C está sobre $\mathbb{F}_2$, segue que o número de elementos de C é $|C| = q^k = 2^3 = 8$. Para determinar a matriz teste de paridade, basta verificarmos que H é formada pela matriz $-A^t$, 2×3, dada por

$$-A^t = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

e a matriz identidade 2×2. Logo,

$$H = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix}.$$

Para saber se um vetor $\mathbf{v}$ pertence a C, basta verificar a igualdade $H\mathbf{v}^t = 0$. Assim

$$H\mathbf{v}_1^t = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1+0+1+0+0 \\ 0+1+1+0+1 \end{pmatrix} = \begin{pmatrix} 2 \\ 3 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

e

$$H\mathbf{v}_2^t = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1+0+0+1+0 \\ 0+1+0+0+1 \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

portanto, $\mathbf{v}_1 \notin C$ e $\mathbf{v}_2 \in C$. □

A matriz teste de paridade de um código contém informações sobre o valor do peso d do código.

Proposição 4.33. *Seja H a matriz teste de paridade de um código C. Temos que o peso de C é maior do que ou igual a s se, e somente se, quaisquer $s-1$ colunas de H são linearmente independentes.*

Demonstração: Ver [15], página 98. □

Teorema 4.34. *Seja H a matriz teste de paridade de um código C. Temos que o peso de C é igual a s se, e somente se, quaisquer $s-1$ colunas de H são linearmente independentes e existem s colunas de H linearmente dependentes.*

Demonstração: Ver [15], página 98. □

Corolário 4.35. *(Limitante de Singleton) Os parâmetros (n,k,d) de um código linear satisfazem a desigualdade*

$$d \leq n - k + 1.$$

Demonstração: Se H é uma matriz teste de paridade, então será matriz geradora de um código dual com dimensão $n-k$. Com isso, H terá no máximo $n-k$ colunas linearmente independentes e pelo teorema anterior isso significa que $\omega(C) \leq n-k+1$. Mas $\omega(C) = d$, logo $d \leq n - k + 1$.

4.4 Decodificação

Chamamos de *decodificação* o processo de detecção e correção de erros em um determinado código. O método geral para decodificar códigos lineares que veremos a seguir é um aperfeiçoamento de um método criado, na década de 60, por D. Slepian do Laboratório Bell de Tecnologia. O método original de Slepian tinha um custo computacional muito elevado e os aperfeiçoamentos serviram para reduzir esse custo computacional.

Definimos, inicialmente o vetor erro $\mathbf{e}$, que é a diferença entre o vetor recebido $\mathbf{r}$ e o vetor transmitido $\mathbf{c}$, ou seja,

$$\mathbf{e} = \mathbf{r} - \mathbf{c}.$$

Por exemplo, dado um código sobre $\mathbb{F}_2$, se, ao transmitir a palavra $\mathbf{c} = (10011)$, recebermos a palavra $\mathbf{r} = (10111)$, então

$$\mathbf{e} = (10111) - (10011) = (00100).$$

Veja que o peso do vetor erro se refere ao número de erros cometidos numa palavra durante o processo de transmissão e recepção.

Considere H a matriz teste de paridade do código. Temos que $H\mathbf{c}^t = 0$, assim

$$H\mathbf{e}^t = H(\mathbf{r}^t - \mathbf{c}^t) = H\mathbf{r}^t - H\mathbf{c}^t = H\mathbf{r}^t.$$

Isso mostra que palavra recebida e o vetor erro possuem a mesma síndrome.

Vamos denotar por h^i a i-ésima coluna de H. Se $\mathbf{e} = (\alpha_1 ... \alpha_n)$, então

$$H\mathbf{e}^t = H\mathbf{r}^t = \alpha_1 h^1 + \alpha_2 h^2 + \cdots + \alpha_n h^n = \sum_{i=1}^{n} \alpha_i h^i.$$

Lema 4.36. *Seja C um código linear em K^n com capacidade de correção κ. Se $\boldsymbol{r} \in K^n$ e $\boldsymbol{c} \in C$ são tais que $d(\boldsymbol{c}, \boldsymbol{r}) \leq \kappa$, então existe um único vetor $\boldsymbol{e}$ com $\omega(\boldsymbol{e}) \leq \kappa$, cuja síndrome é igual à síndrome de $\boldsymbol{r}$ e tal que $\boldsymbol{c} = \boldsymbol{r} - \boldsymbol{e}$.*

Demonstração: Como $\mathbf{e} = \mathbf{r} - \mathbf{c}$, temos $\omega(\mathbf{e}) = d(\mathbf{c}, \mathbf{r}) \leq \kappa$. Agora mostraremos que este vetor $\mathbf{e}$ é único. De fato, suponhamos que existam $\mathbf{e} = (\alpha_1, ..., \alpha_n)$ e $\mathbf{f} = (\beta_1, ..., \beta_n)$ tais que $\omega(\mathbf{e}) \leq \kappa$ e $\omega(\mathbf{f}) \leq \kappa$ e tenham mesma síndrome que $\mathbf{r}$. Logo, sendo H uma matriz teste de paridade de C, temos

$$H\mathbf{e}^t = H\mathbf{f}^t \Longrightarrow \sum_{i=1}^{n} \alpha_i h^i = \sum_{i=1}^{n} \beta_i h^i \Longrightarrow \alpha_1 h^1 + \cdots + \alpha_n h^n = \beta_1 h^1 + \cdots + \beta_n h^n \Longrightarrow$$
$$(\alpha_1 - \beta_1)h^1 + \cdots + (\alpha_n - \beta_n)h^n = 0,$$

há assim uma relação de dependência linear entre 2κ ($\leq d - 1$) colunas de H. Mas, quaisquer $d - 1$ colunas de H são linearmente independentes, Teorema 4.34, logo $\alpha_i = \beta_i$ para todo i, portanto $\mathbf{e} = \mathbf{f}$. □

Assim, o problema colocado é como encontrar esse erro único com base em $H\mathbf{r}^t$.

Vamos determinar agora como encontrar o erro $\mathbf{e}$, quando $\omega(\mathbf{e}) \leq 1$, ou seja, quando ocorre no máximo um erro na transmissão da palavra.

Seja C um código com distância mínima $d \geq 3$ e tal que o vetor erro $\mathbf{e}$ foi introduzido entre a palavra transmitida $\mathbf{c}$ e a palavra recebida $\mathbf{r}$.

Se $H\mathbf{e}^t = 0$, então $\mathbf{r} \in C$ e tomamos $\mathbf{r} = \mathbf{c}$.

Suponhamos $H\mathbf{e}^t \neq 0$, assim $\omega(\mathbf{e}) = 1$ e, logo $\mathbf{e}$ tem uma única coordenada não nula. Consideremos, então, que $\mathbf{e} = (0, ..., \alpha, ..., 0)$ com $\alpha \neq 0$ na i-ésima posição. Logo,

$$H\mathbf{e}^t = \alpha h^i,$$

onde h^i é a i-ésima coluna de H. Dessa forma, mesmo não conhecendo $\mathbf{e}$, mas sabendo que

$$H\mathbf{e}^t = H\mathbf{r}^t = \alpha h^i,$$

podemos determinar $\mathbf{e}$ de modo que será o vetor com todas as componentes nulas exceto a i-ésima componente que é α. Note que i acima é bem determinado, pois $d \geq 3$. □

Assim estabelecemos, a seguir, o algoritmo de decodificação em códigos corretores de um único erro.

Seja H a matriz teste de paridade do código C e seja $\mathbf{r}$ um vetor recebido. (Suponha $d \geq 3$.)

(i) Calcule $H\mathbf{r}^t$;

(ii) Se $H\mathbf{r}^t = 0$, aceite $\mathbf{r}$ como sendo a palavra transmitida;

(iii) Se $H\mathbf{r}^t = \mathbf{s}^t \neq 0$, compare $\mathbf{s}^t$ com as colunas de H;

(iv) Se existirem i e α tais que $\mathbf{s}^t = \alpha h^i$, para $\alpha \in K$, então $\mathbf{e}$ é a n-upla com α na posição i e zeros nas outras posições. Corrija $\mathbf{r}$ pondo $\mathbf{c} = \mathbf{r} - \mathbf{e}$;

(v) Se o contrário de (iv) ocorrer, então mais de um erro foi cometido.

Exemplo 4.4 Considere o código C do exemplo 4.3 cuja matriz teste de paridade é

$$H = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix}.$$

. Supondo que seja recebida a palavra $\mathbf{r} = (10110)$, verifiquemos que houve um erro e determinemos a palavra enviada $\mathbf{c}$.

Temos que $H\mathbf{e}^t = H\mathbf{r}^t$, assim

$$H\mathbf{e}^t = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1+0+1+1+0 \\ 0+0+1+0+0 \end{pmatrix} = \begin{pmatrix} 3 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} = 1.h^3$$

onde h^3 é a terceira coluna de H. Assim, segue que $\mathbf{e} = (00100)$. Portanto,

$$\mathbf{c} = \mathbf{r} - \mathbf{e} = (10110) - (00100) = (10010).$$

□

Dado $\mathbf{v} \in K^n$, defina

$$\mathbf{v} + C = \{\mathbf{v} + \mathbf{c};\ \mathbf{c} \in C\}.$$

O conjunto do tipo $\mathbf{v} + C$ é chamado *classe lateral* de $\mathbf{v}$ segundo C. Veja que

$$\mathbf{v} + C = C \Leftrightarrow \mathbf{v} \in C.$$

Lema 4.37. *Os vetores $\boldsymbol{u}$ e $\boldsymbol{v}$ de K^n têm a mesma síndrome se, e somente se, $\boldsymbol{u} \in \boldsymbol{v} + C$.*

Demonstração: Os vetores $\mathbf{u}$ e $\mathbf{v}$ têm a mesma síndrome se, e somente se, $H\mathbf{u}^t = H\mathbf{v}^t \Longleftrightarrow H(\mathbf{u}-\mathbf{v})^t = 0 \Longleftrightarrow \mathbf{u}-\mathbf{v} \in C \Longleftrightarrow \mathbf{u}-\mathbf{v} = \mathbf{c}, \mathbf{c} \in C \Longleftrightarrow \mathbf{u} = \mathbf{v}+\mathbf{c}, \mathbf{c} \in C \Longleftrightarrow \mathbf{u} \in \mathbf{v} + C$. □

Proposição 4.38. *Seja C um (n,k)-código linear. As seguintes afirmações são verdadeiras:*

i) $\boldsymbol{v} + C = \boldsymbol{w} + C \Leftrightarrow \boldsymbol{v} - \boldsymbol{w} \in C$;

ii) $(\boldsymbol{v} + C) \cap (\boldsymbol{w} + C) \neq \emptyset \Rightarrow \boldsymbol{v} + C = \boldsymbol{w} + C$;

iii) $\bigcup_{\boldsymbol{v} \in K^n} (\boldsymbol{v} + C) = K^n$;

iv) $|(\boldsymbol{v}+C)| = |C| = q^k$.

Demonstração:
i) Temos que $v + C = w + C \Longrightarrow v + c = w + c'$, com $c, c' \in C \Longrightarrow v - w = c' - c \in C$, logo $v - w \in C$. Por outro lado, se $v - w \in C$, então existem $c, c' \in C$ tais que $c + (v - w) = c' \Longrightarrow v + c = w + c'$. Mas, temos que $v + c \in v + C$ e $w + c' \in w + C$, portanto $v + C = w + C$.

ii) Seja $x \in (v + C) \cap (w + C)$, logo $x \in v + C$ e $x \in w + C$, ou seja, existem $c, c' \in C$ tais que $x = v + c$ e $x = w + c'$, assim $v + c = w + c' \Longrightarrow v - w = c' - c \ \in C$, portanto, de (i), segue que $v + C = w + C$.

iii) Temos que $v \in K^n$ e $C \subset K^n$, logo $\bigcup_{\mathbf{v}\in K^n}(\mathbf{v} + C) \subset K^n$. Por outro lado, como C é um subespaço vetorial de K^n, então $0 \in C$. Assim, para todo $v \in K^n$, temos que v pode ser escrito como $v + 0$, logo v pertence a uma classe lateral de $v + C$, portanto, $K^n \subset \bigcup_{\mathbf{v}\in K^n}(\mathbf{v} + C)$.

iv) Consideremos a função f tal que

$$\begin{array}{rccc} f: & C & \longrightarrow & \mathbf{v} + C \\ & \mathbf{x} & \mapsto & v + \mathbf{x} = \mathbf{y} \end{array}$$

Assim, dados x_1 e $x_2 \in C$ temos

$$f(x_1) = f(x_2) \Rightarrow v + x_1 = v + x_2 \Rightarrow x_1 = x_2.$$

Ou seja, f é injetiva.

Temos ainda que

$$\forall\ \mathbf{y} = v + \mathbf{x}\ \in \mathbf{v} + C,\ \exists\ \mathbf{x} \in C \text{ tal que } f(\mathbf{x}) = \mathbf{y}.$$

Assim, f é sobrejetiva.

Portanto f é bijetiva e consequentemente $|(\mathbf{v} + C)| = |C| = q^k$. □

De (ii)-(iv), da proposição anterior, segue que o número de classes laterais segundo C é

$$\frac{q^n}{q^k} = q^{n-k}.$$

Observe ainda que o Lema 4.37 nos dá uma correspondência biunívoca entre síndromes e classes laterais. Isto é, todos os elementos de uma mesma classe lateral têm a mesma síndrome, enquanto elementos de classes laterais diferentes possuem síndromes diferentes.

Definição 4.31. *Um vetor de peso mínimo numa classe lateral é chamado de elemento líder dessa classe.*

Proposição 4.39. *Seja C um código linear em K^n com distância mínima d. Se $\boldsymbol{u} \in K^n$ é tal que*

$$\omega(\boldsymbol{u}) \leq \left\lfloor \frac{d-1}{2} \right\rfloor = \kappa,$$

então $\boldsymbol{u}$ é o único elemento líder de sua classe.

Demonstração: Suponhamos dois vetores $u, v \in K^n$ com $\omega(u) \leq \kappa$ e $\omega(v) \leq \kappa$ tais que u e v sejam da mesma classe segundo C. Logo, $u - v \in C$, e daí

$$\omega(u - v) \leq \omega(u) + \omega(v) \leq \kappa + \kappa = 2\kappa \leq d - 1.$$

Neste caso, temos que o vetor $u - v \in C$ tem peso menor que d, ou seja, $d(u, v) \leq d - 1 < d$. Logo, $u - v = 0 \Rightarrow u = v$. □

Para encontrar os líderes das classes, determinamos os elementos $\mathbf{v}$ de modo que $\omega(\mathbf{v}) \leq \left\lfloor \frac{d-1}{2} \right\rfloor = \kappa$. Tais elementos são líderes de uma única classe. Os líderes escolhidos são os de peso $\leq \kappa$, os outros são desconsiderados.

Discutiremos agora, um algoritmo para correção de mensagens onde, durante a transmisão, o número de erros sofridos seja menor ou igual a $\kappa = \left\lfloor \frac{d-1}{2} \right\rfloor$, que é a capacidade de correção do código.

Inicialmente devemos encontrar os elementos $\mathbf{v}$ em K^n, tais que $\omega(\mathbf{v}) \leq \kappa$. E logo a seguir determinar suas síndromes, colocando-as numa tabela. Consideremos uma palavra recebida $\mathbf{r}$.

Algoritmo de Decodificação

(1) Determine a síndrome $\mathbf{s}^t = H\mathbf{r}^t$;

(2) Se $\mathbf{s}$ se encontra na tabela, considere ℓ como elemento líder da classe que $\mathbf{s}$ determina. Substitua $\mathbf{r}$ por $\mathbf{r} - \ell$;

(3) Se $\mathbf{s}$ não se encontra na tabela, significa que a mensagem sofreu mais que κ erros.

Seja $\mathbf{r}$ a palavra recebida, e consideremos $\mathbf{c}$ e $\mathbf{e}$, a mensagem transmitida e o vetor erro, respectivamente. Temos que $H\mathbf{e}^t = H\mathbf{r}^t$, assim a classe lateral em que $\mathbf{e}$ pode ser encontrado fica conhecida através da síndrome de $\mathbf{r}$. Se $\omega(\mathbf{e}) \leq \kappa$, então $\mathbf{e}$ será o elemento líder único de sua classe, dado por ℓ, dessa forma o localizamos na tabela, e assim conseguimos determinar $\mathbf{c}$, pois $\mathbf{c} = \mathbf{r} - \mathbf{e} = \mathbf{r} - \ell$.

Exemplo 4.5 Considere o código binário C com matriz geradora

$$G = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}.$$

Queremos determinar a dimensão, o comprimento e o número de elementos de C, construir uma matriz teste de paridade H de C e determinar a distância mínima d de C.

Além disso, suponhamos que as seguintes informações são dadas:

espaço	00000	A	10000	B	01000	C	00100	D	00010	E	00001
F	11000	G	10100	H	10010	I	10001	J	01100	L	01010
M	01001	N	00110	O	00101	P	00011	Q	11100	R	10110
S	10101	T	11010	U	11001	V	01110	X	00111	Z	11110

Decodificaremos a mensagem recebida abaixo, admitindo que no máximo um erro é introduzido em cada palavra transmitida.

r: 011001100 011111001 110100101 010100010
101000110 110010001 100011010

Inicialmente devemos transformar a matriz

$$G = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

na forma padrão $(Id_k \mid A)$.

Através de escalonamentos nas linhas de G encontramos a matriz

$$G' = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{pmatrix}$$

que é uma matriz geradora na forma padrão de um código C' equivalente a C.

A partir das informações anteriores temos que a dimensão de C é $k = 5$, o comprimento do código é $n = 9$ e o número de elementos é $M = q^k$, como estamos trabalhando com um código binário temos que $q = 2$, logo $M = 2^5 = 32$ elementos.

Pela Proposição 4.26 a matriz teste de paridade é da forma $H = (-A^t \mid Id_{n-k})$. Assim,

$$H = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Na matriz H descrita acima podemos observar que quaisquer duas colunas são linearmente independentes, enquanto três colunas são linearmente dependentes, logo pelo Teorema 4.33 temos que $\omega(C) = 3$, ou seja, a distância mínima é $d = 3$.

Primeiramente devemos transformar o código da fonte em código de canal. Tal processo é feito por meio do acréscimo de redundâncias, através do produto entre o código da fonte e a matriz geradora do código. Por exemplo, dada a fonte A, cujo código da fonte é 10000, temos que o código do canal será

$$(10000) \cdot G = (10000) \cdot \begin{pmatrix} 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} = 110010000.$$

Repetindo o processo descrito acima para todas as fontes, obtemos a tabela a seguir:

Fonte	Código da Fonte	Cód. do Canal	Fonte	Código da Fonte	Cód. do Canal
espaço	00000	000000000	M	01001	101000110
A	10000	110010000	N	00110	101101001
B	01000	100100010	O	00101	110100101
C	00100	111000001	P	00011	011001100
D	00010	010101000	Q	11100	101110011
E	00001	001100100	R	10110	011111001
F	11000	010110010	S	10101	000110101
G	10100	001010001	T	11010	000011010
H	10010	100111000	U	11001	011010110
I	10001	111110100	V	01110	001001011
J	01100	011100011	X	00111	100001101
L	01010	110001010	Z	11110	111011011

Para decodificar as mensagens recebidas, inicialmente utilizaremos os vetores v_i tais que $\omega(v_i) \leq 1$ (pois estamos considerando que no máximo um erro foi introduzido), e calculamos suas respectivas síndromes por meio do produto $H \cdot v_i^t$, onde H é a matriz teste de paridade e v_i são os erros (e_i) líderes de cada classe. Dessa forma podemos contruir a seguinte tabela:

Erro (e_i) Líder	Síndrome ($H \cdot e_i^t$)	Erro (e_i) Líder	Síndrome ($H \cdot e_i^t$)
000000000	0000	000010000	1010
000000001	0001	000100000	1111
000000010	0010	001000000	1011
000000100	0100	010000000	0111
000001000	1000	100000000	1101

Agora, para encontrar as palavras recebidas (r_i), calcularemos suas respectivas síndromes $H \cdot r_i^t$ e os erros (e_i), comparando com a tabela anterior e determinando as palavras transmitidas (c_i), identificando assim, suas respectivas fontes. Por exemplo, supondo que a palavra $r = 111111011$ seja recebida, assim

$$H \cdot r_i^t = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \end{pmatrix} = 1111$$

logo, $e = 000100000$, consequentemente $c = r - e = 111111011 - 000100000 = 111011011$, onde o código da fonte é 11110, cuja fonte correspondente é Z.

Utilizando o processo acima desenvolvido podemos decodificar a mensagem enviada.

Pal. rec. (r_i)	Síndrome ($H \cdot r_i^t$)	Erro (e_i) Líder	Pal. tr. ($c_i = r_i - e_i$)	Fonte
011001100	0000	000000000	011001100	P
011111001	0000	000000000	011111001	R
110100101	0000	000000000	110100101	O
010100010	1010	000010000	010110010	F
101000110	0000	000000000	101000110	M
110010001	0001	000000001	110010000	A
100011010	1101	100000000	000011010	T

Assim, a mensagem transmitida é **PROFMAT**.

5 EXEMPLOS DE CÓDIGOS LINEARES

Veremos neste capítulo um pouco sobre o Código de Hamming, um dos códigos lineares mais utilizados e também alguns códigos que estão presentes em nosso dia a dia e que usam dígitos extras para controle de erros.

5.1 Código de Hamming

Como vimos na Seção 3.2, os primeiros estudos sobre Códigos Corretores de Erros foram desenvolvidos por Richard W. Hamming, na década de 40, onde necessitava de um código que fosse capaz de detectar e corrigir um erro durante a transmissão de mensagens. Veremos a partir de agora, o que é e como funciona o Código de Hamming.

Definição 5.32. *Os Códigos de Hamming de ordem m são construídos sobre o corpo finito* $\mathbb{F}_2$ *e determinados pela matriz teste de paridade* H_m*, onde as colunas desta matriz são todos os vetores não nulos de* $\mathbb{F}_2^m$*.*

Se C um Código de Hamming de ordem m, o comprimento de C é $n = 2^m - 1$ e a dimensão é $k = n - m = 2^m - 1 - m = 2^m - m - 1$. Assim, dizemos que um código de Hamming é um código linear binário $C(2^m - 1, 2^m - m - 1)$.

Observando a matriz teste de paridade de um código de Hamming, vemos que quaisquer duas colunas de H_m são linearmente independentes, enquanto que três colunas são linearmente dependentes. Dessa forma, pelo Teorema 4.33, segue que o $\omega(C) = 3$, ou seja, a distância mínima do código é $d = 3$. Consequentemente, pelo Teorema 3.17, temos que o código detecta até $d-1 = 3-1 = 2$ erros e corrige até $\kappa = \left\lfloor \frac{d-1}{2} \right\rfloor = \left\lfloor \frac{3-1}{2} \right\rfloor = \frac{2}{2} = 1$ erro.

Pela definição de código de Hamming, temos que a matriz teste de paridade possui $2^m - 1$ colunas, isto é, H_m é de ordem $m \times (2^m - 1)$. Além disso, por se tratar de um código linear, temos que a matriz teste de paridade H_m de um código de Hamming assume a forma padrão $H_m = (-A^t | Id_m)$. E assim, a sua correspondente matriz geradora na forma padrão é $G_m = (Id_k | A)$, de ordem $(2^m - m - 1) \times (2^m - 1)$, satisfazendo a equação $G_m . H_m^t = 0$.

Proposição 5.40. *Todo código de Hamming é perfeito.*

Demonstração: No código de Hamming temos $d = 3$, $\kappa = 1$ e $q = 2$. Seja $\mathbf{c}$ em $\mathbb{F}_2^n$, temos pelo Lema 3.15, página 42, que

$$|D(\mathbf{c},\kappa)| = \sum_{i=0}^{\kappa} \binom{n}{i} (q-1)^i \Rightarrow |D(\mathbf{c},1)| = \sum_{i=0}^{1} \binom{n}{i} (q-1)^i = \binom{n}{0} (2-1)^0 + \binom{n}{1} (2-1)^1 = 1+n.$$

Assim,

$$|\bigcup_{\mathbf{c}\in C} D(\mathbf{c},1)| = [1+n]\, 2^k = [1+2^m-1]\, 2^{n-m} = 2^m.2^n.2^{-m} = 2^0.2^n = 2^n,$$

portanto,

$$\bigcup_{\mathbf{c}\in C} D(\mathbf{c},1) = \mathbb{F}_2^n.$$

□

Sendo $d = 3$ a distância mínima de um código de Hamming, temos que

$$d = n - k + 1 \Rightarrow 3 = n - k + 1 \Rightarrow n - k = 2 \Rightarrow m = 2.$$

Isto é, a igualdade na Cota de Singleton se verifica para $m = 2$, logo o código de Hamming de ordem 2 é MDS.

5.1.1 O Código de Hamming C(7,4)

Vejamos agora como funciona o processo de codificação e decodificação de um código de Hamming $C(2^m - 1, 2^m - m - 1)$ para o caso $m = 3$, isto é, $C(7,4)$. Aqui seguiremos a ideias das referências [8] e [13].

Para transmitir uma mensagem em $C(7,4)$ em um canal binário simétrico, tomamos uma palavra $x = x_1x_2x_3x_4 \in \mathbb{F}_2^4$ e a transformamos em $y = y_1y_2y_3y_4y_5y_6y_7 \in \mathbb{F}_2^7$. Para esse processo usaremos a codificação $x \cdot G_3$, onde G_3 é uma matriz geradora de C na forma padrão. Tomando

$$G_3 = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix},$$

temos

$$x \cdot G_3 = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix} =$$

$$\begin{pmatrix} x_1 & x_2 & x_3 & x_4 & x_1+x_2+x_4 & x_1+x_3+x_4 & x_2+x_3+x_4 \end{pmatrix} = y.$$

Portanto, a codificação é

$$\begin{aligned} y_1 &= x_1 \\ y_2 &= x_2 \\ y_3 &= x_3 \\ y_4 &= x_4 \\ y_5 &= x_1 + x_2 + x_4 \\ y_6 &= x_1 + x_3 + x_4 \\ y_7 &= x_2 + x_3 + x_4. \end{aligned}$$

Supondo que desejamos enviar a palavra $x = 0111$, então a codificação será

$$\begin{aligned} y_1 &= 0 \\ y_2 &= 1 \\ y_3 &= 1 \\ y_4 &= 1 \\ y_5 &= 0+1+1 = 2 = 0 \\ y_6 &= 0+1+1 = 2 = 0 \\ y_7 &= 1+1+1 = 3 = 1 \end{aligned}$$

assim, a palavra codificada é $y = 0111001$.

Lembremos que as operações são realizadas sobre o corpo finito $\mathbb{F}_2$.

Vejamos agora como decodificar uma mensagem recebida. Suponha um canal binário, no qual é usada a codificação vista anteriormente. Seja $y = 1111111$ a palavra recebida, então podemos nos questionar: será que houve erro durante a transmissão? Qual a palavra enviada? Comparando a palavra recebida com a codificação dada, temos

$$\begin{aligned} x_1 &= 1 = y_1 \\ x_2 &= 1 = y_2 \\ x_3 &= 1 = y_3 \\ x_4 &= 1 = y_4 \\ x_1 + x_2 + x_4 &= 1+1+1 = 1 = y_5 \\ x_1 + x_3 + x_4 &= 1+1+1 = 1 = y_6 \\ x_2 + x_3 + x_4 &= 1+1+1 = 1 = y_7 \end{aligned} \quad .$$

Observamos assim que não houve erro durante a transmissão da palavra e além disso, a palavra enviada foi $x = x_1x_2x_3x_4$, ou seja, $x = 1111$.

Suponhamos agora que recebemos uma palavra $y = 0000111$, dessa forma, comparando com a codificação usada, temos

$$\begin{aligned} x_1 = 0 = y_1 \\ x_2 = 0 = y_2 \\ x_3 = 0 = y_3 \\ x_4 = 0 = y_4 \\ x_1 + x_2 + x_4 = 0 + 0 + 0 = 0 \neq 1 = y_5 \\ x_1 + x_3 + x_4 = 0 + 0 + 0 = 0 \neq 1 = y_6 \\ x_2 + x_3 + x_4 = 0 + 0 + 0 = 0 \neq 1 = y_7 \end{aligned} .$$

Vemos então que houve um erro, pois a codificação não condiz com a palavra recebida. Mas onde foi este erro? Note que o erro ocorre nos últimos três termos e que o dígito x_4 é comum às três equações da codificação dada. Assim, somos levados a crer que o erro está neste dígito, o que de fato é verdade, pois ao trocarmos o valor de x_4, isto é, substituirmos o 0 por 1, então vemos que as três equações são satisfeitas

$$\begin{aligned} x_1 + x_2 + x_4 = 0 + 0 + 1 = 1 = y_5 \\ x_1 + x_3 + x_4 = 0 + 0 + 1 = 1 = y_6 \\ x_2 + x_3 + x_4 = 0 + 0 + 1 = 1 = y_7 \end{aligned} .$$

Portanto, a palavra transmitida é $x = 0001$.

Veremos agora como funciona o processo de codificação e decodificação do código de Hamming usando as matrizes teste de paridade e geradora do código.

5.1.2 Codificando e Decodificando Códigos de Hamming

Como sabemos, um Código de Hamming $C(2^m - 1, 2^m - m - 1)$ é determinado por uma matriz teste de paridade H_m e a ela podemos associar a matriz geradora G_m correspondente. Assim, para codificar a palavra a ser transmitida $c \in \mathbb{F}_2^{2^m - m - 1}$, basta multiplicar pela matriz geradora G_m e encontramos $c \cdot G_m \in \mathbb{F}_2^{2^m - 1}$. As m letras adicionadas em c, após a codificação, é a redundância e servem exatamente para detectar e corrigir possíveis erros que podem ocorrer durante a transmissão da palavra.

Para decodificar uma palavra recebida r realizamos o processo a seguir.

Podemos considerar a palavra r como sendo uma matriz linha de ordem $1 \times (2^m - 1)$. Dada uma matriz teste de paridade H_m, temos, por definição, que será da ordem $m \times (2^m - 1)$. Assim, ao realizarmos o produto

$$H_m \cdot r^t$$

o resultado obtido será uma matriz coluna de ordem $m \times 1$. Esse processo de multiplicação de matrizes é o mesmo visto na Seção 2.3, Capítulo 2.

Se não houve erro durante a transmissão, então esta matriz resultante será a matriz coluna nula. Porém, se ocorreu um erro durante o processo de envio da palavra, a matriz resultante será uma coluna h_i, $i = 1, ..., 2^m - 1$, da matriz teste de paridade H_m.

Dessa forma, veremos que na palavra transmitida ocorreu um erro na coordenada r_i, $i = 1, ..., 2^m - 1$, e assim podemos corrigí-lo e determinar r', que será a codificação exata da palavra c transmitida.

Após determinar r' podemos identificar qual foi a palavra transmitida, uma vez que os primeiros $2^m - m - 1$ dígitos da codificação representam a palavra enviada.

Para entender esse processo vejamos um exemplo.

Exemplo 5.1 Consideremos o código de Hamming $C(7, 4)$, ou seja, o caso $m = 3$. Uma matriz teste de paridade na forma padrão é dada por

$$H_3 = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}.$$

Suponhamos que recebemos a palavra $r = 1010101$. Assim

$$H_3 \cdot r^t = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1+0+1+0+1+0+0 \\ 1+0+0+0+0+0+0 \\ 1+0+1+0+0+0+1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}.$$

Note que a matriz coluna resultante é igual a coluna h_1 de H_3, isto é, a palavra r sofreu um erro na coordenada r_1, portanto a palavra correta é $r' = 0010101$.

Caso queiramos determinar a palavra enviada, basta lembrarmos que em $C(7, 4)$, os quatro primeiros dígitos representam a palavra transmitida, enquanto os três últimos dígitos são apenas redundâncias, que servem exatamente para detecção e correção de eventuais erros que possam acontecer. Dessa forma, observamos a partir de r' que a palavra transmitida foi $c = 0010$.

Exemplo 5.2 Dado o código de Hamming $C(2^m - 1, 2^m - m - 1)$, determinemos para $m = 4$, as matrizes geradora e teste de paridade. Encontrar também qual foi a palavra transmitida, sabendo que recebemos $r = 111110000011111$.

Para $m = 4$, temos $2^m - 1 = 2^4 - 1 = 16 - 1 = 15$ e $2^m - m - 1 = 2^4 - 4 - 1 = 16 - 4 - 1 = 11$. Logo estamos trabalhando sobre o código de Hamming $C(15, 11)$. Uma matriz teste de paridade para este código pode ser da forma

$$H_4 = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

A matriz geradora correspondente será

$$G_4 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{pmatrix}.$$

Agora determinaremos a palavra transmitida. Inicialmente, devemos calcular $H_4 \cdot r^t$, onde $r = 111110000011111$. Logo,

$$H_4 \cdot r^t = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 6 \\ 5 \\ 5 \\ 5 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \end{pmatrix}.$$

Como $H_4 \cdot r^t \neq 0$, temos que houve um erro na transmissão. Mas, percebemos que a matriz resultante é igual a coluna h_2 de H_4, isto é, o erro ocorreu na segunda coordenada de r. Assim, a palavra correta a ser recebida é $r' = 101110000011111$. Consequentemente, a palavra transmitida foi $c = 10111000001$.

5.2 Alguns códigos do cotidiano

Veremos agora alguns exemplos de códigos que estão presentes diariamente em nossas vidas sem nem mesmo percebermos. Os códigos estudados possuem elementos capazes de detectar erros, tais elementos são chamados dígitos de verificação (ou de controle). Vejamos.

Exemplo 5.4 Os livros possuem um número de identificação único chamado ISBN (*International Standard Book Number*). Este código consiste de 10 dígitos da forma $a_1a_2 - a_3a_4a_5a_6a_7 - a_8a_9 - a_{10}$. Os três primeiros segmentos identificam o grupo linguístico, a editora e o volume, enquanto o último dígito é de verificação e pode ser escolhido entre $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, X\}$, onde X representa 10 em algarismos romanos. O símbolo a_{10} é escolhido como o resto da divisão de $\sum_{i=1}^{9} i.a_i$ por 11. Vejamos como encontrar o dígito verificador do ISBN $85 - 87571 - 26 - a_{10}$.

Devemos inicialmente calcular a soma $\sum_{i=1}^{9} i.a_i$ e depois dividir por 11. Assim, temos

$$\sum_{i=1}^{9} i.a_i = 1 \times 8 + 2 \times 5 + 3 \times 8 + 4 \times 7 + 5 \times 5 + 6 \times 7 + 7 \times 1 + 8 \times 2 + 9 \times 6 =$$
$$8 + 10 + 24 + 28 + 25 + 42 + 7 + 16 + 54 =$$
$$214 = 11 \times 19 + 5.$$

Logo, $a_{10} = 5$, e o número procurado é 85-87571-26-5.

Podemos observar também que soma $\sum_{i=1}^{10} i.a_i$ é divisível por 11. Para verificar esta afirmação precisamos mostrar que existe $m \in \mathbb{Z}$ tal que $\sum_{i=1}^{10} i.a_i = 11m$. Como pela definição a_{10} é o resto da divisão de $\sum_{i=1}^{9} i.a_i$ por 11, então dado $q \in \mathbb{Z}$, temos

$$\sum_{i=1}^{9} i.a_i = 11q + a_{10} \Rightarrow \sum_{i=1}^{9} i.a_i + 10.a_{10} = 11q + a_{10} + 10.a_{10} \Rightarrow$$
$$\sum_{i=1}^{10} i.a_i = 11q + 11.a_{10} \Rightarrow \sum_{i=1}^{10} i.a_i = 11(q + a_{10}).$$

Mas $q \in \mathbb{Z}$ e $a_{10} \in \mathbb{Z}$, logo $q + a_{10} = m \in \mathbb{Z}$. Portanto,

$$\sum_{i=1}^{10} i.a_i = 11m.$$

Além disso, este código pode detectar um erro em um dígito. De fato, como vimos anteriormente o resultado da soma $\sum_{i=1}^{10} i.a_i$ é um múltiplo de 11 e $a_i \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$. Assim, ao modificar o valor de algum a_i para a_k, onde $a_k \in \{0, 1, ..., 9\} \setminus \{a_i\}$, para que o resultado do somatório permaneça um múltiplo de 11, é necessário que a diferença entre os termos alterados seja um múltiplo de 11, isto é,

$$i.a_i - i.a_k = 11q, \ q \in \mathbb{Z}.$$
$$\Rightarrow i(a_i - a_k) = 11q.$$

Mas, a_i e a_k pertencem ao conjunto $\{0,1,2,3,4,5,6,7,8,9\}$, assim, o único múltiplo de 11 que pode satisfazer a equação é o 0 (zero). Dessa forma,

$$i(a_i - a_k) = 0.$$

Como $1 \leq i \leq 10$, a única possibilidade é

$$a_i - a_k = 0 \Rightarrow a_i = a_k.$$

Portanto, se $a_i \neq a_k$, então o resultado da soma não será um múltiplo de 11, e assim pode-se detectar o erro.

Um tipo de erro comum que se pode notar ao trabalharmos com muitos números é a inversão de dois dígitos. Por exemplo, o código 70-12345-67-5 pode ser erroneamente digitado como 70-12354-67-5. Vejamos que o código permite detectar esse tipo de erro desde que os dois dígitos consecutivos não sejam iguais (caso em que não ocorre o erro). Consideremos no somatório $\sum_{i=1}^{10} i.a_i$, dois dígitos consecutivos a_i e a_{i+1}. Dentro da soma estes termos aparecerão como

$$i.a_i + (i+1).a_{i+1}.$$

Suponhamos que houve a inversão desses dígitos, assim a soma seria

$$i.a_{i+1} + (i+1).a_i.$$

Lembramos que todos os outros termos da soma permanecerão iguais. Assim, para que o resultado final continue sendo um múltiplo de 11, devemos ter

$$\begin{gathered} i.a_i + (i+1).a_{i+1} = i.a_{i+1} + (i+1).a_i \Rightarrow \\ i.a_i + i.a_{i+1} + a_{i+1} = i.a_{i+1} + i.a_i + a_i \Rightarrow \\ i.a_i + i.a_{i+1} + a_{i+1} - i.a_i - i.a_{i+1} = a_i \Rightarrow \\ a_{i+1} = a_i. \end{gathered}$$

Portando, desde que $a_i \neq a_{i+1}$, então a soma final será diferente do esperado e assim o código detectará o erro.

Exemplo 5.4 O número do CPF (Cadastro de Pessoa Física) é composto de onze dígitos, onde os dois últimos são dígitos de verificação. Denotaremos por a_i o dígito na posição i, para $i = 1, 2, ..., 11$. Assim, temos que um número de CPF assume a forma $a_1a_2a_3a_4a_5a_6a_7a_8a_9 - a_{10}a_{11}$. Um número de CPF será válido se os dígitos de verificação obedecerem as seguintes condições:

- a_{10} é o resto da divisão por 11 de $\sum_{i=1}^{9} i.a_i$.

- a_{11} é o resto da divisão por 11 de $\sum_{i=2}^{10}(i-1).a_i$.

Como em cada caso estamos nos referindo a um único dígito, então se a_{10} ou a_{11} for igual a 10, substituímos este valor por 0. Vamos determinar os dígitos verificadores do CPF $103452281 - a_{10}a_{11}$. Primeiro determinaremos a_{10}, para isso devemos encontrar o resultado da soma indicada, assim

$$\sum_{i=1}^{9} i.a_i = 1 \times a_1 + 2 \times a_2 + 3 \times a_3 + 4 \times a_4 + 5 \times a_5 + 6 \times a_6 + 7 \times a_7 + 8 \times a_8 + 9 \times a_9 =$$
$$1 \times 1 + 2 \times 0 + 3 \times 3 + 4 \times 4 + 5 \times 5 + 6 \times 2 + 7 \times 2 + 8 \times 8 + 9 \times 1 =$$
$$1 + 0 + 9 + 16 + 25 + 12 + 14 + 64 + 9 = 150.$$

Mas

$$150 = 11 \times 13 + 7.$$

Logo $a_{10} = 7$.

Agora encontraremos a_{11}, para isso calculamos

$$\sum_{i=2}^{10}(i-1).a_i = (2-1).a_2 + (3-1).a_3 + (4-1).a_4 + (5-1).a_5 + (6-1).a_6 +$$
$$(7-1).a_7 + (8-1).a_8 + (9-1).a_9 + (10-1).a_{10} =$$
$$1 \times 0 + 2 \times 3 + 3 \times 4 + 4 \times 5 + 5 \times 2 + 6 \times 2 + 7 \times 8 + 8 \times 1 + 9 \times 7 =$$
$$0 + 6 + 12 + 20 + 10 + 12 + 56 + 8 + 63 = 187.$$

E

$$187 = 11 \times 17 + 0.$$

Assim, $a_{11} = 0$.

Portanto, o número procurado é $103452281 - 70$.

Assim com o ISBN, o código do CPF também é capaz de detectar erro em um dígito. De fato, temos que a soma dos primeiros dez dígitos é um múltiplo de 11, isto é, existe $m \in \mathbb{Z}$ tal que

$$\sum_{i=1}^{10} i.a_i = 11m \Rightarrow 1.a_1 + \sum_{i=2}^{10} i.a_i = 11m \Rightarrow \sum_{i=2}^{10} i.a_i = 11m - a_1.$$

Pela definição temos que existe $q \in \mathbb{Z}$ tal que

$$\sum_{i=2}^{10}(i-1).a_i = 11q + a_{11} \Rightarrow \sum_{i=2}^{10}(ia_i - a_i) = 11q + a_{11} \Rightarrow$$
$$\sum_{i=2}^{10} ia_i - \sum_{i=2}^{10} a_i = 11q + a_{11} \Rightarrow 11m - a_1 - \sum_{i=2}^{10} a_i = 11q + a_{11} \Rightarrow$$
$$11m - 11q = a_1 + \sum_{i=2}^{10} a_i + a_{11} \Rightarrow \sum_{i=1}^{11} a_i = 11(m-q).$$

Como $m \in \mathbb{Z}$ e $q \in \mathbb{Z}$, então $m - q \in \mathbb{Z}$. Portanto, $\sum_{i=1}^{11} a_i$ é um múltiplo de 11.

Seguindo as ideias do exemplo anterior, vemos que o código do CPF pode detectar um erro em um dígito.

Exemplo 5.5 O método de construção dos números de cartão de crédito foram introduzidos pela IBM (*International Business Machines*) como veremos a seguir. Considere um número com n dígitos, $a_1, ..., a_n$, com $a_i \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$. Este número será válido se o resultado da soma $\sum_{i=1}^{n} c_i$ for múltiplo de 10, onde c_i é dado como se segue:

- se i é ímpar, definimos $c_i = a_i$;
- se i é par e $2a_i < 10$, definimos $c_i = 2a_i$;
- se i é par e $2a_i > 10$, definimos $c_i = 2a_i - 9$.

Como exemplo vamos determinar o 16º dígito de um cartão de crédito tais que os 15 primeiros dígitos são 1234 5678 1234 567. Temos que o cartão possui 16 dígitos e pela definição, o resultado da soma $\sum_{i=1}^{16} c_i$ deve ser um múltiplo de 10. Temos ainda que, pelas definições do c_i que, para i ímpar

$$\begin{aligned} c_1 &= a_1 = 1 & \qquad c_9 &= a_9 = 1 \\ c_3 &= a_3 = 3 & \qquad c_{11} &= a_{11} = 3 \\ c_5 &= a_5 = 5 & \qquad c_{13} &= a_{13} = 5 \\ c_7 &= a_7 = 7 & \qquad c_{15} &= a_{15} = 7. \end{aligned}$$

Agora, para i par, temos
$a_2 = 2 \Rightarrow 2a_2 = 4 < 10 \Rightarrow c_2 = 4$
$a_4 = 4 \Rightarrow 2a_4 = 8 < 10 \Rightarrow c_4 = 8$
$a_6 = 6 \Rightarrow 2a_6 = 12 > 10 \Rightarrow c_6 = 2a_6 - 9 = 12 - 9 = 3 \Rightarrow c_6 = 3$
$a_8 = 8 \Rightarrow 2a_8 = 16 > 10 \Rightarrow c_8 = 2a_8 - 9 = 16 - 9 = 7 \Rightarrow c_8 = 7$
$a_{10} = 2 \Rightarrow 2a_{10} = 4 < 10 \Rightarrow c_{10} = 4$
$a_{12} = 4 \Rightarrow 2a_{12} = 8 < 10 \Rightarrow c_{12} = 8$
$a_{14} = 6 \Rightarrow 2a_{14} = 12 > 10 \Rightarrow c_{14} = 2a_{14} - 9 = 12 - 9 = 3 \Rightarrow c_{14} = 3.$
Assim,

$$\sum_{i=1}^{16} c_i = c_1 + c_2 + c_3 + c_4 + c_5 + c_6 + c_7 + c_8 + c_9 + c_{10} + c_{11} + c_{12} + c_{13} + c_{14} + c_{15} + c_{16} =$$
$$1 + 4 + 3 + 8 + 5 + 3 + 7 + 7 + 1 + 4 + 3 + 8 + 5 + 3 + 7 + c_{16} = 69 + c_{16}.$$

Como o resultado da soma deve ser um múltiplo de 10 e $c_i \in \{0, 1, 2, ..., 9\}$, então devemos ter

$$69 + c_{16} = 70 \Rightarrow c_{16} = 70 - 69 \Rightarrow c_{16} = 1.$$

Mas 16 é par então $c_{16} = 2a_{16}$ ou $c_{16} = 2a_{16} - 9$. Para o primeiro caso teríamos

$$c_{16} = 2a_{16} \Rightarrow 1 = 2a_{16} \Rightarrow a_{16} = \frac{1}{2},$$

o que não pode ocorrer, pois $a_i \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$.
Para o segundo caso, temos

$$c_{16} = 2a_{16} - 9 \Rightarrow 1 = 2a_{16} - 9 \Rightarrow 2a_{16} = 10 \Rightarrow a_{16} = 5.$$

Portanto, o valor correto é $a_{16} = 5$. Assim, o número completo do cartão de crédito é 1234 5678 1234 5675.

Uma característica importante é que este método também pode detectar erro em um dos dígitos. De fato, usando a mesma ideia do exemplo 5.3, temos pela definição que $\sum_{i=1}^{n} c_i$ deve ser um múltiplo de 10. Assim, ao alterarmos um dígito a_i para a_k, onde esses termos são diferentes, para que o resultado da soma continue sendo divisível por 10, é necessário que a diferença entre os correspondentes c_i e c_k seja um múltiplo de 10. Mas sabemos que $c_i \in \{0, 1, 2, ..., 9\}$, logo a única possibilidade é $c_i = c_k \Rightarrow a_i = a_k$, um absurdo, pois consideramos $a_i \neq a_k$. Portanto, ao alterarmos o valor de um dígito, o resultado da soma $\sum_{i=1}^{n} c_i$ deixará de ser um múltiplo de 10, e assim o código detectará o erro.

Temos ainda que um erro comum é a inversão de dois dígitos consecutivos. Vamos mostrar que o método da IBM é capaz de detectar tais erros se dois dígitos consecutivos não são o mesmo (caso onde não há erro) e se ambos não estão no conjunto $\{0, 9\}$. Consideremos dois dígitos consecutivos a_i e a_{i+1}. Sem perda de generalidade vamos considerar i ímpar, assim, $i+1$ é par, logo temos, pela definição, duas situações $c_i = a_i$ e $c_{i+1} = 2a_{i+1}$ ou $c_i = a_i$ e $c_{i+1} = 2a_{i+1} - 9$. Ao invertermos as posições de a_i e a_{i+1}, para que o resultado de $\sum_{i=1}^{n} c_i$ se mantenha, devemos observar a soma $c_i + c_{i+1}$ deve ser igual em duas ocasiões, $c_i = a_i$ e $c_i = a_{i+1}$. Logo, para o primeiro caso, temos

$$a_i + 2a_{i+1} = a_{i+1} + 2a_i \Rightarrow a_i + a_{i+1} + a_{i+1} = a_{i+1} + a_i + a_i \Rightarrow a_{i+1} = a_i.$$

Ou

$$a_i + 2a_{i+1} - 9 = a_{i+1} + 2a_i - 9 \Rightarrow a_i + a_{i+1} + a_{i+1} - 9 = a_{i+1} + a_i + a_i - 9 \Rightarrow a_{i+1} = a_i.$$

Assim, se $a_i \neq a_{i+1}$, então o resultado da soma $\sum_{i=1}^{n} c_i$ será diferente do esperado e assim o código irá detectar o erro.

Vejamos o caso em que os dígitos consecutivos são 0 e 9. Temos que nesta situação o código não detectará o erro. De fato, dado i ímpar (para i par o processo é análogo), tomando $a_i = 0$ e $a_{i+1} = 9$, temos

$$c_i = a_i = 0 \text{ e } c_{i+1} = 2a_{i+1} - 9 = 2.9 - 9 = 18 - 9 = 9.$$

Ou, tomando $a_i = 9$ e $a_{i+1} = 0$

$$c_i = a_i = 9 \text{ e } c_{i+1} = 2a_{i+1} = 2.0 = 0.$$

Em ambos os casos vemos que $c_i + c_{i+1} = 9$. Portanto, ao inverter esses dois dígitos o código não identificará que houve erro.

6 CONSIDERAÇÕES FINAIS

Vimos ao longo deste trabalho um pouco sobre os Códigos Corretores de Erro e sua importância na transmissão de informações de forma confiável, uma vez que conseguem detectar e corrigir erros em canais de comunicação, preservando a mensagem originalmente enviada. Apresentamos também alguns aspectos históricos sobre o surgimento desses códigos e mostramos que são desenvolvidos mediante uma matemática "básica" vista tanto no Ensino Médio como no Superior.

Observamos a constante presença da Álgebra (vetorial, linear e abstrata) na construção dos Códigos Lineares, no qual destacamos o Código de Hamming. Os elementos dos códigos podem ser enxergados como vetores, o alfabeto em que está inserido é um corpo finito e durante a codificação usamos as transformações lineares. Vale ressaltar também o uso contínuo de matrizes, como por exemplo as matrizes geradora e teste de paridade de um código, que são a base para o processo de codificação e decodificação.

Os códigos vistos no último capítulo (ISBN, CPF e cartão de crédito) trazem em sua composição uma matemática ainda mais simples como as operações básicas adição, subtração, multiplicação e divisão.

Enfatizamos que este estudo é apenas uma introdução à Teoria dos Códigos Corretores de Erros, área esta bastante abrangente. Essa teoria pode ser abordada pelo viés algébrico, o qual vimos neste trabalho, e também na perspectiva geométrica. As referências [15] e [8], fazem um estudo mais aprofundado destes dois conceitos da Teoria dos Códigos, algébrico e geométrico, respectivamente.

Pretendemos em pesquisas futuras aprofundar nosso conhecimento sobre os Códigos Corretores de Erros e estudar outros tipos de códigos não discutidos neste trabalho, sejam eles baseados em argumentos algébricos ou geométricos.

Diante do estudo realizado sobre os Códigos Corretores de Erros, almejamos que professores e alunos de Ensino Básico e Superior, ao lerem este trabalho, possam observar a beleza da Matemática e o quanto ela está presente em nossa vida. E que, apesar de muitas vezes parecer complicada e desnecessária, ela é essencial para o progresso da humanidade através da tecnologia e inovação.

REFERÊNCIAS

[1] Bahia, F., *Um Primeiro Curso sobre Códigos Corretores de Erro*, Anais do I Encontro de Matemática Aplicada e Computacional, p. 149-169, São João Del Rei, 2010.

[2] Bollauf, M. F., *Códigos, Reticulados e Aplicações em Criptografia*, Dissertação de Mestrado, Unicamp, 2015.

[3] Carvalho, S. M. G. de, *Matrizes, Determinantes e Polinômios: Aplicações em códigos corretores de erros, como estratégia motivacional para o ensino de matemática*, Dissertação de Mestrado PROFMAT, Porto Velho, 2014.

[4] Filho, M. F. A., *Geometria Analítica e Álgebra*, Fortaleza, Edições Livro Técnico e Premius Editora, 2001.

[5] Garcia, A. e Lequain, Y., *Elementos de Álgebra*, 6ª ed., Rio de Janeiro, IMPA, 2018.

[6] Gonçalves, A., *Introdução à Álgebra*, 6ª ed., Rio de Janeiro, IMPA, 2017.

[7] Hefez, A., *Curso de Álgebra volume 1*, 3ª ed., Rio de Janeiro, IMPA, 2002.

[8] Lavor, C. C. [et al.], *Uma Introdução à Teoria de Códigos* - São Carlos, SP: SBMAC, 2006.

[9] Lima, E. L., *Álgebra Linear*, 9ª ed., Rio de Janeiro, IMPA, 2018.

[10] Milies, C. P., *Breve Introdução à Teoria dos Códigos Corretores de Erros*, disponível em http://www.kurims.kyoto-u.ac.jp/EMIS/journals/em/docs/coloquios/NE-1.04.pdf.

[11] Neves, L. X., *Uma Introdução à Teoria dos Códigos Corretores de Erros*, Monografia de Graduação - UESC, Ilhéus, 2003.

[12] Nogueira, J. A. P., *Códigos Corretores de Erro*, Monografia de Especialização - URCA, Juazeiro do Norte, 2017.

[13] Rousseau, C., e Saint-Aubin, Y.; tradução de Miguel V. S. Frasson, *Matemática e Atualidade volume 1*, 1ª ed., Rio de Janeiro, SBM, 2015.

[14] Steinbruch, A. e Winterle, P., *Álgebra Linear*, 3ª ed, Pearson, 2006.

[15] Villela, M. L. T. e Hefez, A., *Códigos Corretores de Erros*, IMPA, 2ª ed., Rio de Janeiro, 2008.

[16] Zahn, M., *Introdução à Álgebra*, Rio de Janeiro, Editora Ciência Moderna Ltda., 2013.