O princípio da inclusão-exclusão e o cálculo de permanentes

The inclusion-exclusion principle and the calculation of permanents

El principio de inclusión-exclusión y el cálculo de permanentes

José Ricardo Gonçalves de Mendonça[1]
Universidade de São Paulo, São Paulo, SP, Brasil
https://orcid.org/0000-0002-5516-0568
http://lattes.cnpq.br/8792749813872106

Resumo: Neste artigo revisamos o princípio da inclusão-exclusão (PIE) sob os pontos de vista conjuntista e algébrico e discutimos sua aplicação ao cálculo de permanentes, um assunto que normalmente não é abordado em cursos de graduação. A apresentação procura ser rigorosa porém elementar e acessível a alunos dos anos iniciais de cursos de licenciatura ou bacharelado em matemática, ciências e engenharias, exigindo somente familiaridade com notação de conjuntos, aritmética e álgebra de matrizes. No tratamento do cálculo de permanentes, apresentamos o algoritmo de Ryser, um dos desenvolvimentos mais espetaculares na abordagem de problemas combinatoriais difíceis, cuja complexidade algorítmica discutimos brevemente. O artigo inclui exemplos, notas complementares e um programa em Python que implementa o algoritmo de Ryser usando códigos de Gray para o cálculo de permanentes, juntamente com sua discussão.

Palavras-chave: matemática discreta; análise combinatória; partição de conjuntos; algoritmo de Ryser; problemas #P-completos.

Abstract: In this article we review the inclusion-exclusion principle (PIE) from set theoretical and algebraic points of view and discuss its application to the calculation of permanents, a subject that is not normally covered in undergraduate courses. The presentation is intended to be rigorous but elementary and accessible to first-year students in mathematics, science, and engineering programs, requiring only familiarity with set notation, arithmetic and matrix algebra. In dealing with the calculation of permanents, we present the Ryser algorithm, one of the most spectacular developments in the approach to difficult combinatorial problems, whose computational complexity we briefly discuss. The article contains examples, complementary notes, and a program in Python that implements Ryser's algorithm using Gray codes to calculate permanents, accompanied by its discussion.

Keywords: discrete mathematics; combinatorial analysis; set partition; Ryser's algorithm; #P-complete problems.

Resumen: En este artículo revisamos el principio de inclusión-exclusión (PIE) desde los puntos de vista conjuntista y algebraico y discutimos su aplicación al cálculo de permanentes, un tema que normalmente no se trata en los cursos de pregrado. La presentación busca ser rigurosa pero elemental y accesible para los estudiantes de los primeros años de programas de matemáticas, ciencias e ingeniería, y que sólo requiera familiaridad con la notación de conjuntos, la aritmética y el álgebra matricial. Al abordar el cálculo de permanentes, presentamos el algoritmo de Ryser, uno de los desarrollos más espectaculares en el tratamiento de problemas combinatorios difíciles, cuya complejidad computacional discutimos brevemente. El artículo contiene ejemplos, notas complementarias y un programa en Python que implementa el algoritmo de Ryser utilizando códigos de Gray para calcular permanentes, acompañado de su discusión.

[1] **Currículo sucinto**: Bacharel e mestre em Física do Estado Sólido pela Universidade de São Paulo, doutor em Física Estatística pela Universidade Federal de São Carlos, docente da Universidade de São Paulo e orientador credenciado junto aos programas de pós-graduação em Modelagem de Sistemas Complexos (EACH/USP), Matemática Aplicada (IME/USP) e PROFMAT (SBM@ICMC/USP). **Contribuição de autoria**: Obtenção de financiamento, conceituação, metodologia, investigação, validação e visualização, software, escrita – primeira redação, escrita – revisão e edição. **Contato**: jricardo@usp.br.

MENDONÇA, José Ricardo Gonçalves de. O princípio da inclusão-exclusão e o cálculo de permanentes. **REMAT**: Revista Eletrônica da Matemática, Bento Gonçalves, RS, v. 10, n. 2, p. e3005, 30 de setembro de 2024.
https://doi.org/10.35819/remat2024v10i2id6987

Palabras clave: matemáticas discretas; análisis combinatorio; partición de conjuntos; algoritmo de Ryser; problemas #P-completos.

Data de submissão: 27 de novembro de 2023.

Data de aprovação: 3 de maio de 2024.

1 Introdução

Em análise combinatória elementar somos apresentados a dois princípios básicos de contagem: o **princípio da adição** e o **princípio da multiplicação**. O princípio da adição afirma que o número de elementos de uma união de n conjuntos finitos $A_1, \ldots, A_n$ disjuntos entre si ($A_i \cap A_j = \varnothing$ para $i \neq j$) é dado pela soma dos números de elementos em cada conjunto,

$$|A_1 \cup \cdots \cup A_n| = \sum_{i=1}^{n} |A_i| \,, \tag{1}$$

onde $|A_i|$ indica o número de elementos de A_i. Já o princípio da multiplicação estabelece que o número de elementos do produto cartesiano de n conjuntos finitos $A_1, \ldots, A_n$, disjuntos ou não, é dado pelo produto dos números de elementos em cada conjunto,

$$|A_1 \times \cdots \times A_n| = \prod_{i=1}^{n} |A_i| \,. \tag{2}$$

A partir desses dois princípios básicos é possível deduzir a maioria das fórmulas para o número de permutações e combinações de um conjunto de objetos (com ou sem repetições, em que a ordem dos objetos importa ou não) encontradas nos livros-texto de matemática para o ensino fundamental e médio (Hazzan, 2013; Morgado *et al.*, 2020).

Quando dois conjuntos A e B não são disjuntos, o princípio da soma não pode ser aplicado imediatamente, pois a soma $|A| + |B|$ conta os elementos que pertencem simultaneamente a ambos os conjuntos duas vezes. Nesse caso, o número de elementos em $A \cap B$ deve ser subtraído da soma uma vez e obtemos a expressão bem conhecida $|A \cup B| = |A| + |B| - |A \cap B|$. Esse caso simples fornece a primeira aplicação de um outro princípio fundamental de contagem, o **princípio da inclusão-exclusão**, normalmente abreviado por PIE tanto em português quanto em inglês. O PIE nos permite calcular o número de elementos na união (ou, equivalentemente, na interseção de complementos) de um número qualquer de subconjuntos de um conjunto finito, disjuntos entre si ou não, e constitui uma poderosa ferramenta matemática.

Neste trabalho, revisamos o PIE e discutimos sua aplicação ao cálculo de permanentes, um

assunto que normalmente não é abordado em cursos de licenciatura ou bacharelado em matemática. Alunos de ciência da computação ocasionalmente são introduzidos aos permanentes em disciplinas de matemática discreta e teoria da computação devido à sua relação com a teoria da complexidade computacional. A apresentação procura ser rigorosa, porém elementar e acessível a alunos dos anos iniciais de cursos de licenciatura ou bacharelado em matemática, ciências e engenharias. Na Seção 2 apresentamos o PIE em sua versão conjuntista, mais conhecida, e algébrica, juntamente com exemplos clássicos de sua aplicação: o crivo de Eratóstenes (Subseção 2.3) e a função totiente de Euler (Subeção 2.4). Na Seção 3 introduzimos os permanentes, formas multilineares totalmente simétricas semelhantes aos determinantes que possuem inúmeras aplicações em análise combinatória. Na Subseção 3.2 apresentamos o algoritmo de Ryser, um dos desenvolvimentos mais espetaculares na abordagem de problemas combinatoriais difíceis, e calculamos o permanente de uma matriz retangular usando esse algoritmo. Após uma breve exposição da relação entre o cálculo de permanentes com a teoria da complexidade computacional na Subseção 3.3, exibimos na Subseção 3.4 uma implementação do algoritmo de Ryser em Python usando códigos de Gray e a testamos em duas famílias de matrizes para as quais se conhece o valor exato do permanente. Na Seção 4 concluímos o artigo com algumas observações acerca do PIE e indicamos referências para seu estudo em nível mais avançado.

2 O princípio da inclusão-exclusão

2.1 A abordagem conjuntista

O princípio por trás do PIE para contar corretamente o número de elementos em $A_1 \cup \cdots \cup A_n$ consiste em primeiro superestimar grosseiramente o número de elementos na união pela inclusão indiscriminada de todos os elementos em cada conjunto, em seguida realizar uma correção simples da estimativa pela exclusão dos elementos que não deveriam ter sido incluídos, depois corrigir essa exclusão pela inclusão dos elementos que não deveriam ter sido excluídos e assim por diante, incluindo e excluindo alternadamente elementos até que na contagem final restem somente os elementos que deveriam ser contados uma única vez. Uma das características mais evidentes de resultados obtidos através do PIE é a presença de somas de termos com sinais alternados.

O teorema a seguir estabelece rigorosamente uma das formas elementares do PIE.

Teorema 2.1 (Princípio da inclusão-exclusão)**.** *Suponha que temos N objetos e que cada objeto pode possuir ou não uma ou mais das propriedades $a_1, \ldots, a_n$. Denotando o número de objetos com*

as k propriedades $a_{i_1}, \ldots, a_{i_k}$ por $N(a_{i_1}, \ldots, a_{i_k})$, $1 \leqslant k \leqslant n$, o número de objetos que não possuem qualquer das propriedades $a_1, \ldots, a_n$ é dado por

$$N - \sum_{i_1} N(a_{i_1}) + \sum_{i_1<i_2} N(a_{i_1}, a_{i_2}) - \cdots + (-1)^k \sum_{i_1<\cdots<i_k} N(a_{i_1}, \ldots, a_{i_k}) + \cdots + (-1)^n N(a_1, \ldots, a_n), \quad (3)$$

onde cada índice i_k nos somatórios assume todos os possíveis valores $1, \ldots, n$.

Prova. *Para demonstrar o PIE na forma acima, basta verificar quantas vezes um objeto qualquer é contado pela expressão (3). Se um objeto não possui qualquer das propriedades, ele é contado uma única vez no termo N em (3) e não contribui para os termos restantes. Um objeto que possui somente uma das propriedades, por sua vez, é contado uma vez em N e uma vez em $\sum_{i_1} N(a_{i_1})$ e, portanto, contribui com $1 - 1 = 0$ para a soma. Já um objeto que possui exatamente $m \geqslant 2$ propriedades é contado uma vez no termo N, m vezes no termo $\sum_{i_1} N(a_{i_1})$, $\binom{m}{2}$ vezes no termo $\sum_{i_1<i_2} N(a_{i_1}, a_{i_2})$ e assim sucessivamente, isto é, ele é contado $\binom{m}{k}$ vezes em cada termo $\sum_{i_1<\cdots<i_k} N(a_{i_1}, \ldots, a_{i_k})$, $0 \leqslant k \leqslant m$. A contribuição total desse objeto para a soma vale, portanto,*

$$1 - m + \binom{m}{2} - \cdots + (-1)^m \binom{m}{m} = \sum_{k=0}^{m} (-1)^k \binom{m}{k} = (-1+1)^m = 0, \quad (4)$$

pelo teorema binomial.[1] *Dessa forma, a expressão (3) conta somente os objetos que não possuem qualquer das propriedades $a_1, \ldots, a_n$, como queríamos demonstrar.* □

Observação 2.2. *Se truncarmos a expressão (3) após um termo negativo (respectivamente, positivo), obtemos uma cota inferior (respectivamente, superior) para o número de objetos que não possuem qualquer das propriedades $a_1, \ldots, a_n$. Isso significa que cada termo negativo da soma corrige um excesso de contagem e cada termo positivo corrige uma deficiência de contagem até aquele ponto.*

Vamos dar um exemplo de aplicação do PIE nessa forma.

Exemplo 2.3. *Seja uma coleção de 50 pedras semipreciosas das quais 25 são translúcidas (T), 30 são vermelhas (V), 20 são redondas (R), 18 são translúcidas e vermelhas (T, V), 12 são translúcidas e redondas (T, R), 15 são vermelhas e redondas (V, R) e 8 são translúcidas, vermelhas e redondas (T, V, R). Quantas pedras não são nem translúcidas, nem vermelhas, nem redondas? Uma aplicação*

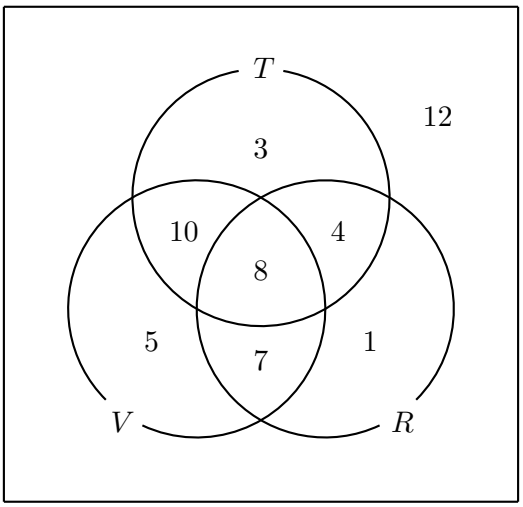
[1] O teorema binomial de Newton estabelece que $(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k}$, onde $\binom{n}{k} = n!/k!(n-k)!$ é o coeficiente binomial usual. Colocando $x = -1$ e $y = 1$ nesta expressão obtemos a identidade empregada na equação (4).

direta do PIE fornece

$$\begin{aligned} N_0 &= N - N(T) - N(V) - N(R) + N(T,V) + N(T,R) + N(V,R) - N(T,V,R) \\ &= 50 - 25 - 30 - 20 + 18 + 12 + 15 - 8 = 12. \end{aligned} \tag{5}$$

Outra maneira de resolver esse problema simples seria desenhar o diagrama de Venn dos conjuntos T, V e R e suas interseções, conforme a Figura 1. Esse dispositivo visual, no entanto, só é prático quando o número de subconjuntos envolvidos no cálculo é pequeno, no máximo 4 ou 5.

Figura 1: Solução do Exemplo 2.3 em termos de diagramas de Venn. Como a soma dos números que aparecem em cada subconjunto disjunto do diagrama vale 38, temos $50 - 38 = 12$ pedras que não pertencem a qualquer das categorias T, V ou R ou suas combinações.



Fonte: Elaboração do autor (2023).

Como o diagrama de Venn empregado no Exemplo 2.3 sugere, podemos interpretar o PIE em termos conjuntistas da seguinte forma. Suponha que $A_1, \ldots, A_r$ sejam subconjuntos de um conjunto Ω tais que $A_i = \{x \in \Omega \colon x \text{ possui a propriedade } a_i\}$. Quantos elementos não possuem qualquer das propriedades a_i? Esses são os elementos que estão em $\overline{A}_1 \cap \cdots \cap \overline{A}_r$. Seguindo a receita do PIE, para contá-los tomamos todos os elementos de Ω e subtraímos os elementos que estão em pelo menos um A_i, adicionamos aqueles que estão em pelo menos dois A_i, subtraímos aqueles que aparecem em pelo menos três A_i e assim por diante, e daí ficamos com

$$\left|\overline{A}_1 \cap \cdots \cap \overline{A}_r\right| = |\Omega| - \sum_{i_1} |A_{i_1}| + \sum_{i_1 < i_2} |A_{i_1} \cap A_{i_2}| - \cdots + (-1)^r |A_1 \cap \cdots \cap A_r|. \tag{6}$$

Observação 2.4. *A expressão acima pode ser escrita de maneira mais compacta como*

$$\left|\overline{A}_1 \cap \cdots \cap \overline{A}_r\right| = \sum_{J \subseteq [n]} (-1)^{|J|} \left|A_J\right|, \tag{7}$$

onde $[n]$ *é uma notação usual em matemática discreta para o conjunto* $\{1, \ldots, n\}$*,* $J \subseteq [n]$ *denota todos os* 2^n *subconjuntos de* $[n]$ *e* $A_J = \bigcap_{j \in J} A_j$*, com a convenção de que* $A_\varnothing = \Omega$*.*

A fórmula do PIE talvez mais conhecida, principalmente em contextos elementares, é aquela usada para calcular $|A_1 \cup \cdots \cup A_r|$, isto é, para determinar o número de objetos que possuem pelo menos uma das propriedades $a_1, \ldots, a_r$. Como, por definição de conjunto complementar, $\Omega = (A_1 \cup \cdots \cup A_r) \cup (\overline{A_1 \cup \cdots \cup A_r})$ e pela lei de DeMorgan $\overline{A_1 \cup \cdots \cup A_r} = \overline{A}_1 \cap \cdots \cap \overline{A}_r$, encontramos que $|A_1 \cup \cdots \cup A_r| = |\Omega| - \left|\overline{A}_1 \cap \cdots \cap \overline{A}_r\right|$ e daí, pela equação (6),

$$|A_1 \cup \cdots \cup A_r| = \sum_{i_1} |A_{i_1}| - \sum_{i_1 < i_2} |A_{i_1} \cap A_{i_2}| + \cdots + (-1)^{r-1} |A_1 \cap \cdots \cap A_r|, \tag{8}$$

que generaliza a expressão $|A \cup B| = |A| + |B| - |A \cap B|$ para um número arbitrário de conjuntos.

2.2 Uma forma simbólica para o PIE

Podemos obter a fórmula do PIE através de um método conhecido como **método simbólico**. Dados N objetos que podem possuir ou não uma ou mais das propriedades $a_1, \ldots, a_r$ e denotando o número de objetos com as $k \leqslant r$ propriedades $a_{i_1}, \ldots, a_{i_k}$ por $N(a_{i_1} \cdots a_{i_k})$, podemos denotar simbolicamente o número de objetos que não possuem qualquer das propriedades $a_1, \ldots, a_r$ por

$$N_0 = N(a_1' \cdots a_r') = N((1 - a_1) \cdots (1 - a_r)), \tag{9}$$

onde a_i significa "possui a propriedade i" e $a_i' = 1 - a_i$ significa "não possui a propriedade i". Expandindo o produto $(1 - a_1) \cdots (1 - a_r)$ encontramos

$$N_0 = N(a_1' \cdots a_r') = N(1 - a_1 - \cdots - a_r + a_1 a_2 + \cdots + a_{r-1} a_r - \cdots + (-1)^r a_1 \cdots a_r), \tag{10}$$

e atuando linearmente com o "operador contagem" N sobre a soma ficamos com

$$N_0 = N - N(a_1) - \cdots - N(a_r) + N(a_1 a_2) + \cdots + N(a_{r-1} a_r) - \cdots + (-1)^r N(a_1 \cdots a_r), \tag{11}$$

onde colocamos $N(1) = N$. Essa é exatamente a mesma expressão (3) que encontramos inicialmente para o PIE. Algumas dessas expressões, assim como diversas extensões e exemplos de aplicação, foram primeiramente obtidas nessa forma pelo matemático norte-americano Hassler Whitney (1907–1989) no início dos anos 1930 (Whitney, 1932). Riordan (2002) oferece uma exposição detalhada.

Uma das vantagens do método simbólico é que podemos considerar o número de objetos que possuem as propriedades $a_{i_1}, \cdots, a_{i_p}$ e não possuem as propriedades $a_{j_1}, \cdots, a_{j_q}$, com $p + q = r$, de maneira relativamente trivial: basta considerar

$$N(a_{i_1} \cdots a_{i_p} a'_{j_1} \cdots a'_{j_q}) = N(a_{i_1} \cdots a_{i_p}(1 - a_{j_1}) \cdots (1 - a_{j_q})). \tag{12}$$

Por exemplo, se queremos calcular o número de objetos com as propriedades a_1 e a_3 e sem as propriedades a_2 e a_4 calculamos

$$N(a_1 a'_2 a_3 a'_4) = N(a_1(1 - a_2)a_3(1 - a_4)) = N(a_1 a_3) - N(a_1 a_2 a_3) - N(a_1 a_3 a_4) + N(a_1 a_2 a_3 a_4). \tag{13}$$

Uma vez obtida a expressão desejada, calculamos os termos $N(a_{i_1} \cdots a_{i_p})$ como $\left| A_{i_1} \cap \cdots \cap A_{i_p} \right|$.

Outra vantagem do método simbólico é que ele permite obter uma grande variedade de funções geratrizes, que podem ser manipuladas formalmente (somadas, compostas, derivadas, expandidas em séries *etc.*) e terem o significado combinatorial de seus termos recuperado ao final (Riordan, 2002).

2.3 O crivo de Eratóstenes

Eratóstenes (276–194 a.C.) foi um dos bibliotecários da famosa Biblioteca de Alexandria e é lembrado principalmente pela sua medição da circunferência da Terra e pela invenção do crivo numérico que leva seu nome. O **crivo de Eratóstenes**, descrito pela primeira vez na obra do obscuro matemático grego Nicomedes (280–210 a.C.), consiste de um dispositivo prático para encontrar todos os números primos p no intervalo $2 \leqslant p \leqslant n$ pela eliminação recursiva de todos os números compostos no intervalo de tal forma que os números que sobrevivem ao crivo são todos primos. O crivo de Eratóstenes foi posteriormente aperfeiçoado de muitas maneiras diferentes, levando à introdução de diversas funções aritméticas – por exemplo, a **função totiente de Euler**, que exploramos a seguir – e a métodos sofisticados conhecidos como **métodos de crivo** amplamente empregados em teoria dos números e suas aplicações (Hefez, 2016; Schroeder, 2009; Tenenbaum; France, 2000).

O Teorema 2.5 estabelece a expressão matemática para o crivo de Eratóstenes; sua demonstração envolve a aplicação do PIE.

Teorema 2.5 (Crivo de Eratóstenes)**.** *O número $\pi(n)$ de números primos entre 2 e n é dado por*

$$\pi(n) = (n-1+k) - \sum_i \left\lfloor \frac{n}{p_i} \right\rfloor + \sum_{i<j} \left\lfloor \frac{n}{p_i p_j} \right\rfloor - \cdots + (-1)^k \left\lfloor \frac{n}{p_1 \cdots p_k} \right\rfloor, \tag{14}$$

onde $p_1, \ldots, p_k$ são os números primos entre 2 e $\sqrt{n}$ e $\lfloor x \rfloor$ denota o maior inteiro menor ou igual a x.

Para demonstrar o Teorema 2.5, vamos primeiro estabelecer um importante lema auxiliar.

Lema 2.6. *Para encontrar os números primos p no intervalo $2 \leqslant p \leqslant n$, basta eliminar todos os múltiplos dos números primos entre 2 e $\sqrt{n}$. Os números restantes no intervalo após a eliminação são todos primos.*

Prova (Lema 2.6)**.** *Se n é um número composto $n = pq$ com $2 \leqslant p \leqslant q$, então $p^2 \leqslant pq = n$, de onde segue que $p \leqslant \sqrt{n}$. Isso nos permite concluir que para encontrar todos os números primos $2 \leqslant p \leqslant n$ basta eliminar os múltiplos dos números primos entre 2 e $\sqrt{n}$, já que nenhum número maior que $\sqrt{n}$ e menor ou igual a n que sobra depois da eliminação possui fator primo menor ou igual a $\sqrt{n}$ tampouco pode ser produto de dois números maiores que $\sqrt{n}$.* □

Observação 2.7. *Os números primos obtidos pelo procedimento do Lema 2.6 não correspondem a todos os números primos no intervalo $2, \ldots, n$ porque o procedimento, exatamente como descrito, em princípio elimina também os números primos p no intervalo $2 \leqslant p \leqslant \sqrt{n}$. Essa observação é relevante na obtenção da fórmula (16) para o crivo de Eratóstenes.*

Na demonstração do Lema 2.6 aparecem números $n = pq$ com p e q primos. Números desse tipo, dados pelo produto de dois números primos muito grandes de aproximadamente mesma quantidade (atualmente, centenas) de dígitos, são empregados em criptografia porque são mais difíceis de fatorar, fornecendo maior segurança criptográfica (Hefez, 2016; Schroeder, 2009).

De posse do Lema 2.6 podemos demonstrar a fórmula para o crivo de Eratóstenes como uma simples aplicação do PIE.

Prova (Teorema 2.5)**.** *Sejam $p_1, \ldots, p_k$ os números primos entre 2 e $\sqrt{n}$ e seja $N(p_1 \cdots p_j) = \lfloor n/p_1 \cdots p_j \rfloor$ o número de múltiplos de $p_1 \cdots p_j$, $j = 1, \ldots, k$, entre 2 e n. Pelo Lema 2.6 e pelo PIE (3), o número de inteiros entre 2 e n que não são múltiplos de nenhum primo $p_1, \ldots, p_k$ vale*

$$(n-1) - \sum_i N(p_i) + \sum_{i<j} N(p_i p_j) - \cdots + (-1)^k N(p_1 \cdots p_k), \tag{15}$$

onde o termo $(n-1)$ corresponde à quantidade de números inteiros entre 2 e n. O número dado por (15), no entanto, ainda não corresponde à quantidade de números primos entre 2 e n, pois falta incluir

na contagem os k números primos $p_1, \ldots, p_k$ eles próprios. Assim, o número $\pi(n)$ de números primos entre 2 e n é dado, finalmente, por

$$\pi(n) = (n - 1 + k) - \sum_i \left\lfloor \frac{n}{p_i} \right\rfloor + \sum_{i<j} \left\lfloor \frac{n}{p_i p_j} \right\rfloor - \cdots + (-1)^k \left\lfloor \frac{n}{p_1 \cdots p_k} \right\rfloor, \tag{16}$$

que é a expressão matemática para o crivo de Eratóstenes. □

Exemplo 2.8. *Seja $n = 170$. Neste caso, os números primos entre 2 e $\sqrt{170}$ são 2, 3, 5, 7, 11 e 13. Para empregar a fórmula (16), podemos considerar somente os termos até $N(3, 5, 11)$, porque qualquer combinação dos primos 2, 3, 5, 7, 11 e 13 envolvendo números maiores que $p_i p_j p_k = 3 \cdot 5 \cdot 11$ ou mais de 3 fatores (por exemplo, $p_i p_j p_k p_\ell = 2 \cdot 3 \cdot 5 \cdot 7$) será maior que 170. O resto é simples aritmética: temos $N(2) = \lfloor 170/2 \rfloor = 85$, $N(3) = \lfloor 170/3 \rfloor = 56$, ..., $N(2, 3) = \lfloor 170/6 \rfloor = 28$ e assim por diante até $N(3, 5, 11) = \lfloor 170/165 \rfloor = 1$. Juntando tudo obtemos*

$$\begin{aligned}
\sum_i \left\lfloor \frac{n}{p_i} \right\rfloor &= 85 + 56 + 34 + 24 + 15 + 13 = 227, \\
\sum_{i<j} \left\lfloor \frac{n}{p_i p_j} \right\rfloor &= 28 + 17 + 12 + 7 + 6 + 11 + 8 + 5 + 4 + 4 + 3 + 2 + 2 + 1 + 1 = 111, \\
\sum_{i<j<k} \left\lfloor \frac{n}{p_i p_j p_k} \right\rfloor &= 5 + 4 + 2 + 2 + 2 + 1 + 1 + 1 + 1 + 1 = 20,
\end{aligned} \tag{17}$$

de onde concluímos, pelo crivo de Eratóstenes (16), que existem $\pi(170) = (170-1+6)-227+111-20 = 39$ números primos entre 2 e 170. O leitor pode querer verificar quais são esses primos.

2.4 A função totiente de Euler

A função totiente ou ϕ de Euler é uma importante função em teoria dos números que possui relação com o crivo de Eratóstenes. A função totiente fornece o número de inteiros positivos $1 \leqslant k \leqslant n$ tais que k e n são primos entre si, isto é, tais que $\mathrm{mdc}(k, n) = 1$. Por convenção tomamos $\phi(1) = 1$, de modo que $\phi(n) \geqslant 1$ para todo $n \geqslant 1$. Assim, $\phi(1) = \phi(2) = 1$, $\phi(3) = \phi(4) = 2$, $\phi(5) = 4$ e $\phi(6) = 2$. Obviamente, $\phi(p) = p - 1$ para todo número primo p, já que, por definição de número primo, $\mathrm{mdc}(k, p) = 1$ para todo $1 \leqslant k \leqslant p - 1$. Pela definição da função totiente, $n - \phi(n)$ conta o número de inteiros positivos menores ou iguais a n que possuem pelo menos um fator primo em comum com n.

O Teorema 2.9 estabelece a expressão matemática para a função totiente de Euler. Sua demonstração emprega o PIE na forma (6).

Teorema 2.9 (Função totiente de Euler)**.** *A função totiente de Euler, que fornece o número de inteiros*

positivos $1 \leqslant k \leqslant n$ tais que k e n são primos entre si, isto é, tais que $\mathrm{mdc}(k,n) = 1$, é dada por

$$\phi(n) = n - \sum_i \frac{n}{p_i} + \sum_{i<j} \frac{n}{p_i p_j} - \cdots + (-1)^k \frac{n}{p_1 \cdots p_k} = n \prod_{i=1}^{k} \Big(1 - \frac{1}{p_i}\Big), \tag{18}$$

onde os números $p_1, \ldots, p_k$ são os fatores primos distintos de n.

Prova. *Lembremos inicialmente que pelo teorema fundamental da aritmética todo número inteiro $n \geqslant 2$ pode ser escrito como um produto único $n = p_1^{a_1} \cdots p_k^{a_k}$, com $2 \leqslant p_1 < \cdots < p_k \leqslant n$ números primos e $a_i \geqslant 1$ expoentes inteiros positivos. Sejam agora os conjuntos $A_p = \{k \colon k \equiv 0 \bmod p,\ 1 \leqslant k \leqslant n\}$, onde a notação $k \equiv 0 \bmod p$ se lê "k é congruente a 0 módulo p" e indica que a divisão inteira k/p possui resto zero. Em outras palavras, A_p é o conjunto dos números $1 \leqslant k \leqslant n$ que são divisíveis pelo número primo p. Em termos dos conjuntos A_p, o número de inteiros positivos $1 \leqslant k \leqslant n$ tais que k e n são primos entre si é dado por $\phi(n) = \big|\overline{A}_{p_1} \cap \cdots \cap \overline{A}_{p_k}\big|$, expressão que pelo PIE, equação (6), pode ser escrita como*

$$\phi(n) = \big|\overline{A}_{p_1} \cap \cdots \cap \overline{A}_{p_k}\big| = n - \sum_i |A_{p_i}| + \sum_{i<j} \big|A_{p_i} \cap A_{p_j}\big| - \cdots + (-1)^k \left|A_{p_1} \cap \cdots \cap A_{p_k}\right|. \tag{19}$$

Para calcular os termos $|A_{p_i}|$, repare que existem n/p_i números divisíveis por p_i entre 1 e n, de forma que $|A_{p_i}| = n/p_i$; aqui a divisão é exata, pois n é múltiplo de p_i. Da mesma forma, existem $\big|A_{p_i} \cap A_{p_j}\big| = n/p_i p_j$ números entre 1 e n que são simultaneamente divisíveis por p_i e p_j, e assim por diante. Inserindo esses valores na fórmula (19) encontramos finalmente que

$$\phi(n) = n - \sum_i \frac{n}{p_i} + \sum_{i<j} \frac{n}{p_i p_j} - \cdots + (-1)^k \frac{n}{p_1 \cdots p_k} = n \prod_{i=1}^{k} \Big(1 - \frac{1}{p_i}\Big) = n \prod_{p \mid n} \Big(1 - \frac{1}{p}\Big), \tag{20}$$

onde, na última expressão, $p \mid n$ se lê "p divide n" e entende-se tacitamente que os p são primos. □

Existe outra maneira de chegar à expressão (20) para $\phi(n)$. A função totiente de Euler é uma **função multiplicativa**, o que em teoria dos números significa que $\phi(mn) = \phi(m)\phi(n)$ para quaisquer dois inteiros m e n primos entre si. Assim, dado um número $n = p_1^{a_1} \cdots p_k^{a_k}$, temos que

$$\phi(n) = \phi(p_1^{a_1} \cdots p_k^{a_k}) = \prod_{i=1}^{k} \phi(p_i^{a_i}) = \prod_{i=1}^{k} (p_i^{a_i} - p_i^{a_i - 1}) = \prod_{i=1}^{k} p_i^{a_i} \Big(1 - \frac{1}{p_i}\Big) = n \prod_{i=1}^{k} \Big(1 - \frac{1}{p_i}\Big), \tag{21}$$

onde empregamos a identidade $\phi(p^a) = p^a - p^{a-1}$ válida para todo número primo p, pois $\mathrm{mdc}(k, p^a) = 1$ se e somente se k não é múltiplo de p e existem p^{a-1} múltiplos de p no intervalo $1 \leqslant k \leqslant p^a$.

A relação próxima que existe entre o crivo de Eratóstenes e a função totiente de Euler pode

ser percebida informalmente da seguinte forma. Ao aplicar o crivo de Eratóstenes, inicialmente eliminamos da lista de inteiros todos os múltiplos de 2, restando após a remoção desses múltiplos aproximadamente $\frac{1}{2}n$ números, já que cerca de $1/2$ dos inteiros até n são divisíveis por 2. A seguir, removemos da lista os múltiplos de 3, e como cerca de $1/3$ dos inteiros ímpares restantes na lista são divisíveis por 3, ficamos com cerca de $\frac{2}{3} \cdot \frac{1}{2}\, n$ números. Pelo mesmo raciocínio, a remoção dos múltiplos de p elimina cerca de $1/p$ dos inteiros da lista, deixando passar pelo crivo cerca de $1 - 1/p$ dos números restantes. Podemos esperar, portanto, que o número de inteiros até n que não foram peneirados no crivo de Eratóstenes pelos números primos até $\sqrt{n}$, isto é, o número $\pi(n)$ de números primos menores ou iguais a n, valha aproximadamente o inteiro mais próximo de

$$\pi(n) \approx n \prod_{p \leqslant \sqrt{n}} \Big(1 - \frac{1}{p}\Big). \tag{22}$$

Esse resultado não é equivalente ao crivo de Eratóstenes, pois as frações $1/p$ eliminadas em cada etapa no procedimento descrito acima não são exatas. Por exemplo, vimos no Exemplo (2.8) que $\pi(170) = 39$, enquanto a aproximação (22) fornece $\pi(170) \approx 33$. O cálculo do valor correto de $\pi(n)$ constitui um dos maiores desafios da teoria dos números e, de fato, motivou grande parte dos desenvolvimentos na área, com alcance por toda a matemática pura e aplicada. Ao leitor interessado no assunto, que envolve propriedades analíticas profundas da função zeta de Riemann, recomendamos o fascinante livro de Schroeder (2009) e Tenenbaum e France (2000).

Concluímos esta seção com uma observação acerca do nome da função totiente de Euler.

Observação 2.10. *O nome aparentemente bizarro de "função totiente de Euler" para a função $\phi(n)$ foi cunhado pelo matemático inglês James Joseph Sylvester (1814–1897) para indicar que a função $\phi(n)$ contava o número total de divisores de n. Sylvester fez uma conexão entre as palavras latinas quotiens (quantos) e totiens (tantos), e assim como quotiens entrou para a língua inglesa como quotient, totiens acabou resultando, em inglês, na palavra totient, forma que se espalhou para outras línguas. Em português, podemos pensar que a palavra totiente transmite a noção de "tantos quantos".*

3 PIE e o cálculo de permanentes

3.1 Permanentes

Mesmo entre pós-graduandos e pesquisadores, é comum encontrar pessoas que desconhecem completamente a definição, as propriedades e as aplicações dos permanentes. Apesar de muito menos conhecido que o determinante, o permanente de uma matriz possui inúmeras aplicações em

análise combinatória, teoria dos grafos, teoria da computação, probabilidade e estatística e, portanto, também nas áreas de aplicação dessas disciplinas. A partir dos anos 1950, aproximadamente, permanentes se tornaram ferramentas matemáticas importantes no estudo de processos estocásticos espaciais, modelos bidimensionais sobre grafos planares (por exemplo, na caracterização do recobrimento de um reticulado por monômeros, dímeros ou poliominos mais complicados) e mecânica quântica (por exemplo, na representação de sistemas de muitos bósons) (Reale, 2018). Permanentes e algumas de suas principais aplicações são discutidos em Barvinok (2016), Brualdi e Ryser (1991), Minc (1978) (uma referência enciclopédica), Ryser (1963) e van Lint e Wilson (2001).

O permanente de uma matriz quadrada $A = (a_{ij})$ de ordem n é dado por (Minc, 1978)

$$\operatorname{per}(A) = \sum_{\sigma \in S} \prod_{i=1}^{n} a_{i\sigma(i)}, \tag{23}$$

onde a soma se estende sobre todas as $n!$ permutações (bijeções) $\sigma \colon [n] \to [n]$. Vemos que a fórmula para o permanente de A é semelhante à fórmula para o determinante de A,

$$\det(A) = \sum_{\sigma \in S} (-1)^{\sigma} \prod_{i=1}^{n} a_{i\sigma(i)}, \tag{24}$$

sem os sinais $(-1)^{\sigma}$ oriundos da paridade da permutação σ.

Observação 3.1 (Paridade de uma permutação)**.** *A paridade $(-1)^{\sigma}$ de uma permutação σ, frequentemente denotada também por $\operatorname{sgn}(\sigma)$ ou $\varepsilon(\sigma)$, é dada pela paridade de seu* ***número de inversões*** *$\sigma(i) > \sigma(j)$ com $i < j$. Por exemplo, a permutação $\sigma = 2314$ possui paridade $(-1)^{\sigma} = 1$, pois σ possui duas inversões, $\sigma(1) > \sigma(3)$ e $\sigma(2) > \sigma(3)$; já a permutação $\sigma = 3142$ possui três inversões e sua paridade vale $(-1)^{\sigma} = -1$. Permutações de paridade positiva $(-1)^{\sigma} = 1$ são denominadas pares, enquanto permutações de paridade negativa $(-1)^{\sigma} = -1$ são denominadas ímpares.*

Em comum, tanto determinantes quanto permanentes podem ser calculados através do **desenvolvimento de Laplace**, que permite expressar o determinante (ou o permanente) de uma matriz A em termo dos determinantes (ou permanentes) de submatrizes de A. O desenvolvimento de Laplace para o determinante de uma matriz A de ordem n é dado por

$$\det(A) = \sum_{i=1}^{n} a_{ij} (-1)^{i+j} \det(A_{ij}) = \sum_{j=1}^{n} a_{ij} (-1)^{i+j} \det(A_{ij}), \tag{25}$$

onde A_{ij} é a submatriz de ordem $n-1$ que se obtém excluindo a i-ésima linha e a j-ésima coluna de A. O determinante de A_{ij} é chamado de **menor do elemento a_{ij}** e o produto $(-1)^{i+j} \det(A_{ij})$

é denominado **cofator de a_{ij}**. Repare que no primeiro somatório da equação (25) fixamos uma coluna j qualquer para desenvolver o determinante "pelas linhas", enquanto no segundo somatório fixamos uma linha i qualquer para desenvolver o determinante "pelas colunas". Analogamente, o desenvolvimento de Laplace para o permanente de uma matriz A assume a forma mais simples

$$\operatorname{per}(A) = \sum_{i=1}^{n} a_{ij} \operatorname{per}(A_{ij}) = \sum_{j=1}^{n} a_{ij} \operatorname{per}(A_{ij}). \tag{26}$$

O desenvolvimento de Laplace para o cálculo de determinantes e permanentes permite explorar a estrutura dos zeros e as simetrias das matrizes para reduzir a complexidade dos cálculos (Brualdi; Ryser, 1991; Minc, 1978; Reale, 2018).

Por outro lado, a diferença aparentemente superficial envolvendo "somente alguns sinais" entre o permanente e o determinante possui enormes consequências. Diferentemente do determinante, que possui interpretação geométrica como um volume em n dimensões ou como um fator de escala associado a matrizes de transformações lineares, o permanente não possui interpretação geométrica imediata, embora ocorra associado a problemas envolvendo o volume de poliedros convexos (Barvinok, 2016). Além disso, a propriedade multiplicativa $\det(AB) = \det(A)\det(B)$ não vale para permanentes. Enquanto $\det(A)$ segue mudando de sinal a cada transposição de linhas ou colunas de A, o permanente não se altera pela transposições de linhas ou colunas devido à ausência do fator $(-1)^{\sigma}$ em sua definição. O permanente de A também não se anula necessariamente se duas linhas forem iguais ou se qualquer subconjunto de linhas for linearmente dependente. Assim, a adição de um múltiplo de uma linha de A a outra não deixa $\operatorname{per}(A)$ invariante, de forma que enquanto o determinante de uma matriz pode ser calculado por eliminação gaussiana em $O(n^3)$ operações, o cálculo do permanente em princípio é uma operação de ordem $O(n \cdot n!)$, uma diferença brutal.

Observação 3.2 (Notação O)**.** *A notação $f(x) = O(g(x))$ envolvendo duas funções f e g significa que a função f certamente não cresce a uma taxa maior que g, isto é, existe uma constante C e um número x_0 tal que para todo $x > x_0$ podemos afirmar que $|f(x)| < Cg(x)$. A notação implica que f pode crescer na mesma proporção que g ou mais lentamente; ambas as possibilidades são válidas. Na análise de algoritmos, quantificamos a eficiência de um algoritmo através da maneira como o seu número de operações básicas – na maior parte dos casos, adição e multiplicação de dígitos – aumenta à medida que aumentamos o tamanho da entrada. Dizer que um algoritmo é de ordem $O(n^3)$ significa que se dobrarmos o tamanho n do problema (por exemplo, a ordem de uma matriz ou o número de vértices de um grafo), o número de operações necessárias para resolvê-lo (proporcional ao tempo de execução em uma máquina sequencial) cresce oito vezes.*

Diversos problemas envolvem permanentes de matrizes cujos elementos assumem somente os valores 0 ou 1; um caso simples aparece no Exemplo 3.3. Um problema computacional de grande relevância tanto teórica quanto prática desse tipo é o do número de **emparelhamentos perfeitos** (***perfect matchings***) de um grafo bipartido G. Um emparelhamento (*matching*) em G é um conjunto de arestas mutuamente disjuntas, isto é, que não incidem sobre os mesmos vértices. Pode-se mostrar que o número de emparelhamentos perfeitos em um grafo bipartido G qualquer com um número par de vértices é dado por $\mathrm{per}(A_G)$, onde A_G é a matriz de adjacência de G. Para conceitos em teoria dos grafos especificamente relacionados a esse assunto, veja Brualdi e Ryser (1991). No problema dos emparelhamentos perfeitos em grafos bipartidos surge uma relação interessante entre o determinante e o permanente: se $\det(A_G) \neq 0$ então existem $\mathrm{per}(A_G)$ emparelhamentos perfeitos em G.

Vamos examinar um exemplo simples de enumeração combinatória que possui solução dada pelo permanente de uma matriz 0-1: o número de **sistemas de representantes distintos** (**SRD**) de uma coleção de conjuntos. SRD podem ser usados para construir quadrados latinos, que são matrizes $n \times n$ com n elementos distintos aparecendo exatamente uma vez em cada linha e coluna da matriz (Brualdi; Ryser, 1991; Ryser, 1963; van Lint; Wilson, 2001). SRD e quadrados latinos são úteis, por exemplo, no *design* de experimentos e em códigos corretores de erro.

Exemplo 3.3 (Sistema de representantes distintos)**.** *Seja $A_1, \ldots, A_n$ uma coleção de n conjuntos não necessariamente disjuntos ou distintos entre si cuja união $\Omega = A_1 \cup \cdots \cup A_n$ consiste dos n elementos $a_1, \ldots, a_n$. Por um sistema de representantes distintos (SRD) da coleção $A_1, \ldots, A_n$ entendemos uma lista $r_1, \ldots, r_n$ de elementos de Ω distintos entre si escolhidos de tal forma que $r_i \in A_i$ para todo $i = 1, \ldots, n$. Cada elemento r_i é um representante do conjunto A_i no SRD. Quantos SRD da coleção $A_1, \ldots, A_n$ existem? Podemos responder essa a questão introduzindo a matriz $A = (a_{ij})$ de ordem n com elementos $a_{ij} = 1$ se $a_i \in A_j$ e $a_{ij} = 0$ em caso contrário. Como permutações são bijeções, isto é, não existe $\sigma(i) = \sigma(j)$ com $i \neq j$, cada produto $a_{1\sigma(1)}a_{2\sigma(2)} \cdots a_{n\sigma(n)} = 1$ corresponde a um SRD possível. Dessa forma, o número total de SRD da coleção $A_1, \ldots, A_n$ que podem ser formados é dado pela soma de $a_{1\sigma(1)}a_{2\sigma(2)} \cdots a_{n\sigma(n)}$ sobre todas as permutações possíveis dos índices $1, \ldots, n$,*

$$|\mathit{SRD}| = \sum_{\sigma \in S} a_{1\sigma(1)}a_{2\sigma(2)} \cdots a_{n\sigma(n)} = \sum_{\sigma \in S} \prod_{i=1}^{n} a_{i\sigma(i)}, \tag{27}$$

*isto é, pelo permanente de A. O **teorema de Hall** afirma que a coleção de conjuntos $A_1, \ldots, A_n$ possui um SRD se e somente se para cada $1 \leqslant k \leqslant n$ a união $A_{i_1} \cup \cdots \cup A_{i_k}$ de quaisquer k conjuntos da coleção tem cardinalidade pelo menos k (van Lint; Wilson, 2001).*

Ao contrário do determinante, o permanente é bem definido para uma matriz retangular $A_{m \times n}$,

$m < n$, por uma generalização de sua definição (23): ao invés de tomar a soma sobre todas as permutações de $\{1, \ldots, n\}$, basta tomar a soma sobre $\sigma \in \mathsf{Inj}(R, C)$, o conjunto de todas as aplicações injetoras $\sigma\colon R \to C$, onde $R = \{1, \ldots, m\}$ é o índice das linhas e $C = \{1, \ldots, n\}$ é o índice das colunas. O permanente de A, neste caso, é definido pela expressão

$$\operatorname{per}(A) = \sum_{\sigma \in \mathsf{Inj}(R,C)} \prod_{i=1}^{m} a_{i\sigma(i)}. \tag{28}$$

O número $|\mathsf{Inj}(R, C)|$ de aplicações injetoras $R \to C$ corresponde ao número de m-permutações de n objetos, normalmente denotado na literatura didática por $A_{n,m}$ (Hazzan, 2013). Assim,

$$|\mathsf{Inj}(R, C)| = n(n-1)\cdots(n-(m-1)) = \frac{n!}{(n-m)!}. \tag{29}$$

Se $m = n$, recuperamos as $n!$ aplicações bijetoras. Se $m > n$, $|\mathsf{Inj}(R, C)| = 0$, pois não é possível mapear m elementos distintos de R para n elementos distintos de C sem repetição. No entanto, como $\operatorname{per}(A) = \operatorname{per}(A^T)$, essa não é uma dificuldade na definição do permanente de uma matriz retangular, e podemos sempre considerar que $m \leqslant n$ sem perda de generalidade.

3.2 A fórmula de Ryser

Uma das aplicações mais impressionantes do PIE foi dada por Herbert John Ryser (1923–1985) de maneira despretensiosa em um conjunto de notas de aula publicadas no volume 14 da *The Carus Mathematical Monographs* (Ryser, 1963). Nessas notas, usando o PIE Ryser estabeleceu uma fórmula para o cálculo exato do permanente de uma matriz que era mais eficiente do que todos os outros métodos conhecidos até então e, exceto por algumas melhorias incrementais, até hoje. A derivação da fórmula de Ryser considera uma generalização do PIE que atribui pesos aos subconjuntos, e embora seja possível compreendê-la sem maiores dificuldades, ela não será dada aqui devido à quantidade de notação e definições que precisaria ser introduzida. O leitor interessado em sua derivação deve consultar Brualdi e Ryser (1991), Ryser (1963) e van Lint e Wilson (2001).

Dada uma matriz $A_{m\times n}$, $m \leqslant n$, a fórmula de Ryser para o permanente de A é dada por

$$\operatorname{per}(A) = \sum_{k=1}^{m} (-1)^{m-k} \binom{n-k}{n-m} \sum_{\substack{J \subseteq C \\ |J|=k}} P(A_J), \tag{30}$$

onde $P(A_J)$ é o produto das somas dos elementos das linhas da matriz A_J formada pelas colunas de

A com índices em J,

$$P(A_J) = \prod_{i=1}^{m} \Big(\sum_{j \in J} a_{ij} \Big). \tag{31}$$

Se a matriz A for quadrada ($m = n$), o coeficiente $\binom{n-k}{n-m} = \binom{n-k}{0} = 1$ e, após fatorarmos o sinal $(-1)^{n-k}$ como $(-1)^n(-1)^k$, a equação (30) se torna

$$\operatorname{per}(A) = (-1)^n \sum_{k=1}^{n} (-1)^k \sum_{\substack{J \subseteq C \\ |J|=k}} P(A_J). \tag{32}$$

Nesta forma, o método de Ryser requer $O(2^{n-1}n^2)$ operações para o cálculo do permanente.

Exemplo 3.4. *Vamos calcular detalhadamente o permanente da matriz retangular*

$$A = \begin{pmatrix} 1 & -1 & 4 & 0 \\ 5 & 0 & 1 & -2 \end{pmatrix} \tag{33}$$

pela fórmula de Ryser. Essa matriz é do tipo $m \times n = 2 \times 4$, de maneira que o conjunto dos índices de linha $R = \{1, 2\}$ e o conjunto dos índices de coluna $C = \{1, 2, 3, 4\}$. A soma sobre as linhas em (30) possui, portanto, dois termos, $k = 1$ e $k = 2$. Quando $k = 1$, os subconjuntos $J \subseteq C$ com $|J| = 1$ são $J = \{1\}$, $\{2\}$, $\{3\}$ e $\{4\}$. Por exemplo, quando $J = \{1\}$, temos $A_J = \binom{1}{5}$ e a equação (31) fornece

$$P(A_J) = \prod_{i=1}^{2} \sum_{j \in \{1\}} a_{ij} = \prod_{i=1}^{2} a_{i1} = a_{11} \cdot a_{21} = 1 \cdot 5 = 5. \tag{34}$$

Procedendo da mesma forma, encontramos que os produtos das somas dos elementos das linhas da matrizes formadas pelas colunas de A com índices em $J \subseteq C$ com $|J| = 1$ valem, respectivamente, 5, 0, 4 e 0. Assim, pela equação (30) a contribuição desses termos para o permanente vale

$$(-1)^{m-k} \binom{n-k}{n-m} \sum_{\substack{J \subseteq C \\ |J|=k}} P(A_J) = (-1)^{2-1} \binom{4-1}{4-2} (5 + 0 + 4 + 0) = -27. \tag{35}$$

Quando $k = 2$, os subconjuntos $J \subseteq C$ com $|J| = 2$ são $J = \{1, 2\}$, $\{1, 3\}$, $\{1, 4\}$, $\{2, 3\}$, $\{2, 4\}$ e $\{3, 4\}$. Por exemplo, quando $J = \{1, 3\}$, temos $A_J = \left(\begin{smallmatrix} 1 & 4 \\ 5 & 1 \end{smallmatrix}\right)$ e a equação (31) fornece

$$P(A_J) = \prod_{i=1}^{2} \sum_{j \in \{1,3\}} a_{ij} = \prod_{i=1}^{2} (a_{i1} + a_{i3}) = (a_{11} + a_{13}) \cdot (a_{21} + a_{23}) = (1 + 4) \cdot (5 + 1) = 30. \tag{36}$$

Procedendo da mesma forma encontramos que os produtos das somas dos elementos das linhas da

matrizes formadas pelas colunas de A com índices em $J \subseteq C$ com $|J| = 2$ valem, respectivamente, 0*,* 30*,* 3*,* 3*,* 2 *e* -4*. Assim, a contribuição desses termos para o permanente vale*

$$(-1)^{m-k}\binom{n-k}{n-m}\sum_{\substack{J\subseteq C\\|J|=k}} P(A_J) = (-1)^{2-2}\binom{4-2}{4-2}(0+30+3+3+2-4) = 34. \tag{37}$$

A partir dessas somas parciais encontramos, finalmente, que $\operatorname{per}(A) = -27 + 34 = 7$*.*

3.3 Complexidade algorítmica dos permanentes

Na teoria da **complexidade computacional**, os problemas são classificados conforme sua dificuldade aumenta com o tamanho da entrada. Uma solução eficiente significa que o problema pode ser resolvido em tempo polinomial ou menos. Por exemplo, o cálculo do determinante de uma matriz de ordem n pode ser realizado eficientemente em $O(n^3)$ operações por eliminação gaussiana.

Nossa experiência cotidiana mostra que é mais difícil resolver um problema matemático do que verificar se determinada solução proposta é válida. Existe um universo, real ou platônico, no qual resolver qualquer problema não é significativamente mais difícil do que verificar uma solução para ele? A existência de um tal universo significaria que **P** $=$ **NP**, onde **P** representa o conjunto de problemas que podem ser solucionados de forma eficiente em uma máquina de Turing determinística e **NP** representa o conjunto de problemas para os quais soluções podem ser verificadas de forma eficiente no mesmo tipo de máquina (Arora; Barak, 2009; Goldreich, 2008).

Determinar se **P** $=$ **NP** ou não constitui um dos problemas em aberto mais importantes na ciência da computação. A grande maioria dos especialistas acredita que **P** $\neq$ **NP**, já que após décadas de esforços, ninguém foi capaz de encontrar um algoritmo de tempo polinomial para qualquer problema **NP-completo**. Problemas **NP**-completos formam uma classe de problemas em **NP** equivalentes entre si considerados os mais difíceis. Vários **teoremas de hierarquia** sugerem que existem limitações intrínsecas à capacidade de algoritmos em diferentes classes, tornando ainda mais improvável que **P** $=$ **NP**. Uma demonstração rigorosa desse fato, no entanto, ainda não existe.

Em 1979, Valiant (1979a; 1979b) mostrou que permanentes ocupam um lugar especial na teoria da complexidade computacional ao provar que o cálculo do permanente de matrizes 0-1 é um problema de uma classe de complexidade algorítmica própria que ele denominou **#P-completo** (lê-se "*number* P completo" ou "*sharp* P completo"), o equivalente combinatorial a um problema de decisão **NP**-completo. Na verdade, problemas **#P**-completos são ainda mais difíceis de atacar que problemas **NP**-completos: enquanto muitos problemas **NP**-completos são fáceis de decidir em

instâncias aleatórias, esse não parece ser o caso para problemas de enumeração combinatória **#P**-completos. Um algoritmo para resolver um problema **#P**-completo em tempo polinomial, caso existisse, implicaria que **P** $=$ **NP** (Arora; Barak, 2009; Barvinok, 2016; Goldreich, 2008; Valiant, 1979a; Valiant, 1979b).

Pela sua definição (23), a complexidade algorítmica do cálculo do permanente é de ordem $O(n \cdot n!)$. Em 1963, no entanto, Ryser propôs um algoritmo baseado no PIE que reduziu a complexidade desse cálculo para $O(2^{n-1}n^2)$, uma redução teórica e prática espetacular (Ryser, 1963). Melhorias incrementais posteriores reduziram a complexidade do algoritmo de Ryser para matrizes quadradas a $O(2^{n-1}n)$ (Nijenhuis, Wilf, 1978). Para $n = 20$, por exemplo, $O(n \cdot n!) \sim 10^{20}$, enquanto $O(2^{n-1}n) \sim 10^7$, um número 10 trilhões de vezes menor. Para matrizes 0-1, métodos ainda mais eficientes foram desenvolvidos empregando operações lógicas e aritmética binária (Kalmann, 1982). A complexidade algorítmica do cálculo do permanente, porém, continua sendo exponencial. O mérito do algoritmo de Ryser e suas variantes foi ter reduzido a fórmula (23) aos seus elementos computacionais essenciais, eliminando redundâncias e revelando sua verdadeira ordem de complexidade algorítmica.

3.4 O algoritmo de Ryser em Python

O Programa R implementa o algoritmo de Ryser para o cálculo do permanente de uma matriz quadrada $n \times n$ segundo a fórmula (32) na linguagem de programação Python 3.9.14 em um MacBook Pro com processador M1 Pro rodando sob o sistema operacional MacOS Sonoma 14.1.1.

Nossa implementação faz uso de **códigos de Gray** para indexar os subconjuntos $J \subseteq [\,n\,]$ de colunas da matriz. Códigos de Gray são enumerações binárias de objetos nas quais números adjacentes diferem em apenas um bit e possuem propriedades teóricas e aplicadas de enorme utilidade. Por exemplo, uma ordenação de Gray dos números binários de 0 a 7 é $\texttt{000} \prec \texttt{001} \prec \texttt{011} \prec \texttt{010} \prec \texttt{110} \prec \texttt{111} \prec \texttt{101} \prec \texttt{100}$; essa ordenação não é única. Como podemos identificar cada subconjunto $J \subseteq [\,n\,]$ com o número binário $b_1 + 2b_2 + \cdots + 2^{n-1}b_n$, onde $b_j = 1$ se $j \in J$ e $b_j = 0$ se $j \notin J$, $j = 1, \ldots, n$, o emprego de códigos de Gray em princípio permite selecionar subconjuntos de colunas de maneira a minimizar as mudanças entre seleções consecutivas. Essa propriedade está na raiz da otimização do algoritmo de Ryser sugerida por Nijenhuis e Wilf (1978). Enquanto nossa implementação do algoritmo de Ryser é de ordem $O(2^{n-1}n^2)$, a versão de Nijenhuis e Wilf é de ordem $O(2^{n-1}n)$. Também usamos aritmética binária em alguns lugares; por exemplo, escrevemos 2^n como `1<<n`, que corresponde a n deslocamentos à esquerda dos bits de $\texttt{1} \equiv \texttt{0}\cdots\texttt{01}$.

Testamos nossa implementação do algoritmo de Ryser em duas classes de matrizes para as quais se conhece o permanente: as **matrizes de desarranjo** $\boldsymbol{D_n}$, com elementos $d_{ij} = 1 - \delta_{ij}$, e as

matrizes tridiagonais T_n, com elementos $t_{ij} = \delta_{i,j+1} + \delta_{ij} + \delta_{i,j-1}$, onde o delta de Kronecker vale $\delta_{ij} = 1$ quando $i = j$ e $\delta_{ij} = 0$ quando $i \neq j$. Explicitamente, temos

$$D_n = \begin{pmatrix} 0 & 1 & 1 & 1 & \cdots & 1 \\ 1 & 0 & 1 & 1 & \cdots & 1 \\ 1 & 1 & 0 & 1 & \cdots & 1 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 1 & \cdots & 1 & 1 & 0 & 1 \\ 1 & \cdots & 1 & 1 & 1 & 0 \end{pmatrix}, \qquad T_n = \begin{pmatrix} 1 & 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 1 & 1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & 1 & 1 \\ 0 & \cdots & 0 & 0 & 1 & 1 \end{pmatrix}. \tag{38}$$

O permanente das matrizes D_n vale $\operatorname{per}(D_n) = !n$, o número de desarranjos de n objetos, isto é, o número de n-permutações $\sigma(1)\ldots\sigma(n)$ sem nenhum ponto fixo $\sigma(i) = i$ (por exemplo, as permutações 35124 e 41532 são desarranjos), enquanto o permanente das matrizes T_n vale $\operatorname{per}(T_n) = F_{n+1}$, o $(n+1)$-ésimo número de Fibonacci, $n \geqslant 0$. O cálculo desses permanentes aparece em Minc (1978).

Observação 3.5. *O número de desarranjos de n objetos, denominado **subfatorial de n** e denotado por $!n$, pode ser calculado pelo PIE. Seja $N(i_1, \ldots, i_k)$ o número de permutações de n objetos que fixam as posições $\sigma(i_1) = i_1$, ..., $\sigma(i_k) = i_k$. Temos $N(i_1, \ldots, i_k) = (n-k)!$, já que ao fixarmos k índices restam ainda $n-k$ índices que podem ser permutados à vontade. Como existem $\binom{n}{k}$ maneiras diferentes de escolher os índices i_1, ..., i_k em $N(i_1, \ldots, i_k)$, o número de permutações que não fixam nenhum índice, isto é, o número $!n$ de desarranjos, é dado pelo PIE por*

$$!n = n! - \binom{n}{1}(n-1)! + \cdots + (-1)^n \binom{n}{n} 0! = \sum_{k=0}^{n} (-1)^k \binom{n}{k} (n-k)! = n! \sum_{k=0}^{n} \frac{(-1)^k}{k!}. \tag{39}$$

Vemos dessa expressão que para $n \gg 1$ cerca de $!n/n! \simeq e^{-1} \simeq 37\%$ de todas as permutações são desarranjos. Os primeiros valores de $!n$ para $n \geqslant 0$ são $1, 0, 1, 2, 9, 44, 265$ etc. Para $n \geqslant 1$ vale a aproximação $!n = \lfloor (n! + 1)/e \rfloor$. Já os números de Fibonacci são definidos pela relação de recorrência $F_0 = 0$, $F_1 = 1$ e $F_{n+1} = F_n + F_{n-1}$ para $n \geqslant 1$, formando a sequência $0, 1, 1, 2, 3, 5, 8, 13$ etc.

Nossos testes mostraram que o programa calcula corretamente os permanentes de D_n e T_n para todos os valores de n executados. Não é possível, no entanto, usar valores de n muito grandes por causa do tempo de execução. Por exemplo, o Programa R calcula o valor de $\operatorname{per}(D_{20}) = 895.014.631.192.902.121$ em $88{,}124$ s de tempo de CPU, enquanto o cálculo de $\operatorname{per}(T_{20}) = 10.946$ leva $87{,}002$ s de tempo de CPU para ser executado. Os tempos de execução do cálculo dos permanentes de D_n e T_n são praticamente os mesmos, dependendo somente de n e não da estrutura da matriz.

As otimizações sugeridas por Nijenhuis e Wilf (1978) ou por Kalmann (1982) para matrizes 0-1 teoricamente reduzem esses tempos de execução em até $O(n)$ vezes para matrizes de ordem n. A Figura 2 ilustra a tela do console do IDE Spyder 5.5.10 resultante da execução do Programa R.

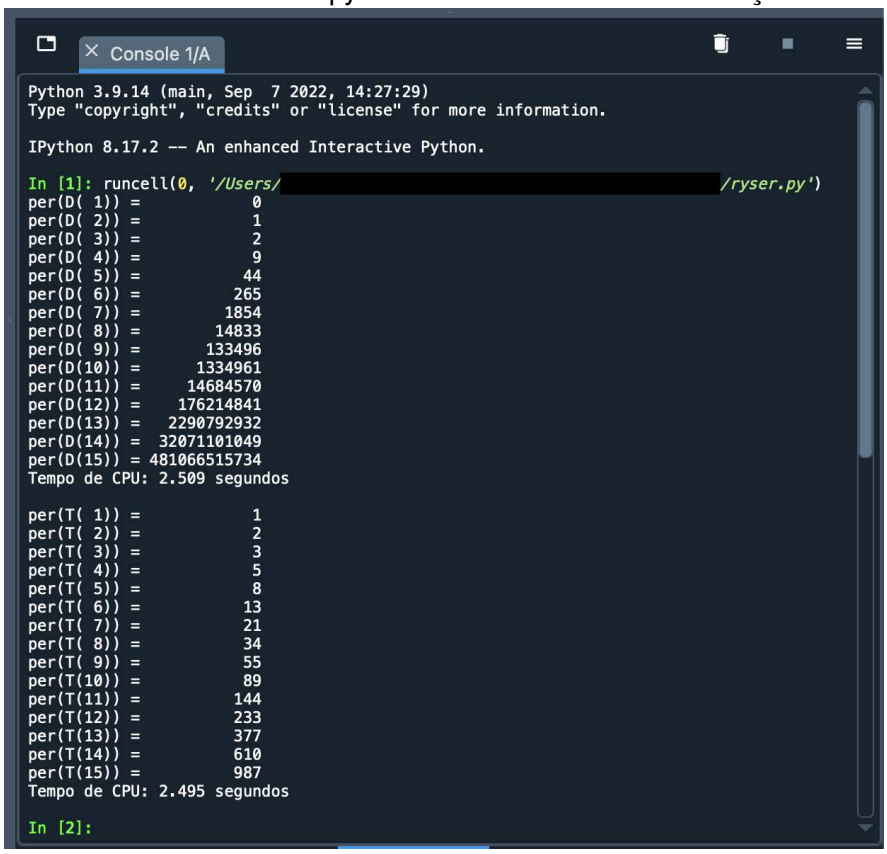
Programa R: Algoritmo de Ryser para o cálculo de permanentes em Python.

```
import time
import numpy as np

# Calculo de per(A) pelo algoritmo de Ryser
def ryser (matA):

    # Verifica se a matriz de entrada e quadrada
    m, n = matA.shape
    if m != n:
       raise ValueError ('A matriz deve ser quadrada!')

    # Codigo de Gray de 0 ate 2**n-1
    gray = [i^(i>>1) for i in range (1<<n)]

    # Inicio do calculo de per(A)
    perm = 0
    # Loop nos subconjuntos J de C = [n] em ordem de Gray
    for setJ in gray:
        # Calcula |J| e o sinal (-1)**|J|
        absJ = bin(setJ).count('1')
        sign = (-1)**absJ
        # Calcula o produto das somas P(A(J))
        prod = 1
        # Loop sobre as linhas i de A
        for i in range (n):
            soma = 0
            # Loop sobre as colunas j de A
            for j in range (n):
                # Seleciona somente as colunas j em J
                if setJ & (1<<j):
                   soma += matA[i][j]
            prod *= soma
        # Adiciona o termo ((-1)**|J|)*P(A(J)) ao calculo de per(A)
        perm += sign*prod
    # Retorna o valor de per(A) calculado pelo algoritmo de Ryser
    return ((-1)**n)*perm
```

```
# Teste da rotina ryser (matA) com matrizes de desarranjo: per(D(n)) = !n
# Instante inicial de execucao
t0 = time.process_time()
for n in range (1,16):
    matD = np.ones((n,n),dtype=int)-np.eye(n,dtype=int)
    perD = ryser (matD)
    print ('per(D(', f'{n:2d}', '))␣=␣', f'{perD:12d}', sep='')
# Instante final de execucao
t1 = time.process_time()
print('Tempo␣de␣CPU:', f'{(t1-t0):.3f}', 'segundos\n')

# Teste da rotina ryser (matA) com matrizes tridiagonais: per(T(n)) = F(n+1)
# Instante inicial de execucao
t0 = time.process_time()
for n in range (1,16):
    matT = np.eye(n,dtype=int)+np.eye(n,k=1,dtype=int)+np.eye(n,k=-1,dtype=int)
    perT = ryser (matT)
    print ('per(T(', f'{n:2d}', '))␣=␣', f'{perT:12d}', sep='')
# Instante final de execucao
t1 = time.process_time()
print('Tempo␣de␣CPU:', f'{(t1-t0):.3f}', 'segundos')
```

Fonte: Elaboração do autor (2023).

Figura 2: Tela do console do IDE Spyder 5.5.10 resultante da execução do Programa R.



Fonte: Elaboração do autor (2023).

4 Conclusões

Neste artigo, fizemos uma exposição do princípio da inclusão-exclusão (PIE) em nível elementar e acessível a alunos de início de graduação. A partir do PIE, deduzimos a expressão matemática para o crivo de Eratóstenes e introduzimos a função totiente de Euler, fundamental em aritmética e teoria dos números. A título de exemplo mais sofisticado de aplicação do PIE, mostramos (mas não deduzimos) a fórmula de Ryser para o cálculo do permanente de uma matriz. Aproveitamos a oportunidade para discutir as principais propriedades dos permanentes, explicamos sua relação com sistemas de representantes distintos (SRD) e calculamos explicitamente o permanente de uma matriz retangular usando a fórmula de Ryser. Também discutimos como o cálculo do permanente constitui o problema mais paradigmático da classe de complexidade algorítmica **#P**-completo. Finalmente, exibimos uma implementação em Python da fórmula de Ryser para matrizes quadradas e testamos a rotina no cálculo do permanente de matrizes para as quais se pode determinar o permanente de maneira analítica. Nossa implementação do algoritmo de Ryser emprega códigos de Gray mas não inclui as otimizações sugeridas por Nijenhuis e Wilf (1978), embora seja relativamente trivial

implementá-las, o que convidamos o leitor a fazer. Da mesma forma, a extensão do Programa R para o cálculo do permanente de matrizes retangulares constitui um interessante exercício para o leitor.

Existem diversas derivações do PIE que podem ser aplicadas a uma enorme variedade de problemas em enumeração combinatória. Sua versão mais geral envolve a atribuição de pesos (também denominados medidas) aos produtos $\bigcap_i A_i$, tornando-a extremamente versátil. O leitor interessado em aprofundar seu conhecimento sobre o PIE deve consultar os excelentes textos de Brualdi e Ryser (1991), Riordan (2002), Ryser (1963) e van Lint e Wilson (2001).

Referências

ARORA, Sanjeev; BARAK, Boaz. **Computational Complexity**: A Modern Approach. Cambridge, UK: Cambridge University Press, 2009. ISBN: 978-0-521-42426-4.

BARVINOK, Alexander. **Combinatorics and Complexity of Partition Functions**. [*S. l.*]: Springer, 2016. v. 30. ISBN: 978-3-319-51828-2. DOI: https://doi.org/10.1007/978-3-319-51829-9.

BRUALDI, Richard A.; RYSER, Herbert John. **Combinatorial Matrix Theory**. Cambridge, UK: Cambridge University Press, 1991. v. 39. Encyclopedia of Mathematics and its Applications. ISBN: 0-521-32265-0.

GOLDREICH, Oded. **Computational Complexity**: A Conceptual Perspective. Cambridge, UK: Cambridge University Press, 2008. ISBN: 978-0-521-88473-0.

HAZZAN, Samuel. **Fundamentos de Matemática Elementar**: Combinatória e Probabilidade. 8. ed. São Paulo, SP: Atual, 2013. v. 5. ISBN: 978-8-535-71750-1.

HEFEZ, Abramo. **Aritmética**. 2. ed. Rio de Janeiro, RJ: SBM, 2016. v. 8. Coleção PROFMAT. ISBN 978-85-8337-105-2

KALMANN, Ralph. A method for finding permanents of 0, 1 matrices. **Mathematics of Computation**, Providence, RI, v. 38, n. 157, p. 167-170, 1982. DOI: https://doi.org/10.1090/S0025-5718-1982-0637294-0.

MINC, Marvin. **Permanents**. London: Addison-Wesley, 1978. v. 6. Encyclopedia of Mathematics and its Applications. ISBN: 0-201-13505-1.

MORGADO, Augusto César; CARVALHO, João Bosco Pitombeira de; CARVALHO, Paulo Cezar Pinto; FERNANDEZ, Pedro. **Análise Combinatória e Probabilidade**. 11. ed. Rio de Janeiro, RJ: SBM, 2020. Coleção do Professor de Matemática. ISBN 978-65-990-3953-9.

NIJENHUIS, Albert; WILF, Herbert Saul. **Combinatorial Algorithms for Computers and Calculators**. 2. ed. New York: Academic Press, 1978.

REALE, Fábio Tosetto. **Métodos de Monte Carlo para amostragem de permutações com restrições e aplicações**. Orientador: José Ricardo Gonçalves de Mendonça. 2018. 57 f. Dissertação (Mestrado em Modelagem de Sistemas Complexos) – Escola de Artes, Ciências e Humanidades, Universidade de São Paulo, São Paulo, 2018. DOI: https://doi.org/10.11606/D.100.2018.tde-06092018-144335.

RIORDAN, John. **Introduction to Combinatorial Analysis**. Mineola, NY: Dover, 2002. ISBN 0-486-42536-3.

RYSER, Herbert John. **Combinatorial Mathematics**. Rahway, NJ: The Mathematical Association of America, 1963. v. 14. Carus Mathematical Monographs.

SCHROEDER, Manfred Robert. **Number Theory in Science and Communication**: With Applications in Cryptography, Physics, Digital Information, Computing, and Self-Similarity. 5. ed. Berlin: Springer, 2009. ISBN: 978-3-540-85297-1. DOI: http://doi.org/10.1007/978-3-540-85298-8.

TENENBAUM, Gérald; FRANCE, Michel Mendès. **The Prime Numbers and Their Distribution**. Providence, RI: AMS, 2000. v. 6. Student Mathematical Library. ISBN: 978-0-821-81647-9.

VALIANT, Leslie Gabriel. The complexity of computing the permanent. **Theoretical Computer Science**, Amsterdam, NL, v. 8, n. 2, p. 189-201, Mar. 1979. DOI: https://doi.org/10.1016/0304-3975(79)90044-6.

VALIANT, Leslie Gabriel. The complexity of enumeration and reliability problems. **SIAM Journal on Computing**, v. 8, n. 3, p. 410-421, Aug. 1979. DOI: https://doi.org/10.1137/0208032.

VAN LINT, Jacobus Hendricus; WILSON, Richard Michael.**A Course in Combinatorics**. 2. ed. Cambridge, UK: Cambridge University Press, 2001. ISBN: 978-0-521-80340-3. Disponível em: http://www.cambridge.org/9780521803403. Acesso em: 23 set. 2024.

WHITNEY, Hassler. A logical expansion in mathematics. **Bulletin of the American Mathematical Society**, New York, v. 38, n. 8, p. 572-579, Aug. 1932. DOI: https://doi.org/10.1090/S0002-9904-1932-05460-X.

Agradecimentos

O autor agradece à Dra. Yeva Gevorgyan (KAUST) pelo auxílio com o pacote TikZ e aos dois revisores cujos comentários contribuíram para melhorar a apresentação final do manuscrito. Este trabalho contou com o apoio da Fundação de Amparo à Pesquisa do Estado de São Paulo – FAPESP por meio do processo AP.R 2020/04475-7.