

矩阵广义迹的拉伸运算方法探究

武真真

东明县职业中等专业学校, 山东 菏泽

收稿日期: 2025年11月11日; 录用日期: 2025年12月5日; 发布日期: 2025年12月12日

摘要

基于矩阵广义迹定义, 本文研究了利用内积运算和矩阵的拉伸运算求解矩阵广义迹的新方法, 通过严格的数学推导给予证明过程, 并结合具体实例验证该方法的可行性。

关键词

矩阵广义迹, 内积, 拉伸运算

Exploration of the Stretching Operation Method for Generalized Trace of Matrix

Zhenzhen Wu

Dongming Vocational Secondary School, Heze Shandong

Received: November 11, 2025; accepted: December 5, 2025; published: December 12, 2025

Abstract

Based on the definition of matrix generalized trace, this paper studies new methods to calculate of matrix generalized trace by using Inner product and Stretching Operation, then provide the rigorous mathematical process and verify the feasibility of the method through examples.

Keywords

Matrix Generalized Trace, Inner Product, Stretching Operation

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

文章引用: 武真真. 矩阵广义迹的拉伸运算方法探究[J]. 应用数学进展, 2025, 14(12): 239-243.

DOI: 10.12677/aam.2025.1412503

1. 引言

作为线性代数中的基本概念，矩阵迹的定义为方阵主对角线元素之和，其特征表现为：特征多项式 $n-1$ 次项系数的 -1 倍或者方阵特征值的总和。然而一般矩阵也有它自身的迹，称其为广义迹。有很多专家对矩阵广义迹已开展多方面研究，包括但不限于：一般矩阵的高次广义迹与行列式的关系研究[1]、两种特殊矩阵 Hadamard 积 Kronecker 积的广义迹与拉伸运算之间的研究[2]、分块矩阵广义迹的研究及应用[3]、用递归算法计算高阶矩阵广义迹的研究[4]、矩阵迹的不等式研究[5]、矩阵拉伸运算的研究[6]及矩阵广义迹的计算方法[7]等，但现有研究尚未系统阐述矩阵广义迹的拉伸运算方法。本文在向量化(vec 算子)的基础上，重点研究内积运算结合拉伸运算求解矩阵广义迹并进行推广，提供严谨的推导步骤，并利用实例检验该方法的可行性。

2. 基础理论

矩阵广义迹的定义[1]:

对于任意的 $m \times n$ 矩阵 $A = (a_{ij})_{m \times n}$ ，其广义迹可定义为：
$$\text{tr}(A) = \sum_{i=1}^{\min(m,n)} a_{ii}。$$

向量化定义:

设 $A = (a_{ij})_{m \times n}$ ，则按照每列拼接起来组成的向量：

$$\text{Vec}(A) = (a_{11}, a_{21}, \dots, a_{m1}, a_{12}, a_{22}, \dots, a_{m2}, \dots, a_{1n}, a_{2n}, \dots, a_{mn})^T$$

拉伸运算[6]: F 为数域，矩阵 $A = (a_{ij}) \in F^{m \times n}$ ，

矩阵 A 按照行顺列顺拉伸为 $\sum_{i=1}^m \sum_{j=1}^n a_{ij} e_{(i-1) \times n + j}$ 的形式，记作： $\text{rscs}(A)$ 。

矩阵 A 按照行顺列逆拉伸为 $\sum_{i=1}^m \sum_{j=1}^n a_{ij} e_{(i-1) \times n + n + 1 - j}$ 的形式，记作： $\text{rsci}(A)$ 。

矩阵 A 按照行逆列顺拉伸为 $\sum_{i=1}^m \sum_{j=1}^n a_{ij} e_{(m-i) \times n + j}$ 的形式，记作： $\text{rics}(A)$ 。

矩阵 A 按照行逆列逆的拉伸为 $\sum_{i=1}^m \sum_{j=1}^n a_{ij} e_{(m-i) \times n + n + 1 - j}$ 的形式，记作： $\text{rici}(A)$ 。

矩阵 A 按照列顺行顺的拉伸为 $\sum_{i=1}^m \sum_{j=1}^n a_{ij} e_{(j-1) \times m + i}^T$ 的形式，记作： $\text{csrs}(A)$ 。

矩阵 A 按照列顺行逆拉伸为 $\sum_{i=1}^m \sum_{j=1}^n a_{ij} e_{(j-1) \times m + m + 1 - i}^T$ 的形式，记作： $\text{csri}(A)$ 。

矩阵 A 按照列逆行顺拉伸为 $\sum_{i=1}^m \sum_{j=1}^n a_{ij} e_{(n-j) \times m + i}^T$ 的形式，记作： $\text{cirs}(A)$ 。

矩阵 A 按照列逆行逆拉伸为 $\sum_{i=1}^m \sum_{j=1}^n a_{ij} e_{(n-j) \times m + m + 1 - i}^T$ 的形式，记作： $\text{ciri}(A)$ 。

下面结合实例直观阐述以上八大拉伸运算形式：

设矩阵 $A = \begin{bmatrix} 1 & 0 & 4 & 0 \\ 1 & 2 & 0 & 1 \\ 0 & 5 & 3 & 2 \end{bmatrix}$ ，展开为

$$\text{rscs}(A) = (1, 0, 4, 0, 1, 2, 0, 1, 0, 5, 3, 2)$$

$$\text{rsci}(A) = (0, 4, 0, 1, 1, 0, 2, 1, 2, 3, 5, 0)$$

$$rics(A) = (0, 5, 3, 2, 1, 2, 0, 1, 1, 0, 4, 0)$$

$$rici(A) = (2, 3, 5, 0, 1, 0, 2, 1, 0, 4, 0, 1)$$

$$csrs(A) = (1, 1, 0, 0, 2, 5, 4, 0, 3, 0, 1, 2)^T$$

$$csri(A) = (1, 1, 0, 5, 2, 0, 3, 0, 4, 2, 1, 0)^T$$

$$cirs(A) = (0, 1, 2, 4, 0, 3, 0, 2, 5, 1, 1, 0)^T$$

$$ciri(A) = (2, 1, 0, 3, 0, 4, 5, 2, 0, 0, 1, 1)^T$$

3. 矩阵广义迹的拉伸运算方法探究

矩阵广义迹是矩阵理论中的一个核心概念，而广义迹的计算可以结合多种运算来求出，下面探索运用拉伸运算方法求出矩阵广义迹。

定理 设 $A = (a_{ij})_{m \times n}$ ，则有 $trA = \left\langle rscs(A_{m \times n}), rscs \left(\begin{bmatrix} E_{\min(m,n)} & O_{\min(m,n) \times [n-\min(m,n)]} \\ O_{[m-\min(m,n)] \times \min(m,n)} & O_{[m-\min(m,n)] \times [n-\min(m,n)]} \end{bmatrix} \right) \right\rangle$ 。

证明： 设 $A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$ ，将其按照行顺序进行拉伸，有

$$rscs(A) = (a_{11}, a_{12}, \cdots, a_{1n}, a_{21}, a_{22}, \cdots, a_{2n}, \cdots, a_{m1}, a_{m2}, \cdots, a_{mn})$$

对 m, n 大小分类讨论：

若 $m = n$ 时，有 $rscs(A) = (a_{11}, a_{12}, \cdots, a_{1n}, a_{21}, a_{22}, \cdots, a_{2n}, \cdots, a_{m1}, a_{m2}, \cdots, a_{mm})$ ；

$$\text{设 } E_m = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}, \text{ 有 } rscs(E_m) = \left(\overbrace{1, 0, 0, \cdots, 0}^{m\text{个}}, \overbrace{1, 0, 0, \cdots, 0}^{m\text{个}}, \cdots, 1 \right)。$$

$$\therefore trA = a_{11} + a_{22} + \cdots + a_{mm} = \langle rscs(A), rscs(E_m) \rangle。$$

若 $m < n$ 时，有

$$rscs(A_m) = (a_{11}, a_{12}, \cdots, a_{1n}, a_{21}, a_{22}, \cdots, a_{2n}, \cdots, a_{m1}, a_{m2}, \cdots, a_{mm})；$$

$$\text{设 } E_m = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}, \text{ 有 } [E_m, O_{m \times (n-m)}] = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & \cdots & 0 \end{bmatrix},$$

$$rscs([E_m, O_{m \times (n-m)}]) = \left(\overbrace{1, 0, 0, \cdots, 0}^{n\text{个}}, \overbrace{1, 0, 0, \cdots, 0}^{n\text{个}}, \cdots, \overbrace{1, 0, 0, \cdots, 0}^{(n-m)\text{个}} \right),$$

$$\therefore trA = \langle rscs(A_{m \times n}), rscs([E_m, O_{m \times (n-m)}]) \rangle。$$

若 $m > n$ 时，有

$$rscs(A_n) = (a_{11}, a_{12}, \cdots, a_{1n}, a_{21}, a_{22}, \cdots, a_{2n}, \cdots, a_{n1}, a_{n2}, \cdots, a_{nn})；$$

$$\text{设 } E_n = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}, \text{ 有 } \begin{bmatrix} E_n \\ O_{(m-n) \times n} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix},$$

$$rscs \left(\begin{bmatrix} E_n \\ O_{(m-n) \times n} \end{bmatrix} \right) = \left(\overbrace{1, 0, 0, \cdots, 0}^{n \uparrow}, \overbrace{1, 0, 0, \cdots, 0}^{n \uparrow}, \overbrace{1, 0, 0, \cdots, 0}^{(m-n) \uparrow} \right),$$

$$\therefore trA = \left\langle rscs(A_{m \times n}), rscs \left(\begin{bmatrix} E_n \\ O_{(m-n) \times n} \end{bmatrix} \right) \right\rangle.$$

$$\text{综上可得 } trA = \left\langle rscs(A_{m \times n}), rscs \left(\begin{bmatrix} E_{\min(m,n)} & O_{\min(m,n) \times [n-\min(m,n)]} \\ O_{[m-\min(m,n)] \times \min(m,n)} & O_{[m-\min(m,n)] \times [n-\min(m,n)]} \end{bmatrix} \right) \right\rangle.$$

同理可证

$$\begin{aligned} trA &= \left\langle csrs(A_{m \times n}), csrs \left(\begin{bmatrix} E_{\min(m,n)} & O_{\min(m,n) \times [n-\min(m,n)]} \\ O_{[m-\min(m,n)] \times \min(m,n)} & O_{[m-\min(m,n)] \times [n-\min(m,n)]} \end{bmatrix} \right) \right\rangle \\ trA &= \left\langle rsci(A_{m \times n}), rsci \left(\begin{bmatrix} E_{\min(m,n)} & O_{\min(m,n) \times [n-\min(m,n)]} \\ O_{[m-\min(m,n)] \times \min(m,n)} & O_{[m-\min(m,n)] \times [n-\min(m,n)]} \end{bmatrix} \right) \right\rangle \\ trA &= \left\langle rics(A_{m \times n}), rics \left(\begin{bmatrix} E_{\min(m,n)} & O_{\min(m,n) \times [n-\min(m,n)]} \\ O_{[m-\min(m,n)] \times \min(m,n)} & O_{[m-\min(m,n)] \times [n-\min(m,n)]} \end{bmatrix} \right) \right\rangle \\ trA &= \left\{ rici(A_{m \times n}), rici \left(\begin{bmatrix} E_{\min(m,n)} & O_{\min(m,n) \times [n-\min(m,n)]} \\ O_{[m-\min(m,n)] \times \min(m,n)} & O_{[m-\min(m,n)] \times [n-\min(m,n)]} \end{bmatrix} \right) \right\} \\ trA &= \left\langle csri(A_{m \times n}), csri \left(\begin{bmatrix} E_{\min(m,n)} & O_{\min(m,n) \times [n-\min(m,n)]} \\ O_{[m-\min(m,n)] \times \min(m,n)} & O_{[m-\min(m,n)] \times [n-\min(m,n)]} \end{bmatrix} \right) \right\rangle \\ trA &= \left\langle cirs(A_{m \times n}), cirs \left(\begin{bmatrix} E_{\min(m,n)} & O_{\min(m,n) \times [n-\min(m,n)]} \\ O_{[m-\min(m,n)] \times \min(m,n)} & O_{[m-\min(m,n)] \times [n-\min(m,n)]} \end{bmatrix} \right) \right\rangle \\ trA &= \left\langle ciri(A_{m \times n}), ciri \left(\begin{bmatrix} E_{\min(m,n)} & O_{\min(m,n) \times [n-\min(m,n)]} \\ O_{[m-\min(m,n)] \times \min(m,n)} & O_{[m-\min(m,n)] \times [n-\min(m,n)]} \end{bmatrix} \right) \right\rangle \end{aligned}$$

矩阵广义迹的拉伸运算方法的步骤(以行顺列顺拉伸运算为例):

第一步: 已知 $m \times n$ 阶的矩阵 A , 首先采用行顺列顺拉伸运算将矩阵 A 拉伸, 即

$$rscs(A) = (a_{11}, a_{12}, \cdots, a_{1n}, a_{21}, a_{22}, \cdots, a_{2n}, \cdots, a_{m1}, a_{m2}, \cdots, a_{mn});$$

第二步: 观察 m, n 的大小, 假设 $m < n$ 则取相应的矩阵 $[E_m, O_{m \times (n-m)}]$,

将其运用行顺列顺拉伸运算将其拉伸,

$$rscs\left(\left[E_m, O_{m \times (n-m)}\right]\right) = \left(\overbrace{1, 0, 0, \dots, 0}^{n\text{个}}, \overbrace{1, 0, 0, \dots, 0}^{n\text{个}}, \overbrace{1, 0, 0, \dots, 0}^{(n-m)\text{个}} \right);$$

第三步: 计算 $rscs(A_{m \times n})$ 和 $rscs\left(\left[E_m, O_{m \times (n-m)}\right]\right)$ 的内积, 即 $\left\langle rscs(A_{m \times n}), rscs\left(\left[E_m, O_{m \times (n-m)}\right]\right) \right\rangle$, 即可求出矩阵的广义迹。

4. 例题验证

设矩阵 $A = \begin{bmatrix} 1 & 0 & 4 & 0 \\ 1 & 2 & 0 & 1 \\ 0 & 5 & 3 & 2 \end{bmatrix}$, 运用拉伸运算求矩阵 A 的广义迹。

解 采用行顺列顺将矩阵 A 拉伸, 即 $rscs(A) = (1, 0, 4, 0, 1, 2, 0, 1, 0, 5, 3, 2)$,

$\therefore$ 矩阵 A 是 3×4 阶的, $\therefore$ 取相对应的矩阵 $[E_3, O_{3 \times 1}] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$, 采用行顺列顺将矩阵 $[E_3, O_{3 \times 1}]$ 拉伸

可得 $rscs([E_3, O_{3 \times 1}]) = (1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0)$,

$\therefore rscs(A)$ 与 $rscs([E_3, O_{3 \times 1}])$ 的内积为 $\left\langle rscs(A), rscs([E_3, O_{3 \times 1}]) \right\rangle = 1 + 0 + 0 + 0 + 0 + 2 + 0 + 0 + 0 + 3 + 0 = 6$ 。

故 $trA = \left\langle rscs(A), rscs([E_3, O_{3 \times 1}]) \right\rangle = 6$ 。

同理, 此题还可以采用另外七种拉伸运算进行求解。

5. 小结

目前, 关于矩阵广义迹的理论研究已取得丰富成果, 然而在拉伸运算结合内积的方法中计算矩阵广义迹这一领域的研究尚存在一定的发展空间。基于此, 本文选择该研究方向展开深入探讨, 旨在通过系统分析拉伸运算结合内积在矩阵广义迹计算中应用效果的认知。拉伸运算除了运用到矩阵广义迹的计算中, 应该还可以运用分块矩阵中, 比如与 Hadamard 积和 Kronecker 积迹运算性质相结合, 当然其他数学运算方法也可能适用于矩阵广义迹的计算, 这将是未来数学应用研究的重要方向。

参考文献

- [1] 杨楠, 刘兴祥, 岳育英. $m \times n$ 矩阵 k 次广义迹[J]. 河南科学, 2012, 30(2): 149-152.
- [2] 刘兴祥, 李姣, 朱磊等. 矩阵的两种特殊运算的广义迹及拉伸运算的关系[J]. 河南科学, 2014, 32(1): 7-9.
- [3] 刘兴祥, 朱磊, 李姣, 程雯姬. 分块矩阵的广义迹及其应用[J]. 河南科学, 2014, 32(11): 2212-2213.
- [4] 毛建耀. 矩阵的广义迹[J]. 天津师范大学学报(自然科学版), 2002, 22(1): 29-32.
- [5] 杨楠, 刘兴祥, 岳育英. 矩阵迹的推广[J]. 延安大学学报(自然科学版), 2012, 31(1): 20-21.
- [6] 张莉. 矩阵拉伸运算的研究[D]: [硕士学位论文]. 延安: 延安大学, 2016.
- [7] 刘兴祥, 武真真. 矩阵广义迹的计算方法[J]. 延安大学学报(自然科学版), 2020, 39(3): 19-22.