

# 从线性代数到张量计算

陈新宇（蒙特利尔大学）& 赵熙乐（电子科技大学）

2022 年 12 月



# 目录

<b>第一章 代数结构</b>	<b>7</b>
1.1 向量与矩阵 . . . . .	7
1.2 张量 . . . . .	7
1.3 特殊代数结构 . . . . .	7
1.4 实际应用：时空数据中的代数结构 . . . . .	7



# 前言

在过去的数十年间，随着信号处理、机器学习与数值计算等领域的快速发展，张量计算已从以线性代数为基础的矩阵计算中逐步拓展开来，相关研究贯穿信号处理、机器学习等众多领域。随着大量张量计算算法涌现出来，我们不难发现：这些算法大多建立在张量分解的基础上。本着以张量计算这一概念为中心，本文将从线性代数出发，讲述张量计算相关的一系列内容。为了提高读者的阅读体验，笔者进行了以下尝试：

- **化繁为简**。将线性代数以及张量计算的范畴限定在实空间中。另外，严格来说，向量和矩阵属于低阶张量，为区分概念，我们默认常提到的张量特指高阶张量（阶数大于或等于 3）。
- **由浅入深**。从基本的线性代数内容展开，通过循序渐进的方式引出一系列矩阵分解与张量分解技术，使读者体会到线性代数的巨大价值。
- **熟能生巧**。本文在撰写过程中尽可能考虑初学者的学习历程，在全文中设计一系列难度适中的例题让读者更直观理解一系列理论，并通过练习熟练掌握相应内容。

笔者深感自身才疏学浅，对于线性代数和张量计算的认识具有一定的局限性，请广大读者批评指正。另外，全文内容设置的合理性也有待考究，需要等待读者的检验。尽管如此，笔者愿竭心力，在后续版本中逐步更新与完善本文，如有建议或疑问，请在 GitHub 开源项目<https://github.com/xinychen/tensor-book>的问答区与笔者进行互动交流。

## 作者声明：

- 撰写本文的初衷在于传播知识，为感兴趣的读者提供参考素材。
- 禁止将本文放在其他网站上供人下载，唯一下载网站为<https://xinychen.github.io/books/tensor-book.pdf>。
- 禁止将本文用于任何形式的商业活动。



# 第一章 代数结构

长期以来，线性代数一直作为机器学习中最为重要的数学工具之一，被人们广泛用于开发各类机器学习算法。线性代数本质上是以向量与矩阵为基本代数结构，本书要讨论的张量分解等模型则主要以张量为基本代数结构。在过去的数十年间，借助线性代数这一基本数学工具，机器学习中涌现出了很多经典的代数模型，这其中不乏矩阵分解、主成分分析，而张量分解在某种程度上可看作是矩阵分解的一种衍生物。

近年来，张量分解在机器学习的众多问题中得到了很好的应用，但关于张量的一些计算与我们所熟悉的线性代数却大相径庭，同时，张量计算相比以矩阵计算为主导的线性代数更为抽象，这使得很多与张量分解相关的内容看起来晦涩难懂。实际上，向量与矩阵都是张量的特例，可以被定义为低阶张量。一般而言，向量是第 1 阶张量，英文表述为 first-order tensor；矩阵是第 2 阶张量，英文表述为 second-order tensor；第 3 阶或者更高阶数的张量被称为高阶张量，英文表述为 higher-order tensor。在各类文献中，通常提到的张量都是特指高阶张量，当然，这在本书的叙述中也不例外。需要注意的是，在各类程序语言中，人们更愿意将张量称为多维数组。

在一个矩阵中，某一元素的位置可以说是“第  $i$  行、第  $j$  列”，即要描述某一元素的位置需用到行和列索引构成的组合  $(i, j)$ 。类似地，在一个第 3 阶张量中，描述某一元素的位置需用到三个索引构成的组合，例如  $(i, j, k)$ 。在处理稀疏矩阵或稀疏张量时，用索引来标记元素的位置会节省下一些不必要的存储开支。

## 1.1 向量与矩阵

## 1.2 张量

## 1.3 特殊代数结构

## 1.4 实际应用：时空数据中的代数结构