张量的基本知识

Tensor Basic Knowledge and Operation

Made by: @BuG_17

Vector

向量（矢量，Vector）是由单独的一行或一列数据组成的排列，用小写加粗的字母表示。向量是一阶的张量（Tensor）. 例如：

$$\boldsymbol{a} = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}$$

表示一个长度为2的vector.

Vector Operations:

- Vector Addition（向量加法）：

  2 vectors:

$$\boldsymbol{a} = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} \quad \boldsymbol{b} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

Then：

$$\boldsymbol{a} + \boldsymbol{b} = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = \begin{pmatrix} a_1 + b_1 \\ a_2 + b_2 \end{pmatrix}$$

- Vector Products（向量乘法）：

  Inner Product（向量内积）：

$$\langle \boldsymbol{a}, \boldsymbol{b} \rangle = \left\langle \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}, \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} \right\rangle = a_1 b_1 + a_2 b_2$$

  Outer Product（向量外积）：

$$\boldsymbol{a} \circ \boldsymbol{b} = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} \circ \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = \begin{pmatrix} a_1 b_1 & a_1 b_2 \\ a_2 b_1 & a_2 b_2 \end{pmatrix}$$

Vector

向量范数（Norm）：Vector的长度。

欧几里得向量范数（Euclidean Vector Norm）：与Vector的几何长度紧密相关，定义：

$$\| \boldsymbol{v} \| = \left(\sum_{i=1}^{n} v_i^2 \right)^{1/2} = \sqrt{\langle \boldsymbol{v}, \boldsymbol{v} \rangle}$$

为Vector的欧几里得范数.

规范化（Normalization）：如果一个Vector是规范化的，那么它的长度（范数）为1，即它具有单位长度。

Matrix

矩阵（Matrix）是数据的二维排列形式，是二阶的张量（Tensor），用大写加粗的字母表示。例如，对于一个m行（row）n列（column）的矩阵，表示为：

$$\boldsymbol{A}_{m\times n}=\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$

此外，矩阵的第 i 行被标记为 $\boldsymbol{A}_{\mathrm{i}*}$，第 j 列被标记为 $\boldsymbol{A}_{*\mathrm{j}}$.

Matrix Operations:

- Matrix Addition（矩阵加法）：类似于Vector加法，将对应位置的元素相加，要求两个矩阵的各个维度元素个数相同。

- Matrix Products（矩阵乘法）：

  Usual Matrix Multiplication（矩阵相乘）：对于矩阵 $\boldsymbol{A}$ 与矩阵 $\boldsymbol{B}$ 的相乘，要求 $\boldsymbol{A}$ 的column数等于 $\boldsymbol{B}$ 的row数。

Matrix

Let:

$$\boldsymbol{A}_{I\times K}=\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1K} \\ \boxed{a_{21} \quad a_{22} \quad \cdots \quad a_{2K}} \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1} & a_{I2} & \cdots & a_{IK} \end{pmatrix} \quad \boldsymbol{B}_{K\times J}=\begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1J} \\ b_{21} & b_{22} & \cdots & b_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ b_{K1} & b_{K2} & \cdots & b_{KJ} \end{pmatrix}$$

Then:

$$\boldsymbol{A}_{I\times K}\boldsymbol{B}_{K\times J}=\boldsymbol{C}_{I\times J}$$

For element c_{ij}, it equals $\langle A_{i*}, B_{*j}\rangle$.

Matrix

哈达玛积（Hadamard product）：

- 对于矩阵$\boldsymbol{A}$与矩阵$\boldsymbol{B}$的哈达玛积，要求两个矩阵的各个维度元素个数相同。
- 作为计算结果的矩阵$\boldsymbol{C}$，其维度与各个维度上元素的个数与原始矩阵相同。
- 用符号 $*$ 表示。

Matrix

Likewise, let:

$$\boldsymbol{A}_{I\times J}=\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1J} \\ a_{21} & a_{22} & \cdots & a_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1} & a_{I2} & \cdots & a_{IJ} \end{pmatrix} \quad \boldsymbol{B}_{I\times J}=\begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1J} \\ b_{21} & b_{22} & \cdots & b_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ b_{I1} & b_{I2} & \cdots & b_{IJ} \end{pmatrix}$$

Then:

$$\begin{aligned} \boldsymbol{A}_{I\times J} * \boldsymbol{B}_{I\times J} &= \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1J} \\ a_{21} & a_{22} & \cdots & a_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1} & a_{I2} & \cdots & a_{IJ} \end{pmatrix} * \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1J} \\ b_{21} & b_{22} & \cdots & b_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ b_{I1} & b_{I2} & \cdots & b_{IJ} \end{pmatrix} \\ &= \begin{pmatrix} a_{11}b_{11} & a_{12}b_{12} & \cdots & a_{1J}b_{1J} \\ a_{21}b_{21} & a_{22}b_{22} & \cdots & a_{2J}b_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1}b_{I1} & a_{I2}b_{I2} & \cdots & a_{IJ}b_{IJ} \end{pmatrix} \\ &= \boldsymbol{C}_{I\times J} \end{aligned}$$

Matrix

克罗内克积（Kronecker product）（直积，张量积）：

- 任意两个矩阵$\boldsymbol{A}$与$\boldsymbol{B}$，都可以做Kronecker积。
- 假设$\boldsymbol{A}$是一个 $I \times J$ 矩阵， $\boldsymbol{B}$是一个 $K \times L$ 矩阵，作为计算结果的矩阵$\boldsymbol{C}$，其维度为 $IK \times JL$.
- 用符号 $\otimes$ 表示。

Matrix

Likewise, let:

$$A_{I\times J}=\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1J}\\ a_{21} & a_{22} & \cdots & a_{2J}\\ \vdots & \vdots & \ddots & \vdots\\ a_{I1} & a_{I2} & \cdots & a_{IJ}\end{pmatrix}\quad B_{K\times L}=\begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1L}\\ b_{21} & b_{22} & \cdots & b_{2L}\\ \vdots & \vdots & \ddots & \vdots\\ b_{K1} & b_{K2} & \cdots & b_{KL}\end{pmatrix}$$

Then:

$$\begin{aligned}A_{I\times J}\otimes B_{K\times L}&=\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1J}\\ a_{21} & a_{22} & \cdots & a_{2J}\\ \vdots & \vdots & \ddots & \vdots\\ a_{I1} & a_{I2} & \cdots & a_{IJ}\end{pmatrix} * \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1L}\\ b_{21} & b_{22} & \cdots & b_{2L}\\ \vdots & \vdots & \ddots & \vdots\\ b_{K1} & b_{K2} & \cdots & b_{KL}\end{pmatrix}\\
&=\begin{pmatrix} a_{11}\begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1L}\\ b_{21} & b_{22} & \cdots & b_{2L}\\ \vdots & \vdots & \ddots & \vdots\\ b_{K1} & b_{K2} & \cdots & b_{KL}\end{pmatrix} & \cdots & a_{1J}\begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1L}\\ b_{21} & b_{22} & \cdots & b_{2L}\\ \vdots & \vdots & \ddots & \vdots\\ b_{K1} & b_{K2} & \cdots & b_{KL}\end{pmatrix}\\ \vdots & \ddots & \vdots\\ a_{I1}\begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1L}\\ b_{21} & b_{22} & \cdots & b_{2L}\\ \vdots & \vdots & \ddots & \vdots\\ b_{K1} & b_{K2} & \cdots & b_{KL}\end{pmatrix} & \cdots & a_{IJ}\begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1L}\\ b_{21} & b_{22} & \cdots & b_{2L}\\ \vdots & \vdots & \ddots & \vdots\\ b_{K1} & b_{K2} & \cdots & b_{KL}\end{pmatrix}\end{pmatrix}\\
&=C_{IK\times JL}\end{aligned}$$

Matrix

KR积（Khatri-Rao product）：

- 如果矩阵$\boldsymbol{A}$与矩阵$\boldsymbol{B}$可以做KR积，则它们的列数相同。
- 假设$\boldsymbol{A}$是一个 $I \times K$ 矩阵， $\boldsymbol{B}$是一个 J × K 矩阵，作为计算结果的矩阵$\boldsymbol{C}$，其维度为 $IJ \times K$.
- 用符号 ⊙ 表示。

Matrix

Likewise, let:

$$
\boldsymbol{A}_{I\times K}=\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1K} \\ a_{21} & a_{22} & \cdots & a_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1} & a_{I2} & \cdots & a_{IK} \end{pmatrix} \quad \boldsymbol{B}_{J\times K}=\begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1K} \\ b_{21} & b_{22} & \cdots & b_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ b_{J1} & b_{J2} & \cdots & b_{JK} \end{pmatrix}
$$

Then:

$$
\begin{aligned}
\boldsymbol{A}_{I\times K}\odot\boldsymbol{B}_{J\times K} &= \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1K} \\ a_{21} & a_{22} & \cdots & a_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1} & a_{I2} & \cdots & a_{IK} \end{pmatrix} \odot \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1K} \\ b_{21} & b_{22} & \cdots & b_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ b_{J1} & b_{J2} & \cdots & b_{JK} \end{pmatrix} \\
&= \left(\left(\begin{pmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{I1} \end{pmatrix} \otimes \begin{pmatrix} b_{11} \\ b_{21} \\ \vdots \\ b_{J1} \end{pmatrix}\right) \cdots \left(\begin{pmatrix} a_{1K} \\ a_{2K} \\ \vdots \\ a_{IK} \end{pmatrix} \otimes \begin{pmatrix} b_{1K} \\ b_{2K} \\ \vdots \\ b_{JK} \end{pmatrix}\right)\right) \\
&= \begin{pmatrix} a_{11}\begin{pmatrix} b_{11} \\ b_{21} \\ \vdots \\ b_{J1} \end{pmatrix} & \cdots & a_{1K}\begin{pmatrix} b_{1K} \\ b_{2K} \\ \vdots \\ b_{JK} \end{pmatrix} \\ a_{21}\begin{pmatrix} b_{11} \\ b_{21} \\ \vdots \\ b_{J1} \end{pmatrix} & \cdots & a_{2K}\begin{pmatrix} b_{1K} \\ b_{2K} \\ \vdots \\ b_{JK} \end{pmatrix} \\ \vdots & \ddots & \vdots \\ a_{I1}\begin{pmatrix} b_{11} \\ b_{21} \\ \vdots \\ b_{J1} \end{pmatrix} & \cdots & a_{IK}\begin{pmatrix} b_{1K} \\ b_{2K} \\ \vdots \\ b_{JK} \end{pmatrix} \end{pmatrix} \\
&= \boldsymbol{C}_{IJ\times K}
\end{aligned}
$$

Matrix

矩阵标量积（Khatri-Rao product）：

- 对于矩阵$\boldsymbol{A}$ 与矩阵$\boldsymbol{B}$ 的标量积，要求两个矩阵的各个维度元素个数相同。
- 假设$\boldsymbol{A}$和$\boldsymbol{B}$都是 $I \times J$ 矩阵， 与向量内积类似地，作为运算结果的c是一个标量。
- 矩阵$\boldsymbol{A}$ 与矩阵$\boldsymbol{B}$ 的标量积用$\langle \boldsymbol{A}, \boldsymbol{B} \rangle$表示。

Definition:

$$\langle \boldsymbol{A}, \boldsymbol{B} \rangle = \sum_{i=1}^{I} \sum_{j=1}^{J} (a_{ij} b_{ij}) = Tr(\boldsymbol{A}^T \boldsymbol{B})$$

Matrix

$$\mathbf{A}_{I\times J} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1J} \\ a_{21} & a_{22} & \cdots & a_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1} & a_{I2} & \cdots & a_{IJ} \end{pmatrix} \quad \mathbf{B}_{I\times J} = \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1J} \\ b_{21} & b_{22} & \cdots & b_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ b_{I1} & b_{I2} & \cdots & b_{IJ} \end{pmatrix}$$

Definition:

$$\langle \boldsymbol{A}, \boldsymbol{B} \rangle = \sum_{i=1}^{I} \sum_{j=1}^{J} (a_{ij} b_{ij}) = Tr(\boldsymbol{A}^T \boldsymbol{B})$$

Matrix

矩阵范数（Norm）：与向量范数类似，与矩阵的标量积有关。主要讨论弗罗贝尼乌斯范数（Frobenius Matrix Norm，F-范数）。例如，对于一个m行（row）n列（column）的矩阵$\boldsymbol{A}$，其定义如下：

$$\|\boldsymbol{A}\|_F=\left(\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2\right)^{1/2}=\sqrt{\langle\boldsymbol{A},\boldsymbol{A}\rangle}$$

此外，对于两个各个维度元素个数相同的矩阵$\boldsymbol{A}$与矩阵$\boldsymbol{B}$，它们差的范数，即$\|\boldsymbol{A}-\boldsymbol{B}\|_F$，定义了这两个矩阵之间的距离。

Matrix Inverses（矩阵求逆）：

对于方程：

$$\boldsymbol{A}\boldsymbol{x}=\boldsymbol{b}$$

其解为：

$$\boldsymbol{x}=\boldsymbol{A}^{-1}\boldsymbol{b}$$

然而，最直接的求逆方法只有当矩阵为方针，并且非奇异（即$|\boldsymbol{A}|\neq 0$）时才可以使用。当矩阵求逆的条件不再存在时，可以使用矩阵的伪逆矩阵（pseudoinverse matrix）（穆尔-彭罗斯广义逆矩阵，Moore-Penrose generalized inverse matrix）：

$$\mathbf{A}^{\dagger}=\begin{cases}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T & \text{when } rank(\mathbf{A}_{m\times n})=n\\ \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1} & \text{when } rank(\mathbf{A}_{m\times n})=m\end{cases}$$

- 如果方程所给出的系统是一致的，那么$\boldsymbol{x}=\boldsymbol{A}^{\dagger}\boldsymbol{b}$是最小欧几里得范数的解；
- 如果方程所给出的系统是不一致的，那么$\boldsymbol{x}=\boldsymbol{A}^{\dagger}\boldsymbol{b}$是最小欧几里得范数的最小二乘解。

Matrix

矩阵的分解（decomposition）：

- 满秩分解

  设矩阵 $A_{m\times n}$ 秩为 r，则存在秩为 r 的矩阵 $B_{m\times r}$，秩为 r 的矩阵 $C_{r\times n}$，使得：

$$A_{m\times n}=B_{m\times r}C_{r\times n}$$

  其中，$B_{m\times r}$称为列满秩矩阵，$C_{r\times n}$称为行满秩矩阵。

  满秩分解不是唯一的。

- QR分解（正交三角分解）

  设矩阵 $A_{m\times n}$ 秩为 n，则它可以唯一地分解为：

$$A_{m\times n}=Q_{m\times n}R_{n\times n}$$

  其中，$Q_{m\times n}$是标准正交向量组矩阵，$R_{n\times n}$ 是正线上三角矩阵。

- 实对称矩阵的特征分解（谱分解、对角化）

  当 n 阶方阵 A 具有 n 个线性无关的特征向量时，可以将 A 特征分解为可逆矩阵与对角矩阵的乘积：

$$A=P\Lambda P^{-1}$$

  其中，P 是对应于 Λ 主对角特征值的特征向量组。

  如果 A 是实对称矩阵，则必然存在特征向量组成的正交矩阵 Q，使得：

$$A=Q\Lambda Q^{T}$$

  虽然任意一个实对称矩阵 A 都有特征分解，但是特征分解可能并不唯一。如果两个或多个特征向量拥有相同的特征值，那么在由这些特征向量产生的生成子空间中，任意一组正交向量都是该特征值对应的特征向量。惯例下通常按降序排列 Λ 的元素。在该约定下，特征分解唯一当且仅当所有的特征值都是唯一的。

Matrix

- 奇异值分解（SVD，Singular Value Decomposition）

　　对于一个秩为 r 的矩阵 $\boldsymbol{A}_{m\times n}$，必存在 $m\times m$ 的正交矩阵 $\boldsymbol{U}_{m\times m}$，$n\times n$ 的正交矩阵 $\boldsymbol{V}_{n\times n}$，$m\times n$ 的矩阵 $\boldsymbol{\Sigma}_{m\times n}$，使得：

$$\boldsymbol{A}_{m\times n}=\boldsymbol{U}_{m\times m}\boldsymbol{\Sigma}_{m\times n}\boldsymbol{V}_{n\times n}^T=\boldsymbol{U}_{m\times m}\begin{pmatrix}\boldsymbol{D}_{r\times r} & \boldsymbol{O}\\ \boldsymbol{O} & \boldsymbol{O}\end{pmatrix}\boldsymbol{V}_{n\times n}^T$$

　　其中，

$$\boldsymbol{D}_{r\times r}=\begin{pmatrix}\sqrt{\lambda_1} & & & \\ & \sqrt{\lambda_2} & & \\ & & \ddots & \\ & & & \sqrt{\lambda_r}\end{pmatrix}_{r\times r}$$

　　$\lambda_1\ge\lambda_2\ge\ldots\ge\lambda_r>0$，为 $\boldsymbol{A}^T\boldsymbol{A}$ 的 r 个非零特征值。$\sqrt{\lambda_1}$，$\sqrt{\lambda_2}$，…，$\sqrt{\lambda_r}$ 称为 $\boldsymbol{A}$ 的正奇异值。$\boldsymbol{A}^T\boldsymbol{A}$ 的 n 个特征值开根号 $\sqrt{\lambda_1}$，$\sqrt{\lambda_2}$，…，$\sqrt{\lambda_r}$，0_1，0_2，…，0_{n-r}（从大到小排列）称为 $\boldsymbol{A}$ 的奇异值。

　　矩阵的奇异值分解一定存在，但不唯一。

- LU分解

　　对于一个可逆方阵 $\boldsymbol{A}$，如果它的所有顺序主子式都不为零时，可以唯一地分解为一个上三角矩阵和一个下三角矩阵的乘积形式，即：

$$\boldsymbol{A}=\boldsymbol{L}\boldsymbol{U}$$

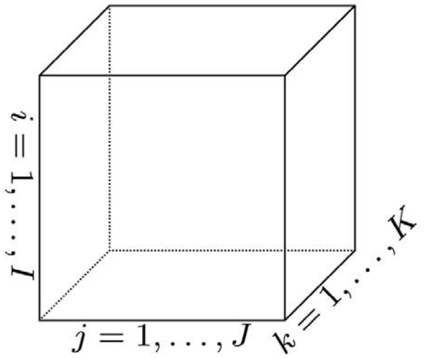
　　其中，$\boldsymbol{L}$，$\boldsymbol{U}$分别是下三角矩阵和上三角矩阵。

　　LU分解在本质上是高斯消元法的一种表达形式。实质上是将A通过初等行变换变成一个上三角矩阵，其变换矩阵就是一个单位下三角矩阵。

Tensor

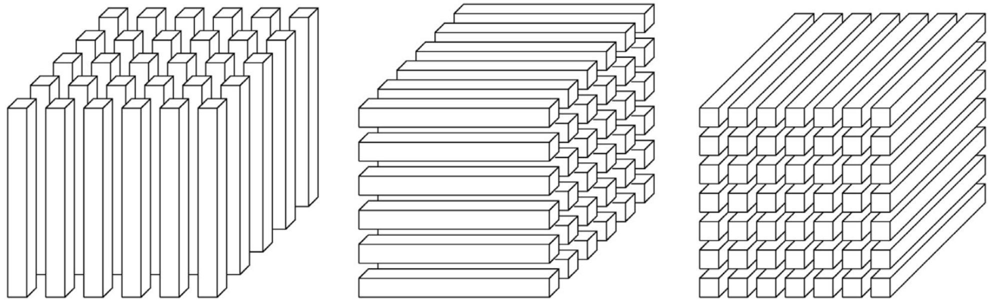
张量（Tensor）是一种多维度的数据排列。一维的张量即向量（Vector），用小写加粗的字母表示。二维的张量即矩阵（Matrix），用大写加粗的字母表示。三维及三维以上的张量用加粗的欧拉体字母（或带下划线的大写加粗的字母）表示。标量用小写字母表示。

与矩阵类似地，下标用于指定张量的某一个或某一组元素。冒号（colon）作为下标，用来指定某一个维度的全部元素。



- **Fiber:**

在N-way张量中，固定N-1个维度，剩余一个维度取任意（所有）元素，这样所得到的元素组就是一个fiber. 例如，对于一个三阶的张量$\mathcal{X} \in \mathbb{R}^{I \times J \times K}$，有column fibers（mode A vectors），row fibers（mode B vectors），tube fibers（mode C vectors），分别用$\boldsymbol{x}_{:jk}$，$\boldsymbol{x}_{i:k}$和$\boldsymbol{x}_{ij:}$表示。

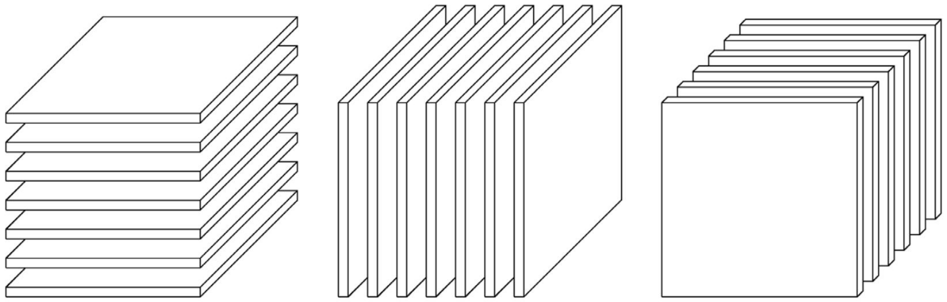


(a) Mode-1 (column) fibers: $\mathbf{x}_{:jk}$ (b) Mode-2 (row) fibers: $\mathbf{x}_{i:k}$ (c) Mode-3 (tube) fibers: $\mathbf{x}_{ij:}$

Tensor

- **Slice:**

　　在N-way张量中，固定N-2个维度，剩余两个维度取任意（所有）元素，这样所得到的元素组就是一个slice. 例如，对于一个三阶的张量$\mathcal{X} \in \mathbb{R}^{I \times J \times K}$，有horizontal slides，lateral slides和frontal slides，分别用$\boldsymbol{X}_{i::}$，$\boldsymbol{X}_{:j:}$和$\boldsymbol{X}_{::k}$表示。除此之外，第k个frontal slide，还可以简写为$\boldsymbol{X}_k$.



(a) Horizontal slices: $\mathbf{X}_{i::}$　(b) Lateral slices: $\mathbf{X}_{:j:}$　(c) Frontal slices: $\mathbf{X}_{::k}$ (or $\mathbf{X}_k$)

张量范数（Norm）：与矩阵的弗罗贝尼乌斯范数（Frobenius Norm）类似。对于一个N-way的张量$\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$，其定义如下:

$$\|\mathcal{X}\| = \left(\sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \cdots \sum_{i_N=1}^{I_N} x_{i_1 i_2 \ldots i_N}^2 \right)^{1/2} = \sqrt{\langle \mathcal{X}, \mathcal{X} \rangle}$$

其中，$\langle \mathcal{X}, \mathcal{X} \rangle$为张量的内积（inner product）. 更普遍地，对于两个同样size的张量$\mathcal{X}, \mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$，有:

$$\langle \mathcal{X}, \mathcal{Y} \rangle = \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \cdots \sum_{i_N=1}^{I_N} x_{i_1 i_2 \ldots i_N} y_{i_1 i_2 \ldots i_N}$$

Tensor

张量的矩阵展开（Matricization）：

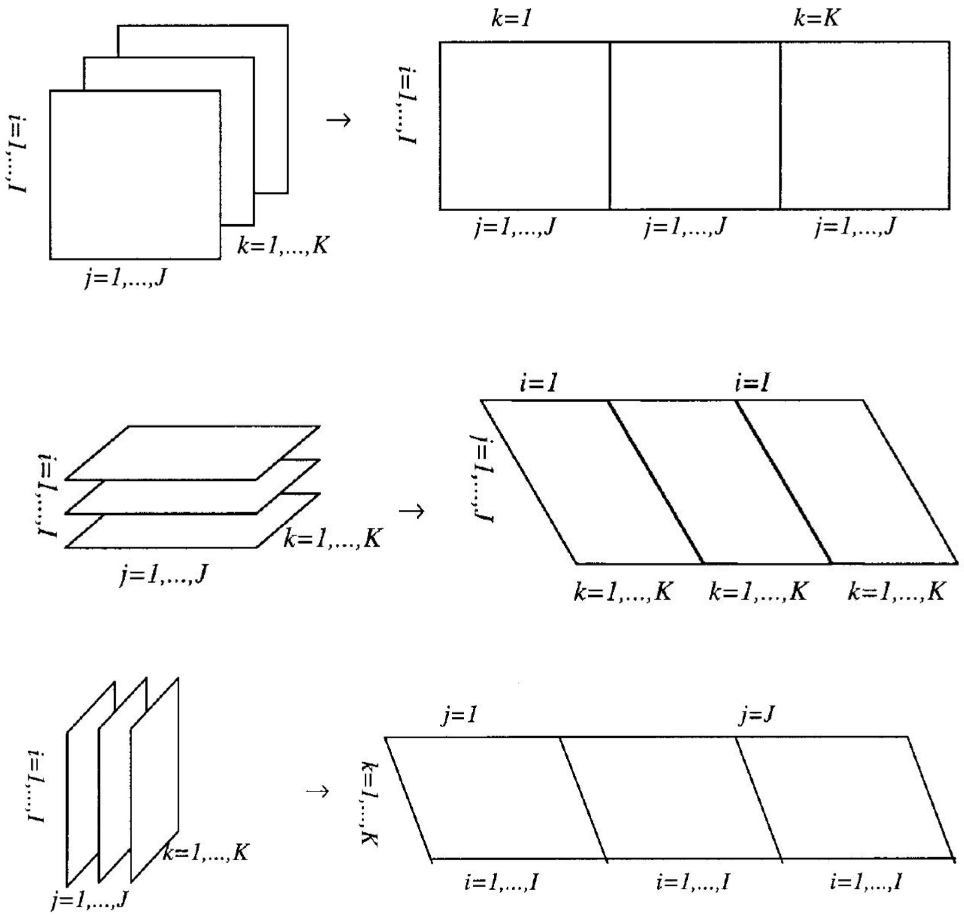
将3维或更高维的张量转换为一个矩阵。实际上是收集所有的mode n vectors，排列成一个matrix. 不同的文献中使用了不同的列展开顺序。事实上，列的具体排列并不重要，只要它在相关计算中是一致的。这里举其中一种说明：

- 三维张量的mode A 矩阵展开（the mode A matricized version of a three-way tensor）：将张量分为mode A fibers，按mode B 嵌套于mode C的方式排列成一个矩阵，这就是三维张量的mode A 矩阵展开。假设张量$\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I \times J \times K}$，则展开后的矩阵$\boldsymbol{X}_a$是一个 $I \times JK$ 矩阵。

类似地，定义：

- 三维张量的mode B 矩阵展开（the mode B matricized version of a three-way tensor）：将张量分为mode B fibers，按mode C 嵌套于mode A的方式排列成一个矩阵，这就是三维张量的mode B 矩阵展开。假设张量$\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I \times J \times K}$，则展开后的矩阵$\boldsymbol{X}_b$是一个 $J \times KI$ 矩阵。
- 三维张量的mode C 矩阵展开（the mode C matricized version of a three-way tensor）：将张量分为mode C fibers，按mode A 嵌套于mode B的方式排列成一个矩阵，这就是三维张量的mode C 矩阵展开。假设张量$\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I \times J \times K}$，则展开后的矩阵$\boldsymbol{X}_c$是一个 $K \times IJ$ 矩阵。

Tensor


i=1,...,I
k=1,...,K
j=1,...,J
k=1
k=K
j=1,...,J
j=1,...,J
j=1,...,J
i=1,...,I
k=1,...,K
j=1,...,J
i=1
i=I
j=1,...,J
k=1,...,K
k=1,...,K
k=1,...,K
i=1,...,I
k=1,...,K
j=1,...,J
j=1
j=J
k=1,...,K
i=1,...,I
i=1,...,I
i=1,...,I

Tensor

三维张量的以上三种矩阵展开方式可以通过简单的循环置换互相联系，只需更换张量的mode的排列。例如，将排列 $(i\ j\ k)$ 置换一次，得到 $(j\ k\ i)$，即使得置换后的mode A对应置换前的mode B，置换后的mode B对应置换前的mode C，置换后的mode C对应置换前的mode A，这样就完成了从$\boldsymbol{\mathcal{X}}$展开为$\boldsymbol{X}_a$到$\boldsymbol{\mathcal{X}}$展开为$\boldsymbol{X}_b$的转换。类似地，将mode的排列置换两次，可以将$\boldsymbol{\mathcal{X}}$展开为$\boldsymbol{X}_c$.

假设有一个三维张量：

$$\boldsymbol{\mathcal{X}} = \begin{matrix} 1 & 2 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & -1 \\ -1 & 0 & 0 \end{matrix} \qquad \begin{matrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 1 \end{matrix}$$

则：

$$\boldsymbol{X}_a = \begin{pmatrix} 1 & 2 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & -1 & 1 & 2 & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\boldsymbol{X}_b = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 & -1 & 0 \\ 2 & 1 & 1 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \end{pmatrix}$$

$$\boldsymbol{X}_c = \begin{pmatrix} 1 & 0 & 1 & -1 & 2 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 2 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Tensor

类似地，可以定义三维以上张量的矩阵展开，具体的展开过程与三维张量的展开大致相同。例如对于一个N-way张量， $\boldsymbol{X}_a$（如果维数很多，可以用数字作为下标，如$\boldsymbol{X}_1$，表示mode-1展开）包含了所有mode A 向量，它们按mode B 嵌套在 mode C，mode C 嵌套在 mode D ...的方式排列成一个矩阵，矩阵的size就是 $I \times JKLM ...$; $\boldsymbol{X}_b$包含了所有mode B 向量，它们按mode C 嵌套在 mode D，mode D 嵌套在 mode E ...的方式（此时mode A在嵌套的最外层）排列成一个矩阵，矩阵的size就是 J $\times KLM ...I$，以此类推。

除此之外，还有其他的矩阵展开方式。如对于三维张量的mode B 矩阵展开，也可以按模的顺序，使mode A嵌套于mode C；对于三维张量的mode C 矩阵展开，按模的顺序，使mode A嵌套于mode B.如果使用其他的展开方式，则要对其清晰地描述。例如，如果将一个 $I \times J \times K \times L$ 的张量展开为 $IJ \times KL$ 大小的矩阵，应该描述为：结合mode 1 和mode 2（mode 1 嵌套于 mode 2），mode 3 和mode 4（mode 3 嵌套于 mode 4）的矩阵展开。

Tensor

张量的向量化（Vectorization）：

将3维或更高维的张量用一个向量来表示。实际上是收集（张量展开的）矩阵中所有的列，组合成一个单独的向量。这个过程表示为：

$$\boldsymbol{u} = Vec(\boldsymbol{U})$$

为了得到这个矩阵，首先要对张量进行mode A 展开；为了将张量向量化，再将这个mode A 展开的矩阵向量化。这样一来，可以得到任意维张量的向量化结果。

$$\boldsymbol{x} = Vec(\boldsymbol{X}_a)$$

模 n 秩和张量秩（mode n rank & tensor rank）：

- mode n rank：模 n 空间（mode n space）的秩，写作$rank_n(\boldsymbol{\mathcal{X}})$. 例如，一个$I \times J \times K \times L$ 的张量，其mode A rank（或mode 1 rank）为mode A vector $\boldsymbol{x}_{:jkl}$ 所扩展的空间的秩。具体来说，就是矩阵 $\boldsymbol{X}_a$ 的秩。总的来说，mode n rank就是矩阵 $\boldsymbol{X}_{(n)}$ 的秩，其中矩阵 $\boldsymbol{X}_{(n)}$ 是张量 $\boldsymbol{\mathcal{X}}$ 的mode n 矩阵展开，即：

$$rank_n(\boldsymbol{\mathcal{X}}) = rank(\boldsymbol{X}_{(n)})$$

特别地，由于矩阵的行向量组和列向量组有相同的秩，所以二维张量的mode A rank与mode B rank相等.

Tensor

- tensor rank：将张量分解为秩为1的张量之和。秩为1的张量意味着其中的元素可以被写为$x_{ijkl...} = a_i b_j c_k d_l ...$. 之后，张量的秩就是足以分解该张量的秩1张量的最少个数（这个将张量分解为若干个秩1张量的过程就是CP分解，即CANDECOMP/PARAFAC Decomposition），标记为$rank(\mathcal{X})$. 特别地，如果张量是二阶的（即矩阵），那么它的秩就等于行向量组的秩或列向量组的秩；但是对于三阶或三阶以上的张量，其秩远大于（从不小于）任意mode n rank.

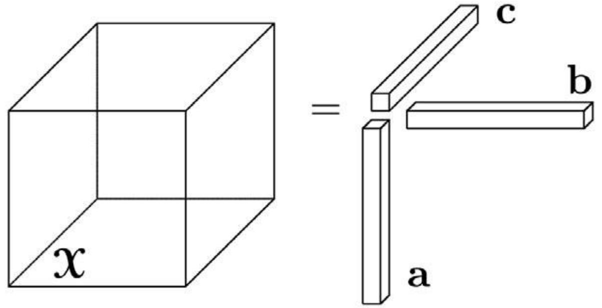
秩为1的张量（rank-1 tensor）：

一个N维的张量$\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$的秩为1，当它可以被写为N个向量的外积，即：

$$\mathcal{X} = \boldsymbol{a}^{(1)} \circ \boldsymbol{a}^{(2)} \circ \ldots \circ \boldsymbol{a}^{(N)}$$

其中，元素

$$x_{i_1 i_2 \ldots i_N} = a_{i_1}^{(1)} a_{i_2}^{(2)} \ldots a_{i_N}^{(N)}$$



Tensor

张量的运算：

- mode n product（模 n 积）：

模 n 积定义为张量与矩阵（或向量）在 mode n 上的乘积。该运算用符号 $\times_n (\bar{\times}_n)$ 表示，n 即在 mode n 上。

假设有张量 $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$，矩阵 $\boldsymbol{U} \in \mathbb{R}^{J \times I_n}$，则它们在 mode n 上的乘积写为：

$$\boldsymbol{\mathcal{X}} \times_n \boldsymbol{U}$$

作为结果的张量

$$\boldsymbol{\mathcal{Y}} \in \mathbb{R}^{I_1 \times \ldots \times I_{n-1} \times J \times I_{n+1} \times \ldots \times I_N}$$

对于其中的每一个元素来说，有：

$$(\boldsymbol{\mathcal{X}} \times_n \boldsymbol{U})_{i_1 \times \ldots \times i_{n-1} \times j \times i_{n+1} \times \ldots \times i_N} = \sum_{i_n=1}^{I_n} x_{i_1 \times \ldots \times i_{n-1} \times i_n \times i_{n+1} \times \ldots \times i_N} u_{j i_n}$$

可以理解为：张量$\boldsymbol{\mathcal{X}}$的每一个 mode n fiber 都被矩阵 $\boldsymbol{U}$ 相乘。如果以矩阵展开的方式表示，则为：

$$\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{X}} \times_n \boldsymbol{U} \Leftrightarrow \mathbf{Y}_{(n)} = \boldsymbol{U} \boldsymbol{X}_{(n)}$$

Tensor

如果是张量与向量的模 n 积，运算与上述类似。

假设有N阶张量 $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$，向量 $\boldsymbol{v} \in \mathbb{R}^{I_n}$，则它们在 mode n 上的乘积写为：

$$\boldsymbol{\mathcal{X}} \bar{\times}_n \boldsymbol{v}$$

作为结果的张量，其阶数为N-1，size为$I_1 \times \ldots \times I_{n-1} (\times 1) \times I_{n+1} \times \ldots \times I_N$，即：

$$\boldsymbol{\mathcal{Y}} \in \mathbb{R}^{I_1 \times \ldots \times I_{n-1} \times I_{n+1} \times \ldots \times I_N}$$

对于其中的每一个元素来说，有：

$$(\boldsymbol{\mathcal{X}} \bar{\times}_n \boldsymbol{v})_{i_1 \times \ldots \times i_{n-1} \times i_{n+1} \times \ldots \times i_N} = \sum_{i_n=1}^{I_n} x_{i_1 \times \ldots \times i_{n-1} \times i_n \times i_{n+1} \times \ldots \times i_N} v_{i_n}$$

可以理解为：张量$\boldsymbol{\mathcal{X}}$的每一个 mode n fiber 都与向量 $\boldsymbol{v}$ 做内积，结果就减少了一个维度。

海南大学 Hainan university

Tensor

张量的分解：

- CP分解（CANDECOMP/PARAFAC Decomposition）：

将张量分解为若干个秩为1的张量（正如上文提到的，可以被写为N个向量的外积）的和。例如，给定一个3维张量$\mathcal{X} \in \mathbb{R}^{I\times J\times K}$，对其进行CP分解：

$$\mathcal{X} \approx \sum_{r=1}^{R} \boldsymbol{a}_r \circ \boldsymbol{b}_r \circ \boldsymbol{c}_r$$

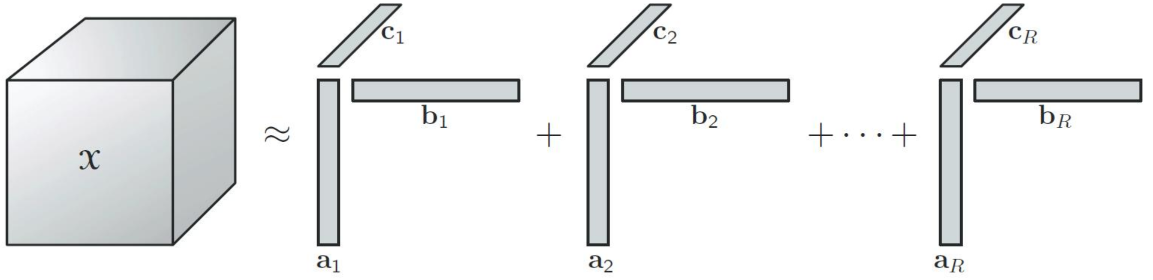
其中，R为正整数，$\boldsymbol{a}_r \in \mathbb{R}^I$，$\boldsymbol{b}_r \in \mathbb{R}^J$，$\boldsymbol{c}_r \in \mathbb{R}^K$，$r = 1, 2, ..., R$.

Name	Proposed by
Polyadic form of a tensor	Hitchcock, 1927 [105]
PARAFAC (parallel factors)	Harshman, 1970 [90]
CANDECOMP or CAND (canonical decomposition)	Carroll and Chang, 1970 [38]
Topographic components model	Möcks, 1988 [166]
CP (CANDECOMP/PARAFAC)	Kiers, 2000 [122]

对于张量中的每一个元素，对应的分解关系为：

$$x_{ijk} \approx \sum_{r=1}^{R} a_{ir} b_{jr} c_{kr}$$

其中，$i = 1, 2, ..., I$，$j = 1, 2, ..., J$，$k = 1, 2, ..., K$



Tensor

- 因子矩阵（factor matrix）：

  组成秩1张量的向量的组合。例如：

$$\boldsymbol{A} = [\boldsymbol{a}_1\ \boldsymbol{a}_2\ \dots\ \boldsymbol{a}_R]$$

类似地，

$$\boldsymbol{B} = [\boldsymbol{b}_1\ \boldsymbol{b}_2\ \dots\ \boldsymbol{b}_R]$$

$$\boldsymbol{C} = [\boldsymbol{c}_1\ \boldsymbol{c}_2\ \dots\ \boldsymbol{c}_R]$$

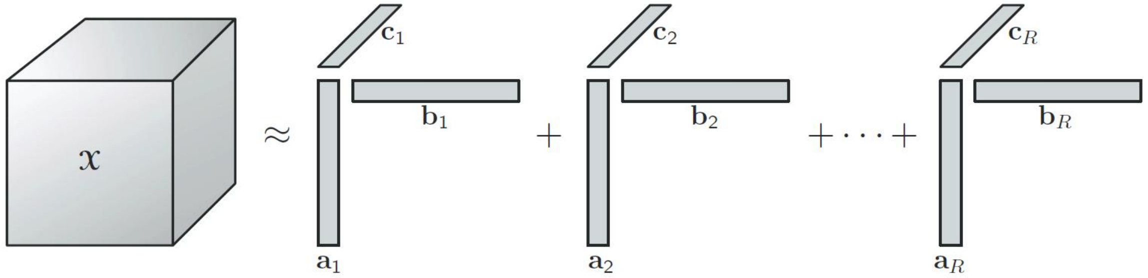
使用因子矩阵，将CP分解写为矩阵展开的形式：

$$\boldsymbol{X}_{(1)} \approx \boldsymbol{A}(\boldsymbol{C} \odot \boldsymbol{B})^T$$

$$\boldsymbol{X}_{(2)} \approx \boldsymbol{B}(\boldsymbol{C} \odot \boldsymbol{A})^T$$

$$\boldsymbol{X}_{(3)} \approx \boldsymbol{C}(\boldsymbol{B} \odot \boldsymbol{A})^T$$

其中，$\odot$为KR积。



这样的三维模型有时也写为frontal slices的形式。

$$\boldsymbol{X}_k \approx \boldsymbol{A}\boldsymbol{D}^{(k)}\boldsymbol{B}^T$$

其中$\boldsymbol{D}^{(k)}$是对角阵，$\boldsymbol{D}^{(k)} \equiv diag(\boldsymbol{C}_{k:})$，$k = 1, 2, \dots, K$，$\boldsymbol{X}_k$中 k 表示第 k 个 slice.

Horizontal slices 和 lateral slices 形式的方程也可以类似地写出。但总体来说，以slice形式表示不容易拓展到三维以上。

Tensor

总的来说，CP模型可以被概括为下面的形式：

$$\mathcal{X} \approx [\![A, B, C]\!] \equiv \sum_{r=1}^{R} \boldsymbol{a}_r \circ \boldsymbol{b}_r \circ \boldsymbol{c}_r$$

可以假定$\boldsymbol{A}$，$\boldsymbol{B}$ 和 $\boldsymbol{C}$ 的列都被规范化为长度1，权重被概括到向量$\boldsymbol{\lambda} \in \mathbb{R}^R$，使得：

$$\mathcal{X} \approx [\![\boldsymbol{\lambda}\,; A, B, C]\!] \equiv \sum_{r=1}^{R} \lambda_r\, \boldsymbol{a}_r \circ \boldsymbol{b}_r \circ \boldsymbol{c}_r$$

以上的分解针对三维模型，因为三维模型普遍适用，可以满足很多需要。事实上，对于N阶张量$\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$，CP分解表示为：

$$\mathcal{X} \approx [\![\boldsymbol{\lambda}\,; \boldsymbol{A}^{(1)}, \boldsymbol{A}^{(2)}, \ldots, \boldsymbol{A}^{(N)}]\!] \equiv \sum_{r=1}^{R} \lambda_r\, \boldsymbol{a}_r^{(1)} \circ \boldsymbol{a}_r^{(2)} \circ \ldots \circ \boldsymbol{a}_r^{(N)}$$

其中$\boldsymbol{\lambda} \in \mathbb{R}^R$，$\boldsymbol{A}^{(n)} \in \mathbb{R}^{I_n \times R}$， $n = 1, 2, \ldots, N.$

在这种情况下，mode n 矩阵展开的形式由下式给出：

$$\boldsymbol{X}_{(n)} \approx \boldsymbol{A}^{(n)} \boldsymbol{\Lambda} \left(\boldsymbol{A}^{(N)} \odot \ldots \odot \boldsymbol{A}^{(n+1)} \odot \boldsymbol{A}^{(n-1)} \odot \ldots \odot \boldsymbol{A}^{(1)}\right)^T$$

其中$\boldsymbol{\Lambda} = diag(\boldsymbol{\lambda})$.

Tensor

CP分解的唯一性：

矩阵作为二阶张量，其分解不具备唯一性，高阶张量的秩分解通常具有唯一性。

- 矩阵CP分解的不唯一性：

假设秩为R的矩阵$\boldsymbol{X} \in \mathbb{R}^{I\times J}$，则这个矩阵的一个秩分解为：

$$\boldsymbol{X} = \boldsymbol{A}\boldsymbol{B}^T = \sum_{r=1}^{R} \boldsymbol{a}_r \circ \boldsymbol{b}_r$$

如 果 矩 阵 $\boldsymbol{X}$ 的 奇 异 值 分 解 （S V D，Singular Value Decomposition）是$\boldsymbol{U\Sigma V}^T$，可以令$\boldsymbol{A} = \boldsymbol{U\Sigma}$， $\boldsymbol{B} = \boldsymbol{V}$，同时，如果令$\boldsymbol{A} = \boldsymbol{U\Sigma W}$， $\boldsymbol{B} = \boldsymbol{VW}$，其中$\boldsymbol{W}$是 $R \times R$ 正交阵，这也同样合理。类似地，还可以乘以任意多的正交阵，因为正交阵具有 $\boldsymbol{W}\boldsymbol{W}^T = \boldsymbol{E}$ 的性质。

- 张量CP分解的唯一性：

上文已经提到，张量的CP分解为：

$$\mathcal{X} \approx [\![\boldsymbol{A},\ \boldsymbol{B},\ \boldsymbol{C}]\!] \equiv \sum_{r=1}^{R} \boldsymbol{a}_r \circ \boldsymbol{b}_r \circ \boldsymbol{c}_r$$

张量的CP分解（排除二阶的矩阵）在一定约束下是具有唯一性的，体现在：唯一可能的和为$\mathcal{X}$的秩1张量的组合，不考虑元素的扩展（scaling）和排列（permutation）不确定性。

排列不确定性指的是秩1张量可以被任意重排。

$$\mathcal{X} = [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!] = [\![\mathbf{A}\boldsymbol{\Pi}, \mathbf{B}\boldsymbol{\Pi}, \mathbf{C}\boldsymbol{\Pi}]\!] \text{ for any } R \times R \text{ permutation matrix } \boldsymbol{\Pi}.$$

扩展不确定性指的是可以缩放每一个向量，只要缩放因子的乘积为1.

$$\mathcal{X} = \sum_{r=1}^{R} (\alpha_r \mathbf{a}_r) \circ (\beta_r \mathbf{b}_r) \circ (\gamma_r \mathbf{c}_r)$$

Tensor

k-rank:

矩阵 $\boldsymbol{A}$ 的 k-rank，记作 $k_{\boldsymbol{A}}$，定义为：最大的 k，使得任意 k 列向量都是线性无关的。

- 张量CP分解唯一性的充分条件：

三阶张量的CP分解为：

$$\mathcal{X} \approx [\![\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]\!] \equiv \sum_{r=1}^{R} \boldsymbol{a}_r \circ \boldsymbol{b}_r \circ \boldsymbol{c}_r$$

该分解唯一，当：

$$k_{\boldsymbol{A}} + k_{\boldsymbol{B}} + k_{\boldsymbol{C}} \geq 2R + 2$$

推广到N阶张量的CP分解：

$$\mathcal{X} = [\![\boldsymbol{A}^{(1)}, \boldsymbol{A}^{(2)}, \ldots, \boldsymbol{A}^{(N)}]\!]$$

$$= \sum_{r=1}^{R} \boldsymbol{a}_r^{(1)} \circ \boldsymbol{a}_r^{(2)} \circ \ldots \circ \boldsymbol{a}_r^{(N)}$$

该分解唯一，当：

$$\sum_{n=1}^{N} k_{\boldsymbol{A}^{(n)}} \geq 2R + (N-1)$$

- 张量CP分解唯一性的必要条件：

三阶张量的CP分解唯一，则：

$$min\,\{rank(\boldsymbol{A} \odot \boldsymbol{B}), rank(\boldsymbol{A} \odot \boldsymbol{C}), rank(\boldsymbol{B} \odot \boldsymbol{C})\} = R$$

推广到N阶张量，CP分解唯一，则：

$$\min_{n=1,\ldots,N} rank\left(\boldsymbol{A}^{(1)} \odot \ldots \odot \boldsymbol{A}^{(n-1)} \odot \boldsymbol{A}^{(n+1)} \odot \ldots \odot \boldsymbol{A}^{(N)}\right) = R$$

由于：

$$rank(\boldsymbol{A} \odot \boldsymbol{B}) \leq rank(\boldsymbol{A} \otimes \boldsymbol{B}) \leq rank(\boldsymbol{A}) \cdot rank(\boldsymbol{B})$$

因此一个更简单的必要条件是：

$$\min_{n=1,\ldots,N} \left(\prod_{\substack{m=1 \\ m \neq n}}^{N} \mathrm{rank}(\mathbf{A}^{(m)}) \right) \geq R.$$

Tensor

CP分解算法实现——ALS算法：

对于给定张量$\mathcal{X}$的CP分解为：

$$\mathcal{X} \approx \sum_{r=1}^{R} \boldsymbol{a}_r^{(1)} \circ \boldsymbol{a}_r^{(2)} \circ \ldots \circ \boldsymbol{a}_r^{(N)}$$

在R确定的情况下，张量的CP分解有多种算法实现，这里介绍主流的交替二乘法（the alternating least squares (ALS)）.

从三阶张量开始介绍，对于给定的三阶张量$\mathcal{X} \in \mathbb{R}^{I \times J \times K}$，找一个秩为$R$的张量$\hat{\mathcal{X}}$，使其尽可能与$\mathcal{X}$接近，即求解如下无约束问题：

$$\min_{\hat{\mathcal{X}}} \|\mathcal{X} - \hat{\mathcal{X}}\|$$

其中$\hat{\mathcal{X}}$为：

$$\hat{\mathcal{X}} = \sum_{r=1}^{R} \lambda_r \, \boldsymbol{a}_r \circ \boldsymbol{b}_r \circ \boldsymbol{c}_r = [\![\boldsymbol{\lambda} ; \boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]\!]$$

ALS算法的实现思路是：

- 固定矩阵 $\boldsymbol{B}, \boldsymbol{C}$，求解最优解 $\boldsymbol{A}$
- 固定矩阵 $\boldsymbol{A}, \boldsymbol{C}$，求解最优解 $\boldsymbol{B}$
- 固定矩阵 $\boldsymbol{A}, \boldsymbol{B}$，求解最优解 $\boldsymbol{C}$

重复以上步骤，直到满足某个收敛条件算法终止，收敛条件可以是：

- ✓ 目标函数的值不再下降或下降很少
- ✓ 因子矩阵（factor matrices）不再变化或变化很少

Tensor

因为固定了其中两个矩阵，因此问题实际是线性最小二乘问题（linear least-squares problem）。假设矩阵 $\boldsymbol{B}, \boldsymbol{C}$ 已经固定，对张量 $\mathcal{X}, \widehat{\mathcal{X}}$ 做mode-1展开，则问题变为：

$$\min_{\widehat{\mathcal{X}}} \left\| \mathcal{X}_{(1)} - \widehat{\mathcal{X}}_{(1)} \right\|$$

其中$\widehat{\mathcal{X}}_{(1)}$的表达式为：

$$\widehat{\mathcal{X}}_{(1)} = \boldsymbol{A}(\boldsymbol{C} \odot \boldsymbol{B})^T$$

因此原问题变为：

$$\min_{\widehat{\mathcal{X}}} \left\| \mathcal{X}_{(1)} - \widehat{\boldsymbol{A}}(\boldsymbol{C} \odot \boldsymbol{B})^T \right\|_F$$

这里计算的是F-范数，其中

$$\widehat{\boldsymbol{A}} = \boldsymbol{A} \cdot diag(\lambda)$$

该问题的最优解为：

$$\widehat{\boldsymbol{A}} = \mathcal{X}_{(1)}[(\boldsymbol{C} \odot \boldsymbol{B})^T]^\dagger$$

其中，†表示矩阵的伪逆，即Moore-Penrose广义逆。

上面的表达式又可以写为：

$$\widehat{\boldsymbol{A}} = \mathcal{X}_{(1)}(\boldsymbol{C} \odot \boldsymbol{B})(\boldsymbol{C}^T\boldsymbol{C} * \boldsymbol{B}^T\boldsymbol{B})^\dagger$$

转换为这样的形式的优势为原先是计算 $R \times JK$ 矩阵（$(\boldsymbol{C} \odot \boldsymbol{B})^T$ 的Moore-Penrose广义逆，现在为计算 $R \times R$ 矩阵 $\boldsymbol{C}^T\boldsymbol{C} * \boldsymbol{B}^T\boldsymbol{B}$ 的Moore-Penrose广义逆。

最后，正规化 $\widehat{\boldsymbol{A}}$ 的列向量，得到 $\boldsymbol{A}$，即：令$\lambda_r = \|\widehat{\boldsymbol{a}}_r\|$，则 $\boldsymbol{a}_r = \widehat{\boldsymbol{a}}_r/\lambda_r$， $r = 1,2, \dots, R$.

Tensor

对于更高阶张量的CP分解，ALS算法如下（其中 R 是事先确定好的，并且因子矩阵可以被任意初始化）：

procedure CP-ALS($\mathcal{X}$,R)
  initialize $\mathbf{A}^{(n)} \in \mathbb{R}^{I_n \times R}$ for $n = 1, \dots, N$
  repeat
    for $n = 1, \dots, N$ **do**
      $\mathbf{V} \leftarrow \mathbf{A}^{(1)\mathsf{T}}\mathbf{A}^{(1)} * \cdots * \mathbf{A}^{(n-1)\mathsf{T}}\mathbf{A}^{(n-1)} * \mathbf{A}^{(n+1)\mathsf{T}}\mathbf{A}^{(n+1)} * \cdots * \mathbf{A}^{(N)\mathsf{T}}\mathbf{A}^{(N)}$
      $\mathbf{A}^{(n)} \leftarrow \mathbf{X}_{(n)}(\mathbf{A}^{(N)} \odot \cdots \odot \mathbf{A}^{(n+1)} \odot \mathbf{A}^{(n-1)} \odot \cdots \odot \mathbf{A}^{(1)})\mathbf{V}^{\dagger}$
      normalize columns of $\mathbf{A}^{(n)}$ (storing norms as $\boldsymbol{\lambda}$)
    end for
  until fit ceases to improve or maximum iterations exhausted
  return $\boldsymbol{\lambda}, \mathbf{A}^{(1)}, \mathbf{A}^{(2)}, \dots, \mathbf{A}^{(N)}$
end procedure

Tensor

再提Tensor Rank：

正如上文中已经提过的，将张量分解为秩为1的张量之和，之后，张量的秩就是足以分解该张量的秩1张量的最少个数，标记为$rank(\mathcal{X})$.

要补充的是，实值（real-valued）张量的秩在实数域和复数域上可能是不同的。例如，对于张量$\mathcal{X}$，它的frontal slices如下：

$$X_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$X_2 = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

其在实数域上的秩分解为：$\mathcal{X} = [\![\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]\!]$，其中：

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & -1 \end{bmatrix}, \boldsymbol{B} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}, \boldsymbol{C} = \begin{bmatrix} 1 & 1 & 0 \\ -1 & 1 & 1 \end{bmatrix}$$

其在复数域上的秩分解为：$\mathcal{X} = [\![\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]\!]$，其中：

$$\boldsymbol{A} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ -i & i \end{bmatrix}, \boldsymbol{B} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ i & -i \end{bmatrix}, \boldsymbol{C} = \begin{bmatrix} 1 & 1 \\ i & -i \end{bmatrix}$$

- 没有直接的算法，可以确定给定的一个特定的张量的秩（这是一个NP-hard问题），除了下面这些特殊情况下的**最大秩（maximum rank）**和**典型秩（typical rank）**（实数域上）。
- 在实践中，张量的秩是通过匹配各种秩为R的CP模型来确定的。

海南大学
Hainan university

Tensor

Tensor size	Maximum rank	Citation
$I \times J \times 2$	$\min\{I, J\} + \min\{I, J, \lfloor \max\{I, J\}/2 \rfloor\}$	[109, 143]
$3 \times 3 \times 3$	5	[143]

Tensor size	Typical rank	Citation
$2 \times 2 \times 2$	$\{2, 3\}$	[143]
$3 \times 3 \times 2$	$\{3, 4\}$	[142, 212]
$5 \times 3 \times 3$	$\{5, 6\}$	[214]
$I \times J \times 2$ with $I \geq 2J$ (very tall)	$2J$	[216]
$I \times J \times 2$ with $J < I < 2J$ (tall)	I	[216]
$I \times I \times 2$ (compact)	$\{I, I+1\}$	[212, 216]
$I \times J \times K$ with $I \geq JK$ (very tall)	JK	[213]
$I \times J \times K$ with $JK - J < I < JK$ (tall)	I	[213]
$I \times J \times K$ with $I = JK - J$ (compact)	$\{I, I+1\}$	[213]

Tensor

最大秩（maximum rank）：可以达到的最大秩。

典型秩（typical rank）：任何一个出现的概率大于零的秩。

对于 $I \times J$ 的矩阵，它的最大秩和典型秩相同，为 $min\{I,J\}$. 对于张量来说，这两者可能不同。此外，对于实数域上的分解，其结果可能会有不止一个典型秩；但是对于复数域上的分解，总是只有一个典型秩。总的来说，对于普遍的三阶张量$\mathcal{X} \in \mathbb{R}^{I \times J \times K}$，其最大秩的上界是已知的：

$$rank(\mathcal{X}) = min\{IJ, IK, JK\}$$

对于 $I \times J \times K$ 的张量，或者更高阶的张量，各个mode的排序不影响张量的秩，即在各模重新排列的情况下，秩是不变的。

在frontal slices上具有对称性的三维张量在实数域上的典型秩：

Tensor size	Typical rank
$I \times I \times 2$	$\{I, I+1\}$
$3 \times 3 \times 3$	4
$3 \times 3 \times 4$	$\{4, 5\}$
$3 \times 3 \times 5$	$\{5, 6\}$
$I \times I \times K$ with $K \geq I(I+1)/2$	$I(I+1)/2$

在frontal slices上有无对称性的三维张量在实数域上的典型秩的比较：

Tensor size	Typical rank with symmetry	Typical rank without symmetry
$I \times I \times 2$ (compact)	$\{I, I+1\}$	$\{I, I+1\}$
$2 \times 3 \times 3$ (compact)	$\{3, 4\}$	$\{3, 4\}$
$3 \times 2 \times 2$ (tall)	3	3
$I \times 2 \times 2$ with $I \geq 4$ (very tall)	3	4
$I \times 3 \times 3$ with $I = 6, 7, 8$ (tall)	6	I
$9 \times 3 \times 3$ (very tall)	6	9

Tensor

Tucker分解（Tucker decomposition）：

Tucker分解是主成分分析（PCA）的高阶形式。Tucker分解的思路是将张量分解为一个核心张量（core tensor），并沿每个模态乘以一个矩阵。

对于一个给定张量$\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$，其Tucker分解为：

$$\mathcal{X} = \mathcal{G} \times_1 A^{(1)} \times_2 A^{(2)} \ldots \times_N A^{(N)}$$

可以简写为：

$$\mathcal{X} \approx [\![\mathcal{G}; A^{(1)}, A^{(2)}, \ldots, A^{(N)}]\!]$$

其中，$A^{(n)} \in \mathbb{R}^{I_n \times R_n}$，$n = 1,2,\ldots,N$，$\mathcal{G} \in \mathbb{R}^{R_1 \times R_2 \times \ldots \times R_N}$，即核心张量。

Table 4.1 *Names for the Tucker decomposition (some specific to three-way and some for N-way).*

Name	Proposed by
Three-mode factor analysis (3MFA/Tucker3)	Tucker, 1966 [226]
Three-mode PCA (3MPCA)	Kroonenberg and De Leeuw, 1980 [140]
N-mode PCA	Kapteyn et al., 1986 [113]
Higher-order SVD (HOSVD)	De Lathauwer et al., 2000 [63]
N-mode SVD	Vasilescu and Terzopoulos, 2002 [229]

对于其中的每个元素，表示为：

$$x_{i_1 i_2 \cdots i_N} = \sum_{r_1=1}^{R_1} \sum_{r_2=1}^{R_2} \cdots \sum_{r_N=1}^{R_N} g_{r_1 r_2 \cdots r_N} \, a^{(1)}_{i_1 r_1} a^{(2)}_{i_2 r_2} \cdots a^{(N)}_{i_N r_N}$$

$$\text{for} \quad i_n = 1, \ldots, I_n, \; n = 1, \ldots, N.$$

矩阵形式的表示：

$$\mathbf{X}_{(n)} = \mathbf{A}^{(n)} \mathbf{G}_{(n)} (\mathbf{A}^{(N)} \otimes \cdots \otimes \mathbf{A}^{(n+1)} \otimes \mathbf{A}^{(n-1)} \otimes \cdots \otimes \mathbf{A}^{(1)})^{\mathsf{T}}.$$

讨论三维张量的Tucker分解：对于张量$\mathcal{X} \in \mathbb{R}^{I \times J \times K}$，其Tucker分解形式如下：

$$\mathcal{X} \approx \mathcal{G} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} = \sum_{p=1}^{P} \sum_{q=1}^{Q} \sum_{r=1}^{R} g_{pqr} \, \mathbf{a}_p \circ \mathbf{b}_q \circ \mathbf{c}_r = [\![\mathcal{G} \,; \mathbf{A}, \mathbf{B}, \mathbf{C}]\!].$$

其中，$A \in \mathbb{R}^{I \times P}$，$B \in \mathbb{R}^{J \times Q}$，$C \in \mathbb{R}^{K \times R}$，是Tucker分解得到的因子矩阵，$\mathcal{G} \in \mathbb{R}^{P \times Q \times R}$代表核心张量，它的元素代表不同因子矩阵之间相互作用的水平。

Tensor

讨论三维张量的Tucker分解：对于张量 $\mathcal{X} \in \mathbb{R}^{I\times J\times K}$，其Tucker分解形式如下：

$$\mathcal{X} \approx \mathcal{G} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} = \sum_{p=1}^{P}\sum_{q=1}^{Q}\sum_{r=1}^{R} g_{pqr}\, \mathbf{a}_p \circ \mathbf{b}_q \circ \mathbf{c}_r = [\![\mathcal{G}\,; \mathbf{A}, \mathbf{B}, \mathbf{C}]\!].$$

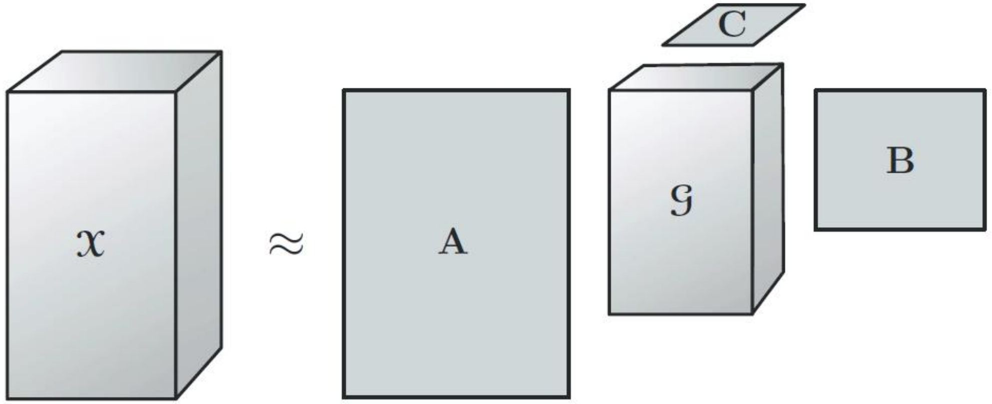
其中，$\boldsymbol{A} \in \mathbb{R}^{I\times P}$， $\boldsymbol{B} \in \mathbb{R}^{J\times Q}$， $\boldsymbol{C} \in \mathbb{R}^{K\times R}$，是Tucker分解得到的因子矩阵，$\mathcal{G} \in \mathbb{R}^{P\times Q\times R}$代表核心张量，它的元素代表不同因子矩阵之间相互作用的水平。

Tucker分解也可以表示如下：

$$\mathcal{X} \approx [\![\mathcal{G}; \boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]\!]$$

其中每个元素：

$$x_{ijk} \approx \sum_{p=1}^{P}\sum_{q=1}^{Q}\sum_{r=1}^{R} g_{pqr}\, a_{ip}\, b_{jq}\, c_{kr} \quad \text{for} \quad i = 1,\ldots,I,\ \ j = 1,\ldots,J,\ \ k = 1,\ldots,K.$$



如果 P,Q,R 小于 I,J,K，核心张量就相当于原始张量的压缩形式，即对原始张量进行降维操作的结果。

Tucker分解的另外一种形式：

$$\mathbf{X}_{(1)} \approx \mathbf{A}\mathbf{G}_{(1)}(\mathbf{C} \otimes \mathbf{B})^{\mathsf{T}},$$
$$\mathbf{X}_{(2)} \approx \mathbf{B}\mathbf{G}_{(2)}(\mathbf{C} \otimes \mathbf{A})^{\mathsf{T}},$$
$$\mathbf{X}_{(3)} \approx \mathbf{C}\mathbf{G}_{(3)}(\mathbf{B} \otimes \mathbf{A})^{\mathsf{T}}.$$

此处$\boldsymbol{G}_{(n)}$代表核心张量的mode-n展开。

如果核心张量是一个超对角张量（即，在对角张量的基础上约束：$P=Q=R$），那么Tucker分解就变成了CP分解。

Tensor

Tucker分解有一些重要的特例：

- Tucker2分解：对于三阶张量，固定一个因子矩阵为单位阵。例如，固定 $\boldsymbol{C}=\boldsymbol{I}$，则Tucker2分解为：

$$\mathcal{X}=\mathcal{G}\times_1\mathbf{A}\times_2\mathbf{B}=[\![\mathcal{G}\,;\mathbf{A},\mathbf{B},\mathbf{I}]\!].$$

- Tucker1分解：进一步，固定两个因子矩阵为单位阵。例如，固定 $\boldsymbol{C}=\boldsymbol{I}$， $\boldsymbol{B}=\boldsymbol{I}$，则Tucker1分解为：

$$\mathcal{X}=\mathcal{G}\times_1\mathbf{A}=[\![\mathcal{G}\,;\mathbf{A},\mathbf{I},\mathbf{I}]\!].$$

这与标准的二维PCA（主成分分析）相同，因为：

$$\mathbf{X}_{(1)}=\mathbf{A}\mathbf{G}_{(1)}.$$

再提mode n rank：

正如上文中已经提过的，模 n 秩是模 n 空间的秩，写作 $rank_n(\mathcal{X})$. 它是矩阵 $\boldsymbol{X}_{(n)}$ 的秩，其中矩阵 $\boldsymbol{X}_{(n)}$ 是张量 $\mathcal{X}$ 的 mode n 矩阵展开，即：

$$rank_n(\mathcal{X})=rank(\boldsymbol{X}_{(n)})$$

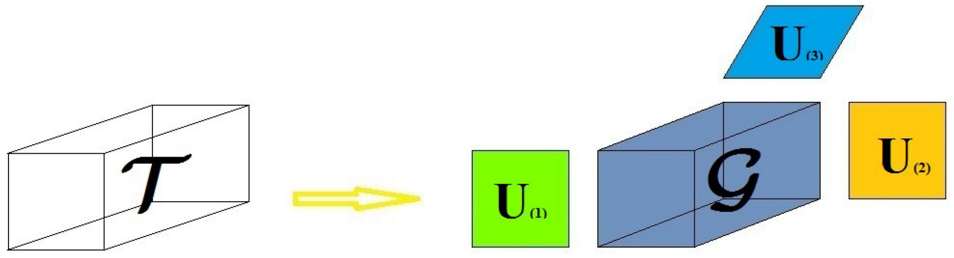
令 $rank_n(\mathcal{X})=R_n$， $n=1,2,\ldots,N$，则 $\mathcal{X}$ 叫做秩$(R_1,R_2,\ldots,R_n)$的张量。R_n 就可以看作是张量 $\mathcal{X}$ 在各个 mode 上 fiber 所构成的空间的维度。

如果 $rank_n(\mathcal{X})=R_n$， $n=1,2,\ldots,N$，则很容易得到 $\mathcal{X}$ 的一个精确秩-$(R_1,R_2,\ldots,R_N)$ Tucker分解；然而如果至少有一个n，使得 $rank_n(\mathcal{X})>R_n$，则通过Tucker分解得到的就是 $\mathcal{X}$ 的一个秩-$(R_1,R_2,\ldots,R_N)$近似。

Tensor

Tucker分解的唯一性不能保证。一般加上一些约束，如分解得到的因子矩阵单位正交约束等。比如HOSVD（High Order SVD）算法，它约束因子矩阵单位正交，核心张量在各模上的slices正交。通过在张量的每一个mode上做SVD分解，对各个mode上的因子矩阵进行求解，最后计算张量在各个mode上的投影之后的张量作为核张量。HOSVD分解的唯一性是可以保证的。它的算法过程如下：

$\mathrm{HOSVD}(\mathcal{X}, R_1, R_2, \cdots, R_N)$
for $k = 1$ **to** N **do**
  $A_{(n)} \leftarrow R_n$ leading left singular vectors of $X_{(n)}$
end for
$\mathcal{G} \leftarrow \mathcal{X} \times_1 A^{(1)T} \times_2 A^{(2)T} \cdots \times_N A^{(N)T}$
return $\mathcal{G}, A^{(1)}, A^{(2)}, \cdots, A^{(N)}$



以三阶张量举例，具体来说，HOSVD首先将张量 $\mathcal{T}$ 在各 mode 上矩阵展开为 $(\boldsymbol{T}_{(1)}, \boldsymbol{T}_{(2)}, \boldsymbol{T}_{(3)})$，将这些矩阵进行奇异值分解，取左奇异矩阵 $(\boldsymbol{U}_{(1)}, \boldsymbol{U}_{(2)}, \boldsymbol{U}_{(3)})$，则张量 $\mathcal{T}$ 的HOSVD为

$$\mathcal{T} = \mathcal{G} \times_1 \boldsymbol{U}^{(1)} \times_2 \boldsymbol{U}^{(2)} \times_3 \boldsymbol{U}^{(3)}$$

其中，核心张量 $\mathcal{G}$ 为原始张量 $\mathcal{T}$ 与求得的左奇异矩阵的逆矩阵（由于矩阵单位正交，逆矩阵亦为转置矩阵）的积，即：

$$\mathcal{G} = \mathcal{T} \times_1 \boldsymbol{U}^{(1)T} \times_2 \boldsymbol{U}^{(2)T} \times_3 \boldsymbol{U}^{(3)T}$$

矩阵形式：

$$\mathcal{G}_{(n)} = \boldsymbol{U}^{(n)T} \mathcal{T}_{(n)} (\boldsymbol{U}^{(N)} \otimes \ldots \otimes \boldsymbol{U}^{(n+1)} \otimes \boldsymbol{U}^{(n-1)} \otimes \ldots \otimes \boldsymbol{U}^{(1)})$$

Tensor

低秩逼近（Low-Rank Approximation）：

在矩阵分析中，用一个更低秩的矩阵来近似一个矩阵是很常见的。例如，寻找一个低秩矩阵，最好地近似给定的一个秩为 r 的矩阵。

假设 $\boldsymbol{A}$ 是秩 r 矩阵，$\boldsymbol{B}$ 是秩 k 矩阵，满足 $k<r$，并且最近似于 $\boldsymbol{A}$，则:

$$\|\boldsymbol{A}-\boldsymbol{B}\|_F=\sqrt{\sigma_{k+1}^2+\ldots+\sigma_r^2}$$

其中，σ_r 表示矩阵 $\boldsymbol{A}$ 的第 k 个奇异值。

类比到张量，寻找一个低秩张量 $\boldsymbol{\mathcal{B}}$，其秩比张量 $\boldsymbol{\mathcal{T}}$ 更低，以使得最小化：$\|\boldsymbol{\mathcal{T}}-\boldsymbol{\mathcal{B}}\|_F$. 尽管在矩阵中，最优的低秩逼近通过奇异值的函数给出，在张量中却没有这样直接的结果。为了计算最优的低秩逼近，需要使用迭代的方法。这一过程用上文中提到的ALS算法来实现。

- 最佳秩1逼近

在分析张量时，最佳秩1逼近是一个重要的工具。最佳秩1逼近的结果提供了一个秩1张量，使得原张量分解为向量的外积，即被向量的外积所近似。

给定张量 $\boldsymbol{\mathcal{T}}\in\mathbb{R}^{I_1\times I_2\times\ldots\times I_N}$，定义二次代价函数（quadratic cost function）：

$$f(\boldsymbol{a},\boldsymbol{b},\boldsymbol{c})=\|\boldsymbol{\mathcal{T}}-\boldsymbol{a}\circ\boldsymbol{b}\circ\boldsymbol{c}\|^2$$

寻找其最佳秩1逼近，即最小化该二次代价函数所得的秩1张量。使用ALS算法计算。

海南大学
Hainan university

THANKS