IFT 6760A - Lecture 9
Tensor Decompositions - Part 1

Scribe(s): Gavin McCracken, Koustuv Sinha **Instructor:** Guillaume Rabusseau

1 Summary

In the previous lecture we looked at Weighted Automata (WA) and their relations with Hidden Markov Models (HMMs) and their use as recurrent models, and corresponding proofs.
This lecture began by introducing notation, definitions and operations on tensors: vectorization, matricization, inner product, norm, n-mode product. This was followed by an introduction to tensor network notation and examples of how to use it and subsequently, the outer, Kronecker, and Khatri-Rao products were introduced. The second half of the lecture introduced the CanDeComp (canonical decomposition)/ParaFac decomposition (parallel factors) (CP) and how to compute it with the Alternating Least-Squares (ALS) algorithm (which is an *approximation* algorithm since computing the CP decomposition is in general an NP-Hard problem). The lecture finished with a series of facts about tensors, and compared them to matrices.

2 Tensor Network Notation

Before we begin, let us review a very simple and intuitive notation to represent vectors, matrices and tensors. Matrices, vectors and tensors can be represented in a intuitive notation known as *Tensor Networks*. This graphical language makes it easy to describe and pictorially reason about operation on tensors and, in quantum physics where it is quite popular, a system, quantum circuits, channels, protocols and more [1]. Figure 1 explains some basic building blocks with tensor networks and the corresponding operators.

3 Definitions

3.1 Tensor and its different views and reshaping

Definition 1 (Order-p Tensor)**.** *A tensor $\mathcal{T}$, is called order-p if: $\mathcal{T} \in \mathbb{R}^{d_1 \times d_2 \times \cdots \times d_p}$, where $\mathcal{T}_{i_1,i_2,\ldots,i_p} \in \mathbb{R}$ for each $i_1 \in [d_1], i_2 \in [d_2], \ldots, i_p \in [d_p]$.*

Remark: An order-0 tensor is a scalar, an order-1 tensor is a vector, an order-2 tensor is a matrix, and an order-3 tensor, $\mathcal{T} \in \mathbb{R}^{d_1 \times d_2 \times d_3}$, is a cube tensor.
Few more general definitions about tensors and their various forms are given below.

Definition 2 (Slices of a tensor)**.** *A* slice *of an order-3 tensor $\mathcal{T}$ is obtained by taking a slice in one direction along the cube. A slice is obtained by fixing one of the indices of a 3rd order tensor and letting the two others* free*. There are three ways of doing this for a 3rd order tensor, leading to the following three kinds of slices:*

$$\mathcal{T}_{j,:,:} \in \mathbb{R}^{d_2 \times d_3}, \mathcal{T}_{:,j,:} \in \mathbb{R}^{d_1 \times d_3}, \mathcal{T}_{:,:,j} \in \mathbb{R}^{d_1 \times d_2}$$

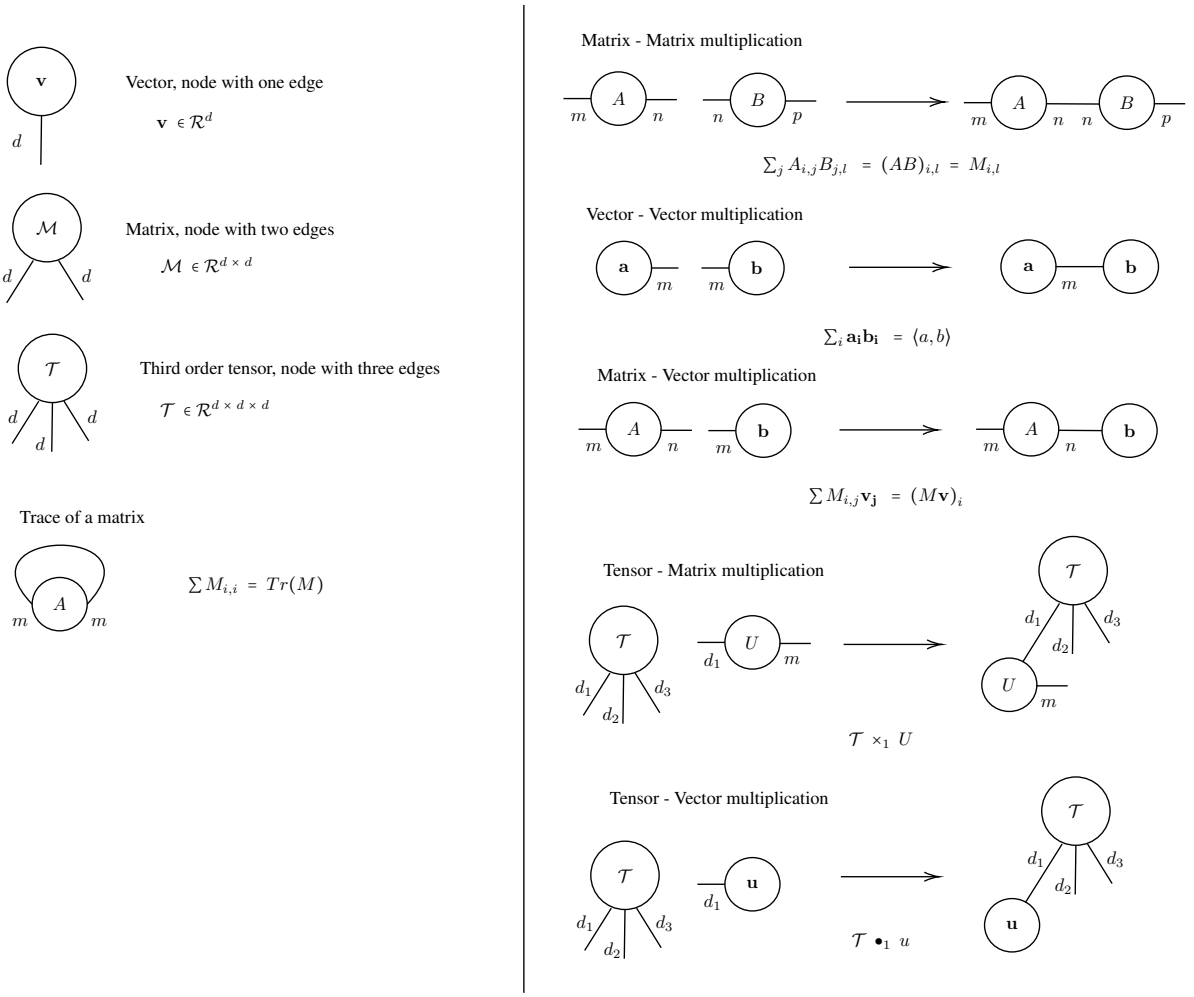

Figure 1: Basic Tensor Network notations

Definition 3 (Fibers of a tensor)**.** *Fixing all but one of the indices of a tensor gives a vector, which is called a* fiber *of the tensor. Again, there are three ways of doing this for a 3rd order tensor, leading to the following three kinds of fibers:*

$$\textit{mode-1 fiber}: \boldsymbol{\mathcal{T}}_{:,i,j} \in \mathbb{R}^{d_1}, \textit{mode-2 fiber}: \boldsymbol{\mathcal{T}}_{i,:,j} \in \mathbb{R}^{d_2}, \textit{mode-3 fiber}: \boldsymbol{\mathcal{T}}_{i,j,:} \in \mathbb{R}^{d_3}$$

As we saw in the definition of slices, we can fix one index of a 3rd order tensor to get a matrix, however that is only a part of the full tensor. We can represent the entire tensor as matrices by flattening it out using the process of *matricization*.

Definition 4 (Matricization)**.** *Given a tensor $\boldsymbol{\mathcal{T}}$, the process of* matricization *reshapes the tensor by flattening it into a matrix by taking all slices along one direction and stacking them together. For an order-3 tensor, we can have three modes of matricization based on which slices we stack together. Concretely,*

$$\boldsymbol{\mathcal{T}}_{(1)} \in \mathbb{R}^{d_1 \times d_2 d_3}, \boldsymbol{\mathcal{T}}_{(2)} \in \mathbb{R}^{d_2 \times d_1 d_3}, \boldsymbol{\mathcal{T}}_{(3)} \in \mathbb{R}^{d_3 \times d_1 d_2}$$

Definition 5 (Vectorization)**.** *Given a tensor $\boldsymbol{\mathcal{T}}$, a* vectorization *reshapes the tensor by flattening it into a vector. This is done by first matricization of the tensor along the first mode, and then stacking the columns of the resulting matrix to obtain a vector. Specifically,* $\operatorname{vec}(\boldsymbol{\mathcal{T}}) = \operatorname{vec}\left(\boldsymbol{\mathcal{T}}_{(1)}\right)$*, where the vectorization of a matrix is defined by:*

$$\operatorname{vec}\left(\begin{bmatrix} | & | & | & \dots & | \\ \mathbf{a}_1 & \mathbf{a}_2 & \mathbf{a}_3 & \dots & \mathbf{a}_n \\ | & | & | & \dots & | \end{bmatrix}\right) = \begin{bmatrix} | \\ \mathbf{a}_1 \\ | \\ \vdots \\ | \\ \mathbf{a}_n \\ | \end{bmatrix}$$

3.2 Products with Tensors

Like matrices, various products can be performed over tensors. Due to the higher order of dimensionality, several new products formulations will also be discussed.

Definition 6 (Inner Product and Norm)**.** *An inner product on a tensor is defined by taking the product between all entries of the two tensors and summing them up. Concretely,*

$$\begin{aligned} \langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}} \rangle &= \sum_{ijk} \boldsymbol{\mathcal{A}}_{ijk} \boldsymbol{\mathcal{B}}_{ijk} \\ &= \langle \operatorname{vec}(\boldsymbol{\mathcal{A}}), \operatorname{vec}(\boldsymbol{\mathcal{B}}) \rangle \end{aligned}$$

From this we can define the Frobenius norm of a tensor as:

$$\begin{aligned} \|\boldsymbol{\mathcal{A}}\|_F^2 &= \langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{A}} \rangle \\ &= \|\operatorname{vec}(\boldsymbol{\mathcal{A}})\|_2^2 \\ &= \|\boldsymbol{\mathcal{A}}_{(1)}\|_F^2 = \|\boldsymbol{\mathcal{A}}_{(2)}\|_F^2 = \|\boldsymbol{\mathcal{A}}_{(3)}\|_F^2 \end{aligned}$$

To multiply tensors with matrices and vectors, we use the *mode-n* product.

Definition 7 (mode-n product)**.** *Let $\boldsymbol{\mathcal{T}} \in \mathbb{R}^{d_1 \times d_2 \times d_3}$, and $\mathbf{U} \in \mathbb{R}^{m_1 \times d_1}$. The mode-1 product of $\boldsymbol{\mathcal{T}}$ with $\mathbf{U}$, denoted by $\boldsymbol{\mathcal{T}} \times_1 \mathbf{U} \in \mathbb{R}^{m_1 \times d_2 \times d_3}$, is defined by:*

$$(\boldsymbol{\mathcal{T}} \times_1 \mathbf{U})_{i_1, i_2, i_3} = \sum_{k=1}^{d_1} \boldsymbol{\mathcal{T}}_{k, i_2, i_3} \mathbf{U}_{i,k},$$

for all $i_1 \in [m_1], i_2 \in [d_2], i_3 \in [d_3]$.

To denote the multiplication of a tensor with a vector, we use the notation $\boldsymbol{\mathcal{T}} \bullet_1 \mathbf{v} = \boldsymbol{\mathcal{T}} \times_1 \mathbf{v}^\top \in \mathbb{R}^{d_2 \times d_3}$.

Observe that we can express the mode-1 product in term of the first matricization of the tensor $\mathcal{T}$:

$$\underbrace{(\mathcal{T} \times_1 \mathbf{U})_{(1)}}_{m_1 \times d_2 d_3} = \underbrace{\mathbf{U}}_{m_1 \times d_1} \underbrace{\boldsymbol{\mathcal{X}}_{(1)}}_{d_1 \times d_2 d_3}$$

Similarly, we can define the mode-n product for any n by the relation:

$$(\boldsymbol{\mathcal{X}} \times_{2n} \mathbf{V})_{(n)} = \mathbf{V}\boldsymbol{\mathcal{X}}_{(n)}$$

Remark 8. *If* $\mathbf{A}$ *is a matrix:*

$$\mathbf{A} \times_1 \mathbf{U} = \mathbf{U}\mathbf{A}$$
$$\mathbf{A} \times_2 \mathbf{V} = \mathbf{A}\mathbf{V}^\top$$

Proposition 9. *The mode-n product is associative:*

$$(\mathcal{T} \times_1 \mathbf{A}) \times_1 \mathbf{B} = \mathcal{T} \times_1 \mathbf{B}\mathbf{A}$$

3.3 More products with Tensors

Definition 10 (Outer-Product)**.** *The* outer product *(or tensor product), is defined as follows. Suppose we have three vectors (order-1 tensors)* $\mathbf{a} \in \mathbb{R}^{d_1}, \mathbf{b} \in \mathbb{R}^{d_2}$*, and* $\mathbf{c} \in \mathbb{R}^{d_3}$*. Their outer product is given by:*

$$\mathcal{T} = \mathbf{a} \circ \mathbf{b} \circ \mathbf{c} \in \mathbb{R}^{d_1 \times d_2 \times d_3}, \text{ where } \mathcal{T}_{i,j,k} = \mathbf{a}_i \mathbf{b}_j \mathbf{c}_k$$

Note: While performing outer product, we typically expand the order of the resulting tensor. For example, in the above example, the final tensor has three modes, each one resulting from one of the order-1 tensors. This generalizes to all kinds of outer products.

Remark: $\mathbf{a} \circ \mathbf{b} = \mathbf{a}\mathbf{b}^T$

Definition 11 (Kronecker Product)**.** *The Kronecker product operation can be seen as an outer product for matrices. For two matrices* $\mathbf{A}^{m \times n}$ *and* $\mathbf{B}^{p \times q}$*, the Kronecker product* $\mathbf{A} \otimes \mathbf{B} \in \mathbb{R}^{mp \times nq}$ *is defined by*

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & \cdots & a_{1n}\mathbf{B} \\ \vdots & \ddots & \vdots \\ a_{m1}\mathbf{B} & \cdots & a_{mn}\mathbf{B} \end{bmatrix}$$

Remark: $\underbrace{\mathbf{a}}_{\mathbb{R}^{m \times 1}} \otimes \underbrace{\mathbf{b}}_{\mathbb{R}^{n \times 1}} = \underbrace{\text{vec}\left(\mathbf{b}\mathbf{a}^T\right)}_{\mathbb{R}^{mn \times 1}} = \text{vec}\left(\mathbf{b} \circ \mathbf{a}\right)$

Properties of Kronecker Product:

1. Assuming compatible dimensions of the matrices, we have
$$(\mathbf{A} \otimes \mathbf{B})(\mathbf{C} \otimes \mathbf{D}) = \mathbf{A}\mathbf{C} \otimes \mathbf{B}\mathbf{D}$$
2. Note that matrix inversion of a Kronecker product can be sped up by using the following fact:
$$(\mathbf{A} \otimes \mathbf{B})^{-1} = \mathbf{A}^{-1} \otimes \mathbf{B}^{-1}$$
(Assuming $\mathbf{A}$ and $\mathbf{B}$ are invertible.)
3. Identity:
$$\text{vec}\left(\mathbf{A}\mathbf{X}\mathbf{B}\right) - (\mathbf{B}^\top \otimes \mathbf{A})\text{vec}\left(\mathbf{X}\right)$$
This identity is very useful to solve equations such as:
$$\mathbf{A}\mathbf{X} + \mathbf{X}\mathbf{B} = \mathbf{C}$$
which can be rewritten as
$$(\mathbf{I} \otimes \mathbf{A})\text{vec}\left(\mathbf{X}\right) + (\mathbf{B}^\top \otimes \mathbf{I})\text{vec}\left(\mathbf{X}\right) = \text{vec}\left(\mathbf{C}\right)$$

4. The Kronecker product allows us to conveniently express a series of n-mode products in the follwing way: if $\mathcal{X} = \mathcal{T} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}$, then

$$\begin{aligned}\mathcal{X}_{(1)} &= \mathbf{A}\mathcal{T}_{(1)}(\mathbf{C} \otimes \mathbf{B})^\top \\ \mathcal{X}_{(2)} &= \mathbf{B}\mathcal{T}_{(2)}(\mathbf{C} \otimes \mathbf{A})^\top \\ \mathcal{X}_{(3)} &= \mathbf{C}\mathcal{T}_{(3)}(\mathbf{B} \otimes \mathbf{A})^\top\end{aligned}$$

Definition 12 (Khatri-Rao Product)**.** *The Khatri-Rao product is defined as the "matching columnwise" Kronecker Product. If $\mathbf{A} \in \mathbb{R}^{m \times R}$, $\mathbf{B} \in \mathbb{R}^{n \times R}$, then Khatri Rao product denoted by $\mathbf{A} \odot \mathbf{B}$ is given by:*

$$\mathbf{A} \odot \mathbf{B} = \underbrace{\begin{pmatrix} | & | & \dots & | \\ \mathbf{a}_1 \otimes \mathbf{b}_1 & \mathbf{a}_2 \otimes \mathbf{b}_2 & \dots & \mathbf{a}_R \otimes \mathbf{b}_R \\ | & | & \dots & | \end{pmatrix}}_{\mathbb{R}^{mn \times R}}$$

Where:

$$\mathbf{A} = \begin{pmatrix} | & \dots & | \\ \mathbf{a}_1 & \dots & \mathbf{a}_R \\ | & \dots & | \end{pmatrix}_{m \times R}$$

and:

$$\mathbf{B} = \begin{pmatrix} | & \dots & | \\ \mathbf{b}_1 & \dots & \mathbf{b}_R \\ | & \dots & | \end{pmatrix}_{n \times R}$$

Note that $\mathbf{a}_i \otimes \mathbf{b}_i$ yields an mn-dimensional vector.

4 The CP Decomposition (CANDECOMP / PARAFAC)

The CP decomposition, or CANDECOMP (canonical decomposition) / PARAFAC (parallel factors) factorizes a tensor into a sum of outer products of vectors (aka a sum of rank one tensors).

Definition 13 (CP Decomposition)**.** *Let $\mathcal{T} \in \mathbb{R}^{d_1 \times d_2 \times d_3}$, then a CP decomposition factorizes $\mathcal{T}$ as a sum of rank one tensors:*

$$\begin{aligned}\mathcal{T} &= \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r \\ \text{for some } \mathbf{a}_1, \dots, \mathbf{a}_R &\in \mathbb{R}^{d_1} \\ \mathbf{b}_1, \dots, \mathbf{b}_R &\in \mathbb{R}^{d_2} \\ \mathbf{c}_1, \dots, \mathbf{c}_R &\in \mathbb{R}^{d_3}\end{aligned}$$

Elementwise, we have $\mathcal{T}_{i,j,k} = \sum_{r=1}^{R} (\mathbf{a}_i)_r (\mathbf{b}_j)_r (\mathbf{c}_k)_r$

Note: We will use the following shorthand notation for the CP decomposition:

$$\mathcal{T} = \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r = [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!]$$

where the three *factor matrices* $\mathbf{A}, \mathbf{B}$ and $\mathbf{C}$ such that:

$$\mathbf{A} = \underbrace{\begin{pmatrix} | & \dots & | \\ \mathbf{a}_1 & \dots & \mathbf{a}_R \\ | & \dots & | \end{pmatrix}}_{\mathbb{R}^{d_1 \times R}}, \quad \mathbf{B} = \underbrace{\begin{pmatrix} | & \dots & | \\ \mathbf{b}_1 & \dots & \mathbf{b}_R \\ | & \dots & | \end{pmatrix}}_{\mathbb{R}^{d_2 \times R}} \text{ and } \mathbf{C} = \underbrace{\begin{pmatrix} | & \dots & | \\ \mathbf{c}_1 & \dots & \mathbf{c}_R \\ | & \dots & | \end{pmatrix}}_{\mathbb{R}^{d_3 \times R}}$$

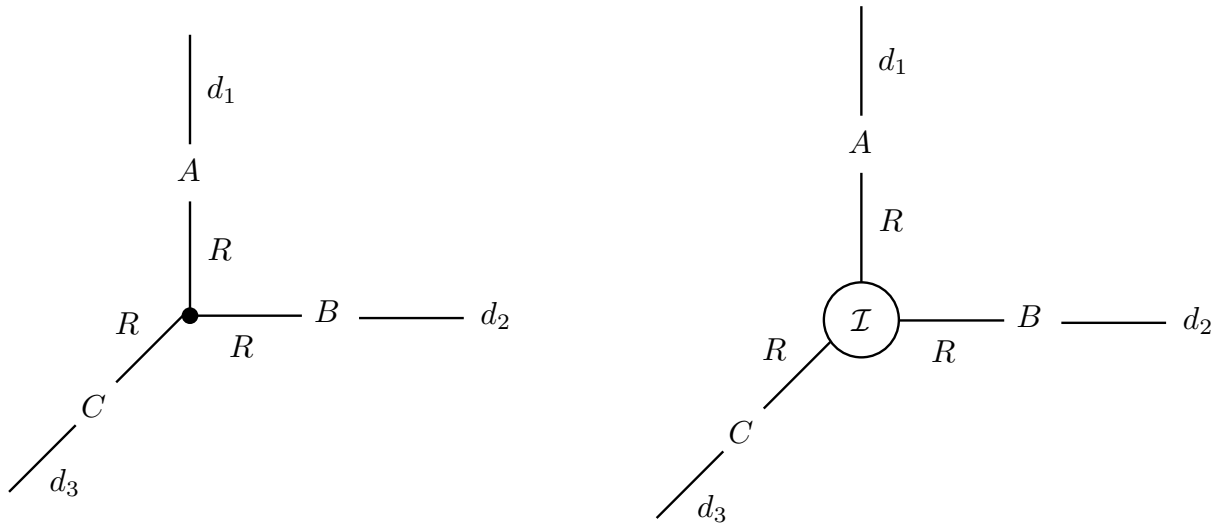

Figure 2: CP decomposition in Tensor Networks

4.1 Properties of CP decomposition:

1. Using the above definition of factor matrices, we can rewrite the matricizations of a CP decomposition in terms of the factor matrices. Recall, $\odot$ is the Khatri Rao product (Definition 12) discussed above.

$$\begin{aligned}&\text{If } \boldsymbol{\mathcal{T}} = [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!] \text{ then,}\\ &\boldsymbol{\mathcal{T}}_{(1)} = \mathbf{A}(\mathbf{C} \odot \mathbf{B})^\top\\ &\boldsymbol{\mathcal{T}}_{(2)} = \mathbf{B}(\mathbf{C} \odot \mathbf{A})^\top\\ &\boldsymbol{\mathcal{T}}_{(3)} = \mathbf{C}(\mathbf{B} \odot \mathbf{A})^\top\end{aligned}$$

2. The *rank* of a tensor $\boldsymbol{\mathcal{T}}$ is defined as the smallest number of rank-one tensors that generate $\boldsymbol{\mathcal{T}}$ as their sum. $Rank_{CP}(\boldsymbol{\mathcal{T}})$ is the smallest R such that $\boldsymbol{\mathcal{T}} = \sum_{n=1}^{R} a_n \circ b_n \circ c_n$ for some $a_1, ..., a_k$

Proposition 14. *For a general third-order tensor $\boldsymbol{\mathcal{T}}$, the following upper bound on its maximum rank holds:*

$$\text{rank}_{CP}(\boldsymbol{\mathcal{T}}) \leq \min\{d_1 d_2, d_2 d_3, d_1 d_3\}$$

Proof.

$$\begin{aligned}\boldsymbol{\mathcal{T}} &= \sum_{i=1}^{d_1}\sum_{j=1}^{d_2}\sum_{k=1}^{d_3} \boldsymbol{\mathcal{T}}_{ijk}\mathbf{e}_i \circ \mathbf{e}_j \circ \mathbf{e}_k\\ &= \sum_{i=1}^{d_1}\sum_{j=1}^{d_2} \mathbf{e}_i \circ \mathbf{e}_j \circ (\sum_{k=1}^{d_3} \boldsymbol{\mathcal{T}}_{ijk}\mathbf{e}_k)\\ \boldsymbol{\mathcal{T}}_{(3)} &= \sum_{i=1}^{d_1}\sum_{j=1}^{d_2} \underbrace{\text{vec}\,(\mathbf{e}_i \circ \mathbf{e}_j)}_{\mathbb{R}^{d_1 d_2 \times 1}} \circ \underbrace{(\sum_{k=1}^{d_3} \boldsymbol{\mathcal{T}}_{ijk}\mathbf{e}_k)}_{\mathbb{R}^{d3}}\end{aligned}$$

Thus, the rank is at most $d_1 d_2$; repeating this for the other two matricizations yields the other results, and the total rank of the tensor is at most the smallest of the three. This bound can be very loose . □

The same thing can be done with an $m \times n$ matrix M, to show that the largest its rank can be is the minimum of its number of rows, or number of columns. Mathematically this is: $\text{Rank}(M) \leq \min\{m, n\}$.

4.2 Computing the CP decomposition:

Given $\mathcal{T} \in \mathbb{R}^{d_1 \times d_2 \times d_3}$, and some target rank, R, finding the best R-rank approximation of $\mathcal{T}$ is an NP-Hard problem:

$$\arg\min_{\mathcal{X}:\text{Rank}_{CP}(x) \leq R} ||\mathcal{T} - \mathcal{X}||_F$$

For additional reading, we suggest Hillar and Lim [2], who show that most tensor problems are NP-Hard. Since for

$$\mathbf{A} \in \mathbb{R}^{d_1 \times R},\ \mathbf{B} \in \mathbb{R}^{d_2 \times R},\ \mathbf{C} \in \mathbb{R}^{d_3 \times R}$$

solving

$$\arg\min_{\mathbf{A} \in \mathbb{R}^{d_1 \times R}, \mathbf{B} \in \mathbb{R}^{d_2 \times R}, \mathbf{C} \in \mathbb{R}^{d_3 \times R}} ||\mathcal{T} - \mathcal{X}||_F$$

is a very difficult problem (it is not jointly convex in $\mathbf{A}, \mathbf{B}, \mathbf{C}$). To tackle this NP-Hard problem, we can instead use a heuristic approach such as the Alternating Least Square (ALS) algorithm. The ALS approach fixes $\mathbf{B}$ and $\mathbf{C}$ to solve for $\mathbf{A}$, and then fixes $\mathbf{A}$ and $\mathbf{C}$ to solve for $\mathbf{B}$, and then fixes $\mathbf{B}$ and $\mathbf{C}$ to solve for $\mathbf{A}$. It repeats this process until some convergence criterion is satisfied (Algorithm 4.2). When we fix all but one matrix, the problem reduces to a linear least square problem:

$$\arg\min_{\hat{\mathbf{A}}} ||\mathcal{T}_{(1)} - \hat{\mathbf{A}}(\mathbf{C} \odot \mathbf{B})^\top||_F$$

The optimal solution of which is given by:

$$\hat{\mathbf{A}} = \mathcal{T}_{(1)}(\underbrace{(\mathbf{C} \odot \mathbf{B})^\top}_{d_2 d_3 \times R})^\dagger$$

Proof. This follows directly from the fact that

$$\arg\min ||\mathbf{A}\mathbf{x} - \mathbf{b}|| = \mathbf{A}^\dagger \mathbf{b}$$

□

Algorithm 1 Alternating Least Squares (ALS)

Input: Tensor $\mathcal{T}$ and target rank R
Output: $\mathbf{A},\ \mathbf{B},\ \mathbf{C}$ each with R columns such that $\mathcal{T} \approx [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!]$
Output:
  Initialize $\mathbf{A},\ \mathbf{B},\ \mathbf{C}$ randomly
  repeat
    $\hat{\mathbf{A}} \leftarrow \arg\min_{\hat{\mathbf{A}}} ||\mathcal{T}_{(1)} - \hat{\mathbf{A}}(\mathbf{C} \odot \mathbf{B})^T||_F$,
    $\hat{\mathbf{B}} \leftarrow \arg\min_{\hat{\mathbf{B}}} ||\mathcal{T}_{(1)} - \hat{\mathbf{B}}(\mathbf{C} \odot \mathbf{A})^T||_F$,
    $\hat{\mathbf{C}} \leftarrow \arg\min_{\hat{\mathbf{C}}} ||\mathcal{T}_{(1)} - \hat{\mathbf{C}}(\mathbf{A} \odot \mathbf{B})^T||_F$,
  until < convergence >

4.3 Some Facts and Notes

- the rank of a tensor $\mathcal{T} \in \mathbb{R}^{d \times d \times d}$ can be larger than d. Of course, this is not the case with matrices.
- $\text{Rank}_{CP}(\mathcal{T})$ can be different over $\mathbb{R}$ and $\mathbb{C}$. This is once again not the case with matrices.
- No Eckart-Young Theorem.
- The best rank R approximation may not even exist: there exists a sequence of tensors, $\mathcal{X}_1, \mathcal{X}_2, \mathcal{X}_3 \ldots$ of rank 2 converging to a tensor of rank 3.
- The CP decomposition is often, but not always, unique.

References

[1] J. Biamonte and V. Bergholm. Tensor networks in a nutshell. *arXiv preprint arXiv:1708.00006*, 2017.

[2] C. J. Hillar and L. Lim. Most tensor problems are NP hard. *CoRR*, abs/0911.1393, 2009. URL `http://arxiv.org/abs/0911.1393`.