Homework 8
CS-526 Learning Theory

Note: The tensor product is denoted by $\otimes$. In other words, for vectors $\underline{a}, \underline{b}, \underline{c}$ we have that $\underline{a} \otimes \underline{b}$ is the square array $a^\alpha b^\beta$ where the superscript denotes the components, and $\underline{a} \otimes \underline{b} \otimes c$ is the cubic array $a^\alpha b^\beta c^\gamma$. We often denote components by superscripts because we need the lower index to label vectors themselves.

Problem 1: Kronecker and Khatri-Rao products

The *Kronecker product* $\otimes_{\text{Kro}}$ of two vectors $\underline{a} \in \mathbb{R}^{I_1}$ and $\underline{b} \in \mathbb{R}^{I_2}$ is a vectorization of the tensor (or outer) product. This amounts to take the $I_1 \times I_2$ array $a^\alpha b^\beta = (\underline{a} \otimes \underline{b})^{\alpha\beta}$ and view it as a vector of size $I_1 I_2$. More precisely, we define the Kronecker product as the column vector:

$$\underline{a} \otimes_{\text{Kro}} \underline{b} = \begin{bmatrix} a^1 \underline{b}^T & \cdots & a^{I_1} \underline{b}^T \end{bmatrix}^T \in \mathbb{R}^{I_1 I_2} .$$

Let $A = \begin{bmatrix} \underline{a}_1 & \cdots & \underline{a}_R \end{bmatrix}$ and $B = \begin{bmatrix} \underline{b}_1 & \cdots & \underline{b}_R \end{bmatrix}$ be matrices of dimensions $I_1 \times R$ and $I_2 \times R$. We define the *Khatri-Rao* product as the $I_1 I_2 \times R$ matrix

$$A \odot_{\text{KhR}} B = \begin{bmatrix} \underline{a}_1 \otimes_{\text{Kro}} \underline{b}_1 & \cdots & \underline{a}_R \otimes_{\text{Kro}} \underline{b}_R \end{bmatrix} .$$

1) Assume that both A and B are full column rank. Prove that the Khatri-Rao product $A \odot_{\text{KhR}} B$ is also full column rank.

2) Explain in detail in which step of Jennrich's algorithm this fact is used (see Figure 1).

Problem 2: Jennrich's type algorithm for order 4 tensors

Consider an order four tensor

$$T = \sum_{r=1}^{R} \underline{a}_r \otimes \underline{b}_r \otimes \underline{c}_r \otimes \underline{d}_r$$

where $A = \begin{bmatrix} \underline{a}_1 & \cdots & \underline{a}_R \end{bmatrix} \in \mathbb{R}^{I_1 \times R}$, $B = \begin{bmatrix} \underline{b}_1 & \cdots & \underline{b}_R \end{bmatrix} \in \mathbb{R}^{I_2 \times R}$, $C = \begin{bmatrix} \underline{c}_1 & \cdots & \underline{c}_R \end{bmatrix} \in \mathbb{R}^{I_3 \times R}$ and $D = \begin{bmatrix} \underline{d}_1 & \cdots & \underline{d}_R \end{bmatrix} \in \mathbb{R}^{I_4 \times R}$ are full column rank.

1) Check that you can apply Jennrich's algorithm to a "flattened" version of T, namely the order three tensor

$$\widetilde{T} = \sum_{r=1}^{R} \underline{a}_r \otimes \underline{b}_r \otimes (\underline{c}_r \otimes_{\text{Kro}} \underline{d}_r) .$$

where $\otimes_{\text{Kro}}$ is the Kronecker product defined in the previous question.

2) Deduce that the rank R as well as the matrices A, B, C, D can be uniquely determined from the four-dimensional array of numbers $T^{\alpha\beta\gamma\delta}$ (up to trivial rank permutation and feature scaling).

4.1.1 Jennrich's Algorithm. If $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{C}$ are all linearly independent (i.e. have full rank), then $\mathcal{X} = \sum_{r=1}^{R} \lambda_r \boldsymbol{a}_r \circledcirc \boldsymbol{b}_r \circledcirc \boldsymbol{c}_r$ is unique up to trivial rank permutation and feature scaling and we can use Jennrich's algorithm to recover the factor matrices [23, 24]. The algorithm works as follows:

(1) Choose random vectors $\boldsymbol{x}$ and $\boldsymbol{y}$.

(2) Take a slice through the tensor by hitting the tensor with the random vector $\boldsymbol{x}$:
$\mathcal{X}(\boldsymbol{I}, \boldsymbol{I}, \boldsymbol{x}) = \sum_{r=1}^{R} \langle \boldsymbol{c}_r, \boldsymbol{x} \rangle \boldsymbol{a}_r \circledcirc \boldsymbol{b}_r = A\mathrm{Diag}(\langle \boldsymbol{c}_r, \boldsymbol{x} \rangle)\boldsymbol{B}^T$.

(3) Take a second slice through the tensor by hitting the tensor with the random vector $\boldsymbol{y}$:
$\mathcal{X}(\boldsymbol{I}, \boldsymbol{I}, \boldsymbol{y}) = \sum_{r=1}^{R} \langle \boldsymbol{c}_r, \boldsymbol{y} \rangle \boldsymbol{a}_r \circledcirc \boldsymbol{b}_r = A\mathrm{Diag}(\langle \boldsymbol{c}_r, \boldsymbol{y} \rangle)\boldsymbol{B}^T$.

(4) Compute eigendecomposition to find $\boldsymbol{A}$:
$\mathcal{X}(\boldsymbol{I}, \boldsymbol{I}, \boldsymbol{x})\, \mathcal{X}(\boldsymbol{I}, \boldsymbol{I}, \boldsymbol{y})^\dagger = A\mathrm{Diag}(\langle \boldsymbol{c}_r, \boldsymbol{x} \rangle)\mathrm{Diag}(\langle \boldsymbol{c}_r, \boldsymbol{y} \rangle)^\dagger \boldsymbol{A}^\dagger$

(5) Compute eigendecomposition to find $\boldsymbol{B}$:
$\mathcal{X}(\boldsymbol{I}, \boldsymbol{I}, \boldsymbol{x})^\dagger\, \mathcal{X}(\boldsymbol{I}, \boldsymbol{I}, \boldsymbol{y}) = (B^T)^\dagger \mathrm{Diag}(\langle \boldsymbol{c}_r, \boldsymbol{x} \rangle)^\dagger \mathrm{Diag}(\langle \boldsymbol{c}_r, \boldsymbol{y} \rangle)\boldsymbol{B}^T$

(6) Pair up the factors and solve a linear system to find $\boldsymbol{C}$.

Figure 1: Jennrich's algorithm (from *Introduction to Tensor Decompositions and their Applications in Machine Learning Review,Rabanser, Shchur, Gunnemann*)