Statistical Learning and Data Science

Xinwei Deng

xdeng@vt.edu

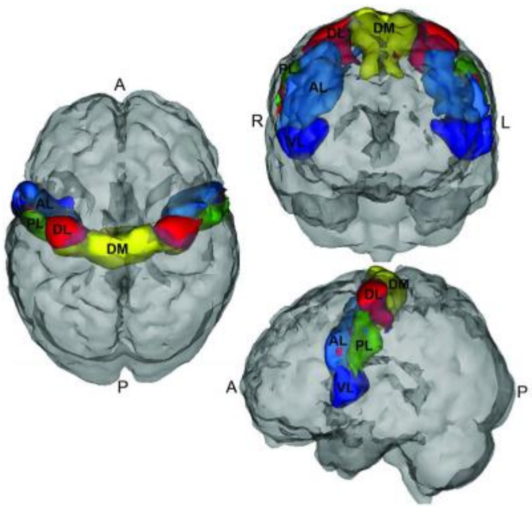
Department of Statistics

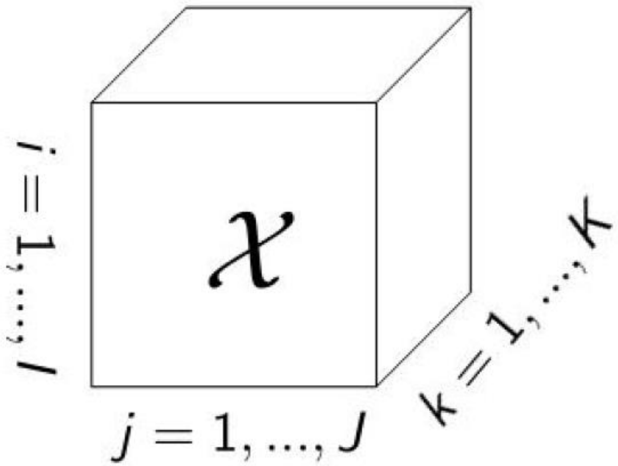
Tensor Regression and Beyond

- Introduction to Tensor Data.
- Tensor and Its Decomposition.
- Tensor Regression
- Modeling and Analysis of Tensor Data
- Future Direction

Tensor:
A Generalization of Matrix

- Tensor: a *multi-way* array.
- Example: fMRI data in Neuroscience: 3D brain images.

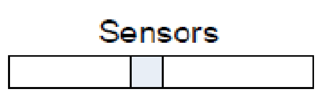
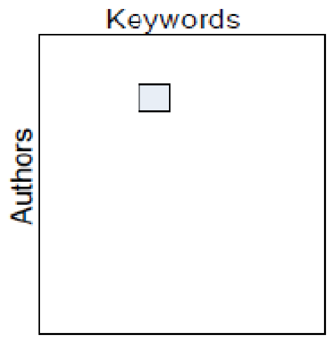
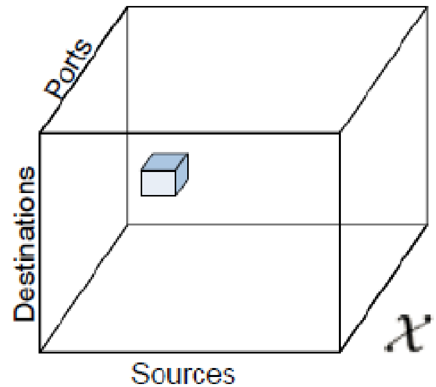




What is Tensor?

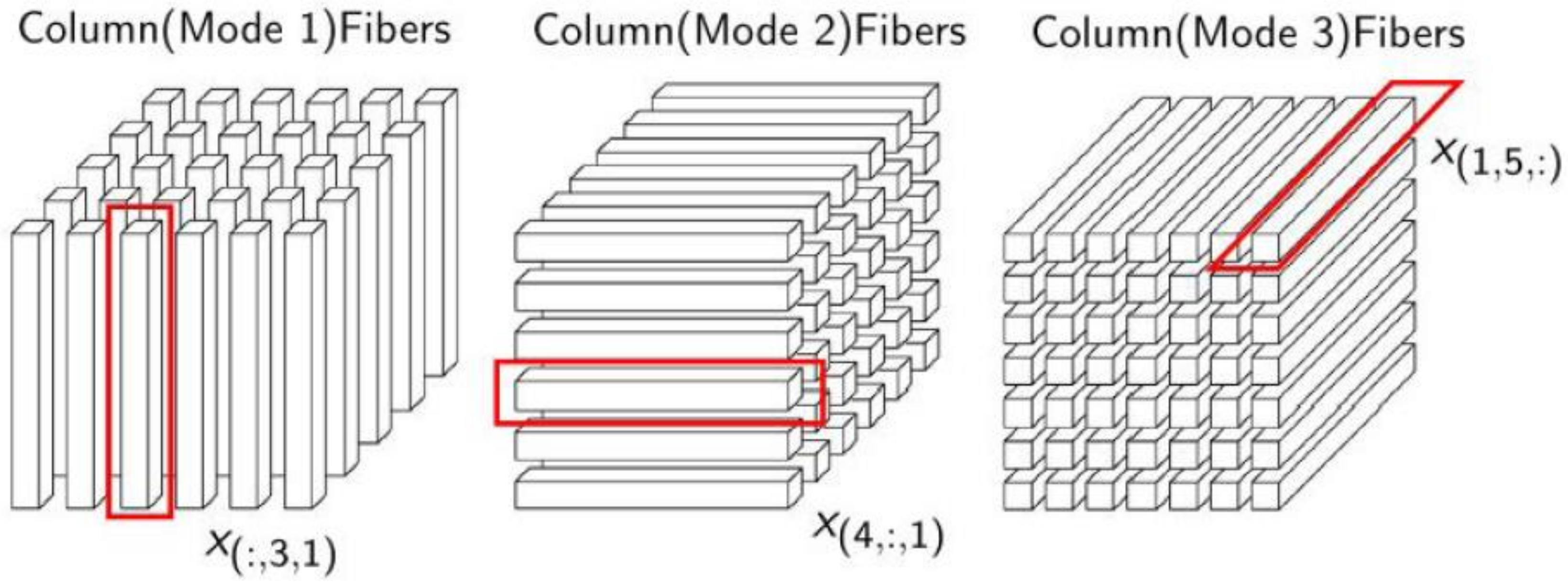
A tensor is formally denoted as $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$

- generalization of vector and matrix
- represented as multi-dimensional array

Order	1st	2^{nd}	3^{rd}
Correspondence	Vector	Matrix	3D array
Example			

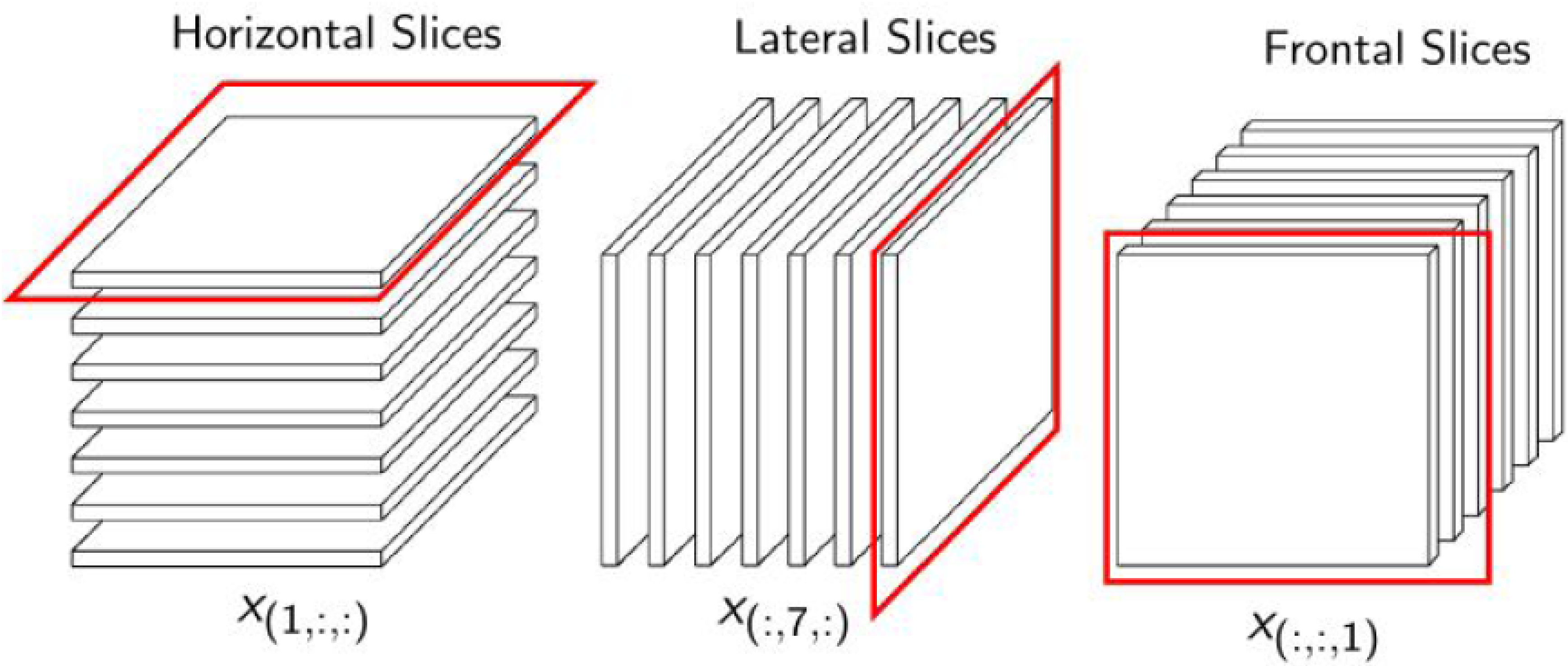
Definition of Fiber

- Fibers are created when fixing all but one index:



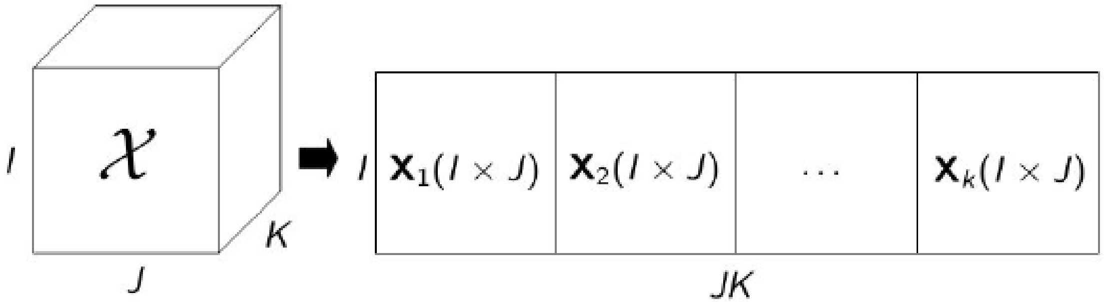
Definition of Slice

- Slices (or slabs) are created when fixing all but two indices.

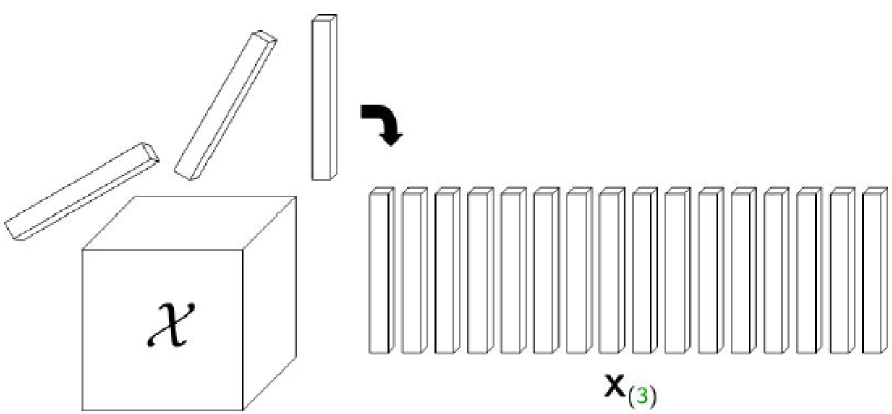


Matricization based on Slices

- Vectorization is to reorder a tensor into a vector.
- Matricization is to reorder a tensor into a matrix.



- One can also think to rearrange the fibers into the columns of a matrix.



The h-Mode Multiplication

Let $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$, $\mathbf{B} \in \mathbb{R}^{M \times J}$, the 2-mode product of $\mathcal{X}$ with $\mathbf{B}$ is defined by

$$\mathcal{Y} = \mathcal{X} \times_2 \mathbf{B} \in \mathbb{R}^{I \times M \times K}$$

Elementwise

$$y_{imk} = \sum_j x_{ijk} b_{mj}$$

In matrix form

$$\mathbf{Y}_{(2)} = \mathbf{B}\mathbf{X}_{(2)}$$

Multiply each row (mode-2) fiber by **B**

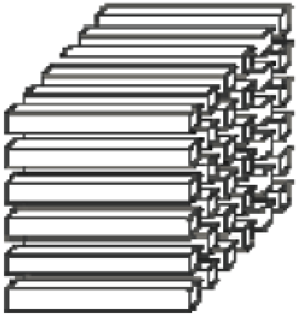
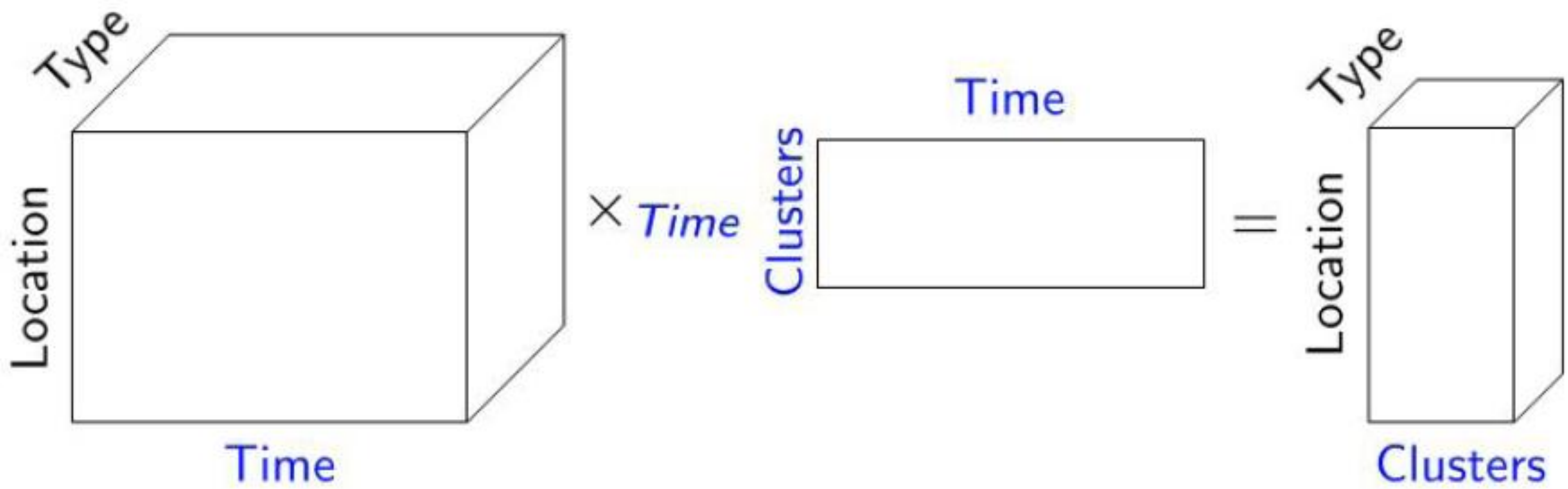


Illustration of h-Mode Multiplication



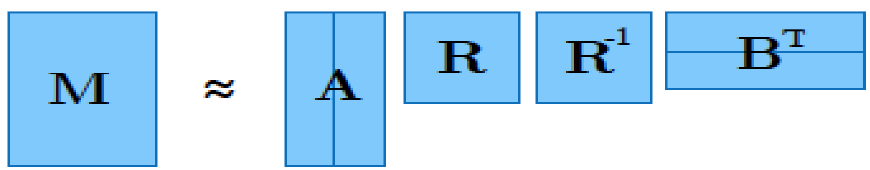
Matrix Decomposition

- Rank decomposition of a matrix

$$M = AB^T \quad \text{with} \quad M \in \mathbb{R}^{n \times m},\ A \in \mathbb{R}^{n \times r},\ B^T \in \mathbb{R}^{r \times m}$$

where r represents the rank of the decomposition.

- The rotation problem



- The optimization does not have a unique solution.

$$\min_{\hat{M}} ||M - \hat{M}|| \quad \text{with} \quad \hat{M} = AB^T$$

- Need certain conditions, such as orthogonality, to make matrix decompositions unique.

Tensor and Its Decomposition

- Tensor is a multi-way array: An order-k tensor can be expressed as

$$\boldsymbol{X} = \left(x_{i_1,i_2,..i_k}\right) \in R^{I_1 \times I_2 \times \cdots \times I_k}.$$

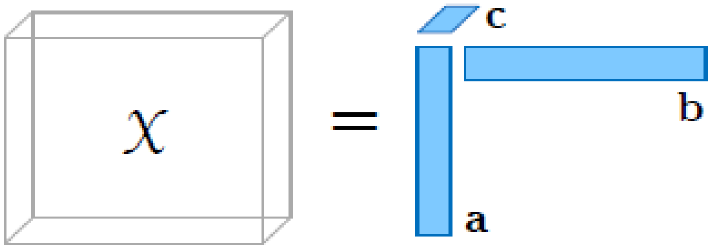
- Inner product vs outer product

$$<a, b> = a^T b \qquad \text{vs} \qquad a \circledcirc b = ab^T$$

- Rank-one Tensor:

$$\boldsymbol{Z} = \boldsymbol{a}^{(1)} \circledcirc \boldsymbol{a}^{(2)} \circledcirc \cdots \circledcirc \boldsymbol{a}^{(k)}$$

$$= \left(a_{i_1}^{(1)} a_{i_2}^{(2)} \ldots a_{i_k}^{(k)}\right)$$



- Tensor rank: The rank of tensor $\boldsymbol{X}$ is defined as the minimum number of rank-one tensors needed to produce $\boldsymbol{X}$ as their sum.

Tensor and Its Decomposition (Con't)

- Thus a tensor $\boldsymbol{X}$ of rank-L can be expressed as

$$\mathcal{X} = \sum_{r=1}^{L} \lambda_r \boldsymbol{a}_r^{(1)} \circledcirc \boldsymbol{a}_r^{(2)} \circledcirc \cdots \circledcirc \boldsymbol{a}_r^{(k)}$$

$$= [\![\boldsymbol{\lambda}; \boldsymbol{A}^{(1)}, \boldsymbol{A}^{(2)}, \cdots, \boldsymbol{A}^{(k)}]\!]$$

  where $A^{(j)} = [\boldsymbol{a}_1^{(j)}, \ldots, \boldsymbol{a}_L^{(j)}]$ is called the *factor matrix*.
- Tensor Decomposition: generalizing the idea of SVD.
  - Canonical Polyadic Decomposition (CPD)
  - Tucker Decomposition (TD)

Norm for Tensors

- An easy norm: *Hilbert–Schmidt norm*, defined as

$$\|A\| = \sqrt{\langle A, A\rangle} = \left(\sum_{i_1,\ldots,i_d=1}^{n_1,\ldots,n_d} |a_{i_1\cdots i_d}|^2\right)^{\frac{1}{2}}.$$

which can be viewed as an extension of Frobenius norm.

- A *spectral norm*, is defined as

$$\|A\|_{\sigma,\mathbb{F}} := \sup\left\{ \frac{|<A, x_1 \odot \ldots \odot x_d>|}{\|x_1\|\cdots\|x_d\|} : 0 \neq x_k \in \mathbb{F}^{n_k} \right\}$$

$$= \sup\{ \quad |<A, u_1 \odot \ldots \odot u_d>| \quad : \|u_k\| = 1\}$$

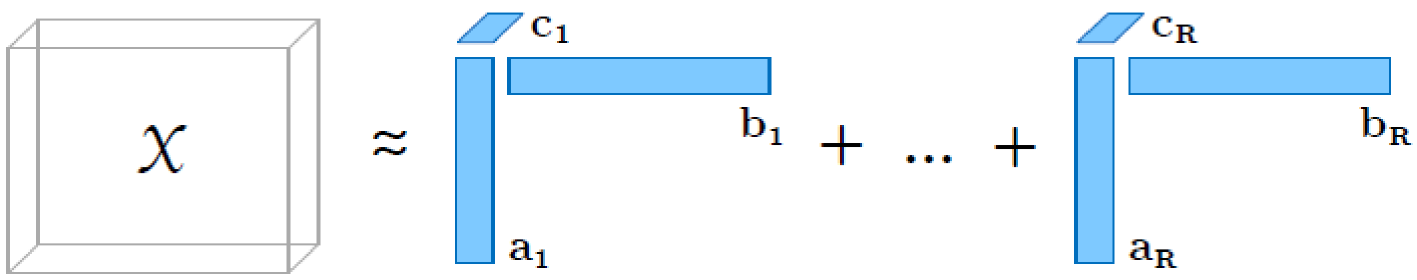
- A singular-value norm: *nuclear norm*, is defined as

$$\|A\|_{*,\mathbb{F}} = \inf\left\{\sum_{i=1}^{r} |\lambda_i| : A = \sum_{i=1}^{r} \lambda_i \boldsymbol{u}_i^{(1)} \odot \ldots \odot \boldsymbol{u}_i^{(d)}, \ ||\boldsymbol{u}_i^{(j)}|| = 1, \quad r \in \mathbb{N}\right\}$$

$$= \inf\left\{\sum_{i=1}^{r} \ ||\boldsymbol{u}_1||\ldots||\boldsymbol{u}_d|| \qquad : A = \sum_{i=1}^{r} \boldsymbol{u}_i^{(1)} \odot \ldots \odot \boldsymbol{u}_i^{(d)} \ \right\}$$

Canonical Polyadic Decomposition (CPD)

- The CPD is rank decomposition, which is to express a tensor as the sum of a finite number of rank-one tensors.

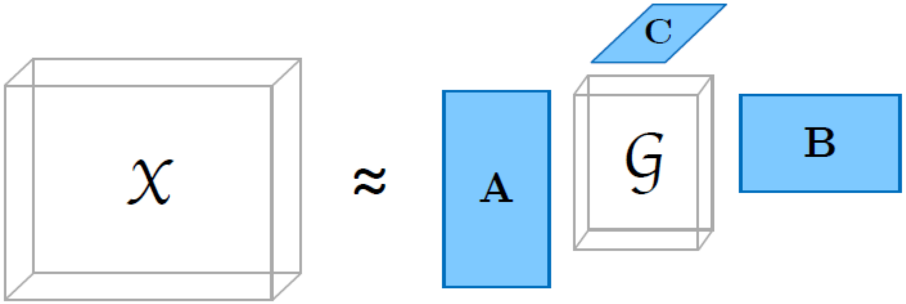


- A order-3 tensor CPD can be formulated as

$$\min_{\hat{\mathcal{X}}} ||\mathcal{X} - \hat{\mathcal{X}}|| \quad \text{where} \quad \hat{\mathcal{X}} = \sum_{r=1}^{R} \boldsymbol{a}_r \circledcirc \boldsymbol{b}_r \circledcirc \boldsymbol{c}_r = [\![A, B, C]\!]$$

Tucker Decomposition (TD)

- Tucker decomposition
  - Decomposes a tensor into a core tensor and multiple matrices which correspond to different core scalings along each mode.
  - Tucker decomposition can be seen as a higher-order PCA.



- A order-3 tensor decomposition can be formulated as

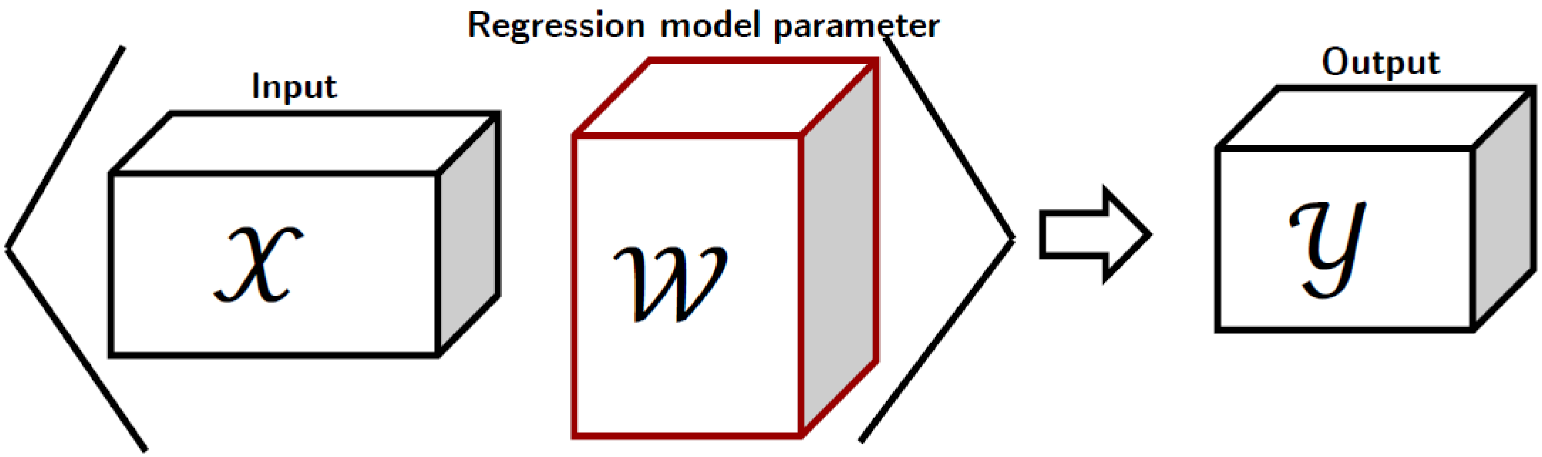
$$\min_{\hat{\mathcal{X}}} \|\mathcal{X} - \hat{\mathcal{X}}\| \quad \text{with} \quad \hat{\mathcal{X}} = \sum_{p=1}^{P}\sum_{q=1}^{Q}\sum_{r=1}^{R} g_{pqr} \; \boldsymbol{a}_p \odot \boldsymbol{b}_q \odot \boldsymbol{c}_r$$

$$= \mathcal{G} \times_1 A \times_2 B \times_3 C$$

$$= [\![\mathcal{G}; A, B, C]\!]$$

Tensor Regression

- Tensor Regression: large-scale supervised learning from multi-way data
- Goal: learn a regression model with multi-linear parameters

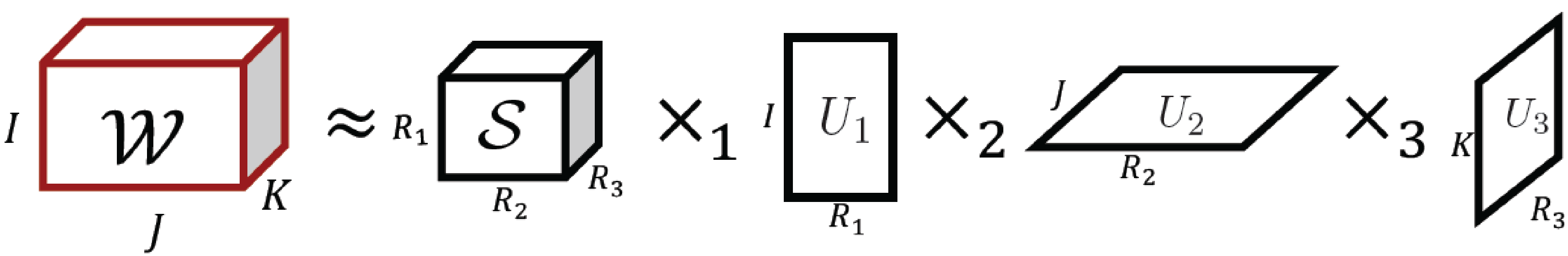


Low-Rank Representation

- Low-rank structures can capture multi-linear correlations.

Collaborative Filtering

- Tucker decomposition: high-order SVD



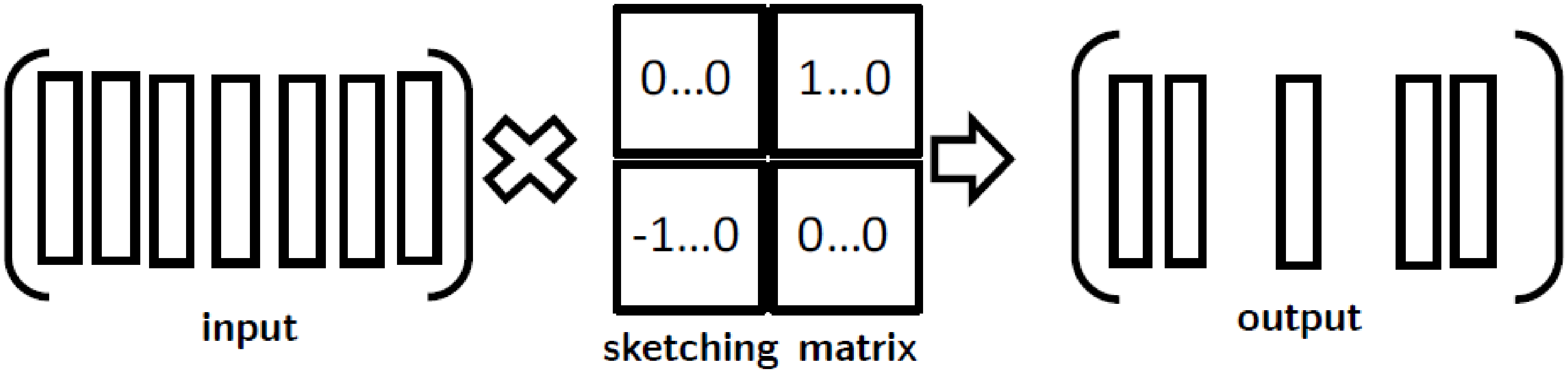
Low-Rank Tensor Regression

- Predictor tensor $\mathcal{X}$; response tensor $\mathcal{Y}$
- Regression model $\langle \mathcal{X}, \mathcal{W} \rangle$: e.g. $\sum_{m=1}^{M} \mathcal{X}_{:,:,m} \mathcal{W}_{:,:,m}$
- Loss function $\mathcal{L}(\hat{\mathcal{Y}}; \mathcal{Y})$: e.g. $\|\hat{\mathcal{Y}} - \mathcal{Y}\|_F^2$
- Goal: Learn a parameter tensor $\mathcal{W}$ with low-rank constraint

$$\mathcal{W}^{\star} = \operatorname{argmin}_{\mathcal{W}} \hat{\mathcal{L}}(f(\mathcal{X}, \mathcal{W}); \mathcal{Y})$$
$$\text{s.t.} \quad \operatorname{rank}(\mathcal{W}) \leq R$$

Subsampled Tensor Projected Gradient (TPG)

- Data $\Rightarrow$ Random sketching [Woodruff 2014]
- Model $\Rightarrow$ Iterative hard thresholding [Thomas and Davies 2009]



- Projected gradient descent: $\mathcal{W}^{k+1} = P_R\left(\mathcal{W}^k - \eta \nabla \mathcal{W}^k\right)$
  1. Gradient descent step
  2. Low-rank projection step

Brief on Random Sketching

- Let us consider a least squares estimation problem,

$$\beta_{OLS} = \arg\min_{\beta \in \mathbb{R}^p} \|Y - X\beta\|_2^2$$

where Y is response vector and X is n-by-p regression matrix.

- Suppose that there is a r-by-n sketching matrix S. Then the sketched problem for parameter estimation

$$\beta_S \in \arg\min_{\beta \in \mathbb{R}^p} \|SY - SX\beta\|_2^2.$$

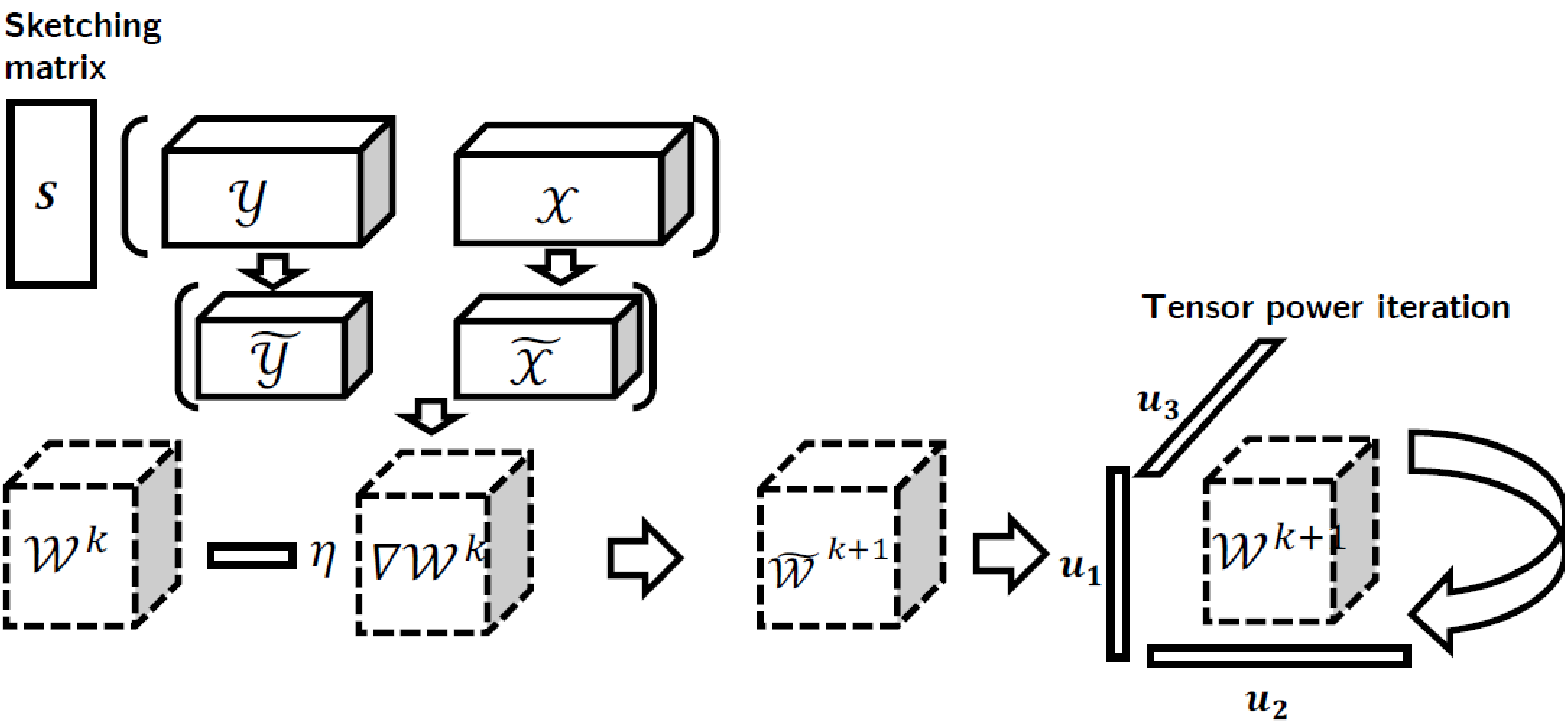
- It is shown that in Drineas et al. (2011, 2012), for any arbitrary (X; Y),

$$\|Y - X\beta_S\|_2^2 \leq (1+\kappa)\|Y - X\beta_{OLS}\|_2^2,$$

with high probability for some pre-specified error parameter $\kappa \in (0,1)$.

Subsampled Tensor Projected Gradient (TPG)

- Random sketching as data subsampling
- Iterative hard thresholding as dimensional reduction



Example: GLM

The standard linear regression model $\mathbf{x} \in \mathbb{R}^p$, $y = \beta^T \mathbf{x} + \alpha + \varepsilon$, $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ can be written

$$\mu = \beta^T \mathbf{x} + \alpha \qquad y \sim \mathcal{N}(\mu, \sigma^2)$$

where $\mu = \mathbb{E}(Y|\mathbf{x})$

A **generalized linear regression model** (**GLM**) extends this to

$$g(\mu) = \beta^T \mathbf{x} + \alpha \qquad y \sim \mathcal{EF}(\mu, \phi)$$

- $\mathcal{EF}(\mu, \phi)$ is any exponential family distribution (e.g. Normal, Poisson, Binomial)
- $g(\cdot)$ is any smooth monotonic link function
- $\beta^T \mathbf{x} + \alpha (= \eta)$ is the linear predictor

Example: GLM with Matrix Predictor

In classical **GLM** Y belongs to an exponential family with **PMF**

$$p(y|\theta,\phi) = \exp\{\frac{y\theta - b(\theta)}{a(\phi)} + c(y,\phi)\}$$

The **GLM** relates $\mathbf{x} \in \mathbb{R}^p$ to the mean $\mu = \mathbb{E}(Y|\mathbf{x})$ by

$$g(\mu) = \eta = \alpha + \beta^T \mathbf{x}$$

The **GLM** for the matrix predictor $\mathbf{X}$ given by

$$g(\mu) = \eta = \alpha + \gamma^T \mathbf{z} + \beta_1^T \mathbf{X} \beta_2$$

Example: GLM with Tensor Predictor

The **GLM** with the systematic part for tensor predictor given by

$$g(\mu) = \eta = \alpha + \gamma^T \mathbf{z} + < \mathcal{B}, \mathcal{X} >$$

- D-dimensional tensor predictor $\mathcal{X} \in \mathbb{R}^{p_1 \times \cdots \times p_D}$
- D-dimensional coefficient tensor $\mathcal{B} \in \mathbb{R}^{p_1 \times \cdots \times p_D}$
- $\mathcal{B}$ has $\prod_{d=1}^{D} p_d$ parameters, which is ultrahigh dimensional and far exceeds sample size

Example: GLM with Tensor Predictor (Con't)

- Univariate outcome Y belongs to exponential family
- Tensor covariate $\mathcal{X} \in \mathbb{R}^{p_1 \times \cdots \times p_D}$
- Assume coefficient tensor $\mathcal{B}$ has a rank-R decomposition $[\mathbf{B}_1, \ldots, \mathbf{B}_D]$ where $\mathbf{B}_d \in \mathbb{R}^{p_d \times R}$

Generalized linear CP tensor regression model (Zhou et al. 2013) with the systematic part given by

$$g(\mu) = \eta = \alpha + \gamma^T \mathbf{z} + < \sum_{r=1}^{R} \beta_1^{(r)} \circ \cdots \circ \beta_D^{(r)}, \mathcal{X} >$$

$$= \alpha + \gamma^T \mathbf{z} + < (\mathbf{B}_D \odot \cdots \odot \mathbf{B}_1)\mathbf{1}_R, vec(\mathcal{X}) >$$

- substantial reduction in dimensionality to the scale of $R \times \sum_{d=1}^{D} p_d$

Parameter Estimation via Regularization

Maximize a regularized log-likelihood function

$$\ell(\alpha, \gamma, \mathbf{B}_1, \ldots, \mathbf{B}_D) - \sum_{d=1}^{D} \sum_{r=1}^{R} \sum_{i=1}^{p_d} P_\lambda(|\beta_{di}^{(r)}|, \rho)$$

- scalar penalty function $P_\lambda(|\beta|, \rho)$
- **power family** $P_\lambda(|x|, \rho) = \rho|\beta|^\lambda$, $\lambda \in (0, 2]$
- in particular lasso ($\lambda = 1$)

Comments

Thank You!