Multi-linear Algebra
– Sketching Tucker Decomposition –
Lecture 17

F. M. Faulstich

26/03/2024

Recall

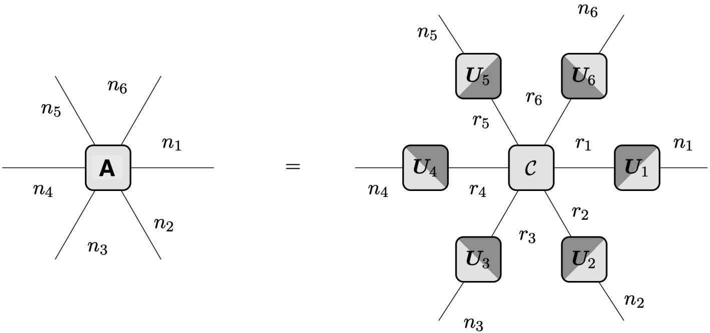
Recall

- Tucker decomposition:

Recall

- Tucker decomposition:
  Let $\mathbf{A} \in \mathbb{R}^{n_1 \times \ldots \times n_d}$. Then

$$\mathbf{A} = \sum_{i_1=1}^{r_1} \cdots \sum_{i_d=1}^{r_d} \mathbf{C}[i_1, \ldots, i_d] \cdot \mathbf{u}_{1,i_1} \otimes \mathbf{u}_{2,i_2} \otimes \cdots \otimes \mathbf{u}_{d,i_1}$$
$$= \mathbf{C} *_1 \mathbf{U}_1 *_2 \mathbf{U}_2 \ldots *_d \mathbf{U}_d$$



STHOSVD

Sequential **T**runcated **H**igher **O**rder SVD (ST-HOSVD)

Algorithm:

Input: Target tensor $\mathbf{A} \in \mathbb{R}^{n_1 \times \dots \times n_d}$, target rank $r = (r_1, ..., r_d)$

Output: Core tensor $\mathbf{C}$, basis matrices $\mathbf{U}_k \in \mathbb{R}^{r_k \times n_k}$ for $1 \leq k \leq d$

Set $\mathbf{A}_0 = \mathbf{A}$

for $k = 1 : d$

$\tilde{\mathbf{U}}_k, \mathbf{\Sigma}_k, \mathbf{V}_k = \mathrm{SVD}(\mathbf{A}_{k-1}^{(k)}, r_k)$

$\mathbf{A}_k^{(k)} = \mathbf{\Sigma}_k \mathbf{V}_k^\top$

$\mathbf{U}_k = \tilde{\mathbf{U}}_k^\top$

$\mathbf{C} = \mathbf{A}_d$

R-SVD

Input: $\mathbf{A} \in \mathbb{R}^{m \times n}$, $r \in \mathbb{N}$, $p \in \mathbb{N}$

Output: Low-rank approximation of $\mathbf{A}$: $\hat{\mathbf{A}} = \hat{\mathbf{U}}\hat{\boldsymbol{\Sigma}}\hat{\mathbf{V}}^\top$

$\boldsymbol{\Omega} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \in \mathbb{R}^{n \times (r+p)}$

$\mathbf{Y} = \mathbf{A}\boldsymbol{\Omega}$

$\mathbf{Q} = \mathrm{ortho}(\mathbf{Y})$ [QR-decomp.]

$\mathbf{B} = \mathbf{Q}^\top \mathbf{A}$

$\hat{\mathbf{U}}\hat{\boldsymbol{\Sigma}}\hat{\mathbf{V}}^\top = \mathrm{SVD}(\mathbf{B}, r)$

Application of R-SVD in STHOSVD

Let's apply R-SVD instead of regular SVD!

R. Minster, A. K. Saibaba, M. E. Kilmer: Randomized algorithms for low-rank tensor decompositions in the Tucker format. SIAM Journal on Mathematics of Data Science. 2(1), 189-215 (2020)

R-STHOSVD

Input: Target tensor $\mathbf{A}$, target rank $(r_1, ..., r_d)$, $p \in \mathbb{N}$

Output: Core tensor $\mathbf{C}$, basis matrices $\mathbf{U}_k \in \mathbb{R}^{r_k \times n_k}$ for $1 \leq k \leq d$

Set $\mathbf{A}_0 = \mathbf{A}$

for $k = 1 : d$

$\quad \tilde{\mathbf{U}}_k,\ \mathbf{\Sigma}_k,\ \mathbf{V}_k = \text{ R-SVD}(\mathbf{A}_{k-1}^{(k)}, r_k, p)$

$\quad \mathbf{A}_k^{(k)} = \mathbf{\Sigma}_k \mathbf{V}_k^\top$

$\quad \mathbf{U}_k = \tilde{\mathbf{U}}_k^\top$

$\mathbf{C} = \mathbf{A}_d$

[1] Tropp, et al. SIAM Journal on Matrix Analysis Applications. 2017

R-STHOSVD

Input: Target tensor $\mathbf{A}$, target rank $(r_1, ..., r_d)$, $p \in \mathbb{N}$

Output: Core tensor $\mathbf{C}$, basis matrices $\mathbf{U}_k \in \mathbb{R}^{r_k \times n_k}$ for $1 \leq k \leq d$

Set $\mathbf{A}_0 = \mathbf{A}$

for $k = 1 : d$

$\quad \tilde{\mathbf{U}}_k,\ \mathbf{\Sigma}_k,\ \mathbf{V}_k =$ R-SVD$(\mathbf{A}_{k-1}^{(k)}, r_k, p)$

$\quad \mathbf{A}_k^{(k)} = \mathbf{\Sigma}_k \mathbf{V}_k^\top$

$\quad \mathbf{U}_k = \tilde{\mathbf{U}}_k^\top$

$\mathbf{C} = \mathbf{A}_d$

- What if both matrix dimensions are large?

[1]Tropp, et al. SIAM Journal on Matrix Analysis Applications. 2017

R-STHOSVD

Input: Target tensor $\mathbf{A}$, target rank $(r_1, ..., r_d)$, $p \in \mathbb{N}$

Output: Core tensor $\mathbf{C}$, basis matrices $\mathbf{U}_k \in \mathbb{R}^{r_k \times n_k}$ for $1 \leq k \leq d$

Set $\mathbf{A}_0 = \mathbf{A}$

for $k = 1 : d$

$\quad \tilde{\mathbf{U}}_k,\ \mathbf{\Sigma}_k,\ \mathbf{V}_k =$ R-SVD$(\mathbf{A}_{k-1}^{(k)}, r_k, p)$

$\quad \mathbf{A}_k^{(k)} = \mathbf{\Sigma}_k \mathbf{V}_k^\top$

$\quad \mathbf{U}_k = \tilde{\mathbf{U}}_k^\top$

$\mathbf{C} = \mathbf{A}_d$

- What if both matrix dimensions are large?
  → Two-sided sketching[1]!

[1] Tropp, et al. SIAM Journal on Matrix Analysis Applications. 2017

Low-rank approximation – two-sided sketch

- Choose sketching parameters $r \leq k \leq \min(l, n)$, $0 < l \leq m$

Algorithm:

Input: $\mathbf{A} \in \mathbb{R}^{m \times n}$, $k, l \in \mathbb{N}$

Output: Rank r approximation of $\mathbf{A}$: $\hat{\mathbf{A}} = \mathbf{Q}\mathbf{X}$

$\boldsymbol{\Omega} = \text{ortho}(\tilde{\boldsymbol{\Omega}}) \leftarrow \tilde{\boldsymbol{\Omega}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \in \mathbb{R}^{n \times k}$

$\boldsymbol{\Psi}^\top = \text{ortho}(\tilde{\boldsymbol{\Psi}}^\top) \leftarrow \tilde{\boldsymbol{\Psi}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \in \mathbb{R}^{l \times m}$

$\mathbf{Y} = \mathbf{A}\boldsymbol{\Omega}$

$\mathbf{W} = \boldsymbol{\Psi}\mathbf{A}$

$\mathbf{Q} = \text{ortho}(\mathbf{Y})$ [QR-decomp.]

$\mathbf{X} = (\boldsymbol{\Psi}\mathbf{Q})^+\mathbf{W}$

sketched-STHOSVD

Idea: Use two-sided skeiched low-rank approximation in R-THOSVD

Algorithm:

Input: Target tensor $\mathbf{A}$, target rank $(r_1, ..., r_d)$, $(l_1, ..., l_d)$

Output: Core tensor $\mathbf{C}$, basis matrices $\mathbf{U}_k \in \mathbb{R}^{r_k \times n_k}$ for $1 \leq k \leq d$

Set $\mathbf{A}_0 = \mathbf{A}$

for $k = 1 : d$

$\mathbf{Q}_k, \mathbf{X}_k =$ 2s-r-SVD$(\mathbf{A}_{k-1}^{(k)}, r_k, l_k)$

Set $\mathbf{A}_k^{(k)} = \mathbf{X}_k$

$\mathbf{U}_k = \mathbf{Q}_k^\top$

$\mathbf{C} = \mathbf{A}_d$

Sketching with subspace power iteration

- Spectrum of materialization decays slowly sketching is not great
- Power iteration technique to enhance the sketching algorithm:
  $\rightarrow$ replace $\mathbf{A}$ by $(\mathbf{A}\mathbf{A}^\top)^q\mathbf{A}$ for some $q \in \mathbb{N}$

$$(\mathbf{A}\mathbf{A}^\top)^q\mathbf{A} = \mathbf{U}\mathbf{\Sigma}^{2q+1}\mathbf{V}^\top$$

$\rightarrow$ Left- and right singular vectors remain the same!
$\rightarrow$ Faster decay rate of singular values
$\rightarrow$ Rounding errors may accumulate
Remedy: Orthonormalizing the columns of the sample matrix between each application of $\mathbf{A}$ and $\mathbf{A}^\top$

Sketching algorithm with subspace power iteration

- Choose sketching parameters $r \le k \le \min(l, n)$, $0 < l \le m$

Algorithm:

Input: $\mathbf{A} \in \mathbb{R}^{m \times n}$, $k, l, q \in \mathbb{N}$
Output: Rank r approximation of $\mathbf{A}$: $\hat{\mathbf{A}} = \mathbf{Q}\mathbf{X}$
$\boldsymbol{\Omega} = \text{ortho}(\tilde{\boldsymbol{\Omega}}) \leftarrow \tilde{\boldsymbol{\Omega}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \in \mathbb{R}^{n \times k}$
$\boldsymbol{\Psi}^\top = \text{ortho}(\tilde{\boldsymbol{\Psi}}^\top) \leftarrow \tilde{\boldsymbol{\Psi}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \in \mathbb{R}^{l \times m}$
$\mathbf{Y} = \mathbf{A}\boldsymbol{\Omega}$
$\mathbf{W} = \boldsymbol{\Psi}\mathbf{A}$
$\mathbf{Q}_0 = \text{ortho}(\mathbf{Y})$ [QR-decomp.]
for $j = 1 : q$
    $\hat{\mathbf{Y}}_j = \mathbf{A}^\top \mathbf{Q}_{j-1}$
    $\hat{\mathbf{Q}}_j = \text{ortho}(\hat{\mathbf{Y}}_j)$
    $\mathbf{Y}_j = \mathbf{A}\hat{\mathbf{Q}}_j$
    $\mathbf{Q}_j = \text{ortho}(\mathbf{Y}_j)$
$\mathbf{Q} = \mathbf{Q}_q$
$\mathbf{X} = (\boldsymbol{\Psi}\mathbf{Q})^+ \mathbf{W}$

sub-sketch-STHOSVD

Use the sketching algorithm with subspace power iteration in STHOSVD

Algorithm:

Input: Target tensor $\mathbf{A}$, target rank $(r_1, ..., r_d)$, $(l_1, ..., l_d)$, $q \in \mathbb{N}$

Output: Core tensor $\mathbf{C}$, basis matrices $\mathbf{U}_k \in \mathbb{R}^{r_k \times n_k}$ for $1 \leq k \leq d$

Set $\mathbf{A}_0 = \mathbf{A}$

for $k = 1 : d$

$\quad \mathbf{Q}_k, \mathbf{X}_k = \text{ sub-sketch-SVD}(\mathbf{A}_{k-1}^{(k)}, r_k, l_k, q)$

$\quad$ Set $\mathbf{A}_k^{(k)} = \mathbf{X}_k$

$\quad \mathbf{U}_k = \mathbf{Q}_k^\top$

$\mathbf{C} = \mathbf{A}_d$

Comparison

Experiment 1:

Hilbert tensor $\mathbf{A} \in \mathbb{R}^{n_1 \times \ldots \times n_d}$:

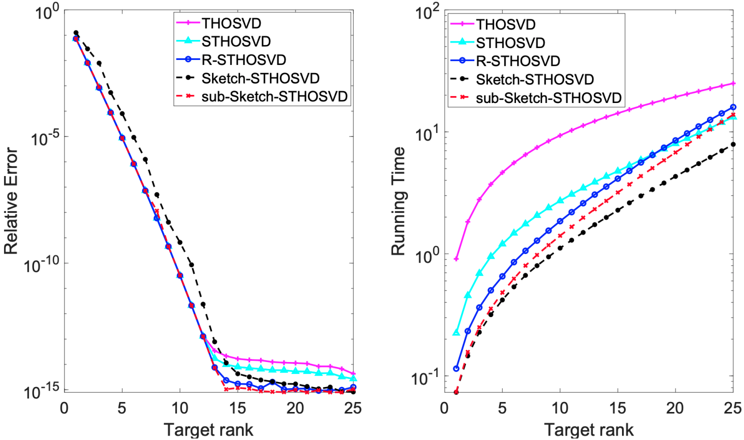
$$\mathbf{A}[i_1, ..., i_d] = \frac{1}{i_1 + ... + i_d}, \qquad 1 \leq i_j \leq n_j,\ 1 \leq j \leq d$$

Tensor parameters:

- $d = 5$
- $n_j = 25,\ j = 1, 2, ..., d$
- Target rank is (r, r, r, r, r), where $r \in [\![1, 25]\!]$

Computational parameters:

- Oversampling $p = 5$
- $l_i = r_i + 2,\ i = 1, 2, ..., d$
- Power parameter $q = 1$



THOSVD
STHOSVD
R-STHOSVD
Sketch-STHOSVD
sub-Sketch-STHOSVD
Relative Error
Target rank
Running Time
Target rank

CPU time (in second) on the Hilbert tensor with a size of $500 \times 500 \times 500$ as the target rank increases:

Target rank	THOSVD	STHOSVD	R-STHOSVD	Sketch-STHOSVD	sub-Sketch-STHOSVD
(10,10,10)	17.18	7.49	0.92	**0.86**	0.98
(20,20,20)	23.13	8.87	1.25	**1.05**	1.48
(30,30,30)	24.91	9.35	1.66	**1.53**	2.16
(40,40,40)	28.05	10.41	1.94	**1.44**	2.11
(50,50,50)	29.44	11.39	2.07	**1.67**	2.43
(60,60,60)	30.14	11.07	2.37	**1.90**	2.77
(70,70,70)	29.44	11.18	2.57	**2.10**	3.02
(80,80,80)	29.65	12.30	3.05	**2.54**	3.75
(90,90,90)	31.11	12.80	3.80	**2.80**	4.33
(100,100,100)	32.22	13.51	4.04	**3.07**	4.61

Relative error on the Hilbert tensor with a size of $500 \times 500 \times 500$ as the target rank increases:

Target rank	THOSVD	STHOSVD	R-STHOSVD	Sketch-STHOSVD	sub-Sketch-STHOSVD
(10,10,10)	2.7354e-06	**2.7347e-06**	**2.7347e-06**	1.1178e-05	2.7568e-06
(20,20,20)	1.1794e-12	**1.1793e-12**	1.1794e-12	7.1408e-12	1.2677e-12
(30,30,30)	4.6574e-15	3.2739e-15	3.2201e-15	4.0641e-15	**2.0182e-15**
(40,40,40)	4.4282e-15	3.4249e-15	2.8212e-15	2.1562e-15	**1.7860e-15**
(50,50,50)	4.1628e-15	3.2342e-15	2.6823e-15	2.3205e-15	**1.8625e-15**
(60,60,60)	4.1214e-15	3.1271e-15	2.3652e-15	2.2920e-15	**1.7472e-15**
(70,70,70)	4.1085e-15	3.0000e-15	2.1761e-15	2.0499e-15	**1.6370e-15**
(80,80,80)	4.0956e-15	3.1350e-15	1.8382e-15	1.8209e-15	**1.6424e-15**
(90,90,90)	4.0792e-15	3.3742e-15	1.8102e-15	1.7193e-15	**1.5264e-15**
(100,100,100)	4.0390e-15	3.0571e-15	1.7323e-15	1.6304e-15	**1.4957e-15**

Comparison

Experiment 2:

Sparse tensor $\mathbf{A} \in \mathbb{R}^{200\times200\times200}$:

$$\mathbf{A} = \sum_{i=1}^{10} \frac{\gamma}{i^2} \mathbf{x}_i \otimes \mathbf{y}_i \otimes \mathbf{z}_i + \sum_{i=11}^{200} \frac{1}{i^2} \mathbf{x}_i \otimes \mathbf{y}_i \otimes \mathbf{z}_i,$$

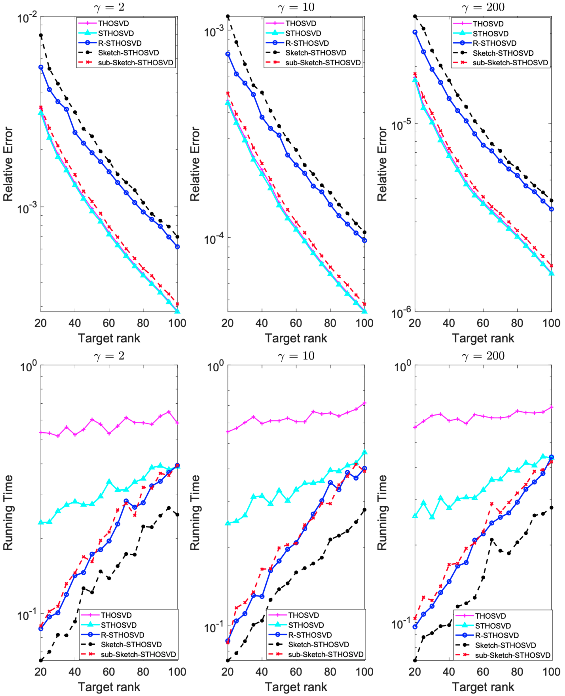
where $\mathbf{x}_i, \mathbf{y}_i, \mathbf{z}_i \in \mathbb{R}^{200}$ are sparse vectors with 5% nonzeros each (generated using the `sprand` command in MATLAB)

Tensor parameters:

- $\gamma = 2, 10, 200$
- $n_j = 25,\ j = 1, 2, ..., d$
- Target rank is (r, r, r), where $r \in [\![20, 100]\!]$

Computational parameters:

- Oversampling $p = 5$
- $l_i = r_i + 2,\ i = 1, 2, ..., d$
- Power parameter $q = 1$


$\gamma = 2$
$\gamma = 10$
$\gamma = 200$
THOSVD
STHOSVD
R-STHOSVD
Sketch-STHOSVD
sub-Sketch-STHOSVD
Relative Error
Running Time
Target rank