

Fast Algorithms for Nonnegative Matrix and Tensor Factorizations

Haesun Park

hpark@cc.gatech.edu

College of Computing

Georgia Institute of Technology

Atlanta, GA 30332, USA

Joint work with Jingu Kim and Krishnakumar Balasubramanian

GTRI, Sep. 3, 2009

Outline

- Fast new algorithms for NLS, NMF, NTF
- Experimental evidence supporting computational efficiency

- Nonnegativity constrained problem formulations and solutions
- Why Nonnegative constraints??
 - Nonnegativity constraints are often physically meaningful and provide natural interpretation.
 - Successful applications include:
 - Pixels in digital images
 - Signal frequency
 - Chemical concentration (Bioinformatics - microarray data analysis)
(Brunet et al., 2004, H. Kim and Park, Bioinformatics 2007, M. Mallick, B.L.Drake, H. Park, et al. Inf. Fusion 2009)
 - Term-document matrix for text analysis
 - Speech and audio processing ...

Nonnegativity Constraint Problems

- NLS (Nonnegative Least Squares):

Given $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^{m \times 1}$,
find $x \in \mathbb{R}_+^{n \times 1}$ such that

$$\min_{x \geq 0} \|Ax - b\|$$

- NMF (Nonnegative Matrix Factorization):

Given $A \in \mathbb{R}_+^{m \times n}$ and a desired rank k ,
find $W \in \mathbb{R}_+^{m \times k}$ and $H \in \mathbb{R}_+^{k \times n}$ such that

$$\min \|A - WH\|_F$$

- NTF (Nonnegative Tensor Factorization):

Given $X \in \mathbb{R}_+^{m \times n \times l}$ and a desired order k ,
find $A \in \mathbb{R}_+^{m \times k}$, $B \in \mathbb{R}_+^{n \times k}$, $C \in \mathbb{R}_+^{l \times k}$ such that

$$\min \|X - [ABC]\|_F^2$$

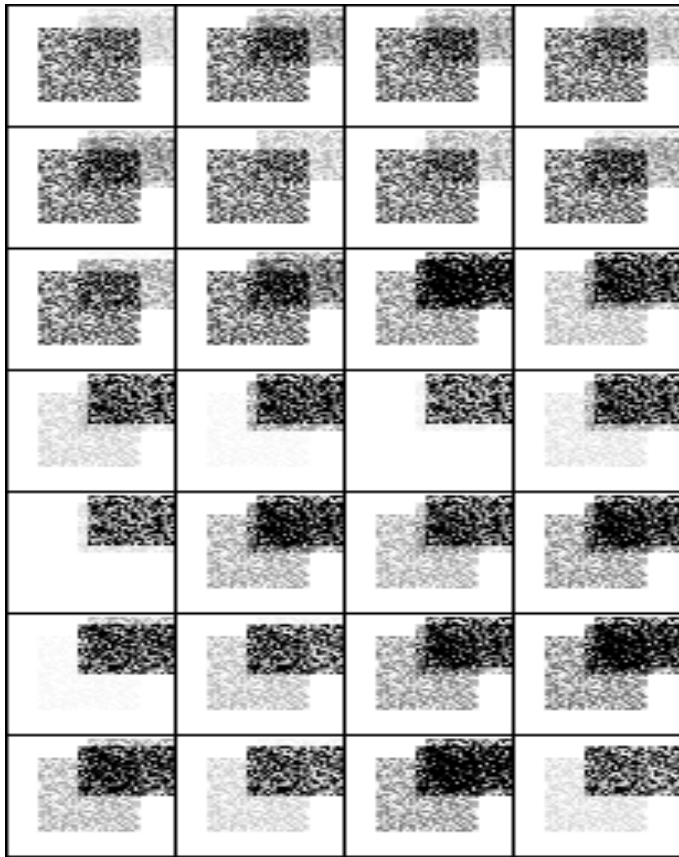
where $[ABC] = \sum_{i=1}^k a_i \circ b_i \circ c_i$, \circ : vector outer product

An NMF Formulation

- Formulation: how to assert $A \approx WH$?
 - $\min_{W \geq 0, H \geq 0} \|A - WH\|_F^2$
 - Alternative formulations exist
- Better Approximation vs. Better Representation/Interpretation
 - SVD (Singular Value Decomposition):
Best Approximation $A = U\Sigma V^T \approx U_k \Sigma_k V_k^T$
 $\|A - U_k \Sigma_k V_k^T\|_F = \min$
 $\|A - U_k \Sigma_k V_k^T\|_F \leq \|A - WH\|_F$
 - NMF: Better Representation/Interpretation
 $\min_{W \geq 0, H \geq 0} \|A - WH\|_F^2$

Factor Recovery by NMF

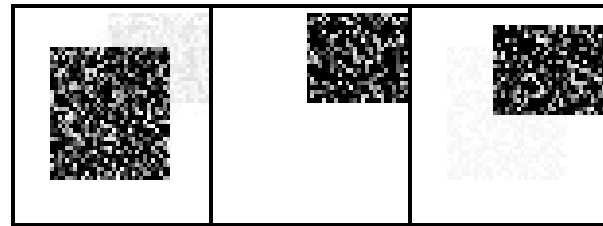
(a) Actual $A_a \in \mathbb{R}_+^{2500 \times 28} = W_a H_a$



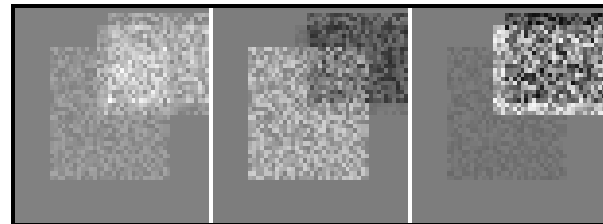
(b) Actual W_a



(c) W obtained from NMF/ANLS



(d) W obtained from SVD



In (a)-(c): zeros: white, larger positive values: darker

In (d): zeros: gray, positive values: lighter, negative values: darker

$(A_a = (U)(\Sigma V^T) = W H)$.

Algorithms for NMF

- Given $A \in \mathbb{R}_+^{m \times n}$ and a desired rank k ,

$$\min_{W, H} \|A - WH\|_F^2, \text{ s.t. } W \geq 0 \text{ and } H \geq 0.$$

- Non-convex optimization
- W and H are not unique (Consider $\hat{W} = WD \geq 0$, $\hat{H} = D^{-1}H \geq 0$).
- Algorithms developed
 - Multiplicative update rules: (Lee and Seung, Nature 99)
 - Alternating Least Squares (ALS): Berry et al 06
 - Alternating Nonnegative Least Squares (ANLS)
 - Lin, 07, Projected gradient descent
 - Kim et al., 07, Quasi-Newton
 - H. Kim and H. Park, SIMAX 08, Active-set based
 - Other algorithms and variants
 - Zdunek, Cichocki, Amari 06, Quasi-Newton
(Cichocki, Zdunek, Phan, Amari: NM and NTF: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation, Wiley, Nov. 2009)
 - Chu and Lin, 07, Low dim polytope approx.

Previous algorithms and drawbacks

- Multiplicative Updating Rules:[Lee and Seung,2001]

$$H_{qj} \leftarrow H_{qj} \frac{(W^T A)_{qj}}{((W^T W)H)_{qj}} \text{ and } W_{iq} \leftarrow W_{iq} \frac{(AH^T)_{iq}}{(W(HH^T))_{iq}}$$

- $\|A - WH\|_F^2$ is nonincreasing.
- Simple implementation, but convergence to a stationary point ??
- Alternating Least Squares (ALS)
 - Fix H and solve for W in $\min \|H^T W^T - A^T\|_F^2$,
and set all negative elements in W to 0.
 - Fix W and solve for H in $\min \|WH - A\|_F^2$,
and set all negative elements in H to 0.
 - The convergence to a stationary point ??
- Alternating Nonnegative Least Squares (ANLS)

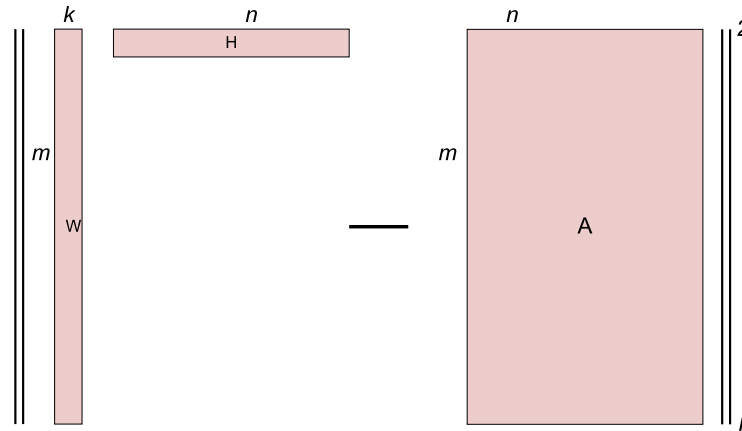
Alternating NLS for NMF

1. Initialize W (or H) with non-negative values.
 2. Iterate the following ANLS until convergence:
 - (a) Fixing W , solve $\min_{H \geq 0} \|WH - A\|_F^2$
 - (b) Fixing H , solve $\min_{W \geq 0} \|H^T W^T - A^T\|_F^2$
 3. The columns of W are normalized to unit L_2 -norm
- Block coordinate descent method in bound-constrained optimization
 - Convergence
 - No matter how many blocks, if each sub problem has a unique solution, then the limit point of the sequence is a stationary point (Powell 73, Bertsekas 99)
 - For two block problems, any limit point of the sequence is a stationary point (Grippo and Siandrone, 2000)
 - It is important to find an optimal solution of 2-(a),(b) at each iteration!
 - Fast algorithm for NLS needed.

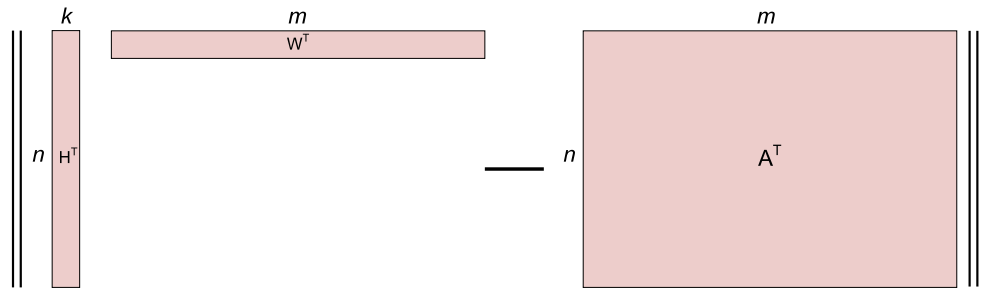
Structure of NLS problems in NMF

- In NMF: Matrix is long and thin, solution vectors short, many right hand side vectors.

- $\min_{H \geq 0} \|WH - A\|_F^2$



- $\min_{W \geq 0} \|H^T W^T - A^T\|_F^2$



NLS with Multiple Right-hand-sides

- $C : m \times k$ and $B : m \times n$ with $m > k$ are Given.
- **LS-Single**: $\min_x \|Cx - b\|_F$
- **LS-Multiple**: $\min_X \|CX - B\|_F$
Inefficient if LS-S is solved n times independently !
 C needs to be processed **only once** (e.g. compute SVD of C once)
- **NLS-Single**: $\min_{x \geq 0} \|Cx - a\|_F$ (Lawson and Hanson 74)
If we know the signs of the components in the final sol. x in advance
e.g. $x = [+ + 0 \cdots 0]^T$, then we only need to solve $\min \|[c_1 c_2]x^{(2)} - b\|_2$
and set the rest of x_i to be zeros.
Active set method: initially $x = 0$, $S_a = \{1, \dots, k\}$, $S_p = \text{null}$
In each step, indices are exchanged between S_a and S_b until the sol. is reached.
- **NLS-Multiple**: $\min_{X \geq 0} \|CX - B\|_F$
 - Apply NLS-S n times? **Inefficient!**

Block principal pivoting algorithm for NLS

- Consider single right-hand side problem: for $x \in \mathbb{R}^{k \times 1}$

$$\min_{x \geq 0} \|Cx - b\|_2^2$$

- KKT (Karush-Kuhn-Tucker) conditions for the above are:

$$y = C^T Cx - C^T b \quad (1a)$$

$$y \geq 0 \quad (1b)$$

$$x \geq 0 \quad (1c)$$

$$x_i y_i = 0, \quad i = 1, \dots, k \quad (1d)$$

- Find x and y that satisfy KKT conditions.

- Repeat:

- Partition $\{1, \dots, k\}$ into two disjoint index sets F and G

- $x_G = 0$

- Solve $x_F = \arg \min_{x_F} \|C_F x_F - b\|_2^2$. Then $y_F = 0$

- $y_G = C_G^T (C_F x_F - b)$

- If $x_F \geq 0$ and $y_G \geq 0$, optimal values are found.
Otherwise, update F and G .

How block principal pivoting works

Update by $C_F^T C_F x_F = C_F^T b$, and $y_G = C_G^T C_F x_F - C_G^T b$.

	x	y
F		0
F		0
F		0
F		0
F		0
G	0	
G	0	
G	0	
G	0	
G	0	

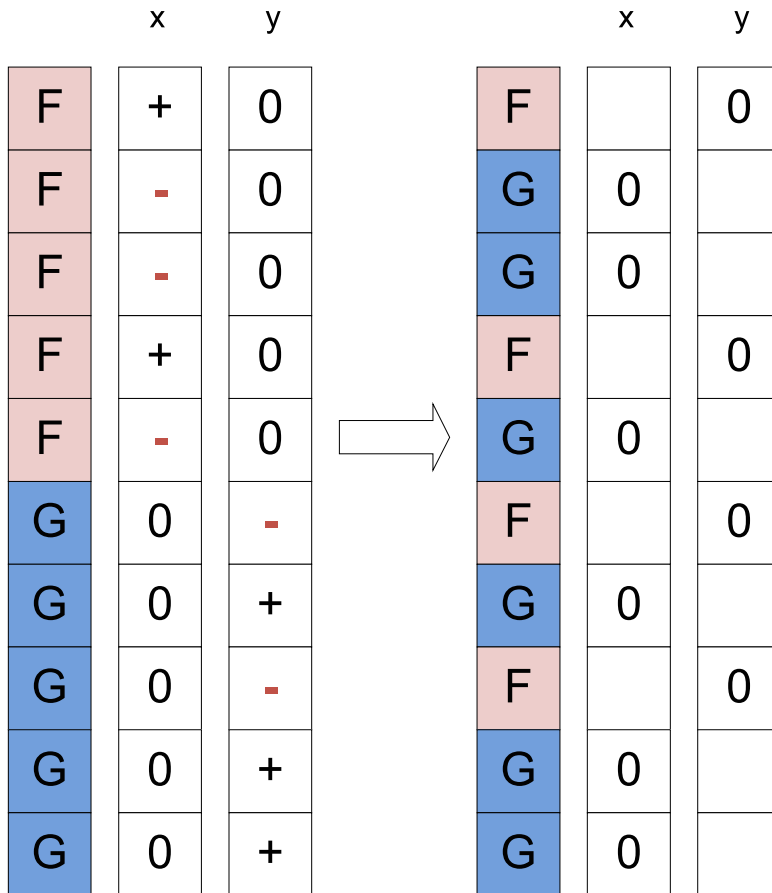
How block principal pivoting works

Update by $C_F^T C_F x_F = C_F^T b$, and $y_G = C_G^T C_F x_F - C_G^T b$.

	x	y
F	+	0
F	-	0
F	-	0
F	+	0
F	-	0
G	0	-
G	0	+
G	0	-
G	0	+
G	0	+

How block principal pivoting works

Update by $C_F^T C_F x_F = C_F^T b$, and $y_G = C_G^T C_F x_F - C_G^T b$.



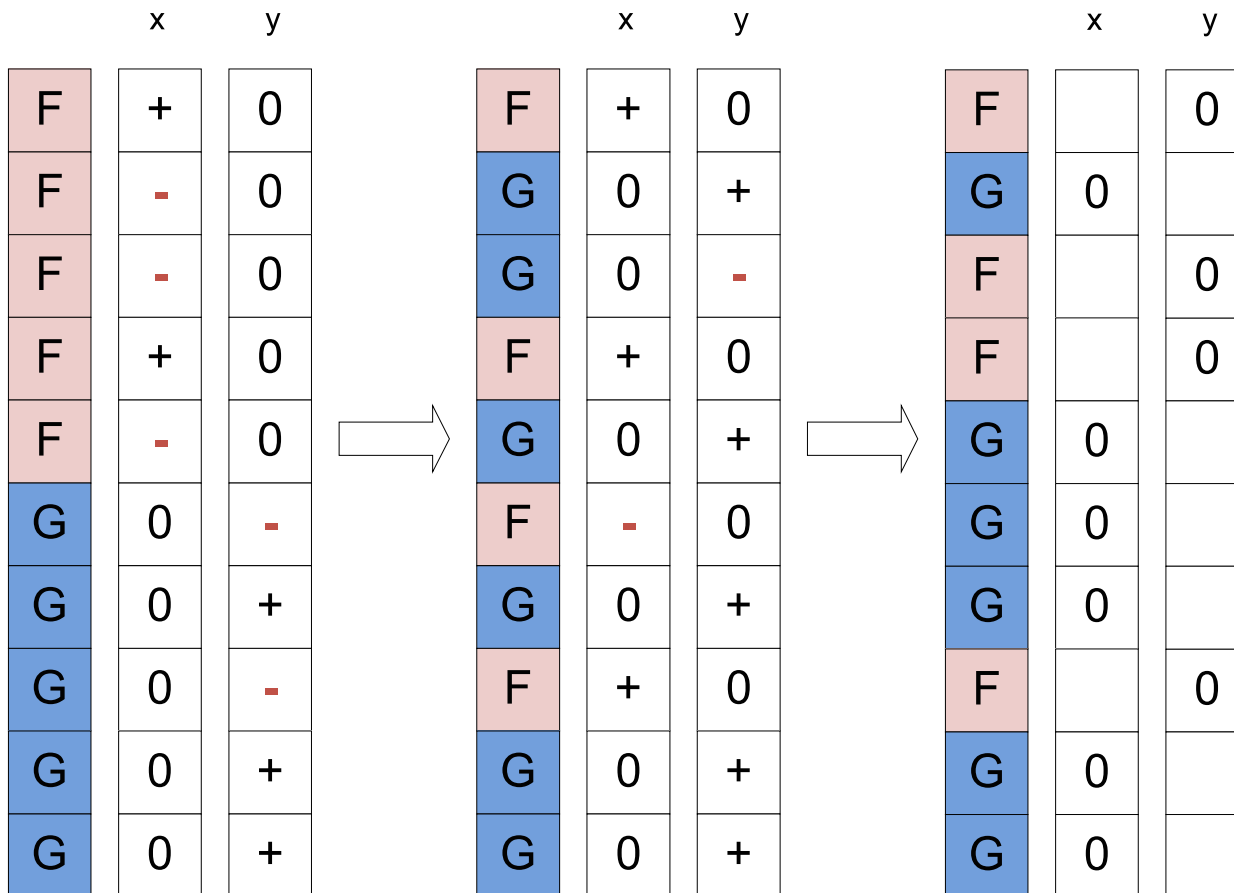
How block principal pivoting works

Update by $C_F^T C_F x_F = C_F^T b$, and $y_G = C_G^T C_F x_F - C_G^T b$.

	x	y		x	y	
F	+	0		F	+	0
F	-	0		G	0	+
F	-	0		G	0	-
F	+	0		F	+	0
F	-	0	→	G	0	+
G	0	-		F	-	0
G	0	+		G	0	+
G	0	-		F	+	0
G	0	+		G	0	+
G	0	+		G	0	+

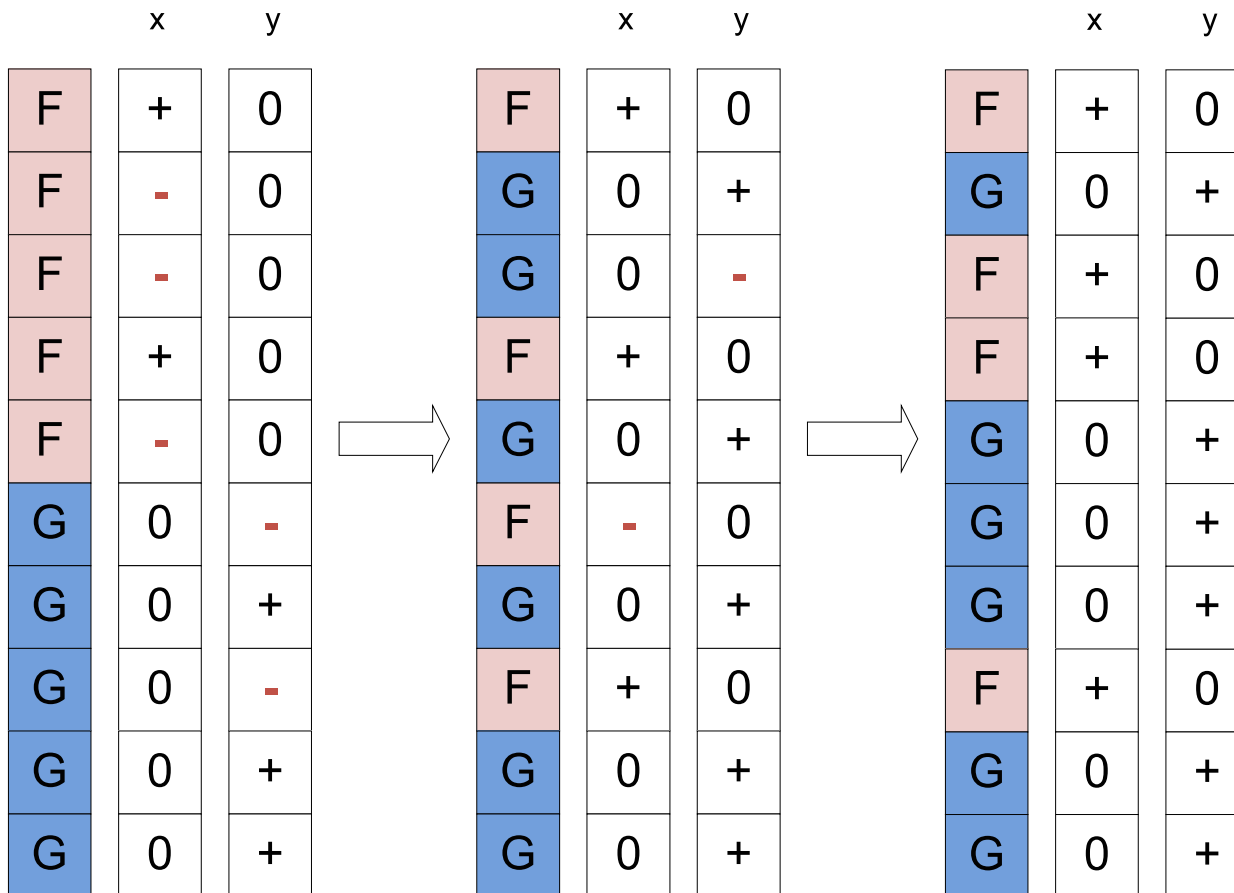
How block principal pivoting works

Update by $C_F^T C_F x_F = C_F^T b$, and $y_G = C_G^T C_F x_F - C_G^T b$.



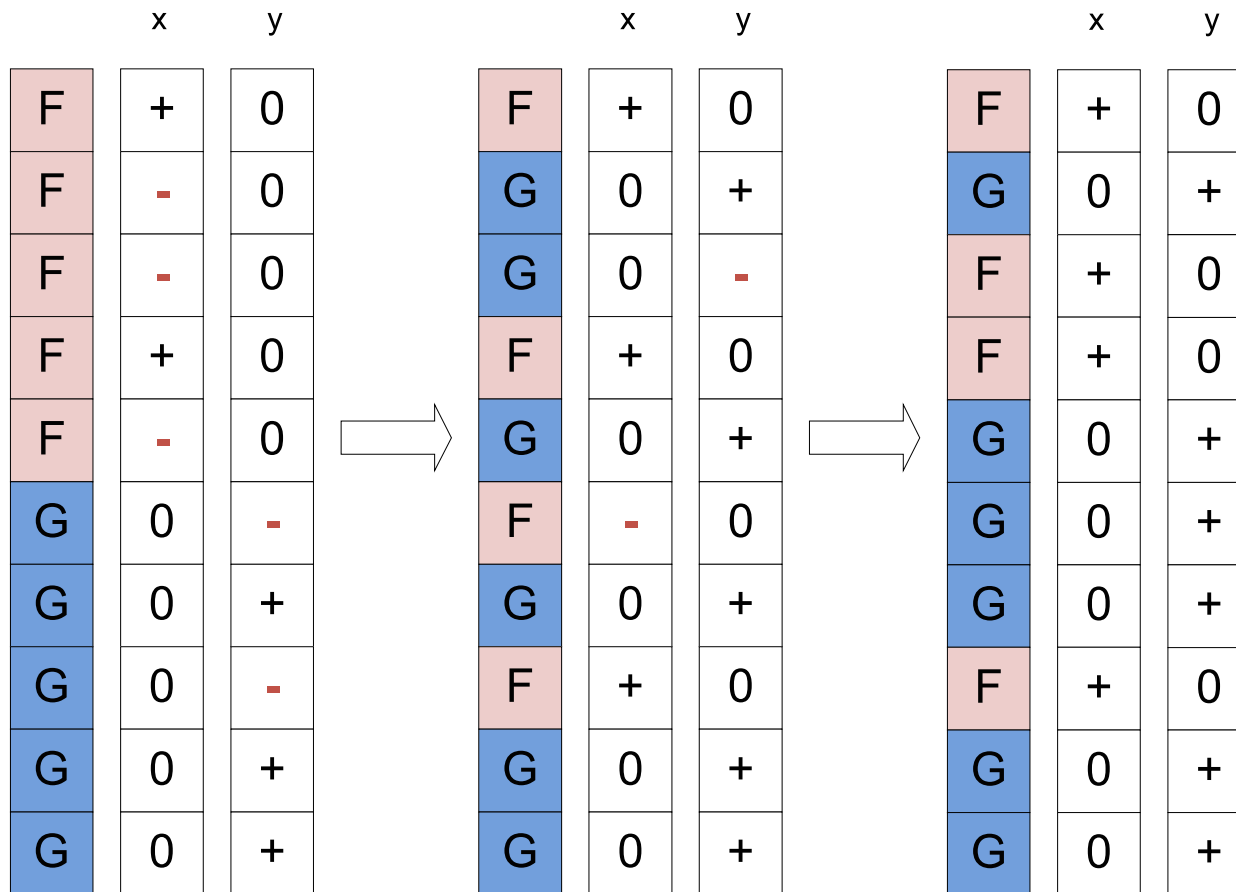
How block principal pivoting works

Update by $C_F^T C_F x_F = C_F^T b$, and $y_G = C_G^T C_F x_F - C_G^T b$.



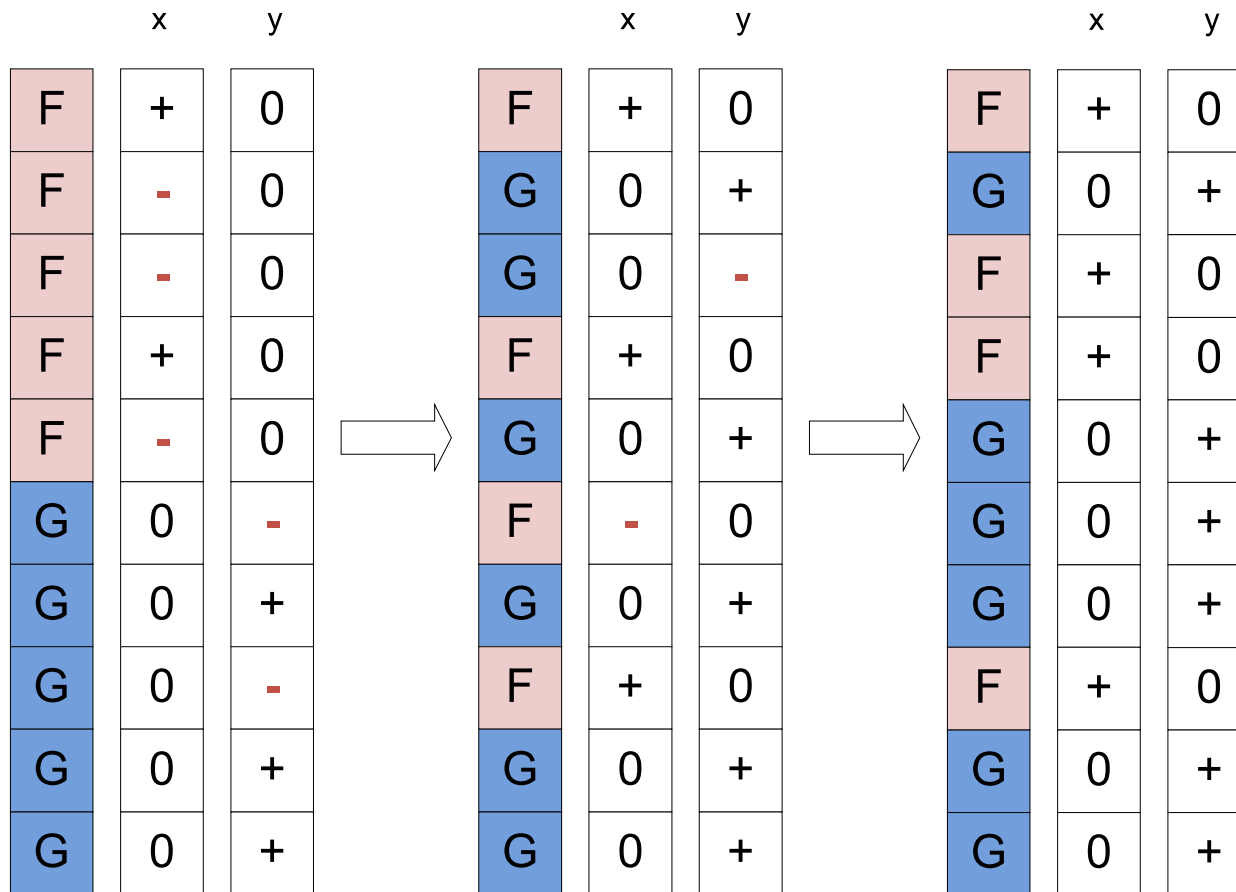
How block principal pivoting works

Update by $C_F^T C_F x_F = C_F^T b$, and $y_G = C_G^T C_F x_F - C_G^T b$.



How block principal pivoting works

Update by $C_F^T C_F x_F = C_F^T b$, and $y_G = C_G^T C_F x_F - C_G^T b$.



Solved!

Problem Solved !

Refining exchange rules

- Active set algorithm is a special instance of single principal pivoting algorithm (H. Kim and Park, SIMAX 08)
- Block exchange rule is not always safe.
 - The residual is not guaranteed to monotonically decrease. Block exchange rule may lead to a cycle (although it occurs rarely).
 - Modification: if the block exchange rule fails to decrease the number of infeasible variables, use a backup exchange rule
 - With this modification, block principal pivoting algorithm finds the solution of NLS in finite number of iterations.

NLS with Multiple right-hand sides

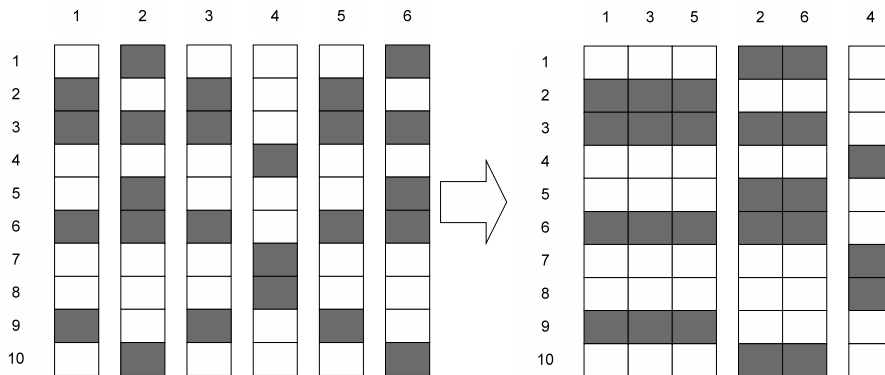
$$\min_{X \geq 0} \|CX - B\|_F^2$$

- It is possible to separately solve for each column of X . SLOW
- Two improvements [Bro and de Jong, 1997, Van Benthem and Keenan, 2004]
 - Precompute $C^T C$ and $C^T B$: updates of x_F and y_G is given by

$$\begin{aligned} C_F^T C_F x_F &= C_F^T b \\ y_G &= C_G^T C_F x_F - C_G^T b. \end{aligned}$$

All coefficients can be directly retrieved from $C^T C$ and $C^T B$!

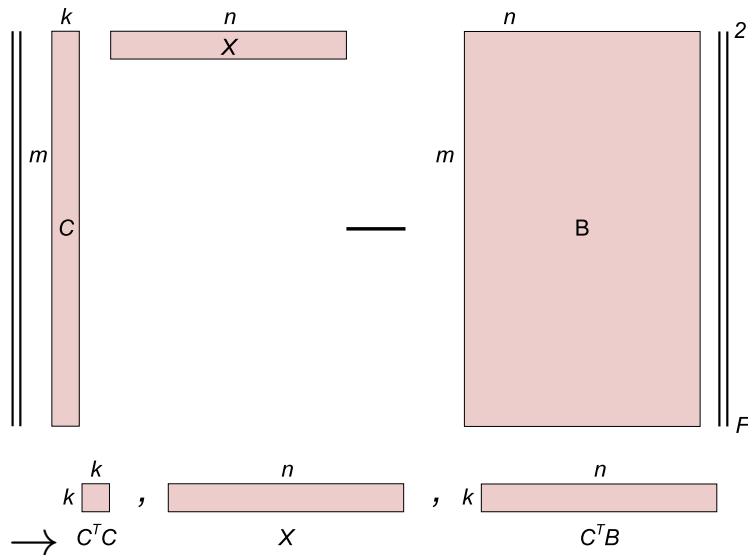
- Exploiting common F and G sets.



NLS with Multiple right-hand sides

$$\min_{X \geq 0} \|CX - B\|_F^2$$

- Note the long and thin structure.



- $C^T C$ and $C^T B$ is small. Storage is not a problem.
- X is flat and wide. \rightarrow More common cases of F and G sets.
- New proposed algorithm for NMF:
ANLS framework + Block principal pivoting algorithm for NLS with improvements for multiple right-hand sides

Extensions

- As other ANLS algorithms, easily extended to other formulations.
- Sparse NMF [H. Kim and Park, 2007] (for clustering)

$$\min_{W, H} \left\{ \|A - WH\|_F^2 + \eta \|W\|_F^2 + \beta \sum_{j=1}^n \|H(:, j)\|_1^2 \right\} \quad (2)$$

subject to $\forall ij, W_{ij}, H_{ij} \geq 0.$

ANLS reformulation [H. Kim and Park, 2007] : alternate the followings

$$\min_{H \geq 0} \left\| \begin{pmatrix} W \\ \sqrt{\beta} e_{1 \times k} \end{pmatrix} H - \begin{pmatrix} A \\ 0_{1 \times n} \end{pmatrix} \right\|_F^2$$
$$\min_{W \geq 0} \left\| \begin{pmatrix} H \\ \sqrt{\eta} I_k \end{pmatrix} W^T - \begin{pmatrix} A^T \\ 0_{k \times m} \end{pmatrix} \right\|_F^2$$

- $A \approx WH$ with sparse H behaves like a clustering algorithm

Clustering Results on TDT2 text corpus

k	dim	#	NMF	SNMF	K-means	NMF	SNMF	K-means
3	4687	306	8	16	14	0.99	0.98	0.95
6	6844	519	34	56	151	0.96	0.95	0.89
9	9136	874	95	124	469	0.95	0.95	0.88
12	10194	1089	164	229	1125	0.94	0.94	0.86
15	11822	1433	351	405	2238	0.94	0.93	0.83
18	13307	1694	604	635	3363	0.94	0.93	0.83

Ave. Computation Time and Purity for clustering

k : number of clusters

dim: dimension of the term-document matrix

#: number of documents

Average of 100 runs

Convergence Criterion

- KKT conditions

$$\min(W, \partial f(W, H)/\partial W) = 0, \min(H, \partial f(W, H)/\partial H) = 0. \quad (3)$$

Δ_o : KKT residual

$$\Delta_o = \sum_{i=1}^m \sum_{q=1}^k |\min(W_{iq}, (\partial f(W, H)/\partial W)_{iq})| + \sum_{q=1}^k \sum_{j=1}^n |\min(H_{qj}, (\partial f(W, H)/\partial H)_{qj})|. \quad (4)$$

- Normalized KKT residual:

$$\Delta = \frac{\Delta_o}{\delta_W + \delta_H}, \quad (5)$$

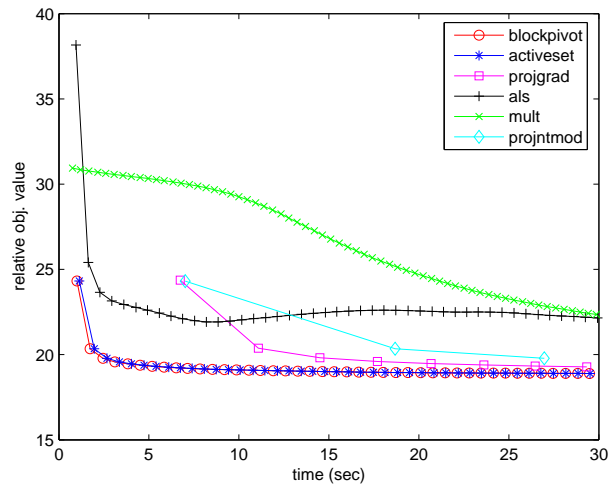
δ_W, δ_H : number of the elements in W and H that did not converge yet.

- Converged if $\Delta \leq \epsilon \Delta_1$

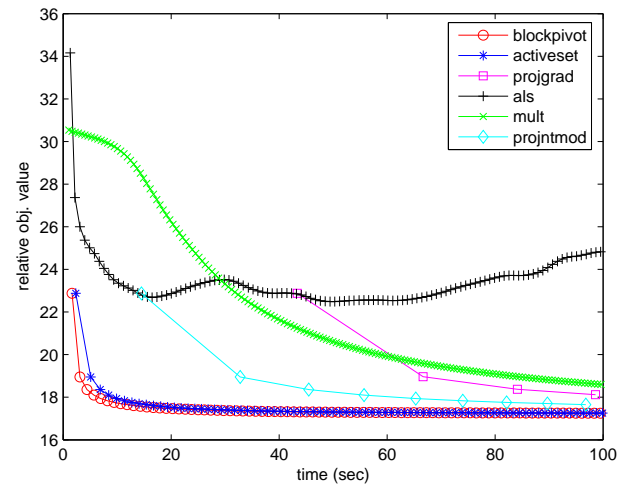
Comparison results (NMF)

- Data: (J.Kim and H. Park, ICDM 08)
 - Synthetic: 300×200 , sparse W and H , $A = WH$ with noise
 - Text: Topic Detection and Tracking 2, randomly select 20 topics, 12617×1491
 - Image: AT &T Facial Data, $(92 \times 112) \times 400$
- NMF algorithms
 - (**mult**) Lee and Seung's multiplicative updating algorithm
 - (**als**) Berry et al.'s alternating least squares algorithm
 - (**lsqnonneg (MATLAB)**) ANLS with Lawson and Hanson's algorithm
 - (**projnewton**) ANLS with Kim et al.'s projected quasi-Newton algorithm
 - (**projgrad**) ANLS with Lin's projected gradient algorithm
 - (**activeset**) ANLS with Kim and Park's active set algorithm
 - (**blockpivot**) ANLS with block principal pivoting algorithm which is proposed in this paper

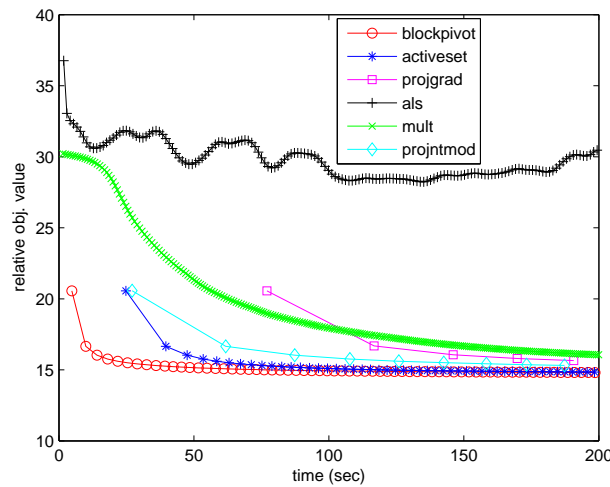
Relative residual and Execution time



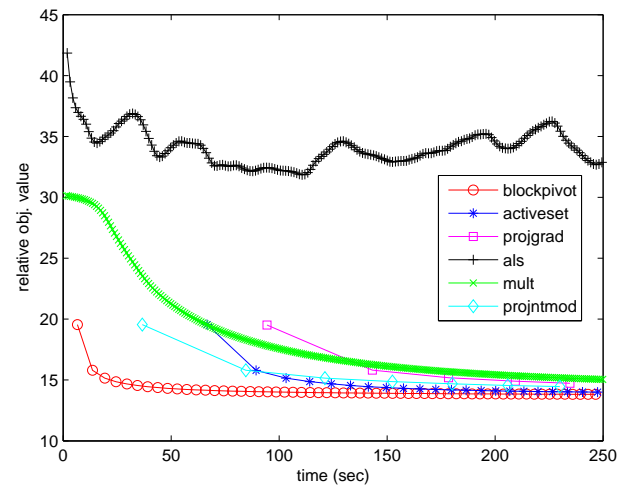
(a) $k=16$



(b) $k=25$



(c) $k=49$



(d) $k=64$

Synthetic Data : 300×200

	k	blockpivot	activeset	projgrad	projnewton	lsqnonneg	als	mult
Time (sec)	10	1.059	1.262	3.415	10.10	114.34	138.15*	105.77*
	20	2.776	3.311	3.588	14.693	305.63*	206.44*	150.17*
	30	4.522	5.841	7.146	19.797	375.80*	223.25*	179.32*
	40	8.436	11.201	51.15	30.53			
	60	32.70	40.88	326.3	110.3			
	80	36.63	50.71	267.0	131.7			
# of iterations	10	31.7	31.7	40.6	37.6	31.7	10000*	10000*
	20	39.3	39.3	56.9	46.7	23.6*	10000*	10000*
	30	47.2	47.2	75.7	57.7	11.9*	10000*	10000*
	40	65.4	65.4	131.4	84.1			
	60	162.6	162.6	334.2	190.9			
	80	121.9	121.9	127.0	135.5			
Relative residual	10	3.805	3.805	3.805	3.805	3.805	3.829*	3.806*
	20	4.178	4.178	4.178	4.178	4.476*	4.326*	4.181*
	30	4.207	4.207	4.207	4.207	9.607*	4.593*	4.224*
	40	4.169	4.169	4.169	4.169			
	60	4.025	4.025	4.286	4.025			
	80	5.361	5.361	5.567	5.443			

Text and Image Data

	Text data 12617×1491					Image data 10304×400				
	k	bp	as	pg	pn	k	bp	as	pg	pn
Time	10	54.5	57.45	69.3	104.6	16	10.64	11.2	60.8	91.12
	20	92.0	102.4	120.8	152.8	25	20.7	24.5	229.9	135.7
	30	81.2	105.0	132.5	127.4	36	31.6	44.8	222.8	189.6
	40	117.5	162.7	190.2	217.8	49	52.8	95.2	326.9	263.6
	50	149.0	229.1	276.7	238.1	64	74.1	195.5	395.7	374.1
	60	139.3	255.7	282.0	257.0	81	107.0	344.9	570.0	601.0
# it	10	22.9	22.9	23.6	23.0	16	15.0	15.0	15.7	15.0
	20	26.5	26.5	31.6	28.5	25	15.7	15.7	16.6	15.7
	30	25.3	25.3	34.5	22.8	36	14.7	14.7	15.4	14.8
	40	28.0	28.0	27.5	30.7	49	14.8	14.8	15.5	15.3
	50	29.3	29.3	31.5	26.4	64	14.9	14.9	15.7	15.5
	60	29.9	26.2	30.0	27.1	81	15.6	15.6	16.2	16.1
RR	10	80.9	80.9	81.0	80.9	16	19.0	19.0	19.0	19.0
	20	74.6	74.6	74.6	74.6	25	17.4	17.4	17.4	17.4
	30	70.8	70.8	70.8	70.8	36	16.1	16.1	16.2	16.194
	40	68.1	68.1	68.2	68.2	49	15.1	15.1	15.1	15.106
	50	65.8	65.8	65.8	65.9	64	14.1	14.1	14.1	14.1
	60	63.8	63.8	63.8	63.9	81	13.2	13.2	13.3	13.2

bp: Block Pivoting

as: Active Set

pg: Projected Gradient

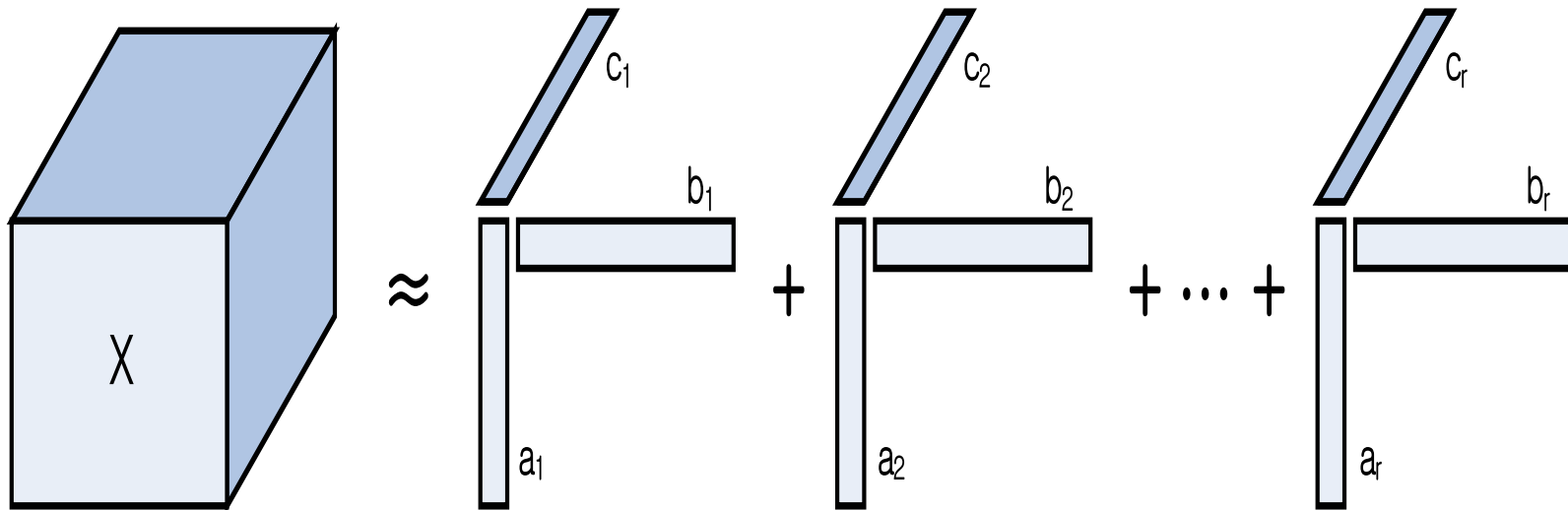
pn: Projected Newton

it: Number of Iterations

RR: Relative Residual

Non-Negative Tensor Factorization

The loading matrices (A, B , and C) can be iteratively estimated by block principal pivoting method.



PARAFAC (PARAllel FACtorization)

Non-Negative Tensor Factorization

- Iterate until a stopping criteria is satisfied:



$$\min_{A \geq 0} \left\| Y_{BC} A^T - X_{(1)} \right\|_F^2$$

where $Y_{BC} = B \odot C$ and $X_{(1)}$ is the $(np) \times m$ unfolded matrix.



$$\min_{B \geq 0} \left\| Y_{AC} B^T - X_{(2)} \right\|_F^2$$

where $Y_{AC} = A \odot C$ and $X_{(2)}$ is the $(mp) \times m$ unfolded matrix.



$$\min_{C \geq 0} \left\| Y_{AB} C^T - X_{(3)} \right\|_F^2$$

where $Y_{AB} = A \odot B$ and $X_{(3)}$ is the $(mn) \times p$ unfolded matrix.

- Can be similarly extended to higher order tensors.

Sparse Non-Negative Tensor Factorization

- This framework can be further extended to the case of Sparse NFT, where we iterate the following ANLS until convergence :

$$\min_{A \geq 0} \left\| \begin{pmatrix} Y_{BC} \\ \sqrt{\alpha_{(1)}} e_{1 \times k} \end{pmatrix} A^T - \begin{pmatrix} X_{(1)} \\ 0_{1 \times m} \end{pmatrix} \right\|_F^2$$

$$\min_{B \geq 0} \left\| \begin{pmatrix} Y_{AC} \\ \sqrt{\alpha_{(2)}} I_{k \times k} \end{pmatrix} B^T - \begin{pmatrix} X_{(2)} \\ 0_{k \times n} \end{pmatrix} \right\|_F^2$$

$$\min_{C \geq 0} \left\| \begin{pmatrix} Y_{AB} \\ \sqrt{\alpha_{(3)}} I_{k \times k} \end{pmatrix} C^T - \begin{pmatrix} X_{(3)} \\ 0_{k \times p} \end{pmatrix} \right\|_F^2$$

Comparison results (NTF)

Algo	K	NTF.blockpivot	NTF.fcnnls	NTF.mupdates
Time(sec)	5	0.6558	3.0233	78.5518
	30	2.1932	11.0865	171.7668
	50	6.9089	24.9563	
SSR	5	270.67	270.67	452.50
	20	270.31	270.31	352.68
	50	250.75	250.75	

Table 1: $\underline{\mathbf{X}} \in \mathbb{R}_+^{50 \times 201 \times 61}$ is a randomly generated tensor. K is the rank used. No. of iterations was 26

Algo	K	NTF.blockpivot	NTF.fcnnls	NTF.mupdates
Time(sec)	9	1.0558	1.9237	308.5518
	50	8.1932	19.0865	
	90	40.9811	87.9563	
SSR	9	1890.67	1865.67	3452.50
	50	1344.33	1344.78	
	90	1266.75	1268.75	

Table 2: $\underline{\mathbf{X}} \in \mathbb{R}_+^{100 \times 433 \times 200}$ is a randomly generated tensor. K is the rank used. No. of iterations was 26

Comparison results (NTF)

Algo	K	NTF.blockpivot	NTF.fcnnls
Time(sec)	3	2.0558	3.9237
	10	18.1932	40.0865

Table 3: $\underline{\mathbf{X}} \in \mathbb{R}_+^{1000 \times 234 \times 654}$ is a randomly generated tensor. K is the rank used. No. of Iterations was 15

Algo	K	SparseNTF.blockpivot	SparseNTF.fcnnls
Time(sec)	10	1.4868	2.9211
	50	10.0558	21.9914
	100	58.1854	90.3214

Table 4: Sparse NTF - $\underline{\mathbf{X}} \in \mathbb{R}_+^{173 \times 234 \times 854}$ is a randomly generated tensor. K is the rank used. No. of Iterations was 20

Summary

- A new algorithm for NMF and its extension to NTF :
ANLS framework + Block principal pivoting algorithm with improvements for multiple right-hand sides
- ANLS with Active-set type algorithms outperform other algorithms in computational experiments
- Importance of sound theoretical foundations in designing efficient algorithms
- Exploit the structure of the problem
- Extension to sparse NMF for clustering
- NLS for Rank Deficient case?
- Efficient algorithm for massive scale data?

Thank you!

www.cc.gatech.edu/~hpark
fodava.gatech.edu