Fast and Guaranteed Tensor Decomposition via Sketching

Yining Wang, Hsiao-Yu Tung, Alex Smola and Anima Anandkumar
Carnegie Mellon University   University of California, Irvine

Introduction

Tensor CP decomposition

- For an $n \times n \times n$ tensor T, find $\{\lambda_i\}$ and $\{u_i\}$ such that $\|T - \sum_{i=1}^k \lambda_i u_i^{\otimes 3}\|_F^2$ is minized.
- Wide applications in data mining and statistical learning of latent variable models.

Robust tensor power method [1]

- Tensor power iteration: $u \leftarrow T(I, u, u)$.
- Challenge: $O(n^3)$ time complexity.

Background

CountSketch [2]: random hash h, σ; $s(x) \in \mathbb{R}^b$:

$$[s(x)]_i = \sum_{h(j)=i} \sigma_j x_j; \quad \hat{x}_j = \sigma_j [s(x)]_{h(j)}.$$

TensorSketch [3]: random $\{h_\ell, \sigma_\ell\}$; $s(T) \in \mathbb{R}^b$:

$$[s(T)]_i = \sum_{h(j_1,j_2,j_3)=i} \sigma_1(j_1)\sigma_2(j_2)\sigma_3(j_3) T_{j_1,j_2,j_3};$$

$$h(j_1,j_2,j_3) = (h_1(j_1) + h_2(j_2) + h_3(j_3)) \mod b;$$

Methods

- **Important facts**:
  - *Linerality*: $s(\mu A + \lambda B) = \mu s(A) + \lambda s(B)$.
  - *Efficient tensorization*: $s(x \otimes y) = s(x) * s(y) = \mathcal{F}^{-1}(\mathcal{F}(s(x)) \circ \mathcal{F}(s(y)))$.
  - *Approximate inner product*: $\langle A, B\rangle \approx \langle s(A), s(B)\rangle = \langle \mathcal{F}(s(A)), \mathcal{F}(s(B))\rangle$.
- **A first attempt for efficiently computing** $v = T(I, u, u)$**:** (assuming $s(T)$ is known)

$$v_i = T(e_i, u, u) = \langle T, \quad e_i \otimes u \otimes u\rangle \approx \langle \mathcal{F}(s(T)), \quad \mathcal{F}(s(e_i)) \circ \mathcal{F}(s(u)) \circ \mathcal{F}(s(u))\rangle.$$

  - Time complexity $O(n^2 + b\log b)$.
  - Can we do even better?
- **The "shifting" trick**

$$v_i \approx \langle \mathcal{F}(s(T)), \quad \mathcal{F}(e_i) \circ \mathcal{F}(u) \circ \mathcal{F}(u)\rangle = \langle \mathcal{F}^{-1}(\mathcal{F}(s(T)) \circ \overline{\mathcal{F}(s(u))} \circ \overline{\mathcal{F}(s(u))}), \quad s(e_i)\rangle.$$

  - $s(e_i)$ only has one non-zero element!
  - Time complexity: $O(n + b\log b)$. Huge improvement over naive methods.
- **Additional techniques**: symmetric hashing using the complex ring, etc.; details in the paper!

Theoretical results

Theorem 1: Fix a symmetric $n \times n \times n$ real tensor T and n-dimensional vector u. Let b be the sketch length and define $\varepsilon_{1,T} = T(u,u,u) - \hat{T}(u,u,u)$, $\varepsilon_{2,T} = T(I,u,u) - \hat{T}(I,u,u)$. Then the following holds:

$$|\varepsilon_{1,T}| = O_P(\|T\|_F/\sqrt{b}), \qquad |[\varepsilon_{2,T}]_i| = O_P(\|T\|_F/\sqrt{b}).$$

Furthermore, for any fixed $w \in \mathbb{R}^n$, $\|w\|_2 = 1$ we have $\langle w, \varepsilon_{2,T}(u)\rangle = O_P(\|T\|_F^2/b)$.

- Complete proof and additional theoretical results for sketching based robust tensor power method can be found in the paper.

Results on synthetic tensor decomposition

Table 3: Squared residual norm on top 10 recovered eigenvectors of 1000d tensors and running time (excluding I/O and sketch building time) for plain (exact) and sketched robust tensor power methods. Two vectors are considered mismatch (wrong) if $\|\boldsymbol{v} - \hat{\boldsymbol{v}}\|_2^2 > 0.1$. A extended version is shown as Table 5 in Appendix A.

		Residual norm					No. of wrong vectors					Running time (min.)				
	$\log_2(b)$:	12	13	14	15	16	12	13	14	15	16	12	13	14	15	16
$\sigma = .01$	$B = 20$	.40	.19	.10	**.09**	.08	8	6	3	**0**	0	.85	1.6	3.5	**7.4**	16.6
	$B = 30$	.26	.10	.09	.08	.07	7	5	2	0	0	1.3	2.4	5.3	11.3	24.6
	$B = 40$	.17	.10	**.08**	.08	.07	7	4	**0**	0	0	1.8	3.3	**7.3**	15.2	33.0
	Exact	.07					0					293.5				

- Additional experimental results on accelerated Alternating Least Squares (ALS) available in the supplementary material!

Results on topic modeling

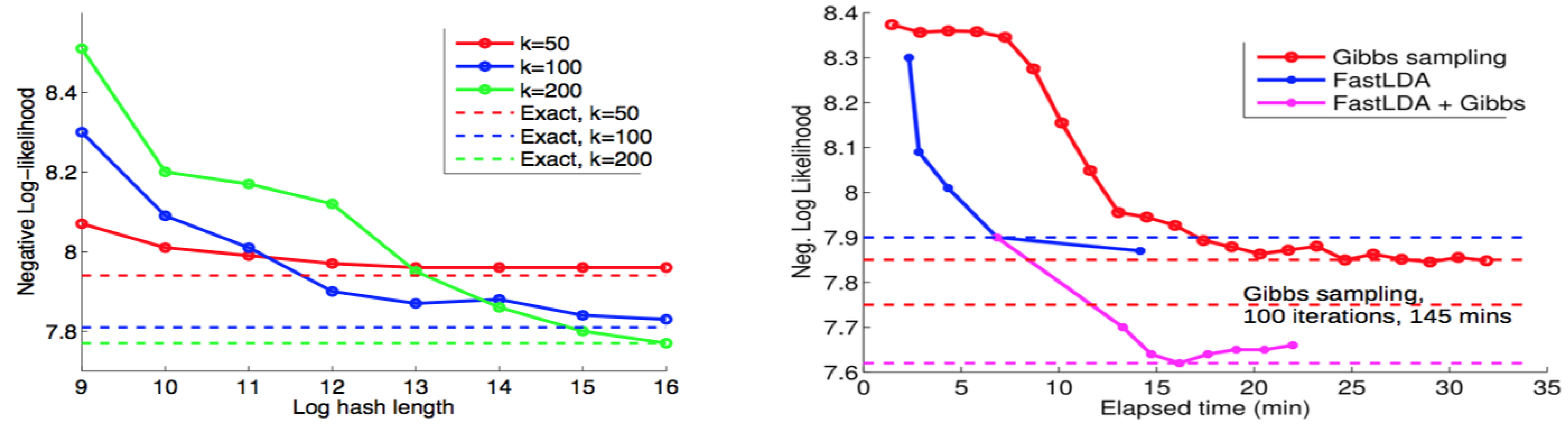

Figure 1: Left: negative log-likelihood for fast and exact tensor power method on Wikipedia dataset. Right: negative log-likelihood for collapsed Gibbs sampling, fast LDA and Gibbs sampling using Fast LDA as initialization.

Table 4: Negative log-likelihood and running time (min) on the *large* Wikipedia dataset for 200 and 300 topics.

k		like.	time	$\log_2 b$	iters	k	like.	time	$\log_2 b$	iters
200	Spectral	7.49	**34**	12	-	300	7.39	**56**	13	-
	Gibbs	6.85	561	-	30		6.38	818	-	30
	Hybrid	**6.77**	144	12	5		**6.31**	352	13	10

References

1. A. Anandkumar, R. Ge, D. Hsu, S. Kakade and M. Telgarsky. **Tensor decompositions for learning latent variable models**. *Journal of Machine Learning Research*, 15(1):2773-2832, 2014.
2. M. Charikar, K. Chen and M. Farah-Colton. **Finding frequent items in data streams.** *Theoretical Computer Science*, 312(1):3-15, 2004.
3. N. Pham and R. Pagh. **Fast and scalable polynomial kernels via explicit feature maps**. In *KDD*, 2013.