Dimensionality reduction via sparse matrices

Jelani Nelson
Harvard

September 19, 2013

based on works with Daniel Kane (Stanford) and Huy Nguyễn (Princeton)

Metric Johnson-Lindenstrauss lemma

Metric JL (MJL) Lemma, 1984

Every set of N points in Euclidean space can be embedded into $O(\varepsilon^{-2} \log N)$-dimensional Euclidean space so that all pairwise distances are preserved up to a $1 \pm \varepsilon$ factor.

Metric Johnson-Lindenstrauss lemma

Metric JL (MJL) Lemma, 1984

Every set of N points in Euclidean space can be embedded into $O(\varepsilon^{-2} \log N)$-dimensional Euclidean space so that all pairwise distances are preserved up to a $1 \pm \varepsilon$ factor.

Uses:

- Speed up geometric algorithms by first reducing dimension of input [Indyk, Motwani '98], [Indyk '01]
- Faster/streaming numerical linear algebra algorithms [Sarlós '06], [LWMRT '07], [Clarkson, Woodruff '09]
- Essentially equivalent to RIP matrices from compressed sensing [Baraniuk et al. '08], [Krahmer, Ward '11] (used for recovery of sparse signals)

How to prove the JL lemma

Distributional JL (DJL) lemma

Lemma

For any $0 < \varepsilon, \delta < 1/2$ *there exists a distribution* $\mathcal{D}_{\varepsilon,\delta}$ *on* $\mathbb{R}^{m \times n}$ *for* $m = O(\varepsilon^{-2} \log(1/\delta))$ *so that for any* u *of unit* ℓ_2 *norm*

$$\mathop{\mathbb{P}}_{\Pi \sim \mathcal{D}_{\varepsilon,\delta}} \left(\left| \|\Pi u\|_2^2 - 1 \right| > \varepsilon \right) < \delta.$$

How to prove the JL lemma

Distributional JL (DJL) lemma

Lemma

For any $0 < \varepsilon, \delta < 1/2$ *there exists a distribution* $\mathcal{D}_{\varepsilon,\delta}$ *on* $\mathbb{R}^{m \times n}$ *for* $m = O(\varepsilon^{-2} \log(1/\delta))$ *so that for any* u *of unit* ℓ_2 *norm*

$$\mathbb{P}_{\Pi \sim \mathcal{D}_{\varepsilon,\delta}} \left(\left| \|\Pi u\|_2^2 - 1 \right| > \varepsilon \right) < \delta.$$

Proof of MJL: Set $\delta = 1/N^2$ in DJL and u as the difference vector of some pair of points. Union bound over the $\binom{N}{2}$ pairs.

How to prove the JL lemma

Distributional JL (DJL) lemma

Lemma

For any $0 < \varepsilon, \delta < 1/2$ there exists a distribution $\mathcal{D}_{\varepsilon,\delta}$ on $\mathbb{R}^{m \times n}$ for $m = O(\varepsilon^{-2} \log(1/\delta))$ so that for any u of unit ℓ_2 norm

$$\mathop{\mathbb{P}}_{\Pi \sim \mathcal{D}_{\varepsilon,\delta}} \left(\left| \|\Pi u\|_2^2 - 1 \right| > \varepsilon \right) < \delta.$$

Proof of MJL: Set $\delta = 1/N^2$ in DJL and u as the difference vector of some pair of points. Union bound over the $\binom{N}{2}$ pairs.

Theorem (Alon, 2003)

For every N, there exists a set of N points requiring target dimension $m = \Omega((\varepsilon^{-2}/\log(1/\varepsilon)) \log N)$.

Theorem (Jayram-Woodruff, 2011; Kane-Meka-N., 2011)

For DJL, $m = \Theta(\varepsilon^{-2} \log(1/\delta))$ is optimal.

Proving the distributional JL lemma

Older proofs

- [Johnson-Lindenstrauss, 1984], [Frankl-Maehara, 1988]: Random rotation, then projection onto first m coordinates.
- [Indyk-Motwani, 1998], [Dasgupta-Gupta, 2003]: Random matrix with independent Gaussian entries.
- [Achlioptas, 2001]: Independent ± 1 entries.
- [Clarkson-Woodruff, 2009]: $O(\log(1/\delta))$-wise independent ± 1 entries.
- [Arriaga-Vempala, 1999], [Matousek, 2008]: Independent entries having mean 0, variance $1/m$, and subGaussian tails

Proving the distributional JL lemma

Older proofs

- [Johnson-Lindenstrauss, 1984], [Frankl-Maehara, 1988]: Random rotation, then projection onto first m coordinates.
- [Indyk-Motwani, 1998], [Dasgupta-Gupta, 2003]: Random matrix with independent Gaussian entries.
- [Achlioptas, 2001]: Independent ± 1 entries.
- [Clarkson-Woodruff, 2009]: $O(\log(1/\delta))$-wise independent ± 1 entries.
- [Arriaga-Vempala, 1999], [Matousek, 2008]: Independent entries having mean 0, variance $1/m$, and subGaussian tails

Downside: Performing embedding is dense matrix-vector multiplication, $O(m \cdot \|x\|_0)$ time

Fast JL Transforms

- [Ailon-Chazelle, 2006]: $x \mapsto PHDx$, $O(n \log n + m^3)$ time
  P random+sparse, H Fourier, D has random ± 1 on diagonal
- Also follow-up works based on similar approach which improve the time while, for some, slightly increasing target dimension [Ailon, Liberty '08], [Ailon, Liberty '11], [Krahmer, Ward '11], [N., Price, Wootters '14], ...

Fast JL Transforms

- [Ailon-Chazelle, 2006]: $x \mapsto PHDx$, $O(n \log n + m^3)$ time
  P random+sparse, H Fourier, D has random ± 1 on diagonal
- Also follow-up works based on similar approach which improve the time while, for some, slightly increasing target dimension [Ailon, Liberty '08], [Ailon, Liberty '11], [Krahmer, Ward '11], [N., Price, Wootters '14], ...

Downside: Slow to embed sparse vectors: running time is $\Omega(\min\{m \cdot \|x\|_0, n \log n\})$.

Where Do Sparse Vectors Show Up?

- **Document as bag of words:** u_i = number of occurrences of word i. Compare documents using cosine similarity.

  n = lexicon size; most documents aren't dictionaries
- **Network traffic:** $u_{i,j}$ = #bytes sent from i to j

  $n = 2^{64}$ (2^{256} in IPv6); most servers don't talk to each other
- **User ratings:** $u_{i,j}$ is user i's score for movie j on Netflix

  $n = \#\text{movies}$; most people haven't rated all movies
- **Streaming:** u receives a stream of updates of the form: "add v to u_i". Maintaining Πu requires calculating $v \cdot \Pi e_i$.
- ...

Sparse JL transforms

One way to embed sparse vectors faster: use sparse matrices.

Sparse JL transforms

One way to embed sparse vectors faster: use sparse matrices.

$s = \#$non-zero entries per column in Π
(so embedding time is $s \cdot \|x\|_0$)

reference	value of s	type
[JL84], [FM88], [IM98], ...	$m \approx 4\varepsilon^{-2}\ln(1/\delta)$	dense
[Achlioptas01]	$m/3$	sparse Bernoulli
[WDALS09]	no proof	hashing
[DKS10]	$\tilde{O}(\varepsilon^{-1}\log^3(1/\delta))$	hashing
[KN10a], [BOR10]	$\tilde{O}(\varepsilon^{-1}\log^2(1/\delta))$	"
[KN12]	$O(\varepsilon^{-1}\log(1/\delta))$	modified hashing

Sparse JL transforms

One way to embed sparse vectors faster: use sparse matrices.

$s = \#$non-zero entries per column in Π
(so embedding time is $s \cdot \|x\|_0$)

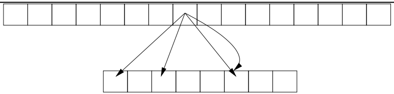
reference	value of s	type
[JL84], [FM88], [IM98], ...	$m \approx 4\varepsilon^{-2}\ln(1/\delta)$	dense
[Achlioptas01]	$m/3$	sparse Bernoulli
[WDALS09]	no proof	hashing
[DKS10]	$\tilde{O}(\varepsilon^{-1}\log^3(1/\delta))$	hashing
[KN10a], [BOR10]	$\tilde{O}(\varepsilon^{-1}\log^2(1/\delta))$	"
[KN12]	$O(\varepsilon^{-1}\log(1/\delta))$	modified hashing

[N., Nguyễn '13]: for any $m \le \mathrm{poly}(1/\varepsilon) \cdot \log N$, $s = \Omega(\varepsilon^{-1}\log N/\log(1/\varepsilon))$ is required, even for metric JL, so [KN12] is optimal up to $O(\log(1/\varepsilon))$.

*[Thorup, Zhang '04] gives $m = O(\varepsilon^{-2}\delta^{-1}), s = 1$.

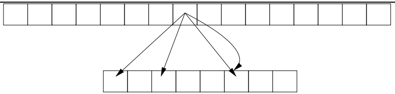
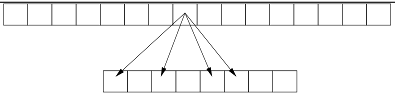
Sparse JL Constructions

[DKS, 2010]

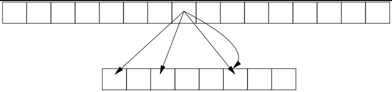
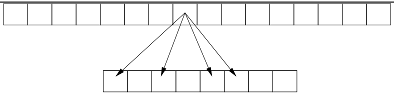
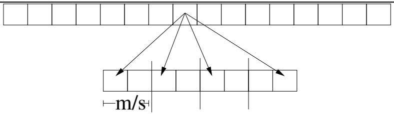


$s = \tilde{\Theta}(\varepsilon^{-1} \log^2(1/\delta))$

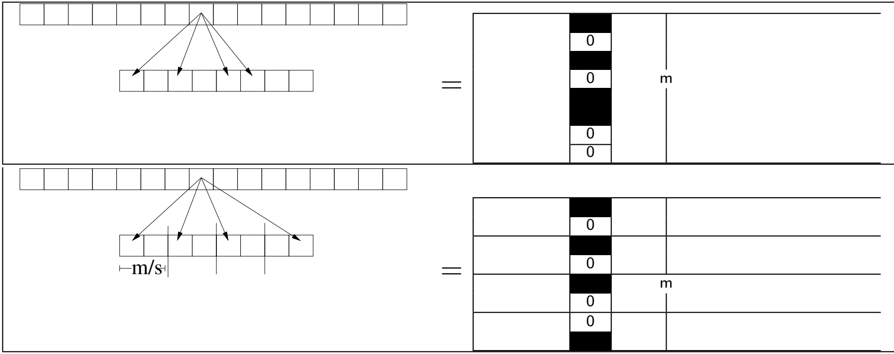
Sparse JL Constructions

[DKS, 2010]		$s = \tilde{\Theta}(\varepsilon^{-1} \log^2(1/\delta))$
[KN12]		$s = \Theta(\varepsilon^{-1} \log(1/\delta))$

Sparse JL Constructions

[DKS, 2010]		$s = \tilde{\Theta}(\varepsilon^{-1} \log^2(1/\delta))$
[KN12]		$s = \Theta(\varepsilon^{-1} \log(1/\delta))$
[KN12]		$s = \Theta(\varepsilon^{-1} \log(1/\delta))$

Sparse JL Constructions (in matrix form)



Each black cell is $\pm 1/\sqrt{s}$ at random

Analysis

- In both constructions, can write $\Pi_{i,j} = \delta_{i,j}\sigma_{i,j}/\sqrt{s}$

$$\|\Pi u\|_2^2 - 1 = \frac{1}{s}\sum_{r=1}^{m}\sum_{i \neq j}\delta_{r,i}\delta_{r,j}\sigma_{r,i}\sigma_{r,j}u_i u_j = \sigma^T B \sigma$$

Analysis

- In both constructions, can write $\Pi_{i,j} = \delta_{i,j}\sigma_{i,j}/\sqrt{s}$

$$\|\Pi u\|_2^2 - 1 = \frac{1}{s}\sum_{r=1}^{m}\sum_{i \neq j} \delta_{r,i}\delta_{r,j}\sigma_{r,i}\sigma_{r,j}u_i u_j = \sigma^T B \sigma$$

$$B = \frac{1}{s} \cdot \begin{bmatrix} B_1 & 0 & \dots & 0 \\ 0 & B_2 & \dots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & \dots & 0 & B_m \end{bmatrix}$$

- $(B_r)_{i,j} = \delta_{r,i}\delta_{r,j}x_i x_j$

Analysis

- In both constructions, can write $\Pi_{i,j} = \delta_{i,j}\sigma_{i,j}/\sqrt{s}$

$$\|\Pi u\|_2^2 - 1 = \frac{1}{s}\sum_{r=1}^{m}\sum_{i\neq j}\delta_{r,i}\delta_{r,j}\sigma_{r,i}\sigma_{r,j}u_i u_j = \sigma^T B\sigma$$

$$B = \frac{1}{s}\cdot \begin{bmatrix} B_1 & 0 & \dots & 0 \\ 0 & B_2 & \dots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & \dots & 0 & B_m \end{bmatrix}$$

- $(B_r)_{i,j} = \delta_{r,i}\delta_{r,j}x_i x_j$
- $\mathbb{P}(|\|\Pi u\|^2 - 1| > \varepsilon) < \varepsilon^{-\ell}\cdot\mathbb{E}\,|\|\Pi u\|^2 - 1|^{\ell}$. Use moment bound for quadratic forms, which depends on $\|B\|, \|B\|_F$ (Hanson-Wright inequality).

What next?

Natural "matrix extension" of sparse JL

[Kane, N. '12]

Theorem

Let $u \in \mathbb{R}^n$ be arbitrary, unit ℓ_2 norm, Π sparse sign matrix. Then

$$\mathbb{P}_{\Pi}\left(\left|\|\Pi u\|^2 - 1\right| > \varepsilon\right) < \delta$$

as long as

$$m \gtrsim \frac{\log(1/\delta)}{\varepsilon^2}, s \gtrsim \frac{\log(1/\delta)}{\varepsilon}, \ell = \log(1/\delta)$$

or

$$m \gtrsim \frac{1}{\varepsilon^2\delta}, s = 1, \ell = 2 \text{ ([Thorup, Zhang'04])}$$

Natural “matrix extension” of sparse JL

[Kane, N. '12]

Theorem

Let $u \in \mathbb{R}^{n\times 1}$ be arbitrary, o.n. cols, Π sparse sign matrix. Then

$$\mathbb{P}_{\Pi}(\|(\Pi u)^T(\Pi u) - I_1\| > \varepsilon) < \delta$$

as long as

$$m \gtrsim \frac{1+\log(1/\delta)}{\varepsilon^2}, s \gtrsim \frac{\log(1/\delta)}{\varepsilon}, \ell = \log(1/\delta)$$

or

$$m \gtrsim \frac{1^2}{\varepsilon^2\delta}, s = 1, \ell = 2$$

Natural "matrix extension" of sparse JL

Conjecture

Theorem

Let $u \in \mathbb{R}^{n \times d}$ be arbitrary, o.n. cols, Π sparse sign matrix. Then

$$\mathbb{P}_{\Pi}(\|(\Pi u)^T(\Pi u) - I_d\| > \varepsilon) < \delta$$

as long as

$$m \gtrsim \frac{d + \log(1/\delta)}{\varepsilon^2}, s \gtrsim \frac{\log(d/\delta)}{\varepsilon}, \ell = \log(d/\delta)$$

or

$$m \gtrsim \frac{d^2}{\varepsilon^2 \delta}, s = 1, \ell = 2$$

Natural "matrix extension" of sparse JL

What we prove [N., Nguyễn '13]

Theorem

Let $u \in \mathbb{R}^{n \times d}$ be arbitrary, o.n. cols, Π sparse sign matrix. Then

$$\mathbb{P}_{\Pi}(\|(\Pi u)^T(\Pi u) - I_d\| > \varepsilon) < \delta$$

as long as

$$m \gtrsim \frac{d \cdot \log^c(d/\delta)}{\varepsilon^2}, s \gtrsim \frac{\log^c(d/\delta)}{\varepsilon} \text{ or } m \gtrsim \frac{d^{1.01}}{\varepsilon^2}, s \gtrsim \frac{1}{\varepsilon}$$

or

$$m \gtrsim \frac{d^2}{\varepsilon^2 \delta}, s = 1$$

Remarks

- [Clarkson, Woodruff '13] was first to show $m = d^2 \cdot \mathrm{polylog}(d/\varepsilon), s = 1$ bound via other methods
- $m = O(d^2/\varepsilon^2), s = 1$ also obtained by [Mahoney, Meng '13].
- $m = O(d^2/\varepsilon^2), s = 1$ also follows from [Thorup, Zhang '04] + [Kane, N. '12] (observed by Nguyễn)

Remarks

- [Clarkson, Woodruff '13] was first to show $m = d^2 \cdot \mathrm{polylog}(d/\varepsilon), s = 1$ bound via other methods
- $m = O(d^2/\varepsilon^2), s = 1$ also obtained by [Mahoney, Meng '13].
- $m = O(d^2/\varepsilon^2), s = 1$ also follows from [Thorup, Zhang '04] + [Kane, N. '12] (observed by Nguyễn)
- What does the "moment method" mean for matrices?

$$\mathop{\mathbb{P}}_{\Pi}(\|(\Pi u)^T(\Pi u) - I_d\| > \varepsilon) < \varepsilon^{-\ell} \cdot \mathbb{E}\,\|(\Pi u)^T(\Pi u) - I_d\|^{\ell}$$

$$\leq \varepsilon^{-\ell} \cdot \mathbb{E}\,\mathrm{tr}(((\Pi u)^T(\Pi u) - I_d)^{\ell})$$

- Classical "moment method" in random matrix theory; e.g. [Wigner, 1955], [Füredi, Komlós, 1981], [Bai, Yin, 1993]

Who cares about this matrix extension?

Motivation for matrix extension of sparse JL

- $\|(\Pi U)^T(\Pi U) - I\| \le \varepsilon$ equivalent to $\|\Pi x\| = (1 \pm \varepsilon)\|x\|$ for all $x \in V$, where V is the subspace spanned by the columns of U (up to changing ε by a factor of 2). "subspace embedding".

Motivation for matrix extension of sparse JL

- $\|(\Pi U)^T(\Pi U) - I\| \le \varepsilon$ equivalent to $\|\Pi x\| = (1 \pm \varepsilon)\|x\|$ for all $x \in V$, where V is the subspace spanned by the columns of U (up to changing ε by a factor of 2). "subspace embedding".
- Subspace embeddings can be used to speed up algorithms for many numerical linear algebra problems on big matrices [Sarlós, 2006], [Dasgupta, Drineas, Harb, Kumar, Mahoney, 2008], [Clarkson, Woodruff, 2009], [Drineas, Magdon-Ismail, Mahoney, Woodruff, 2012], [Clarkson, Woodruff, 2013], [Clarkson, Drineas, Magdon-Ismail, Mahoney, Meng, Woodruff, 2013], [Woodruff, Zhang, 2013], ...

Motivation for matrix extension of sparse JL

- $\|(\Pi U)^T(\Pi U) - I\| \le \varepsilon$ equivalent to $\|\Pi x\| = (1 \pm \varepsilon)\|x\|$ for all $x \in V$, where V is the subspace spanned by the columns of U (up to changing ε by a factor of 2). "subspace embedding".
- Subspace embeddings can be used to speed up algorithms for many numerical linear algebra problems on big matrices [Sarlós, 2006], [Dasgupta, Drineas, Harb, Kumar, Mahoney, 2008], [Clarkson, Woodruff, 2009], [Drineas, Magdon-Ismail, Mahoney, Woodruff, 2012], [Clarkson, Woodruff, 2013], [Clarkson, Drineas, Magdon-Ismail, Mahoney, Meng, Woodruff, 2013], [Woodruff, Zhang, 2013], ...
- Sparse Π: can multiply ΠA in $s \cdot \mathrm{nnz}(A)$ time for big matrix A.

Numerical linear algebra

- $A \in \mathbb{R}^{n \times d}$, $n \gg d$, $rank(A) = r$

Numerical linear algebra

- $A \in \mathbb{R}^{n \times d}$, $n \gg d$, $rank(A) = r$

Classical numerical linear algebra problems

- Compute the **leverage scores** of A, i.e. the ℓ_2 norms of the n standard basis vectors when projected onto the subspace spanned by the columns of A.

Numerical linear algebra

- $A \in \mathbb{R}^{n \times d}$, $n \gg d$, $rank(A) = r$

Classical numerical linear algebra problems

- Compute the **leverage scores** of A, i.e. the ℓ_2 norms of the n standard basis vectors when projected onto the subspace spanned by the columns of A.
- **Least squares regression**: Given also $b \in \mathbb{R}^n$.

$$\text{Compute } x^* = \operatorname{argmin}_{x \in \mathbb{R}^d} \|Ax - b\|_2$$

Numerical linear algebra

- $A \in \mathbb{R}^{n \times d}$, $n \gg d$, $rank(A) = r$

Classical numerical linear algebra problems

- Compute the **leverage scores** of A, i.e. the ℓ_2 norms of the n standard basis vectors when projected onto the subspace spanned by the columns of A.
- **Least squares regression**: Given also $b \in \mathbb{R}^n$.

$$\text{Compute } x^* = \text{argmin}_{x \in \mathbb{R}^d} \|Ax - b\|_2$$

- ℓ_p **regression** $(p \in [1, \infty))$:

$$\text{Compute } x^* = \text{argmin}_{x \in \mathbb{R}^d} \|Ax - b\|_p$$

Numerical linear algebra

- $A \in \mathbb{R}^{n \times d}$, $n \gg d$, $rank(A) = r$

Classical numerical linear algebra problems

- Compute the **leverage scores** of A, i.e. the ℓ_2 norms of the n standard basis vectors when projected onto the subspace spanned by the columns of A.
- **Least squares regression**: Given also $b \in \mathbb{R}^n$.

$$\text{Compute } x^* = \text{argmin}_{x \in \mathbb{R}^d} \|Ax - b\|_2$$

- ℓ_p **regression** $(p \in [1, \infty))$:

$$\text{Compute } x^* = \text{argmin}_{x \in \mathbb{R}^d} \|Ax - b\|_p$$

- **Low-rank approximation**: Given also an integer $1 \leq k \leq d$.

$$\text{Compute } A_k = \text{argmin}_{\text{rank}(B) \leq k} \|A - B\|_F$$

Numerical linear algebra

- $A \in \mathbb{R}^{n \times d}$, $n \gg d$, $rank(A) = r$

Classical numerical linear algebra problems

- Compute the **leverage scores** of A, i.e. the ℓ_2 norms of the n standard basis vectors when projected onto the subspace spanned by the columns of A.
- **Least squares regression**: Given also $b \in \mathbb{R}^n$.
$$\text{Compute } x^* = \text{argmin}_{x \in \mathbb{R}^d} \|Ax - b\|_2$$
- ℓ_p **regression** $(p \in [1, \infty))$:
$$\text{Compute } x^* = \text{argmin}_{x \in \mathbb{R}^d} \|Ax - b\|_p$$
- **Low-rank approximation**: Given also an integer $1 \leq k \leq d$.
$$\text{Compute } A_k = \text{argmin}_{\text{rank}(B) \leq k} \|A - B\|_F$$
- **Preconditioning**: Compute $R \in \mathbb{R}^{d \times d}$ (for $d = r$) so that
$$\forall x \ \|ARx\|_2 \approx \|x\|_2$$

Computationally efficient solutions

Singular Value Decomposition

Theorem

Every matrix $A \in \mathbb{R}^{n \times d}$ of rank r can be written as

$$A = \underbrace{U}_{\substack{\text{orthonorm} \\ \text{columns} \\ n \times r}} \quad \underbrace{\Sigma}_{\substack{\text{diagonal} \\ \text{positive definite} \\ r \times r}} \quad \underbrace{V^T}_{\substack{\text{orthonorm} \\ \text{columns} \\ d \times r}}$$

Can compute SVD in $\tilde{O}(nd^{\omega-1})$ [Demmel, Dumitriu, Holtz, 2007].
$\omega < 2.373\ldots$ is the exponent of square matrix multiplication [Coppersmith, Winograd, 1987], [Stothers, 2010], [Vassilevska-Williams, 2012]

Computationally efficient solutions

$$A = \underbrace{U}_{\substack{orthonorm\\columns\\n\times r}} \underbrace{\Sigma}_{\substack{diagonal\\positive\ definite\\r\times r}} \underbrace{V^T}_{\substack{orthonorm\\columns\\d\times r}}$$

- **Leverage scores**: Output row norms of U.
- **Least squares regression**: Output $V\Sigma^{-1}U^T b$.
- **Low-rank approximation**: Output $U\Sigma_k V^T$.
- **Preconditioning**: Output $R = V\Sigma^{-1}$.

Computationally efficient solutions

$$A = \underbrace{U}_{\substack{orthonorm \\ columns \\ n \times r}} \underbrace{\Sigma}_{\substack{diagonal \\ positive\ definite \\ r \times r}} \underbrace{V^T}_{\substack{orthonorm \\ columns \\ d \times r}}$$

- **Leverage scores**: Output row norms of U.
- **Least squares regression**: Output $V\Sigma^{-1}U^T b$.
- **Low-rank approximation**: Output $U\Sigma_k V^T$.
- **Preconditioning**: Output $R = V\Sigma^{-1}$.

Conclusion: In time $\tilde{O}(nd^{\omega-1})$ we can compute the SVD then solve all the previously stated problems. Is there a faster way?

How to use subspace embeddings

Least squares regression: Let Π be a subspace embedding for the subspace spanned by b and the columns of A. Let $x^* = \operatorname{argmin} \|Ax - b\|$ and $\tilde{x} = \operatorname{argmin} \|\Pi Ax - \Pi b\|$. Then

How to use subspace embeddings

Least squares regression: Let Π be a subspace embedding for the subspace spanned by b and the columns of A. Let $x^* = \operatorname{argmin} \|Ax - b\|$ and $\tilde{x} = \operatorname{argmin} \|\Pi Ax - \Pi b\|$. Then

$$\|\Pi A\tilde{x} - \Pi b\| \leq \|\Pi Ax^* - \Pi b\|$$

How to use subspace embeddings

Least squares regression: Let Π be a subspace embedding for the subspace spanned by b and the columns of A. Let $x^* = \operatorname{argmin} \|Ax - b\|$ and $\tilde{x} = \operatorname{argmin} \|\Pi Ax - \Pi b\|$. Then

$$(1-\varepsilon)\|A\tilde{x}-b\| \leq \underbrace{\|\Pi A\tilde{x} - \Pi b\|}_{\|\Pi(A\tilde{x}-b)\|} \leq \|\Pi Ax^* - \Pi b\|$$

How to use subspace embeddings

Least squares regression: Let Π be a subspace embedding for the subspace spanned by b and the columns of A. Let $x^* = \mathrm{argmin}\, \|Ax - b\|$ and $\tilde{x} = \mathrm{argmin}\, \|\Pi Ax - \Pi b\|$. Then

$$(1-\varepsilon)\|A\tilde{x}-b\| \leq \|\Pi A\tilde{x}-\Pi b\| \leq \|\Pi Ax^*-\Pi b\| \leq (1+\varepsilon)\|Ax^*-b\|$$

$$\Rightarrow \|A\tilde{x} - b\| \leq \left(\frac{1+\varepsilon}{1-\varepsilon}\right) \cdot \|Ax^* - b\|$$

How to use subspace embeddings

Least squares regression: Let Π be a subspace embedding for the subspace spanned by b and the columns of A. Let $x^* = \mathrm{argmin}\,\|Ax - b\|$ and $\tilde{x} = \mathrm{argmin}\,\|\Pi Ax - \Pi b\|$. Then

$$(1-\varepsilon)\|A\tilde{x}-b\| \le \|\Pi A\tilde{x}-\Pi b\| \le \|\Pi Ax^*-\Pi b\| \le (1+\varepsilon)\|Ax^*-b\|$$

$$\Rightarrow \|A\tilde{x} - b\| \le \left(\frac{1+\varepsilon}{1-\varepsilon}\right) \cdot \|Ax^* - b\|$$

Computing SVD of ΠA takes time $\tilde{O}(md^{\omega-1})$, which is much faster than $\tilde{O}(nd^{\omega-1})$ since $m \ll n$.

Back to the analysis

$$\mathbb{P}_{\Pi}\left(\left\|(\Pi U)^T(\Pi U) - I_d\right\| > \varepsilon\right) < \varepsilon^{-\ell} \cdot \mathbb{E}\,\mathrm{tr}(((\Pi U)^T(\Pi U) - I_d)^{\ell})$$

Analysis ($\ell = 2$)
$s = 1$, $m = O(d^2/\varepsilon^2)$

Want to understand $S - I$, $S = (\Pi U)^T(\Pi U)$

Analysis ($\ell = 2$)
$s = 1$, $m = O(d^2/\varepsilon^2)$

Want to understand $S - I$, $S = (\Pi U)^T(\Pi U)$

Let the columns of U be $u^1, \ldots, u^d$
Recall $\Pi_{i,j} = \delta_{i,j}\sigma_{i,j}/\sqrt{s}$

Analysis ($\ell = 2$)
$s = 1$, $m = O(d^2/\varepsilon^2)$

Want to understand $S - I$, $S = (\Pi U)^T(\Pi U)$

Let the columns of U be $u^1, \ldots, u^d$
Recall $\Pi_{i,j} = \delta_{i,j}\sigma_{i,j}/\sqrt{s}$

Some computations yield

$$(S - I)_{k,k'} = \frac{1}{s}\sum_{r=1}^{m}\sum_{i \neq j}\delta_{r,i}\delta_{r,j}\sigma_{r,i}\sigma_{r,j}u_i^k u_j^{k'}$$

Analysis ($\ell = 2$)
$s = 1,\ m = O(d^2/\varepsilon^2)$

Want to understand $S - I$, $S = (\Pi U)^T(\Pi U)$

Let the columns of U be $u^1, \ldots, u^d$
Recall $\Pi_{i,j} = \delta_{i,j}\sigma_{i,j}/\sqrt{s}$

Some computations yield

$$(S - I)_{k,k'} = \frac{1}{s}\sum_{r=1}^{m}\sum_{i \neq j}\delta_{r,i}\delta_{r,j}\sigma_{r,i}\sigma_{r,j}u_i^k u_j^{k'}$$

Computing $\mathbb{E}\,\mathrm{tr}((S - I)^2) = \mathbb{E}\,\|S - I\|_F^2$ is straightforward, and can show $\mathbb{E}\,\|S - I\|_F^2 \leq (d^2 + d)/m$

$$\mathbb{P}(\|S - I\| > \varepsilon) < \frac{1}{\varepsilon^2}\frac{d^2 + d}{m}$$

Analysis ($\ell = 2$)
$s = 1$, $m = O(d^2/\varepsilon^2)$

Want to understand $S - I$, $S = (\Pi U)^T(\Pi U)$

Let the columns of U be $u^1, \ldots, u^d$
Recall $\Pi_{i,j} = \delta_{i,j}\sigma_{i,j}/\sqrt{s}$

Some computations yield

$$(S - I)_{k,k'} = \frac{1}{s}\sum_{r=1}^{m}\sum_{i \neq j}\delta_{r,i}\delta_{r,j}\sigma_{r,i}\sigma_{r,j}u_i^k u_j^{k'}$$

Computing $\mathbb{E}\,\mathrm{tr}((S - I)^2) = \mathbb{E}\,\|S - I\|_F^2$ is straightforward, and can show $\mathbb{E}\,\|S - I\|_F^2 \le (d^2 + d)/m$

$$\mathbb{P}(\|S - I\| > \varepsilon) < \frac{1}{\varepsilon^2}\frac{d^2 + d}{m}$$

Set $m \ge \delta^{-1}(d^2 + d)/\varepsilon^2$ for success probability $1 - \delta$

Analysis (large ℓ)
$s = O_\gamma(1/\varepsilon)$, $m = O(d^{1+\gamma}/\varepsilon^2)$

$$(S - I)_{k,k'} = \frac{1}{s} \sum_{r=1}^{m} \sum_{i \neq j} \delta_{r,i}\delta_{r,j}\sigma_{r,i}\sigma_{r,j} u_i^k u_j^{k'}$$

Analysis (large ℓ) $s = O_\gamma(1/\varepsilon)$, $m = O(d^{1+\gamma}/\varepsilon^2)$

$$(S - I)_{k,k'} = \frac{1}{s} \sum_{r=1}^{m} \sum_{i \neq j} \delta_{r,i}\delta_{r,j}\sigma_{r,i}\sigma_{r,j} u_i^k u_j^{k'}$$

By induction, for any square matrix B and integer $\ell \geq 1$,

$$(B^\ell)_{i,j} = \sum_{\substack{i_1,\ldots,i_{\ell+1} \\ i_1=i, i_{\ell+1}=j}} \prod_{t=1}^{\ell} B_{i_t, i_{t+1}}$$

Analysis (large ℓ)
$s = O_\gamma(1/\varepsilon)$, $m = O(d^{1+\gamma}/\varepsilon^2)$

$$(S-I)_{k,k'} = \frac{1}{s}\sum_{r=1}^{m}\sum_{i\neq j}\delta_{r,i}\delta_{r,j}\sigma_{r,i}\sigma_{r,j}u_i^k u_j^{k'}$$

By induction, for any square matrix B and integer $\ell \geq 1$,

$$(B^\ell)_{i,j} = \sum_{\substack{i_1,\ldots,i_{\ell+1}\\ i_1=i,\, i_{\ell+1}=j}} \prod_{t=1}^{\ell} B_{i_t,i_{t+1}}$$

$$\Rightarrow \mathrm{tr}(B^\ell) = \sum_{\substack{i_1,\ldots,i_{\ell+1}\\ i_1=i_{\ell+1}}} \prod_{t=1}^{\ell} B_{i_t,i_{t+1}}$$

Analysis (large ℓ)
$s = O_\gamma(1/\varepsilon)$, $m = O(d^{1+\gamma}/\varepsilon^2)$

$$\mathbb{E}\,\mathrm{tr}((S-I)^\ell) = \sum_{\substack{i_1 \neq j_1, \ldots, i_\ell \neq j_\ell \\ r_1, \ldots, r_\ell \\ k_1, \ldots, k_{\ell+1} \\ k_1 = k_{\ell+1}}} \left(\mathbb{E} \prod_{t=1}^{\ell} \delta_{r_t, i_t} \delta_{r_t, j_t} \right) \left(\mathbb{E} \prod_{t=1}^{\ell} \sigma_{r_t, i_t} \sigma_{r_t, j_t} \right) \prod_{t=1}^{\ell} u_{i_t}^{k_t} u_{j_t}^{k_{t+1}}$$

Analysis (large ℓ)
$s = O_\gamma(1/\varepsilon)$, $m = O(d^{1+\gamma}/\varepsilon^2)$

$$\mathbb{E}\,\mathrm{tr}((S-I)^\ell) = \sum_{\substack{i_1\neq j_1,\ldots,i_\ell\neq j_\ell\\ r_1,\ldots,r_\ell\\ k_1,\ldots,k_{\ell+1}\\ k_1=k_{\ell+1}}} \left(\mathbb{E}\prod_{t=1}^{\ell}\delta_{r_t,i_t}\delta_{r_t,j_t}\right)\left(\mathbb{E}\prod_{t=1}^{\ell}\sigma_{r_t,i_t}\sigma_{r_t,j_t}\right)\prod_{t=1}^{\ell}u_{i_t}^{k_t}u_{j_t}^{k_{t+1}}$$

The strategy: Associate each monomial in summation above with a graph, group monomials that have the same graph, and estimate the contribution of each graph then do some combinatorics

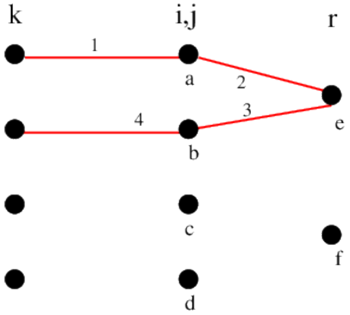
(a common strategy; see [Wigner, 1955], [Füredi, Komlós, 1981], [Bai, Yin, 1993])

Example monomial→graph correspondence

$$\mathrm{tr}((S-I)^{\ell}) = \sum_{\substack{i_1\neq j_1,\ldots,i_\ell\neq j_\ell \\ r_1,\ldots,r_\ell \\ k_1,\ldots,k_{\ell+1} \\ k_1=k_{\ell+1}}} \prod_{t=1}^{\ell} \delta_{r_t,i_t}\delta_{r_t,j_t} \cdot \prod_{t=1}^{\ell} \sigma_{r_t,i_t}\sigma_{r_t,j_t} \cdot \prod_{t=1}^{\ell} u_{i_t}^{k_t} u_{j_t}^{k_{t+1}}$$

$$\ell = 4$$

$$\delta_{r_e,i_a}\delta_{r_e,i_b}\sigma_{r_e,i_a}\sigma_{r_e,i_b}u_{i_a}^{k_1}u_{i_b}^{k_2}$$

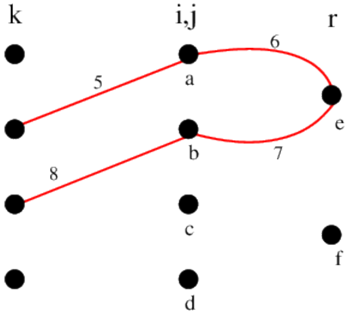


Example monomial→graph correspondence

$$\mathrm{tr}((S-I)^{\ell}) = \sum_{\substack{i_1\neq j_1,\ldots,i_\ell\neq j_\ell \\ r_1,\ldots,r_\ell \\ k_1,\ldots,k_{\ell+1} \\ k_1=k_{\ell+1}}} \prod_{t=1}^{\ell} \delta_{r_t,i_t}\delta_{r_t,j_t} \cdot \prod_{t=1}^{\ell} \sigma_{r_t,i_t}\sigma_{r_t,j_t} \cdot \prod_{t=1}^{\ell} u_{i_t}^{k_t} u_{j_t}^{k_{t+1}}$$

$$\ell = 4$$

$$\times \delta_{r_e,i_a}\delta_{r_e,i_b}\sigma_{r_e,i_a}\sigma_{r_e,i_b} u_{i_a}^{k_2} u_{i_b}^{k_3}$$

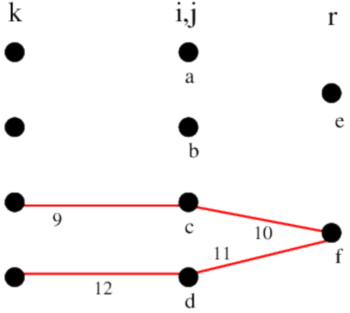


Example monomial→graph correspondence

$$\mathrm{tr}((S-I)^{\ell}) = \sum_{\substack{i_1 \neq j_1, \ldots, i_\ell \neq j_\ell \\ r_1, \ldots, r_\ell \\ k_1, \ldots, k_{\ell+1} \\ k_1 = k_{\ell+1}}} \prod_{t=1}^{\ell} \delta_{r_t, i_t} \delta_{r_t, j_t} \cdot \prod_{t=1}^{\ell} \sigma_{r_t, i_t} \sigma_{r_t, j_t} \cdot \prod_{t=1}^{\ell} u_{i_t}^{k_t} u_{j_t}^{k_{t+1}}$$

$$\ell = 4$$

$$\times \delta_{r_f, i_c} \delta_{r_f, i_d} \sigma_{r_f, i_c} \sigma_{r_f, i_d} u_{i_c}^{k_3} u_{i_d}^{k_4}$$

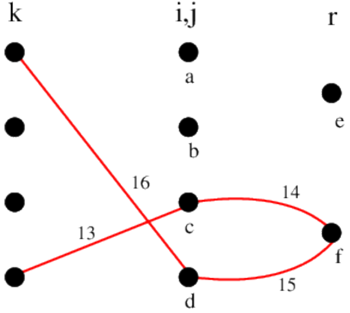


Example monomial→graph correspondence

$$\mathrm{tr}((S-I)^\ell) = \sum_{\substack{i_1 \neq j_1, \ldots, i_\ell \neq j_\ell \\ r_1, \ldots, r_\ell \\ k_1, \ldots, k_{\ell+1} \\ k_1 = k_{\ell+1}}} \prod_{t=1}^{\ell} \delta_{r_t, i_t} \delta_{r_t, j_t} \cdot \prod_{t=1}^{\ell} \sigma_{r_t, i_t} \sigma_{r_t, j_t} \cdot \prod_{t=1}^{\ell} u_{i_t}^{k_t} u_{j_t}^{k_{t+1}}$$

$$\ell = 4$$

$$\times \delta_{r_f, i_c} \delta_{r_f, i_d} \sigma_{r_f, i_c} \sigma_{r_f, i_d} u_{i_c}^{k_4} u_{i_d}^{k_1}$$

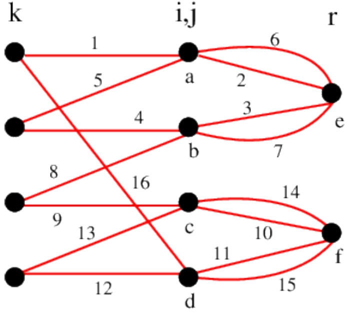


Example monomial→graph correspondence

$$\operatorname{tr}((S-I)^{\ell}) = \sum_{\substack{i_1\neq j_1,\ldots,i_\ell\neq j_\ell\\ r_1,\ldots,r_\ell\\ k_1,\ldots,k_{\ell+1}\\ k_1=k_{\ell+1}}} \prod_{t=1}^{\ell} \delta_{r_t,i_t}\delta_{r_t,j_t} \cdot \prod_{t=1}^{\ell} \sigma_{r_t,i_t}\sigma_{r_t,j_t} \cdot \prod_{t=1}^{\ell} u_{i_t}^{k_t} u_{j_t}^{k_{t+1}}$$

$$\ell = 4$$

$$\begin{aligned} &\delta_{r_e,i_a}\delta_{r_e,i_b}\sigma_{r_e,i_a}\sigma_{r_e,i_b} u_{i_a}^{k_1} u_{i_b}^{k_2} \\ \times &\delta_{r_e,i_a}\delta_{r_e,i_b}\sigma_{r_e,i_a}\sigma_{r_e,i_b} u_{i_a}^{k_2} u_{i_b}^{k_3} \\ \times &\delta_{r_f,i_c}\delta_{r_f,i_d}\sigma_{r_f,i_c}\sigma_{r_f,i_d} u_{i_c}^{k_3} u_{i_d}^{k_4} \\ \times &\delta_{r_f,i_c}\delta_{r_f,i_d}\sigma_{r_f,i_c}\sigma_{r_f,i_d} u_{i_c}^{k_4} u_{i_d}^{k_1} \end{aligned}$$

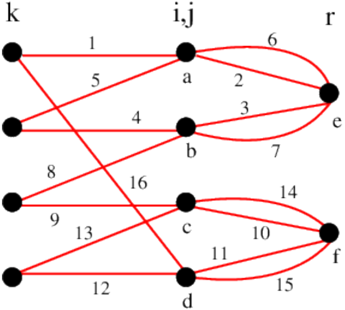


Example monomial→graph correspondence

$$\mathrm{tr}((S-I)^{\ell}) = \sum_{\substack{i_1 \neq j_1, \ldots, i_\ell \neq j_\ell \\ r_1, \ldots, r_\ell}} \prod_{t=1}^{\ell} \delta_{r_t, i_t} \delta_{r_t, j_t} \cdot \prod_{t=1}^{\ell} \sigma_{r_t, i_t} \sigma_{r_t, j_t} \cdot \prod_{t=1}^{\ell} \langle u_{i_t}, u_{i_{t+1}} \rangle$$

$$\ell = 4$$

$$\begin{aligned} &\delta_{r_e,i_a}\delta_{r_e,i_b}\sigma_{r_e,i_a}\sigma_{r_e,i_b} u_{i_a}^{k_1} u_{i_b}^{k_2} \\ \times &\delta_{r_e,i_a}\delta_{r_e,i_b}\sigma_{r_e,i_a}\sigma_{r_e,i_b} u_{i_a}^{k_2} u_{i_b}^{k_3} \\ \times &\delta_{r_f,i_c}\delta_{r_f,i_d}\sigma_{r_f,i_c}\sigma_{r_f,i_d} u_{i_c}^{k_3} u_{i_d}^{k_4} \\ \times &\delta_{r_f,i_c}\delta_{r_f,i_d}\sigma_{r_f,i_c}\sigma_{r_f,i_d} u_{i_c}^{k_4} u_{i_d}^{k_1} \end{aligned}$$



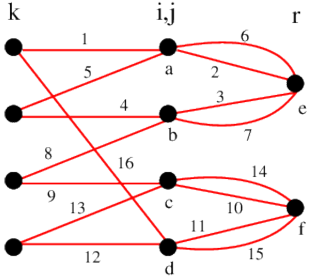
Grouping monomials by graph

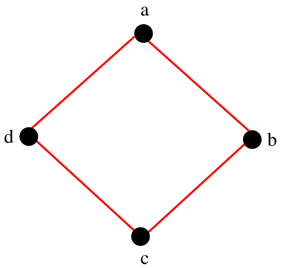
z right vertices, *b* distinct edges between middle and right

$$\mathbb{E}\,\mathrm{tr}((S-I)^{\ell}) = \sum_{\substack{i_1\neq j_1,\ldots,i_\ell\neq j_\ell\\ r_1,\ldots,r_\ell}} \left(\mathbb{E}\prod_{t=1}^{\ell}\delta_{r_t,i_t}\delta_{r_t,j_t}\right)\left(\mathbb{E}\prod_{t=1}^{\ell}\sigma_{r_t,i_t}\sigma_{r_t,j_t}\right)\prod_{t=1}^{\ell}\langle u_{i_t}, u_{i_{t+1}}\rangle$$

$$\leq \sum_{G} m^{z}\left(\frac{s}{m}\right)^{b}\left|\sum_{i_1\neq\ldots\neq i_y}\ \prod_{e=(\alpha,\beta)\in\hat{G}}\left\langle u_{i_\alpha}, u_{i_\beta}\right\rangle\right|$$

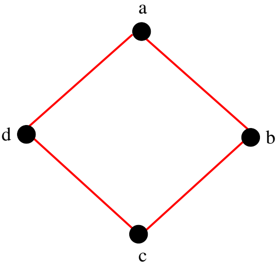
G	$\hat{G}$





Understanding $\hat{G}$

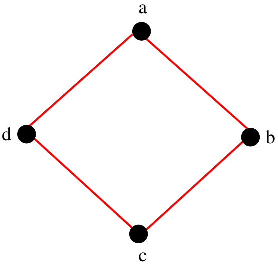
$$F(\hat{G}) = \left| \sum_{i_1 \neq \ldots \neq i_y} \prod_{e=(\alpha,\beta)\in \hat{G}} \left\langle u_{i_\alpha}, u_{i_\beta} \right\rangle \right|$$



Let C be the number of connected components of $\hat{G}$. It turns out the right upper bound for $F(\hat{G})$ is roughly d^C

Understanding $\hat{G}$

$$F(\hat{G}) = \left| \sum_{i_1 \neq \ldots \neq i_y} \prod_{e=(\alpha,\beta)\in\hat{G}} \left\langle u_{i_\alpha}, u_{i_\beta} \right\rangle \right|$$

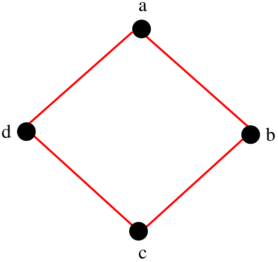


Let C be the number of connected components of $\hat{G}$. It turns out the right upper bound for $F(\hat{G})$ is roughly d^C

- Can get d^C bound if all edges in $\hat{G}$ have even multiplicity

Understanding $\hat{G}$

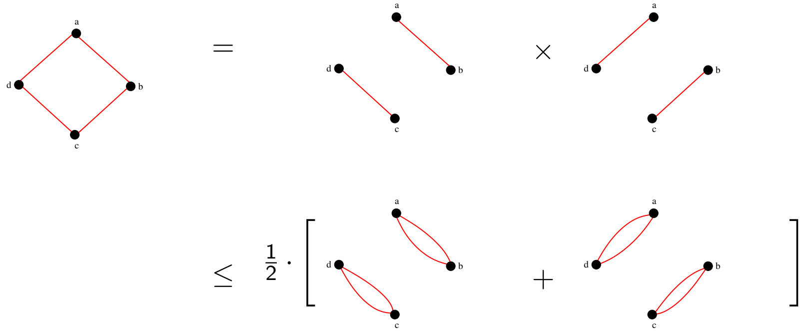
$$F(\hat{G}) = \left| \sum_{i_1 \neq \ldots \neq i_y} \prod_{e=(\alpha,\beta)\in\hat{G}} \left\langle u_{i_\alpha}, u_{i_\beta} \right\rangle \right|$$



Let C be the number of connected components of $\hat{G}$. It turns out the right upper bound for $F(\hat{G})$ is roughly d^C

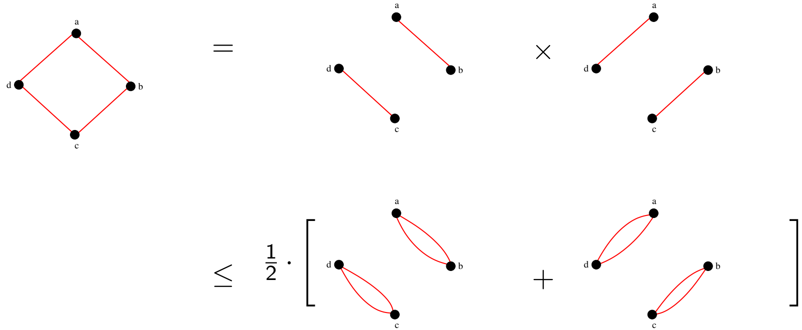
- Can get d^C bound if all edges in $\hat{G}$ have even multiplicity
- How about $\hat{G}$ where this isn't the case, e.g. as above?

Bounding $F(\hat{G})$ with odd multiplicities



Reduces back to case of even edge multiplicities! (AM-GM)

Bounding $F(\hat{G})$ with odd multiplicities



Reduces back to case of even edge multiplicities! (AM-GM)

Caveat: # connected components increased (unacceptable)

AM-GM trick done right

Theorem (Tutte '61, Nash-Williams '61)

Let G be a multigraph with edge-connectivity at least $2k$. Then G must have at least k edge-disjoint spanning trees.

AM-GM trick done right

Theorem (Tutte '61, Nash-Williams '61)

Let G be a multigraph with edge-connectivity at least $2k$. Then G must have at least k edge-disjoint spanning trees.

Using the theorem ($k = 2$)

- If every connected component (CC) of $\hat{G}$ has 2 edge-disjoint spanning trees, we are done

AM-GM trick done right

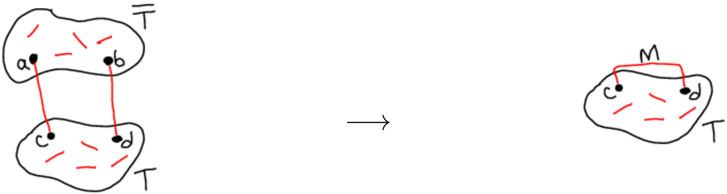
Theorem (Tutte '61, Nash-Williams '61)

Let G be a multigraph with edge-connectivity at least $2k$. Then G must have at least k edge-disjoint spanning trees.

Using the theorem ($k = 2$)

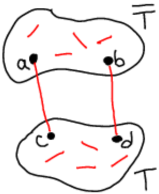
- If every connected component (CC) of $\hat{G}$ has 2 edge-disjoint spanning trees, we are done
- Otherwise, some CC is not 4 edge-connected. Since each CC is Eulerian, there must be a cut of size 2

AM-GM trick done right



$$\sum_{\substack{i_v \\ v \in T}} \left(\prod_{(q,r)\in T} \left\langle u_{i_q}, u_{i_r} \right\rangle \right) u_{i_c}^T \underbrace{\left(\sum_{\substack{i_v \\ v \in \bar{T}}} u_{i_a} \left(\prod_{(q,r)\in \bar{T}} \left\langle u_{i_q}, u_{i_r} \right\rangle \right) u_{i_b}^T \right)}_{M} u_{i_d}$$

AM-GM trick done right



$\longrightarrow$

$$\sum_{\substack{i_v \\ v \in T}} \left(\prod_{(q,r)\in T} \langle u_{i_q}, u_{i_r} \rangle \right) u_{i_c}^T \underbrace{\left(\sum_{\substack{i_v \\ v \in \bar{T}}} u_{i_a} \left(\prod_{(q,r)\in \bar{T}} \langle u_{i_q}, u_{i_r} \rangle \right) u_{i_b}^T \right)}_{M} u_{i_d}$$

- Repeatedly eliminate size-2 cuts until every connected component has two edge-disjoint spanning trees
- Show all M's along the way have bounded operator norm
- Show that even edge multiplicities are still possible to handle when all M's have bounded operator norm

Conclusion

Other recent progress

- Can show any oblivious subspace embedding succeeding with probability $\geq 2/3$ must have $\Omega(d/\varepsilon^2)$ rows [N., Nguyễn]

Other recent progress

- Can show any oblivious subspace embedding succeeding with probability $\geq 2/3$ must have $\Omega(d/\varepsilon^2)$ rows [N., Nguyễn]
- Can show any oblivious subspace embedding with $O(d^{1+\gamma})$ rows must have sparsity $s = \Omega(1/(\varepsilon\gamma))$* [N., Nguyễn]

Other recent progress

- Can show any oblivious subspace embedding succeeding with probability $\geq 2/3$ must have $\Omega(d/\varepsilon^2)$ rows [N., Nguyễn]
- Can show any oblivious subspace embedding with $O(d^{1+\gamma})$ rows must have sparsity $s = \Omega(1/(\varepsilon\gamma))$* [N., Nguyễn]
- Can provide upper bounds on m, s to preserve an arbitrary bounded set $T \subset \mathbb{R}^n$, in terms of the geometry of T, in the style of [Gordon '88], [Klartag, Mendelson '05], [Mendelson, Pajor, Tomczak-Jaegermann '07], [Dirksen '13] (in the current notation, those works analyzed dense Π, i.e. $m = s$) [Bourgain, N.]

Other recent progress

- Can show any oblivious subspace embedding succeeding with probability $\geq 2/3$ must have $\Omega(d/\varepsilon^2)$ rows [N., Nguyễn]
- Can show any oblivious subspace embedding with $O(d^{1+\gamma})$ rows must have sparsity $s = \Omega(1/(\varepsilon\gamma))$* [N., Nguyễn]
- Can provide upper bounds on m, s to preserve an arbitrary bounded set $T \subset \mathbb{R}^n$, in terms of the geometry of T, in the style of [Gordon '88], [Klartag, Mendelson '05], [Mendelson, Pajor, Tomczak-Jaegermann '07], [Dirksen '13] (in the current notation, those works analyzed dense Π, i.e. $m = s$) [Bourgain, N.]

* Has restriction that $1/(\varepsilon\gamma) \ll d$.

Open Problems

- **OPEN:** Improve ω, the exponent of matrix multiplication
- **OPEN:** Find exact algorithm for least squares regression (or any of these problems) in time faster than $\tilde{O}(nd^{\omega-1})$
- **OPEN:** Prove conjecture: to get subsp. embedding with prob. $1-\delta$, can set $m = O((d+\log(1/\delta))/\varepsilon^2), s = O(\log(d/\delta)/\varepsilon)$. Easier: obtain this m with $s = m$ **via moment method**.
- **OPEN:** Show that the tradeoff $m = O(d^{1+\gamma}/\varepsilon^2)$, $s = \mathrm{poly}(1/\gamma)\cdot 1/\varepsilon$ is optimal for any distribution over subspace embeddings (the poly is probably linear)
- **OPEN:** Show that $m = \Omega(d^2/\varepsilon^2)$ is optimal for $s = 1$

  Partial progress: [N., Nguyễn, 2012] shows $m = \Omega(d^2)$