Understanding Sparse JL for Feature Hashing

Meena Jagadeesan (Harvard University)

mjagadeesan@college.harvard.edu

NeurIPS 2019

Dimensionality reduction (ℓ_2-to-ℓ_2)

Informal goal: Project vectors in $\mathbb{R}^n$ to $\mathbb{R}^m$ (for $m << n$) with a linear map while "preserving geometry" (i.e. Euclidean norm distances).

Dimensionality reduction (ℓ_2-to-ℓ_2)

Informal goal: Project vectors in $\mathbb{R}^n$ to $\mathbb{R}^m$ (for $m << n$) with a linear map while "preserving geometry" (i.e. Euclidean norm distances).

Examples: feature hashing, sparse JL transforms (will define soon!)

Dimensionality reduction (ℓ_2-to-ℓ_2)

Informal goal: Project vectors in $\mathbb{R}^n$ to $\mathbb{R}^m$ (for $m << n$) with a linear map while "preserving geometry" (i.e. Euclidean norm distances).

Examples: feature hashing, sparse JL transforms (will define soon!)

Many applications:

- Document classification tasks (Weinberger et al. '09, etc)
- Support Vector Machines (Paul et al. '14)
- k-means/k-medians (Makarychev, Makarychev, Razenshteyn '18)
- Nearest neighbors (Ailon, Chazelle '09, Har-Peled et al. '14, Wei '19)
- Numerical linear algebra (Clarkson and Woodruff '12, Nelson and Nguyen '14, etc.)

Feature hashing (Weinberger et al. '09)

Given vectors in $\mathbb{R}^n$, map to vectors in $\mathbb{R}^m$ where $m << n$.

Feature hashing (Weinberger et al. '09)

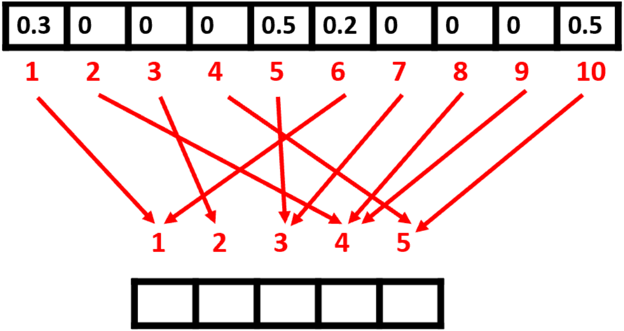
Given vectors in $\mathbb{R}^n$, map to vectors in $\mathbb{R}^m$ where $m << n$.

Use a **hash function** $h : \{1, \ldots, n\} \rightarrow \{1, \ldots, m\}$ on coordinates.

Feature hashing (Weinberger et al. '09)

Given vectors in $\mathbb{R}^n$, map to vectors in $\mathbb{R}^m$ where $m << n$.

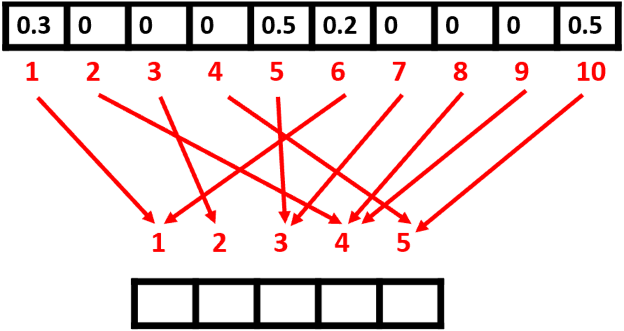
Use a **hash function** $h : \{1, \ldots, n\} \rightarrow \{1, \ldots, m\}$ on coordinates.



Feature hashing (Weinberger et al. '09)

Given vectors in $\mathbb{R}^n$, map to vectors in $\mathbb{R}^m$ where $m << n$.

Use a **hash function** $h : \{1, \ldots, n\} \to \{1, \ldots, m\}$ on coordinates.



But how should collisions be handled?

Feature hashing (Weinberger et al. '09)

Given vectors in $\mathbb{R}^n$, map to vectors in $\mathbb{R}^m$ where $m << n$.

Use a **hash function** $h : \{1, \ldots, n\} \rightarrow \{1, \ldots, m\}$ on coordinates.

Feature hashing (Weinberger et al. '09)

Given vectors in $\mathbb{R}^n$, map to vectors in $\mathbb{R}^m$ where $m << n$.

Use a **hash function** $h : \{1, \ldots, n\} \to \{1, \ldots, m\}$ on coordinates.

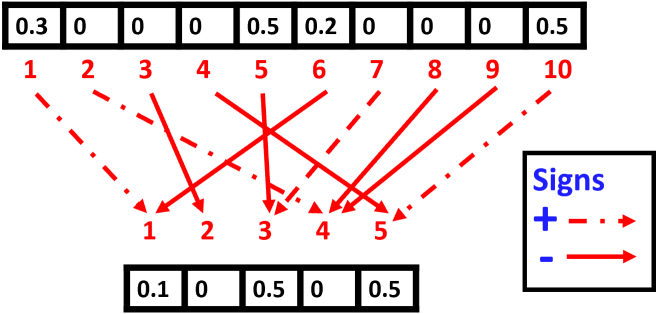
Use **random signs** to handle collisions (unbiased estimator of ℓ_2^2 norm).

Feature hashing (Weinberger et al. '09)

Given vectors in $\mathbb{R}^n$, map to vectors in $\mathbb{R}^m$ where $m << n$.

Use a **hash function** $h : \{1, \dots, n\} \to \{1, \dots, m\}$ on coordinates.

Use **random signs** to handle collisions (unbiased estimator of ℓ_2^2 norm).



Using more than one hash function

Using more than one hash function

Multiple hashing (Weinberger et al. '09):

Using more than one hash function

Multiple hashing (Weinberger et al. '09):

1. Use s hash functions $h_1, h_2, \ldots, h_s : \{1, \ldots, n\} \to \{1, \ldots, m\}$

Using more than one hash function

Multiple hashing (Weinberger et al. '09):

1. Use s hash functions $h_1, h_2, \ldots, h_s : \{1, \ldots, n\} \to \{1, \ldots, m\}$
2. Use random signs to deal with collisions as before.

Using more than one hash function

Multiple hashing (Weinberger et al. '09):

1. Use s hash functions $h_1, h_2, \ldots, h_s : \{1, \ldots, n\} \to \{1, \ldots, m\}$
2. Use random signs to deal with collisions as before.

Can actually do better...

Using more than one hash function

Multiple hashing (Weinberger et al. '09):

1. Use s hash functions $h_1, h_2, \ldots, h_s : \{1, \ldots, n\} \to \{1, \ldots, m\}$
2. Use random signs to deal with collisions as before.

Can actually do better...

Sparse Johnson-Lindenstrauss distributions (Kane, Nelson '12):

Using more than one hash function

Multiple hashing (Weinberger et al. '09):

1. Use s hash functions $h_1, h_2, \ldots, h_s : \{1, \ldots, n\} \to \{1, \ldots, m\}$
2. Use random signs to deal with collisions as before.

Can actually do better...

Sparse Johnson-Lindenstrauss distributions (Kane, Nelson '12):

- *Uniform*: Mildly correlate hash functions so $h_j(i) \neq h_k(i)$.

Using more than one hash function

Multiple hashing (Weinberger et al. '09):

1. Use s hash functions $h_1, h_2, \ldots, h_s : \{1, \ldots, n\} \to \{1, \ldots, m\}$
2. Use random signs to deal with collisions as before.

Can actually do better...

Sparse Johnson-Lindenstrauss distributions (Kane, Nelson '12):

- *Uniform*: Mildly correlate hash functions so $h_j(i) \neq h_k(i)$.
- *Block:* Take $h_i : \{1, \ldots, n\} \to \{(m/s)(i-1)+1, \ldots, (m/s)(i)\}$

Using more than one hash function

Multiple hashing (Weinberger et al. '09):

1. Use s hash functions $h_1, h_2, \ldots, h_s : \{1, \ldots, n\} \to \{1, \ldots, m\}$
2. Use random signs to deal with collisions as before.

Can actually do better...

Sparse Johnson-Lindenstrauss distributions (Kane, Nelson '12):

- *Uniform*: Mildly correlate hash functions so $h_j(i) \neq h_k(i)$.
- *Block:* Take $h_i : \{1, \ldots, n\} \to \{(m/s)(i-1)+1, \ldots, (m/s)(i)\}$

This work (Informal)

Analysis of sparse JL on feature vectors in terms of s.

Using more than one hash function

Multiple hashing (Weinberger et al. '09):

1. Use s hash functions $h_1, h_2, \ldots, h_s : \{1, \ldots, n\} \to \{1, \ldots, m\}$
2. Use random signs to deal with collisions as before.

Can actually do better...

Sparse Johnson-Lindenstrauss distributions (Kane, Nelson '12):

- *Uniform*: Mildly correlate hash functions so $h_j(i) \neq h_k(i)$.
- *Block:* Take $h_i : \{1, \ldots, n\} \to \{(m/s)(i-1)+1, \ldots, (m/s)(i)\}$

This work (Informal)

Analysis of sparse JL on feature vectors in terms of s.
$\implies$ *Sparse JL with ≥ 4 hash functions can have much better norm-preserving properties on feature vectors than feature hashing.*

Mathematical framework

Given vectors in $\mathbb{R}^n$, map to vectors in $\mathbb{R}^m$ where $m << n$.

Mathematical framework

Given vectors in $\mathbb{R}^n$, map to vectors in $\mathbb{R}^m$ where $m << n$.

Use a probability distribution $\mathcal{F}$ over linear maps $f : \mathbb{R}^n \to \mathbb{R}^m$.

Mathematical framework

Given vectors in $\mathbb{R}^n$, map to vectors in $\mathbb{R}^m$ where $m << n$.

Use a probability distribution $\mathcal{F}$ over linear maps $f : \mathbb{R}^n \to \mathbb{R}^m$.

What does it mean to "preserve geometry"?

Mathematical framework

Given vectors in $\mathbb{R}^n$, map to vectors in $\mathbb{R}^m$ where $m << n$.

Use a probability distribution $\mathcal{F}$ over linear maps $f : \mathbb{R}^n \to \mathbb{R}^m$.

What does it mean to "preserve geometry"?

ϵ target error, δ target failure probability

Mathematical framework

Given vectors in $\mathbb{R}^n$, map to vectors in $\mathbb{R}^m$ where $m << n$.

Use a probability distribution $\mathcal{F}$ over linear maps $f : \mathbb{R}^n \to \mathbb{R}^m$.

What does it mean to "preserve geometry"?

ϵ target error, δ target failure probability

For *each* $x \in \mathbb{R}^n$:

$$\mathbb{P}_{f \in \mathcal{F}}[(1 - \epsilon) \|x\|_2 \leq \|f(x)\|_2 \leq (1 + \epsilon) \|x\|_2] > 1 - \delta.$$

Mathematical framework

Given vectors in $\mathbb{R}^n$, map to vectors in $\mathbb{R}^m$ where $m << n$.

Use a probability distribution $\mathcal{F}$ over linear maps $f : \mathbb{R}^n \to \mathbb{R}^m$.

What does it mean to "preserve geometry"?

ϵ target error, δ target failure probability

For *each* $x \in \mathbb{R}^n$:

$$\mathbb{P}_{f \in \mathcal{F}}[(1 - \epsilon) \|x\|_2 \leq \|f(x)\|_2 \leq (1 + \epsilon) \|x\|_2] > 1 - \delta.$$

- Can union bound on a *set* of vectors.

Mathematical framework

Given vectors in $\mathbb{R}^n$, map to vectors in $\mathbb{R}^m$ where $m << n$.

Use a probability distribution $\mathcal{F}$ over linear maps $f : \mathbb{R}^n \to \mathbb{R}^m$.

What does it mean to "preserve geometry"?

ϵ target error, δ target failure probability

For *each* $x \in \mathbb{R}^n$:

$$\mathbb{P}_{f \in \mathcal{F}}[(1-\epsilon)\|x\|_2 \leq \|f(x)\|_2 \leq (1+\epsilon)\|x\|_2] > 1-\delta.$$

- Can union bound on a *set* of vectors.
- Can apply to distances between vectors since f is linear.

Choosing s and m

Notation: ϵ is target error, δ is target failure probability.

Goal: $\mathbb{P}_{f \in \mathcal{F}}[(1-\epsilon)\|x\|_2 \leq \|f(x)\|_2 \leq (1+\epsilon)\|x\|_2] > 1-\delta$.

Choosing s and m

Notation: ϵ is target error, δ is target failure probability.

Goal: $\mathbb{P}_{f \in \mathcal{F}}[(1-\epsilon)\|x\|_2 \leq \|f(x)\|_2 \leq (1+\epsilon)\|x\|_2] > 1-\delta$.

Question

How should the # of hash functions s and dimension m be set?

Choosing s and m

Notation: ϵ is target error, δ is target failure probability.

Goal: $\mathbb{P}_{f \in \mathcal{F}}[(1-\epsilon)\left\|x\right\|_2 \leq \left\|f(x)\right\|_2 \leq (1+\epsilon)\left\|x\right\|_2] > 1-\delta$.

Question

How should the # of hash functions s and dimension m be set?

$m = \Theta(\epsilon^{-2}\log(1/\delta))$, $s = \Theta(\epsilon^{-1}\log(1/\delta))$ (Kane and Nelson '12)

Choosing s and m

Notation: ϵ is target error, δ is target failure probability.

Goal: $\mathbb{P}_{f \in \mathcal{F}}[(1-\epsilon)\left\|x\right\|_2 \leq \left\|f(x)\right\|_2 \leq (1+\epsilon)\left\|x\right\|_2] > 1-\delta$.

Question

How should the # of hash functions s and dimension m be set?

$m = \Theta(\epsilon^{-2}\log(1/\delta))$, $s = \Theta(\epsilon^{-1}\log(1/\delta))$ (Kane and Nelson '12)

But projection time is linear in s...

Choosing s and m

Notation: ϵ is target error, δ is target failure probability.

Goal: $\mathbb{P}_{f \in \mathcal{F}}[(1-\epsilon)\|x\|_2 \leq \|f(x)\|_2 \leq (1+\epsilon)\|x\|_2] > 1-\delta$.

Question

How should the # of hash functions s and dimension m be set?

$m = \Theta(\epsilon^{-2}\log(1/\delta))$, $s = \Theta(\epsilon^{-1}\log(1/\delta))$ (Kane and Nelson '12)

But projection time is linear in s...

$s < \Theta(\epsilon^{-1}\log(1/\delta))$ possible w/ a higher m!

Choosing s and m

Notation: ϵ is target error, δ is target failure probability.

Goal: $\mathbb{P}_{f \in \mathcal{F}}[(1-\epsilon)\|x\|_2 \leq \|f(x)\|_2 \leq (1+\epsilon)\|x\|_2] > 1 - \delta$.

Question

How should the # of hash functions s and dimension m be set?

$m = \Theta(\epsilon^{-2} \log(1/\delta))$, $s = \Theta(\epsilon^{-1} \log(1/\delta))$ (Kane and Nelson '12)

But projection time is linear in s...

$s < \Theta(\epsilon^{-1} \log(1/\delta))$ possible w/ a higher m!

$m = O(\min(2\epsilon^{-2}/\delta, \epsilon^{-2} \log(1/\delta) e^{\Theta(\epsilon^{-1} \log(1/\delta)/s)})$ (Cohen '16).

Sparse JL on feature vectors

Sparse JL on feature vectors

Sometimes a much smaller m works on feature vectors in practice...

Sparse JL on feature vectors

Sometimes a much smaller m works on feature vectors in practice...

In some settings, feature vectors have mass spread out across coordinates.

Sparse JL on feature vectors

Sometimes a much smaller m works on feature vectors in practice...

In some settings, feature vectors have mass spread out across coordinates.

Consider vectors with small ℓ_∞-to-ℓ_2 norm ratio:

$$S_v = \{x \in \mathbb{R}^n \mid \|x\|_\infty \leq v \|x\|_2\},$$

Sparse JL on feature vectors

Sometimes a much smaller m works on feature vectors in practice...

In some settings, feature vectors have mass spread out across coordinates.

Consider vectors with small ℓ_∞-to-ℓ_2 norm ratio:

$$S_v = \{x \in \mathbb{R}^n \mid \|x\|_\infty \leq v \|x\|_2\},$$

$v(m, \epsilon, \delta, s) :=$ inf over $v \in [0, 1]$ s.t. ℓ_2-norm goal is met on $x \in S_v$.

Sparse JL on feature vectors

Sometimes a much smaller m works on feature vectors in practice...

In some settings, feature vectors have mass spread out across coordinates.

Consider vectors with small ℓ_∞-to-ℓ_2 norm ratio:

$$S_v = \{x \in \mathbb{R}^n \mid \|x\|_\infty \leq v \|x\|_2\},$$

$v(m, \epsilon, \delta, s) :=$ inf over $v \in [0, 1]$ s.t. ℓ_2-norm goal is met on $x \in S_v$.

$v(m, \epsilon, \delta, 1)$ understood (Weinberger et al '09,..., Freksen et al. '18)

Sparse JL on feature vectors

Sometimes a much smaller m works on feature vectors in practice...

In some settings, feature vectors have mass spread out across coordinates.

Consider vectors with small ℓ_∞-to-ℓ_2 norm ratio:

$$S_v = \{x \in \mathbb{R}^n \mid \|x\|_\infty \leq v \|x\|_2\},$$

$v(m, \epsilon, \delta, s) :=$ inf over $v \in [0, 1]$ s.t. ℓ_2-norm goal is met on $x \in S_v$.

$v(m, \epsilon, \delta, 1)$ understood (Weinberger et al '09,..., Freksen et al. '18)

This work

Tight bounds on $v(m, \epsilon, \delta, s)$ for a general $s > 1$

Sparse JL on feature vectors

Sometimes a much smaller m works on feature vectors in practice...

In some settings, feature vectors have mass spread out across coordinates.

Consider vectors with small ℓ_∞-to-ℓ_2 norm ratio:

$$S_v = \{x \in \mathbb{R}^n \mid \|x\|_\infty \leq v \|x\|_2\},$$

$v(m, \epsilon, \delta, s) :=$ inf over $v \in [0, 1]$ s.t. ℓ_2-norm goal is met on $x \in S_v$.

$v(m, \epsilon, \delta, 1)$ understood (Weinberger et al '09,..., Freksen et al. '18)

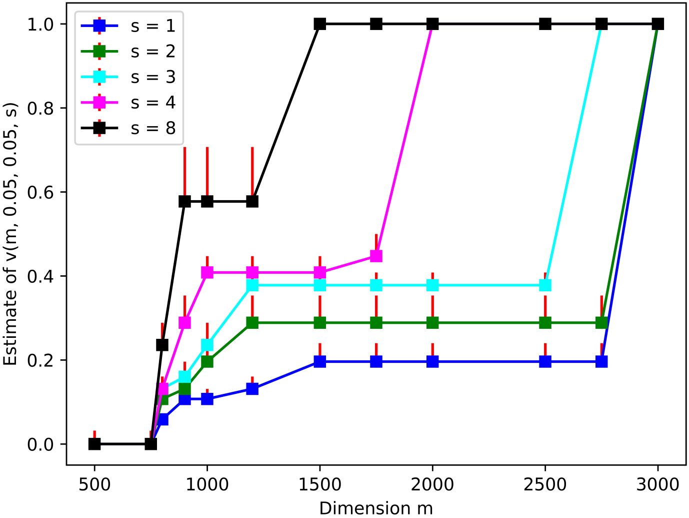
This work

Tight bounds on $v(m, \epsilon, \delta, s)$ for a general $s > 1$

$\implies$ *Characterization of sparse JL performance in terms of ϵ, δ, and ℓ_∞-to-ℓ_2 norm ratio for a general # of hash functions s*

Some intuition for the shape of $v(m, \epsilon, \delta, s)$

Some intuition for the shape of $v(m, \epsilon, \delta, s)$



Main result

Main result

Theorem (Informal)

Under mild conditions, $v(m, \epsilon, \delta, s)$ *is equal to* $f'(m, \epsilon, \ln(1/\delta), s)$*, where*

Main result

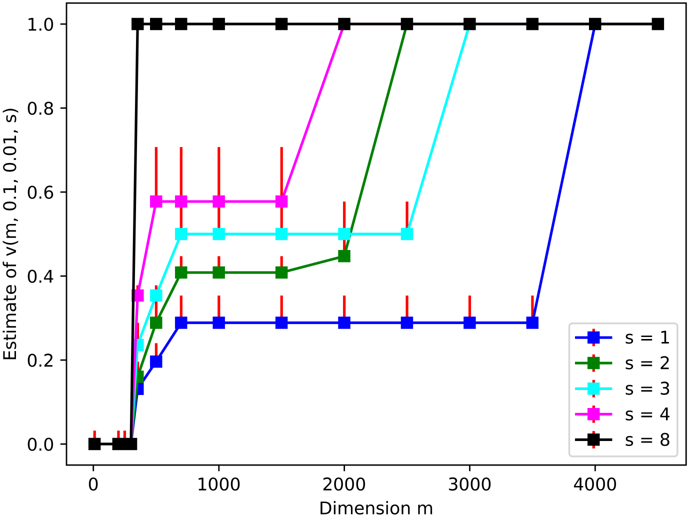
Theorem (Informal)

Under mild conditions, $v(m, \epsilon, \delta, s)$ is equal to $f'(m, \epsilon, \ln(1/\delta), s)$, where

$$f'(m, \epsilon, p, s) = \begin{cases} 1 & \textit{High } m\textit{, full norm preservation} \\ \Theta\left(\sqrt{\epsilon s}\frac{\sqrt{\ln(\frac{m\epsilon^2}{p})}}{\sqrt{p}}\right) & \textit{Medium } m\textit{, middle regime} \\ \Theta\left(\sqrt{\epsilon s}\min\left(\frac{\ln(\frac{m\epsilon}{p})}{p}, \frac{\sqrt{\ln(\frac{m\epsilon^2}{p})}}{\sqrt{p}}\right)\right) & \textit{Medium } m\textit{, middle regime} \\ 0 & \textit{Small } m\textit{, no norm preservation} \end{cases}$$

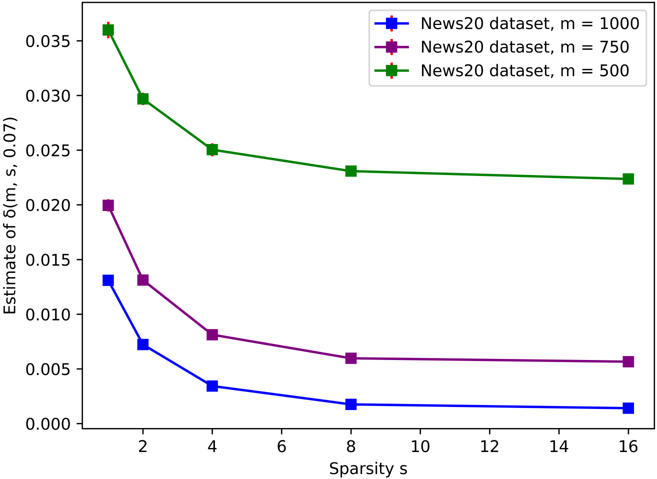
$v(m, \epsilon, \delta, s)$ on more synthetic data

$v(m, \epsilon, \delta, s)$ on more synthetic data

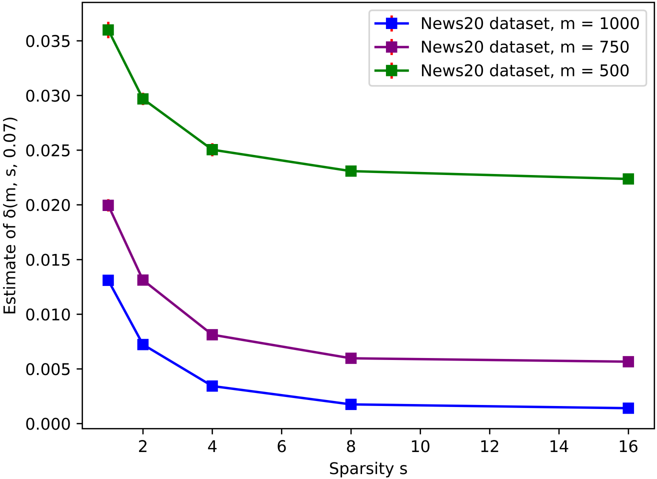


Sparse JL on News20 dataset

Sparse JL on News20 dataset



Sparse JL on News20 dataset



Use sparse JL with more than one hash function!!

Sparse JL as a Sparse Random Projection (KN '12)

Sparse JL as a Sparse Random Projection (KN '12)

$\mathcal{A}_{s,m,n}$ a distribution over $m \times n$ matrices w/ s nonzero entries per column

Sparse JL as a Sparse Random Projection (KN '12)

$\mathcal{A}_{s,m,n}$ a distribution over $m \times n$ matrices w/ s nonzero entries per column

Uniform: Mildly correlate hash functions so $h_j(i) \neq h_k(i)$.

Sparse JL as a Sparse Random Projection (KN '12)

$\mathcal{A}_{s,m,n}$ a distribution over $m \times n$ matrices w/ s nonzero entries per column

Uniform: Mildly correlate hash functions so $h_j(i) \neq h_k(i)$.

Example (Uniform Sparse JL)

Choose s nonzero entries per column; i.i.d signs for nonzero entries

Sparse JL as a Sparse Random Projection (KN '12)

$\mathcal{A}_{s,m,n}$ a distribution over $m \times n$ matrices w/ s nonzero entries per column

Uniform: Mildly correlate hash functions so $h_j(i) \neq h_k(i)$.

Example (Uniform Sparse JL)

Choose s nonzero entries per column; i.i.d signs for nonzero entries

Block: Take $h_i : \{1, \ldots, n\} \to \{(m/s)(i-1)+1, \ldots, (m/s)(i)\}$

Sparse JL as a Sparse Random Projection (KN '12)

$\mathcal{A}_{s,m,n}$ a distribution over $m \times n$ matrices w/ s nonzero entries per column

Uniform: Mildly correlate hash functions so $h_j(i) \neq h_k(i)$.

Example (Uniform Sparse JL)

Choose s nonzero entries per column; i.i.d signs for nonzero entries

Block: Take $h_i : \{1, \ldots, n\} \to \{(m/s)(i-1)+1, \ldots, (m/s)(i)\}$

Example (Block Sparse JL)

Choose one nonzero coordinate per m/s-length block per column; i.i.d signs for nonzero entries

Sparse JL as a Sparse Random Projection (KN '12)

$\mathcal{A}_{s,m,n}$ a distribution over $m \times n$ matrices w/ s nonzero entries per column

Uniform: Mildly correlate hash functions so $h_j(i) \neq h_k(i)$.

Example (Uniform Sparse JL)

Choose s nonzero entries per column; i.i.d signs for nonzero entries

Block: Take $h_i : \{1, \ldots, n\} \to \{(m/s)(i-1)+1, \ldots, (m/s)(i)\}$

Example (Block Sparse JL)

Choose one nonzero coordinate per m/s-length block per column; i.i.d signs for nonzero entries

(r,i)th coordinate is $\eta_{r,i}\sigma_{r,i}/\sqrt{s}$, where $\eta_{r,i} \in \{0,1\}$, $\sigma_{r,i} \in \{-1,1\}$

High-level approach of our analysis

(r, i)th coordinate is $\eta_{r,i}\sigma_{r,i}/\sqrt{s}$, where $\eta_{r,i} \in \{0, 1\}$, $\sigma_{r,i} \in \{-1, 1\}$

High-level approach of our analysis

(r, i)th coordinate is $\eta_{r,i}\sigma_{r,i}/\sqrt{s}$, where $\eta_{r,i} \in \{0, 1\}$, $\sigma_{r,i} \in \{-1, 1\}$

Analyze moments of "error" rv $\|Ax\|_2^2 - 1$ for $\|x\|_2 = 1$:

High-level approach of our analysis

(r, i)th coordinate is $\eta_{r,i}\sigma_{r,i}/\sqrt{s}$, where $\eta_{r,i} \in \{0, 1\}$, $\sigma_{r,i} \in \{-1, 1\}$

Analyze moments of "error" rv $\|Ax\|_2^2 - 1$ for $\|x\|_2 = 1$:

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{1 \leq i \neq j \leq n} \sum_{r=1}^{m} \eta_{r,i}\eta_{r,j}\sigma_{r,i}\sigma_{r,j}x_i x_j$$

This random variable has been repeatedly analyzed in the literature.

High-level approach of our analysis

(r, i)th coordinate is $\eta_{r,i}\sigma_{r,i}/\sqrt{s}$, where $\eta_{r,i} \in \{0, 1\}$, $\sigma_{r,i} \in \{-1, 1\}$

Analyze moments of "error" rv $\|Ax\|_2^2 - 1$ for $\|x\|_2 = 1$:

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{1 \leq i \neq j \leq n} \sum_{r=1}^{m} \eta_{r,i}\eta_{r,j}\sigma_{r,i}\sigma_{r,j}x_i x_j$$

This random variable has been repeatedly analyzed in the literature.

But... existing bounds are limited to $s = 1$ (Freksen et al, etc.)

High-level approach of our analysis

(r,i)th coordinate is $\eta_{r,i}\sigma_{r,i}/\sqrt{s}$, where $\eta_{r,i} \in \{0,1\}$, $\sigma_{r,i} \in \{-1,1\}$

Analyze moments of "error" rv $\|Ax\|_2^2 - 1$ for $\|x\|_2 = 1$:

$$R(x_1,\ldots,x_n) = \frac{1}{s} \sum_{1 \le i \ne j \le n} \sum_{r=1}^{m} \eta_{r,i}\eta_{r,j}\sigma_{r,i}\sigma_{r,j}x_i x_j$$

This random variable has been repeatedly analyzed in the literature.

But... existing bounds are limited to $s = 1$ (Freksen et al, etc.) or limited to $v = 1$ (Kane and Nelson '12, Cohen et al. '18, etc.).

High-level approach of our analysis

(r, i)th coordinate is $\eta_{r,i}\sigma_{r,i}/\sqrt{s}$, where $\eta_{r,i} \in \{0, 1\}$, $\sigma_{r,i} \in \{-1, 1\}$

Analyze moments of "error" rv $\|Ax\|_2^2 - 1$ for $\|x\|_2 = 1$:

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{1 \leq i \neq j \leq n} \sum_{r=1}^{m} \eta_{r,i}\eta_{r,j}\sigma_{r,i}\sigma_{r,j}x_i x_j$$

This random variable has been repeatedly analyzed in the literature.

But... existing bounds are limited to $s = 1$ (Freksen et al, etc.) or limited to $v = 1$ (Kane and Nelson '12, Cohen et al. '18, etc.).

Need tight bounds on $\mathbb{E}[R(x_1, \ldots, x_n)^p]$ on S_v at every threshold v.

Bounding moments of $R(x_1, \ldots, x_n)$ at every threshold v

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{1 \le i \neq j \le n} \sum_{r=1}^{m} \eta_{r,i} \eta_{r,j} \sigma_{r,i} \sigma_{r,j} x_i x_j$$

Bounding moments of $R(x_1, \ldots, x_n)$ at every threshold v

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{1 \leq i \neq j \leq n} \sum_{r=1}^{m} \eta_{r,i} \eta_{r,j} \sigma_{r,i} \sigma_{r,j} x_i x_j$$

Complexities: $\eta_{r,i}$ are correlated,

Bounding moments of $R(x_1, \ldots, x_n)$ at every threshold v

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{1 \le i \ne j \le n} \sum_{r=1}^{m} \eta_{r,i}\eta_{r,j}\sigma_{r,i}\sigma_{r,j}x_i x_j$$

Complexities: $\eta_{r,i}$ are correlated, sum has $\Theta(mn^2)$ terms

Bounding moments of $R(x_1, \ldots, x_n)$ at every threshold v

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{1 \leq i \neq j \leq n} \sum_{r=1}^{m} \eta_{r,i}\eta_{r,j}\sigma_{r,i}\sigma_{r,j}x_i x_j$$

Complexities: $\eta_{r,i}$ are correlated, sum has $\Theta(mn^2)$ terms

Issues with existing approaches:

Bounding moments of $R(x_1, \ldots, x_n)$ at every threshold v

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{1 \le i \neq j \le n} \sum_{r=1}^{m} \eta_{r,i}\eta_{r,j}\sigma_{r,i}\sigma_{r,j}x_i x_j$$

Complexities: $\eta_{r,i}$ are correlated, sum has $\Theta(mn^2)$ terms

Issues with existing approaches:

1. Not clear how to generalize combinatorics of (Kane and Nelson '12, Freksen et al. '18, etc.)

Bounding moments of $R(x_1, \ldots, x_n)$ at every threshold v

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{1 \leq i \neq j \leq n} \sum_{r=1}^{m} \eta_{r,i} \eta_{r,j} \sigma_{r,i} \sigma_{r,j} x_i x_j$$

Complexities: $\eta_{r,i}$ are correlated, sum has $\Theta(mn^2)$ terms

Issues with existing approaches:

1. Not clear how to generalize combinatorics of (Kane and Nelson '12, Freksen et al. '18, etc.)
2. Existing non-combinatorial approaches not sufficiently tight (Cohen et. al '18, Cohen '16, etc.)

Bounding moments of $R(x_1, \ldots, x_n)$ at every threshold v

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{1 \leq i \neq j \leq n} \sum_{r=1}^{m} \eta_{r,i}\eta_{r,j}\sigma_{r,i}\sigma_{r,j}x_i x_j$$

Complexities: $\eta_{r,i}$ are correlated, sum has $\Theta(mn^2)$ terms

Issues with existing approaches:

1. Not clear how to generalize combinatorics of (Kane and Nelson '12, Freksen et al. '18, etc.)
2. Existing non-combinatorial approaches not sufficiently tight (Cohen et. al '18, Cohen '16, etc.)

We use a non-combinatorial approach with Rademacher-specific bounds.

Overview of our approach

$$R(x_1, \ldots, x_n) = \frac{1}{s}\sum_{r=1}^{m} Z_r(x_1, \ldots, x_n) = \frac{1}{s}\sum_{r=1}^{m}\left(\sum_{1 \le i \neq j \le n} \eta_{r,i}\eta_{r,j}\sigma_{r,i}\sigma_{r,j}x_i x_j\right).$$

Overview of our approach

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{r=1}^{m} Z_r(x_1, \ldots, x_n) = \frac{1}{s} \sum_{r=1}^{m} \left(\sum_{1 \leq i \neq j \leq n} \eta_{r,i} \eta_{r,j} \sigma_{r,i} \sigma_{r,j} x_i x_j \right).$$

Lower bound: $\mathbb{E}[Z_r(x_1, \ldots, x_n)^q] = \mathbb{E}_\eta[\mathbb{E}_\sigma[Z_r(x_1, \ldots, x_n)^q]]$

Overview of our approach

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{r=1}^{m} Z_r(x_1, \ldots, x_n) = \frac{1}{s} \sum_{r=1}^{m} \left(\sum_{1 \leq i \neq j \leq n} \eta_{r,i} \eta_{r,j} \sigma_{r,i} \sigma_{r,j} x_i x_j \right).$$

Lower bound: $\mathbb{E}[Z_r(x_1, \ldots, x_n)^q] = \mathbb{E}_\eta[\mathbb{E}_\sigma[Z_r(x_1, \ldots, x_n)^q]]$

1. Suffices to pick "worst" vector in each S_v

Overview of our approach

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{r=1}^{m} Z_r(x_1, \ldots, x_n) = \frac{1}{s} \sum_{r=1}^{m} \left(\sum_{1 \le i \neq j \le n} \eta_{r,i} \eta_{r,j} \sigma_{r,i} \sigma_{r,j} x_i x_j \right).$$

Lower bound: $\mathbb{E}[Z_r(x_1, \ldots, x_n)^q] = \mathbb{E}_\eta[\mathbb{E}_\sigma[Z_r(x_1, \ldots, x_n)^q]]$

1. Suffices to pick "worst" vector in each S_v
2. View $Z_r(v, \ldots, v, 0, \ldots, 0)$ as a quadratic form of ± 1 rvs

Overview of our approach

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{r=1}^{m} Z_r(x_1, \ldots, x_n) = \frac{1}{s} \sum_{r=1}^{m} \left(\sum_{1 \le i \neq j \le n} \eta_{r,i} \eta_{r,j} \sigma_{r,i} \sigma_{r,j} x_i x_j \right).$$

Lower bound: $\mathbb{E}[Z_r(x_1, \ldots, x_n)^q] = \mathbb{E}_\eta[\mathbb{E}_\sigma[Z_r(x_1, \ldots, x_n)^q]]$

1. Suffices to pick "worst" vector in each S_v
2. View $Z_r(v, \ldots, v, 0, \ldots, 0)$ as a quadratic form of ± 1 rvs
   Use known quadratic form moments bounds (Latała '99)

Overview of our approach

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{r=1}^{m} Z_r(x_1, \ldots, x_n) = \frac{1}{s} \sum_{r=1}^{m} \left(\sum_{1 \leq i \neq j \leq n} \eta_{r,i}\eta_{r,j}\sigma_{r,i}\sigma_{r,j}x_i x_j \right).$$

Lower bound: $\mathbb{E}[Z_r(x_1, \ldots, x_n)^q] = \mathbb{E}_\eta[\mathbb{E}_\sigma[Z_r(x_1, \ldots, x_n)^q]]$

1. Suffices to pick "worst" vector in each S_v
2. View $Z_r(v, \ldots, v, 0, \ldots, 0)$ as a quadratic form of ± 1 rvs
   Use known quadratic form moments bounds (Latała '99)
3. Take expectation over $\eta_{r,i}$; carefully combine over $r \in [m]$

Overview of our approach

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{r=1}^{m} Z_r(x_1, \ldots, x_n) = \frac{1}{s} \sum_{r=1}^{m} \left(\sum_{1 \leq i \neq j \leq n} \eta_{r,i} \eta_{r,j} \sigma_{r,i} \sigma_{r,j} x_i x_j \right).$$

Lower bound: $\mathbb{E}[Z_r(x_1, \ldots, x_n)^q] = \mathbb{E}_\eta[\mathbb{E}_\sigma[Z_r(x_1, \ldots, x_n)^q]]$

1. Suffices to pick "worst" vector in each S_v
2. View $Z_r(v, \ldots, v, 0, \ldots, 0)$ as a quadratic form of ± 1 rvs
   Use known quadratic form moments bounds (Latała '99)
3. Take expectation over $\eta_{r,i}$; carefully combine over $r \in [m]$

Upper bound: need to consider $R(x_1, \ldots, x_n)$ for every $x \in S_v$

Overview of our approach

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{r=1}^{m} Z_r(x_1, \ldots, x_n) = \frac{1}{s} \sum_{r=1}^{m} \left(\sum_{1 \leq i \neq j \leq n} \eta_{r,i} \eta_{r,j} \sigma_{r,i} \sigma_{r,j} x_i x_j \right).$$

Lower bound: $\mathbb{E}[Z_r(x_1, \ldots, x_n)^q] = \mathbb{E}_\eta[\mathbb{E}_\sigma[Z_r(x_1, \ldots, x_n)^q]]$

1. Suffices to pick "worst" vector in each S_v
2. View $Z_r(v, \ldots, v, 0, \ldots, 0)$ as a quadratic form of ± 1 rvs
   Use known quadratic form moments bounds (Latała '99)
3. Take expectation over $\eta_{r,i}$; carefully combine over $r \in [m]$

Upper bound: need to consider $R(x_1, \ldots, x_n)$ for every $x \in S_v$

1. Create <u>tractable</u> versions of estimates in (Latała '97, '99)

Overview of our approach

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{r=1}^{m} Z_r(x_1, \ldots, x_n) = \frac{1}{s} \sum_{r=1}^{m} \left(\sum_{1 \le i \ne j \le n} \eta_{r,i} \eta_{r,j} \sigma_{r,i} \sigma_{r,j} x_i x_j \right).$$

Lower bound: $\mathbb{E}[Z_r(x_1, \ldots, x_n)^q] = \mathbb{E}_\eta[\mathbb{E}_\sigma[Z_r(x_1, \ldots, x_n)^q]]$

1. Suffices to pick "worst" vector in each S_v
2. View $Z_r(v, \ldots, v, 0, \ldots, 0)$ as a quadratic form of ± 1 rvs
   Use known quadratic form moments bounds (Latała '99)
3. Take expectation over $\eta_{r,i}$; carefully combine over $r \in [m]$

Upper bound: need to consider $R(x_1, \ldots, x_n)$ for every $x \in S_v$

1. Create <u>tractable</u> versions of estimates in (Latała '97, '99)
   Structure of $Z_r(x_1, \ldots, x_n)$ is helpful

Overview of our approach

$$R(x_1, \ldots, x_n) = \frac{1}{s} \sum_{r=1}^{m} Z_r(x_1, \ldots, x_n) = \frac{1}{s} \sum_{r=1}^{m} \left(\sum_{1 \leq i \neq j \leq n} \eta_{r,i}\eta_{r,j}\sigma_{r,i}\sigma_{r,j}x_i x_j \right).$$

Lower bound: $\mathbb{E}[Z_r(x_1, \ldots, x_n)^q] = \mathbb{E}_\eta[\mathbb{E}_\sigma[Z_r(x_1, \ldots, x_n)^q]]$

1. Suffices to pick "worst" vector in each S_v
2. View $Z_r(v, \ldots, v, 0, \ldots, 0)$ as a quadratic form of ± 1 rvs
   Use known quadratic form moments bounds (Latała '99)
3. Take expectation over $\eta_{r,i}$; carefully combine over $r \in [m]$

Upper bound: need to consider $R(x_1, \ldots, x_n)$ for every $x \in S_v$

1. Create <u>tractable</u> versions of estimates in (Latała '97, '99)
   Structure of $Z_r(x_1, \ldots, x_n)$ is helpful
2. Combine over $r \in [m]$ using (Latała '97)

Conclusion

Tight analysis of $v(m, \epsilon, \delta, s)$ for uniform sparse JL for a general s.

Conclusion

Tight analysis of $v(m, \epsilon, \delta, s)$ for uniform sparse JL for a general s.

Characterization of sparse JL performance in terms of ϵ, δ, and ℓ_∞-to-ℓ_2 norm ratio for a general # of hash functions s.

Conclusion

Tight analysis of $v(m, \epsilon, \delta, s)$ for uniform sparse JL for a general s.

Characterization of sparse JL performance in terms of ϵ, δ, and ℓ_∞-to-ℓ_2 norm ratio for a general # of hash functions s.

Takeaway: use sparse JL w/ ≥ 4 hash functions on feature vectors!

Conclusion

Tight analysis of $v(m, \epsilon, \delta, s)$ for uniform sparse JL for a general s.

Characterization of sparse JL performance in terms of ϵ, δ, and ℓ_∞-to-ℓ_2 norm ratio for a general # of hash functions s.

Takeaway: use sparse JL w/ ≥ 4 hash functions on feature vectors!

Verification on real-world and synthetic data.

Conclusion

Tight analysis of $v(m, \epsilon, \delta, s)$ for uniform sparse JL for a general s.

Characterization of sparse JL performance in terms of ϵ, δ, and ℓ_∞-to-ℓ_2 norm ratio for a general # of hash functions s.

Takeaway: use sparse JL w/ ≥ 4 hash functions on feature vectors!

Verification on real-world and synthetic data.

Hope to see you at the poster session!!!