THE SAMPLE COMPLEXITY OF TOEPLITZ COVARIANCE ESTIMATION

Cameron Musco (Microsoft Research $\rightarrow$ UMass Amherst)

Joint with Yonina Eldar, Jerry Li, and Christopher Musco.

Covariance Estimation Problem. Consider positive semidefinite matrix $T \in \mathbb{R}^{d\times d}$ and distribution $\mathcal{D}$ over d-dimensional vectors with covariance $\mathbb{E}_{x\sim\mathcal{D}}[xx^T] = T$ (i.e., $T_{j,k}$ is the covariance between x_j and x_k).

Covariance Estimation Problem. Consider positive semidefinite matrix $T \in \mathbb{R}^{d \times d}$ and distribution $\mathcal{D}$ over d-dimensional vectors with covariance $\mathbb{E}_{x \sim \mathcal{D}}[xx^T] = T$ (i.e., $T_{j,k}$ is the covariance between x_j and x_k).

Given independent samples $x^{(1)}, \ldots, x^{(n)} \sim \mathcal{D}$, return $\tilde{T}$ with:

$$\|T - \tilde{T}\|_2 \leq \varepsilon \|T\|_2.$$

Covariance Estimation Problem. Consider positive semidefinite Toeplitz matrix $T \in \mathbb{R}^{d\times d}$ and distribution $\mathcal{D}$ over d-dimensional vectors with covariance $\mathbb{E}_{x\sim\mathcal{D}}[xx^T] = T$ (i.e., $T_{j,k}$ is the covariance between x_j and x_k).

Given independent samples $x^{(1)}, \ldots, x^{(n)} \sim \mathcal{D}$, return $\tilde{T}$ with:

$$\|T - \tilde{T}\|_2 \leq \varepsilon \|T\|_2.$$

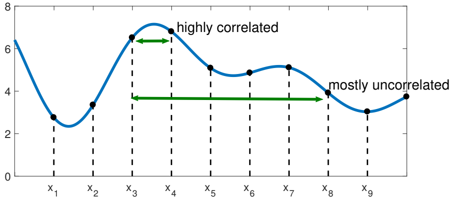
Covariance Estimation Problem. Consider positive semidefinite Toeplitz matrix $T \in \mathbb{R}^{d \times d}$ and distribution $\mathcal{D}$ over d-dimensional vectors with covariance $\mathbb{E}_{x \sim \mathcal{D}}[xx^T] = T$ (i.e., $T_{j,k}$ is the covariance between x_j and x_k).

Given independent samples $x^{(1)}, \ldots, x^{(n)} \sim \mathcal{D}$, return $\tilde{T}$ with:

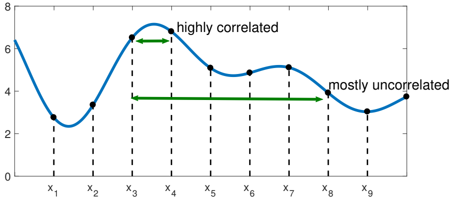
$$\|T - \tilde{T}\|_2 \leq \varepsilon \|T\|_2.$$

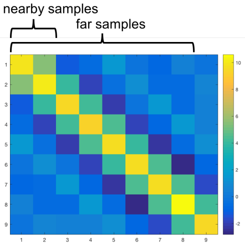
$$T = \begin{bmatrix} a & b & c & d & e \\ b & a & b & c & d \\ c & b & a & b & c \\ d & c & b & a & b \\ e & d & c & b & a \end{bmatrix}$$

Arises often in signal processing, when measurements are taken on a spatial or temporal grid and covariance depends only on the distance between them – i.e., $\mathbb{E}[x_j \cdot x_k] = f(|j-k|)$.

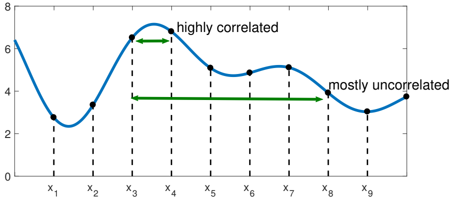


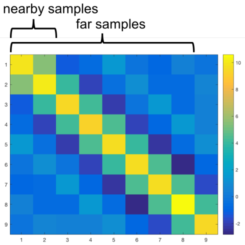
Arises often in signal processing, when measurements are taken on a spatial or temporal grid and covariance depends only on the distance between them – i.e., $\mathbb{E}[x_j \cdot x_k] = f(|j-k|)$.

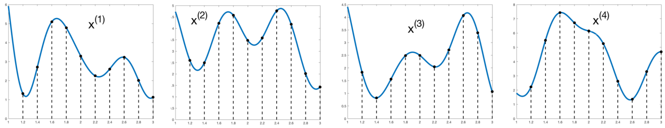




Arises often in signal processing, when measurements are taken on a spatial or temporal grid and covariance depends only on the distance between them – i.e., $\mathbb{E}[x_j \cdot x_k] = f(|j-k|)$.

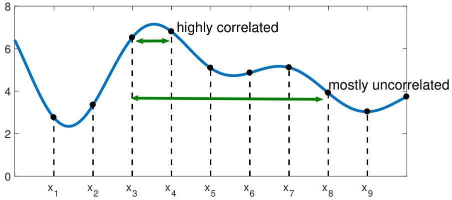


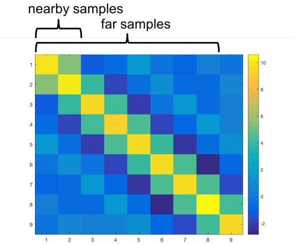


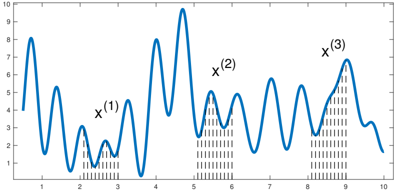


TOEPLITZ COVARIANCE ESTIMATION

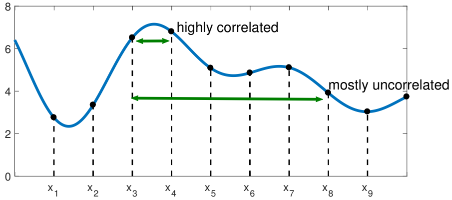
Arises often in signal processing, when measurements are taken on a spatial or temporal grid and covariance depends only on the distance between them – i.e., $\mathbb{E}[x_j \cdot x_k] = f(|j - k|)$.

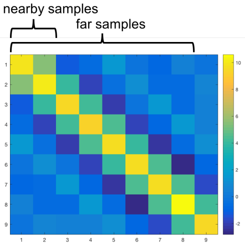






Arises often in signal processing, when measurements are taken on a spatial or temporal grid and covariance depends only on the distance between them – i.e., $\mathbb{E}[x_j \cdot x_k] = f(|j - k|)$.

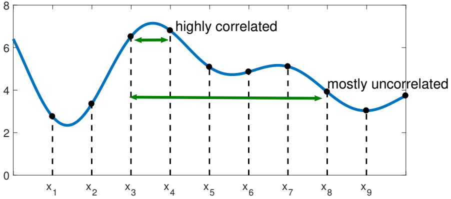


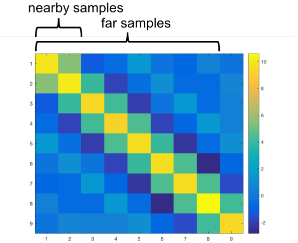


- Applications: spectrum sensing, Doppler radar, direction of arrival estimation, prediction via Gaussian process regression, etc.

TOEPLITZ COVARIANCE ESTIMATION

Arises often in signal processing, when measurements are taken on a spatial or temporal grid and covariance depends only on the distance between them – i.e., $\mathbb{E}[x_j \cdot x_k] = f(|j - k|)$.





- Applications: spectrum sensing, Doppler radar, direction of arrival estimation, prediction via Gaussian process regression, etc.
- Kernel matrices in machine learning are Toeplitz covariance matrices when data points are on a grid.

Want to minimize two types of sample complexity:

Want to minimize two types of sample complexity:

- **Vector sample complexity:** How many samples $x^{(1)}, \ldots, x^{(n)} \sim \mathcal{D}$ are required to estimate T?

Want to minimize two types of sample complexity:

- **Vector sample complexity:** How many samples $x^{(1)}, \ldots, x^{(n)} \sim \mathcal{D}$ are required to estimate T?
- **Entry sample complexity:** How many entries s must be read from each sample $x^{(1)}, \ldots, x^{(n)}$?

Want to minimize two types of sample complexity:

- **Vector sample complexity:** How many samples $x^{(1)}, \ldots, x^{(n)} \sim \mathcal{D}$ are required to estimate T?
- **Entry sample complexity:** How many entries s must be read from each sample $x^{(1)}, \ldots, x^{(n)}$?

In different applications, these complexities correspond to different costs. Typically there is a tradeoff.

Want to minimize two types of sample complexity:

- **Vector sample complexity:** How many samples $x^{(1)}, \ldots, x^{(n)} \sim \mathcal{D}$ are required to estimate T?
- **Entry sample complexity:** How many entries s must be read from each sample $x^{(1)}, \ldots, x^{(n)}$?

In different applications, these complexities correspond to different costs. Typically there is a tradeoff.

- **Total sample complexity**: Total number of entries read, $n \cdot s$.

Want to minimize two types of sample complexity:

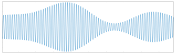
- **Vector sample complexity:** How many samples $x^{(1)}, \ldots, x^{(n)} \sim \mathcal{D}$ are required to estimate T?
- **Entry sample complexity:** How many entries s must be read from each sample $x^{(1)}, \ldots, x^{(n)}$?

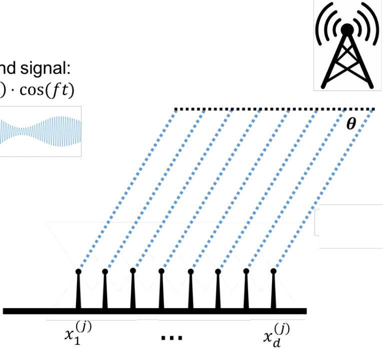
In different applications, these complexities correspond to different costs. Typically there is a tradeoff.

- **Total sample complexity**: Total number of entries read, $n \cdot s$.
- Seems to be interesting even beyond Toeplitz covariance matrices, but not well studied.

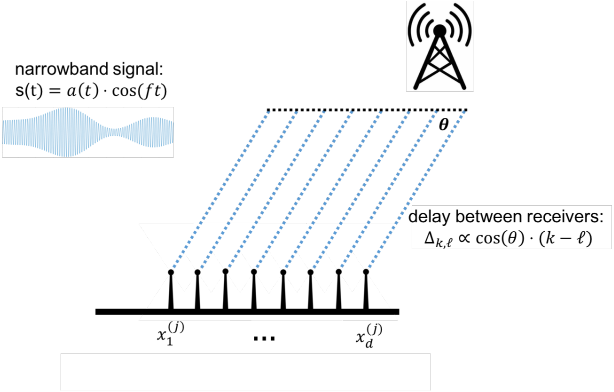
EXAMPLE: DIRECTION OF ARRIVAL ESTIMATION

narrowband signal:
$s(t) = a(t) \cdot \cos(ft)$

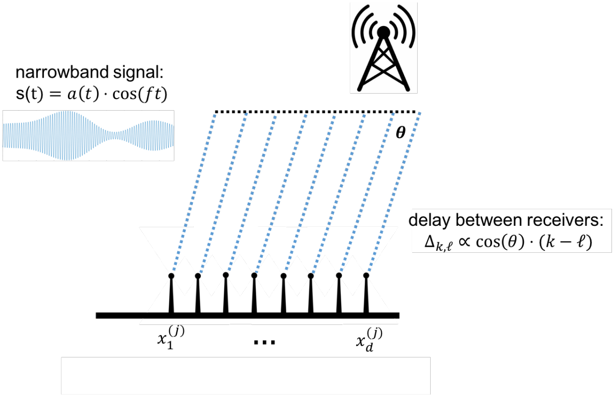




EXAMPLE: DIRECTION OF ARRIVAL ESTIMATION

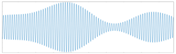


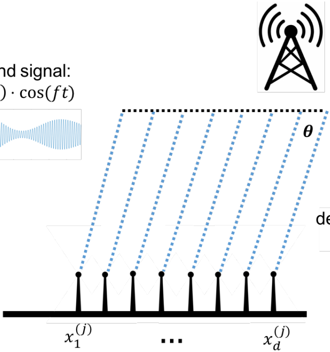
EXAMPLE: DIRECTION OF ARRIVAL ESTIMATION



EXAMPLE: DIRECTION OF ARRIVAL ESTIMATION

narrowband signal:
$s(t) = a(t) \cdot \cos(ft)$



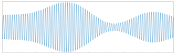


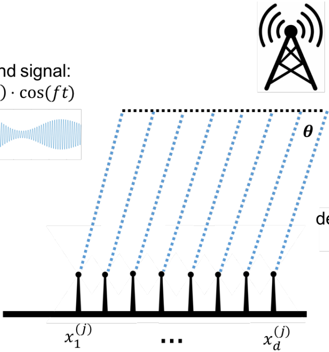
delay between receivers:
$\Delta_{k,\ell} \propto \cos(\theta) \cdot (k - \ell)$

With delay, $\mathbb{E}[x_k^{(j)} \cdot x_\ell^{(j)}] \approx \mathbb{E}[a(t)^2] \cdot \cos(f\Delta_{k,\ell})$

EXAMPLE: DIRECTION OF ARRIVAL ESTIMATION

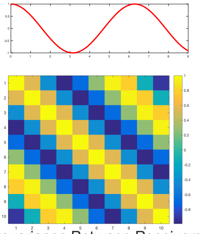
narrowband signal:
$s(t) = a(t) \cdot \cos(ft)$



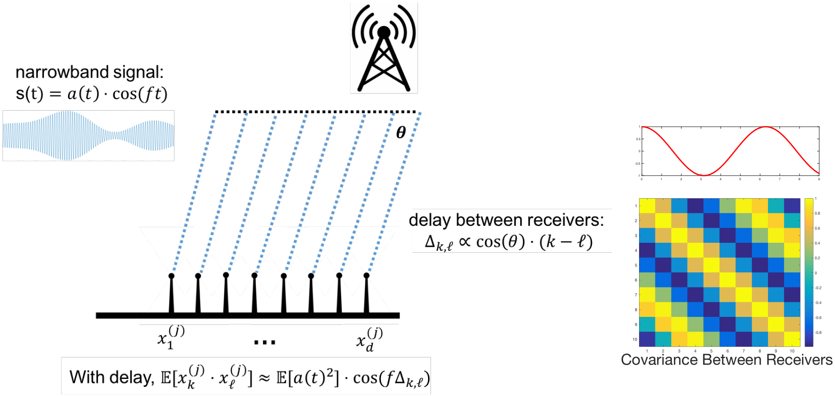


delay between receivers:
$\Delta_{k,\ell} \propto \cos(\theta) \cdot (k - \ell)$

With delay, $\mathbb{E}[x_k^{(j)} \cdot x_\ell^{(j)}] \approx \mathbb{E}[a(t)^2] \cdot \cos(f\Delta_{k,\ell})$

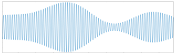


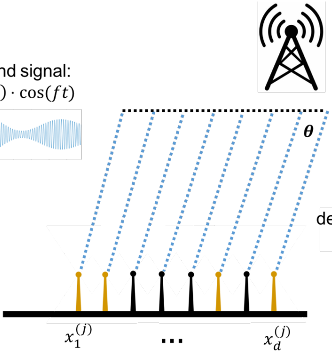
Covariance Between Receivers



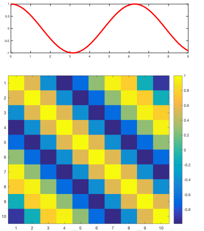
- **Vector sample complexity:** Estimation time (# snapshots).
- **Entry sample complexity:** Number of active receivers.

narrowband signal:
$s(t) = a(t) \cdot \cos(ft)$





delay between receivers:
$\Delta_{k,\ell} \propto \cos(\theta) \cdot (k - \ell)$


Covariance Between Receivers

With delay, $\mathbb{E}[x_k^{(j)} \cdot x_\ell^{(j)}] \approx \mathbb{E}[a(t)^2] \cdot \cos(f\Delta_{k,\ell})$

- **Vector sample complexity:** Estimation time (# snapshots).
- **Entry sample complexity:** Number of active receivers.

Current state: Many algorithms for Toeplitz covariance estimation, but few formal results on sample complexities/tradeoffs.

Current state: Many algorithms for Toeplitz covariance estimation, but few formal results on sample complexities/tradeoffs.

Our contributions:

Current state: Many algorithms for Toeplitz covariance estimation, but few formal results on sample complexities/tradeoffs.

Our contributions:

- Give non-asymptotic sample complexity bounds by analyzing classic algorithms, including those with sublinear entry sample complexity based on sparse ruler measurements.

Current state: Many algorithms for Toeplitz covariance estimation, but few formal results on sample complexities/tradeoffs.

Our contributions:

- Give non-asymptotic sample complexity bounds by analyzing classic algorithms, including those with sublinear entry sample complexity based on sparse ruler measurements.
- Show that sparse ruler methods give sublinear total sample complexity when T is low-rank (e.g., DOA with $k \ll d$ senders).

Current state: Many algorithms for Toeplitz covariance estimation, but few formal results on sample complexities/tradeoffs.

Our contributions:

- Give non-asymptotic sample complexity bounds by analyzing classic algorithms, including those with sublinear entry sample complexity based on sparse ruler measurements.
- Show that sparse ruler methods give sublinear total sample complexity when T is low-rank (e.g., DOA with $k \ll d$ senders).
- Develop improved algorithms in the low-rank setting using techniques from matrix sketching, leverage score-based sampling, and sparse Fourier transforms. Resemble popular 'subspace methods' such as MUSIC and ESPRIT.

Build connections between theoretical computer science and signal processing.

Build connections between theoretical computer science and signal processing.

- Leverage score/effective resistance sampling, sparse Fourier transforms $\iff$ sub-Nyquist sampling, Chebyshev interpolation, active sampling for Gaussian process regression
- Column-based matrix approximation, combinatorial sparsification $\iff$ nonlinear function approximation, Fourier-sparse approximations

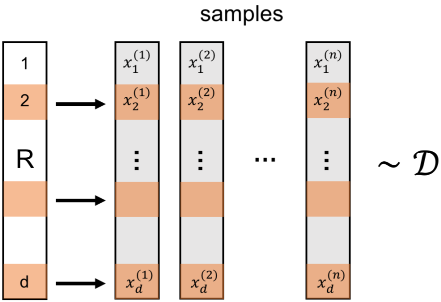
Build connections between theoretical computer science and signal processing.

- Leverage score/effective resistance sampling, sparse Fourier transforms $\iff$ sub-Nyquist sampling, Chebyshev interpolation, active sampling for Gaussian process regression
- Column-based matrix approximation, combinatorial sparsification $\iff$ nonlinear function approximation, Fourier-sparse approximations

Apply tools from TCS to tackle fundamental signal processing problems. *A Universal Sampling Method for Reconstructing Signals with Simple Fourier Transforms* [AKMMVZ STOC '19]

For today, consider algorithms that sample $x^{(1)}, \ldots, x^{(n)} \sim \mathcal{D}$ with covariance T, read a fixed subset of entries $R \subseteq [d]$ from each $x^{(j)}$, and approximate T using $x_R^{(1)}, \ldots, x_R^{(n)} \in \mathbb{R}^{|R|}$.

For today, consider algorithms that sample $x^{(1)}, \ldots, x^{(n)} \sim \mathcal{D}$ with covariance T, read a fixed subset of entries $R \subseteq [d]$ from each $x^{(j)}$, and approximate T using $x_R^{(1)}, \ldots, x_R^{(n)} \in \mathbb{R}^{|R|}$.



Entry sample complexity: $|R|$. Total sample complexity: $|R| \cdot n$.

For today, consider algorithms that sample $x^{(1)}, \ldots, x^{(n)} \sim \mathcal{D}$ with covariance T, read a fixed subset of entries $R \subseteq [d]$ from each $x^{(j)}$, and approximate T using $x_R^{(1)}, \ldots, x_R^{(n)} \in \mathbb{R}^{|R|}$.

How small can R be? I.e., what is the minimal entry sample complexity of such an algorithm?

For today, consider algorithms that sample $x^{(1)}, \ldots, x^{(n)} \sim \mathcal{D}$ with covariance T, read a fixed subset of entries $R \subseteq [d]$ from each $x^{(j)}$, and approximate T using $x_R^{(1)}, \ldots, x_R^{(n)} \in \mathbb{R}^{|R|}$.

How small can R be? I.e., what is the minimal entry sample complexity of such an algorithm?

For general (non-Toeplitz) T, require $|R| = d$.

For today, consider algorithms that sample $x^{(1)}, \ldots, x^{(n)} \sim \mathcal{D}$ with covariance T, read a fixed subset of entries $R \subseteq [d]$ from each $x^{(j)}$, and approximate T using $x_R^{(1)}, \ldots, x_R^{(n)} \in \mathbb{R}^{|R|}$.

How small can R be? I.e., what is the minimal entry sample complexity of such an algorithm?

For general (non-Toeplitz) T, require $|R| = d$.

$$T_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{vs.} \quad T_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

To notice correlation between x_j and x_k must read both.

How small can R be if T is Toeplitz?

How small can R be if T is Toeplitz? Can take advantage of redundancy.

$$T = \begin{bmatrix} a_0 & a_1 & a_2 & \cdots & a_{d-2} & a_{d-1} \\ a_1 & a_0 & a_1 & \cdots & \cdots & a_{d-2} \\ a_2 & a_1 & a_0 & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{d-2} & \cdots & \cdots & \cdots & \cdots & a_1 \\ a_{d-1} & a_{d-2} & \cdots & \cdots & a_1 & a_0 \end{bmatrix}$$

How small can R be if T is Toeplitz? Can take advantage of redundancy.

$$T = \begin{bmatrix} a_0 & a_1 & a_2 & \cdots & a_{d-2} & a_{d-1} \\ a_1 & a_0 & a_1 & \cdots & \cdots & a_{d-2} \\ a_2 & a_1 & a_0 & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{d-2} & \cdots & \cdots & \cdots & \cdots & a_1 \\ a_{d-1} & a_{d-2} & \cdots & \cdots & a_1 & a_0 \end{bmatrix}$$

- $a_1 = \mathbb{E}[x_2 \cdot x_3] = \mathbb{E}[x_d \cdot x_{d-1}]$.

How small can R be if T is Toeplitz? Can take advantage of redundancy.

$$T = \begin{bmatrix} a_0 & a_1 & a_2 & \cdots & a_{d-2} & a_{d-1} \\ a_1 & a_0 & a_1 & \cdots & \cdots & a_{d-2} \\ a_2 & a_1 & a_0 & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{d-2} & \cdots & \cdots & \cdots & \cdots & a_1 \\ a_{d-1} & a_{d-2} & \cdots & \cdots & a_1 & a_0 \end{bmatrix}$$

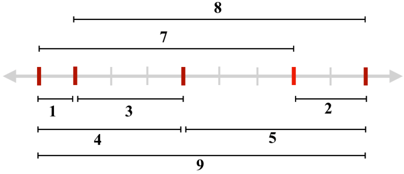
- $a_1 = \mathbb{E}[x_2 \cdot x_3] = \mathbb{E}[x_d \cdot x_{d-1}]$.

Will see that we can achieve $|R| = O(\sqrt{d})$.

Definition (Ruler) A subset $R \subseteq [d]$ is a ruler if for every distance $s \in \{0, \ldots, d-1\}$, there exist $j, k \in R$ with $j - k = s$.

Definition (Ruler) A subset $R \subseteq [d]$ is a ruler if for every distance $s \in \{0, \ldots, d-1\}$, there exist $j, k \in R$ with $j - k = s$.

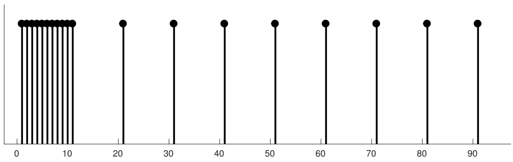
E.g., for $d = 10$, $R = \{1, 2, 5, 8, 10\}$ is a ruler.



Definition (Ruler) A subset $R \subseteq [d]$ is a ruler if for every distance $s \in \{0, \ldots, d-1\}$, there exist $j, k \in R$ with $j - k = s$.

Claim For any d there exists a sparse ruler R with $|R| = 2\sqrt{d}$

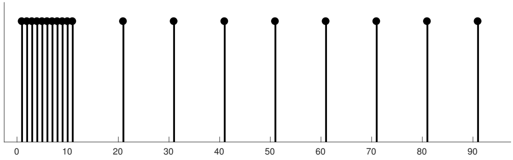
- Suffices to take $R = [1, 2, \ldots, \sqrt{d}] \cup [2\sqrt{d}, 3\sqrt{d}, \ldots, d]$.



Definition (Ruler) A subset $R \subseteq [d]$ is a ruler if for every distance $s \in \{0, \ldots, d-1\}$, there exist $j, k \in R$ with $j - k = s$.

Claim For any d there exists a sparse ruler R with $|R| = 2\sqrt{d}$

- Suffices to take $R = [1, 2, \ldots, \sqrt{d}] \cup [2\sqrt{d}, 3\sqrt{d}, \ldots, d]$.

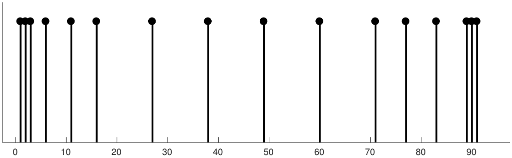


- The best possible leading constant lies between $\sqrt{2 + \frac{4}{3\pi}}$ and $\sqrt{8/3}$ (Erdös, Gal, Leech, '48, '56)

Definition (Ruler) A subset $R \subseteq [d]$ is a ruler if for every distance $s \in \{0, \ldots, d-1\}$, there exist $j, k \in R$ with $j - k = s$.

Claim For any d there exists a sparse ruler R with $|R| = 2\sqrt{d}$

- Suffices to take $R = [1, 2, \ldots, \sqrt{d}] \cup [2\sqrt{d}, 3\sqrt{d}, \ldots, d]$.



- The best possible leading constant lies between $\sqrt{2 + \frac{4}{3\pi}}$ and $\sqrt{8/3}$ (Erdös, Gal, Leech, '48, '56)

$$T = \begin{bmatrix} a_0 & a_1 & a_2 & \cdots & a_{d-2} & a_{d-1} \\ a_1 & a_0 & a_1 & \cdots & \cdots & a_{d-2} \\ a_2 & a_1 & a_0 & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{d-2} & \cdots & \cdots & \cdots & \cdots & a_1 \\ a_{d-1} & a_{d-2} & \cdots & \cdots & a_1 & a_0 \end{bmatrix}$$

- If R is a ruler, for each $s \in \{0, \ldots, d-1\}$, there is at least one $k, \ell \in R$ with $|k - \ell| = s$ and thus with covariance
$$\mathbb{E}[x_k^{(j)} \cdot x_\ell^{(j)}] = a_s.$$

$$T = \begin{bmatrix} a_0 & a_1 & \boxed{a_2} & \cdots & a_{d-2} & \boxed{a_{d-1}} \\ a_1 & a_0 & a_1 & \cdots & \cdots & a_{d-2} \\ a_2 & \boxed{a_1} & \boxed{a_0} & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{d-2} & \cdots & \cdots & \cdots & \cdots & a_1 \\ a_{d-1} & \boxed{a_{d-2}} & \cdots & \cdots & a_1 & a_0 \end{bmatrix}$$

- If R is a ruler, for each $s \in \{0, \ldots, d-1\}$, there is at least one $k, \ell \in R$ with $|k - \ell| = s$ and thus with covariance

$$\mathbb{E}[x_k^{(j)} \cdot x_\ell^{(j)}] = a_s.$$

- Get at least one independent sample of a_s from every $x_R^{(j)}$.

$$T = \begin{bmatrix} a_0 & a_1 & a_2 & \cdots & a_{d-2} & a_{d-1} \\ a_1 & a_0 & a_1 & \cdots & \cdots & a_{d-2} \\ a_2 & a_1 & a_0 & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{d-2} & \cdots & \cdots & \cdots & \cdots & a_1 \\ a_{d-1} & a_{d-2} & \cdots & \cdots & a_1 & a_0 \end{bmatrix}$$

- If R is a ruler, for each $s \in \{0, \ldots, d-1\}$, there is at least one $k, \ell \in R$ with $|k - \ell| = s$ and thus with covariance

$$\mathbb{E}[x_k^{(j)} \cdot x_\ell^{(j)}] = a_s.$$

- Get at least one independent sample of a_s from every $x_R^{(j)}$.
- With enough samples n from $\mathcal{D}$, will converge on an estimate of each a_s and so of the full matrix T.

How many vector samples do we need? What do we pay for the optimal entry sample complexity of sparse rulers?

How many vector samples do we need? What do we pay for the optimal entry sample complexity of sparse rulers?

- How does the total sample complexity compare to methods that read every entry of each $x^{(j)}$, e.g., estimating T with the empricial covariance $\hat{T} = \frac{1}{n}\sum_j x^{(j)}x^{(j)^T}$.

Let $\mathcal{D} = \mathcal{N}(0, T)$ be a d-dimensional Gaussian with $a_0 = 1$.

SOME INTUITION

Let $\mathcal{D} = \mathcal{N}(0, T)$ be a d-dimensional Gaussian with $a_0 = 1$.

- For $n = O\left(\frac{\log d}{\varepsilon^2}\right)$ all estimates of a_s give error $|\varepsilon_s| \leq \varepsilon$.

Let $\mathcal{D} = \mathcal{N}(0, T)$ be a d-dimensional Gaussian with $a_0 = 1$.

- For $n = O\left(\frac{\log d}{\varepsilon^2}\right)$ all estimates of a_s give error $|\varepsilon_s| \leq \varepsilon$.

$$\tilde{T} = \begin{bmatrix} a_0 + \varepsilon_0 & a_1 + \varepsilon_1 & a_2 + \varepsilon_2 & \cdots & a_{d-2} + \varepsilon_{d-2} & a_{d-1} + \varepsilon_{d-1} \\ a_1 + \varepsilon_1 & a_0 + \varepsilon_0 & a_1 + \varepsilon_1 & \cdots & \cdots & a_{d-2} + \varepsilon_{d-2} \\ a_2 + \varepsilon_2 & a_1 + \varepsilon_1 & a_0 + \varepsilon_0 & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{d-2} + \varepsilon_{d-2} & \cdots & \cdots & \cdots & \cdots & a_1 + \varepsilon_1 \\ a_{d-1} + \varepsilon_{d-1} & a_{d-2} + \varepsilon_{d-2} & \cdots & \cdots & a_1 + \varepsilon_1 & a_0 + \varepsilon_0 \end{bmatrix}$$

Let $\mathcal{D} = \mathcal{N}(0, T)$ be a d-dimensional Gaussian with $a_0 = 1$.

- For $n = O\left(\frac{\log d}{\varepsilon^2}\right)$ all estimates of a_s give error $|\varepsilon_s| \leq \varepsilon$.

$$\tilde{T} - T = \begin{bmatrix} \varepsilon_0 & \varepsilon_1 & \varepsilon_2 & \cdots & \varepsilon_{d-2} & \varepsilon_{d-1} \\ \varepsilon_1 & \varepsilon_0 & \varepsilon_1 & \cdots & \cdots & \varepsilon_{d-2} \\ \varepsilon_2 & \varepsilon_1 & \varepsilon_0 & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \varepsilon_{d-2} & \cdots & \cdots & \cdots & \cdots & \varepsilon_1 \\ \varepsilon_{d-1} & \varepsilon_{d-2} & \cdots & \cdots & \varepsilon_1 & \varepsilon_0 \end{bmatrix}$$

Let $\mathcal{D} = \mathcal{N}(0, T)$ be a d-dimensional Gaussian with $a_0 = 1$.

- For $n = O\left(\frac{\log d}{\varepsilon^2}\right)$ all estimates of a_s give error $|\varepsilon_s| \leq \varepsilon$.

$$\tilde{T} - T = \begin{bmatrix} \varepsilon_0 & \varepsilon_1 & \varepsilon_2 & \cdots & \varepsilon_{d-2} & \varepsilon_{d-1} \\ \varepsilon_1 & \varepsilon_0 & \varepsilon_1 & \cdots & \cdots & \varepsilon_{d-2} \\ \varepsilon_2 & \varepsilon_1 & \varepsilon_0 & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \varepsilon_{d-2} & \cdots & \cdots & \cdots & \cdots & \varepsilon_1 \\ \varepsilon_{d-1} & \varepsilon_{d-2} & \cdots & \cdots & \varepsilon_1 & \varepsilon_0 \end{bmatrix}$$

- In the worst case, $\|\tilde{T} - T\|_2 = \varepsilon d$ but if ε_s were independent, $\|\tilde{T} - T\|_2 \leq \varepsilon\sqrt{d}$ [Meckes '07].

SOME INTUITION

Let $\mathcal{D} = \mathcal{N}(0, T)$ be a d-dimensional Gaussian with $a_0 = 1$.

- For $n = O\left(\frac{\log d}{\varepsilon^2}\right)$ all estimates of a_s give error $|\varepsilon_s| \leq \varepsilon$.

$$\tilde{T} - T = \begin{bmatrix} \varepsilon_0 & \varepsilon_1 & \varepsilon_2 & \cdots & \varepsilon_{d-2} & \varepsilon_{d-1} \\ \varepsilon_1 & \varepsilon_0 & \varepsilon_1 & \cdots & \cdots & \varepsilon_{d-2} \\ \varepsilon_2 & \varepsilon_1 & \varepsilon_0 & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \varepsilon_{d-2} & \cdots & \cdots & \cdots & \cdots & \varepsilon_1 \\ \varepsilon_{d-1} & \varepsilon_{d-2} & \cdots & \cdots & \varepsilon_1 & \varepsilon_0 \end{bmatrix}$$

- In the worst case, $\|\tilde{T} - T\|_2 = \varepsilon d$ but if ε_s were independent, $\|\tilde{T} - T\|_2 \leq \varepsilon\sqrt{d}$ [Meckes '07].
- Setting $\varepsilon' = \varepsilon/\sqrt{d}$, $n = \tilde{O}\left(\frac{d}{\varepsilon^2}\right)$ would give

$$\|\tilde{T} - T\|_2 \leq \varepsilon \leq \varepsilon\|T\|_2.$$

Let $\mathcal{D} = \mathcal{N}(0, T)$ be a d-dimensional Gaussian with $a_0 = 1$.

- For $n = O\left(\frac{\log d}{\varepsilon^2}\right)$ all estimates of a_s give error $|\varepsilon_s| \leq \varepsilon$.

$$\tilde{T} - T = \begin{bmatrix} \varepsilon_0 & \varepsilon_1 & \varepsilon_2 & \cdots & \varepsilon_{d-2} & \varepsilon_{d-1} \\ \varepsilon_1 & \varepsilon_0 & \varepsilon_1 & \cdots & \cdots & \varepsilon_{d-2} \\ \varepsilon_2 & \varepsilon_1 & \varepsilon_0 & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \varepsilon_{d-2} & \cdots & \cdots & \cdots & \cdots & \varepsilon_1 \\ \varepsilon_{d-1} & \varepsilon_{d-2} & \cdots & \cdots & \varepsilon_1 & \varepsilon_0 \end{bmatrix}$$

- In the worst case, $\|\tilde{T} - T\|_2 = \varepsilon d$ but if ε_s were independent, $\|\tilde{T} - T\|_2 \leq \varepsilon\sqrt{d}$ [Meckes '07].
- Setting $\varepsilon' = \varepsilon/\sqrt{d}$, $n = \tilde{O}\left(\frac{d}{\varepsilon^2}\right)$ would give

$$\|\tilde{T} - T\|_2 \leq \varepsilon \leq \varepsilon\|T\|_2.$$

Theorem. For any ruler $R \subset [d]$, covariance estimation with R gives $\|\tilde{T} - T\|_2 \leq \varepsilon \|T\|_2$ with entry sample complexity $|R|$ and vector sample complexity $n = \tilde{O}\left(\frac{d}{\varepsilon^2}\right)$.

Theorem. For any ruler $R \subset [d]$, covariance estimation with R gives $\|\tilde{T} - T\|_2 \leq \varepsilon\|T\|_2$ with entry sample complexity $|R|$ and vector sample complexity $n = \tilde{O}\left(\frac{d}{\varepsilon^2}\right)$.

- Vector sample complexity matches the complexity of estimating an unstructured covariance with the empirical covariance but entry sample complexity can be $O(\sqrt{d})$ instead of d.

Theorem. For any ruler $R \subset [d]$, covariance estimation with R gives $\|\tilde{T} - T\|_2 \leq \varepsilon \|T\|_2$ with entry sample complexity $|R|$ and vector sample complexity $n = \tilde{O}\left(\frac{d}{\varepsilon^2}\right)$.

- Vector sample complexity matches the complexity of estimating an unstructured covariance with the empirical covariance but entry sample complexity can be $O(\sqrt{d})$ instead of d.
- Proof uses the Fourier structure of Toeplitz matrices.

Algorithm: For each $s \in \{0, 1\}$ approximate a_s by average over the ruler R:

$$\tilde{a}_s = \frac{1}{n|R_s|} \sum_{j=1}^{n} \sum_{(k,\ell) \in R_s} x_k^{(j)} \cdot x_\ell^{(j)} \text{ where } R_s = \{k, \ell \in R : |k - \ell| = s\}.$$

Let $\tilde{T}$ be the Toeplitz matrix with $\tilde{a}_s$ on its s^{th} diagonal.

Algorithm: For each $s \in \{0, 1\}$ approximate a_s by average over the ruler R:

$$\tilde{a}_s = \frac{1}{n|R_s|} \sum_{j=1}^{n} \sum_{(k,\ell) \in R_s} x_k^{(j)} \cdot x_\ell^{(j)} \text{ where } R_s = \{k, \ell \in R : |k - \ell| = s\}.$$

Let $\tilde{T}$ be the Toeplitz matrix with $\tilde{a}_s$ on its s^{th} diagonal.

Algorithm: For each $s \in \{0, 1\}$ approximate a_s by average over the ruler R:

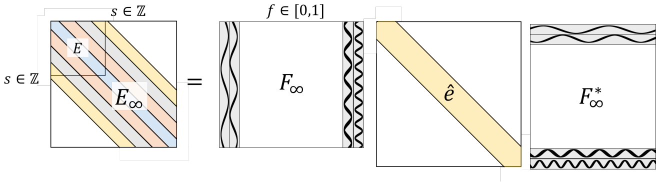
$$\tilde{a}_s = \frac{1}{n|R_s|} \sum_{j=1}^{n} \sum_{(k,\ell) \in R_s} x_k^{(j)} \cdot x_\ell^{(j)} \text{ where } R_s = \{k, \ell \in R : |k - \ell| = s\}.$$

Let $\tilde{T}$ be the Toeplitz matrix with $\tilde{a}_s$ on its s^{th} diagonal.

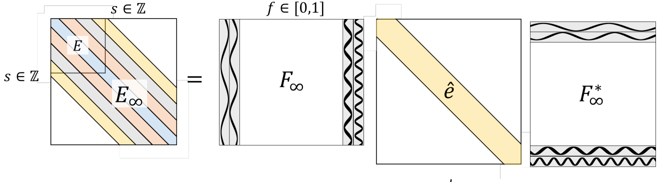
- Let $E = T - \tilde{T}$ and $e = a - \tilde{a}$. We want to bound $\|E\|_2$.

Entry approximation to matrix approximation: Can bound $\|\tilde{T} - T\|_2 = \|E\|_2$ in terms of the Fourier transform of e.

Entry approximation to matrix approximation: Can bound $\|\tilde{T} - T\|_2 = \|E\|_2$ in terms of the Fourier transform of e.

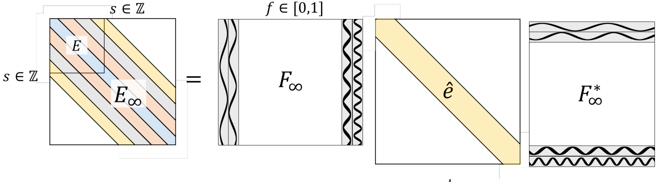


Entry approximation to matrix approximation: Can bound $\|\tilde{T} - T\|_2 = \|E\|_2$ in terms of the Fourier transform of e.

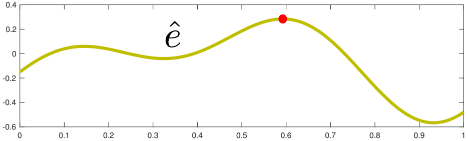


$$\|E\|_2 \leq \|E_\infty\|_2 = \max_{f \in [0,1]} \hat{e} = \max_{f \in [0,1]} \sum_{s=0}^{d} e \cdot \sin(2\pi s f).$$

Entry approximation to matrix approximation: Can bound $\|\tilde{T} - T\|_2 = \|E\|_2$ in terms of the Fourier transform of e.



$$\|E\|_2 \le \|E_\infty\|_2 = \max_{f\in[0,1]} \hat{e} = \max_{f\in[0,1]} \sum_{s=0}^{d} e \cdot \sin(2\pi s f).$$



Formulation as Trace Bound: For fixed f let M_f be the Toeplitz matrix with $(M_f)_{j,k} = \frac{sin(2\pi sf)}{|R_s|}$ when $|j-k| = s$.

Formulation as Trace Bound: For fixed f let M_f be the Toeplitz matrix with $(M_f)_{j,k} = \frac{sin(2\pi sf)}{|R_s|}$ when $|j-k| = s$.

Can rewrite the Fourier transform as:

$$\|\tilde{T} - T\|_2 \leq \max_{f\in[0,1]} \sum_{s=0}^{d} [a_s - \tilde{a}_s] \cdot \sin(2\pi sf) = \max_{f\in[0,1]} \operatorname{tr}\left(T_R - \tilde{T}_R, M_f\right)$$

where $T_R, \tilde{T}_R$ are the principal submatrices of T and $\tilde{T}$ restricted to the indices in the ruler R.

Formulation as Trace Bound: For fixed f let M_f be the Toeplitz matrix with $(M_f)_{j,k} = \frac{sin(2\pi sf)}{|R_s|}$ when $|j-k| = s$.

Can rewrite the Fourier transform as:

$$\|\tilde{T} - T\|_2 \leq \max_{f \in [0,1]} \sum_{s=0}^{d} [a_s - \tilde{a}_s] \cdot \sin(2\pi sf) = \max_{f \in [0,1]} \operatorname{tr}\left(T_R - \tilde{T}_R, M_f\right)$$

where $T_R, \tilde{T}_R$ are the principal submatrices of T and $\tilde{T}$ restricted to the indices in the ruler R.

Formulation as Trace Bound: For fixed f let M_f be the Toeplitz matrix with $(M_f)_{j,k} = \frac{sin(2\pi sf)}{|R_s|}$ when $|j-k| = s$.

Can rewrite the Fourier transform as:

$$\|\tilde{T} - T\|_2 \leq \max_{f \in [0,1]} \sum_{s=0}^{d} [a_s - \tilde{a}_s] \cdot \sin(2\pi sf) = \max_{f \in [0,1]} \operatorname{tr}\left(T_R - \hat{T}_R, M_f\right)$$

where $T_R, \tilde{T}_R$ are the principal submatrices of T and $\tilde{T}$ restricted to the indices in the ruler R.

$$\|\tilde{T}_R - T_R\|_2 \leq \max_{f \in [0,1]} \operatorname{tr}\left(T_R - \hat{T}_R, M_f\right)$$

$$\|\tilde{T}_R - T_R\|_2 \leq \max_{f \in [0,1]} \operatorname{tr}\left(T_R - \hat{T}_R, M_f\right)$$

Concentration Bound: (Hanson-Wright) For fixed f, if $n = \tilde{O}(1/\varepsilon^2)$ can bound the righthand side with high prob. by:

$$\varepsilon\|T_R\|_2 \cdot \|M_f\|_F \leq \varepsilon\|T_R\|_2 \cdot \sqrt{d} \leq \varepsilon\|T\|_2 \cdot \sqrt{d}$$

since each entry of $M_f = \frac{sin(2\pi sf)}{|R_s|}$ for some s so $\|M_f\|_F \leq \sqrt{d}$.

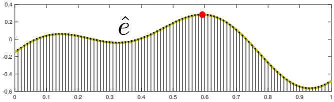
$$\|\tilde{T}_R - T_R\|_2 \leq \max_{f\in[0,1]} \operatorname{tr}\left(T_R - \hat{T}_R, M_f\right)$$

Concentration Bound: (Hanson-Wright) For fixed f, if $n = \tilde{O}(1/\varepsilon^2)$ can bound the righthand side with high prob. by:

$$\varepsilon\|T_R\|_2 \cdot \|M_f\|_F \leq \varepsilon\|T_R\|_2 \cdot \sqrt{d} \leq \varepsilon\|T\|_2 \cdot \sqrt{d}$$

since each entry of $M_f = \frac{sin(2\pi sf)}{|R_s|}$ for some s so $\|M_f\|_F \leq \sqrt{d}$.

- Setting $\varepsilon' = \varepsilon/\sqrt{d}$ and union bounding over a net of f values gives our $n = \tilde{O}(d/\varepsilon^2)$ bound.



$$\|\tilde{T}_R - T_R\|_2 \leq \max_{f \in [0,1]} \operatorname{tr}\left(T_R - \hat{T}_R, M_f\right)$$

Concentration Bound: (Hanson-Wright) For fixed f, if $n = \tilde{O}(1/\varepsilon^2)$ can bound the righthand side with high prob. by:

$$\varepsilon\|T_R\|_2 \cdot \|M_f\|_F \leq \varepsilon\|T_R\|_2 \cdot \sqrt{d} \leq \varepsilon\|T\|_2 \cdot \sqrt{d}$$

since each entry of $M_f = \frac{sin(2\pi sf)}{|R_s|}$ for some s so $\|M_f\|_F \leq \sqrt{d}$.

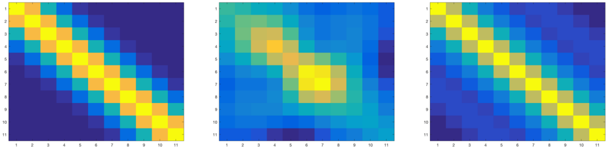
- Setting $\varepsilon' = \varepsilon/\sqrt{d}$ and union bounding over a net of f values gives our $n = \tilde{O}(d/\varepsilon^2)$ bound.
- The more *coverage* R has (the larger the $|R_s|$ is on average), the smaller $\|M_f\|_F$ will be. Let's us interpolate between minimal entry sample complexity and minimal vector sample complexity.

FULL RULER SAMPLE COMPLEXITY

For $R = [d]$, coverage is maximal and $\|M_f\|_F = O(\sqrt{\log d})$, letting us achieve vector sample complexity $n = \tilde{O}\left(\frac{1}{\varepsilon^2}\right)$.

For $R = [d]$, coverage is maximal and $\|M_f\|_F = O(\sqrt{\log d})$, letting us achieve vector sample complexity $n = \tilde{O}\left(\frac{1}{\varepsilon^2}\right)$.

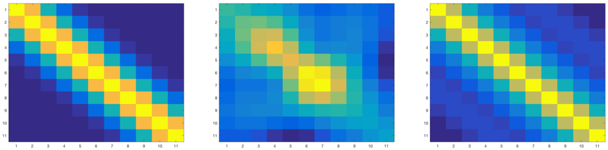
- Algorithm is equivalent to setting $T = \mathrm{avg}\left(\frac{1}{n}\sum x^{(j)} {x^{(j)}}^T\right)$.



True covariance T | Empirical covariance $\hat{T}$ | Improved estimator $avg(\hat{T})$

For $R = [d]$, coverage is maximal and $\|M_f\|_F = O(\sqrt{\log d})$, letting us achieve vector sample complexity $n = \tilde{O}\left(\frac{1}{\epsilon^2}\right)$.

- Algorithm is equivalent to setting $T = \mathrm{avg}\left(\frac{1}{n}\sum x^{(j)}x^{(j)^T}\right)$.

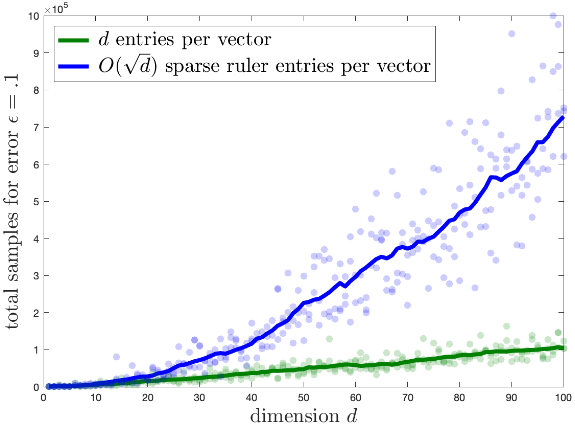


True covariance T Empirical covariance $\hat{T}$ Improved estimator $avg(\hat{T})$

- Improves on sample complexity of just using the empirical covariance by a $\tilde{O}(d)$ factor.

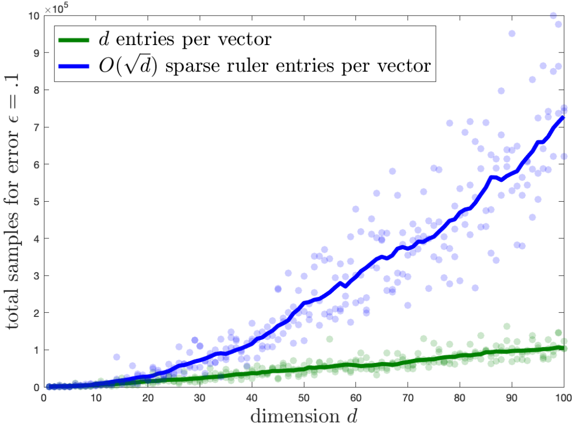
Total sample complexity is $O(\sqrt{d}) \cdot \tilde{O}(d) = \tilde{O}(d^{3/2})$ for sparse ruler vs. $d \cdot \tilde{O}(1) = \tilde{O}(d)$ for full sample estimation.

Total sample complexity is $O(\sqrt{d}) \cdot \tilde{O}(d) = \tilde{O}(d^{3/2})$ for sparse ruler vs. $d \cdot \tilde{O}(1) = \tilde{O}(d)$ for full sample estimation.



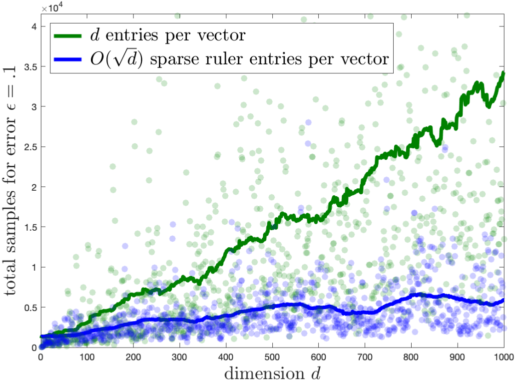
SPARSE RULER VS. FULL RULER

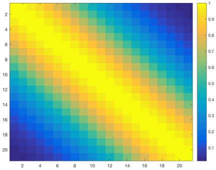
Total sample complexity is $O(\sqrt{d}) \cdot \tilde{O}(d) = \tilde{O}(d^{3/2})$ for sparse ruler vs. $d \cdot \tilde{O}(1) = \tilde{O}(d)$ for full sample estimation.

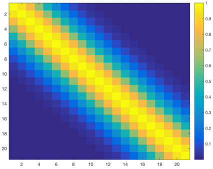


- Prove bounds are tight when T is the identity.

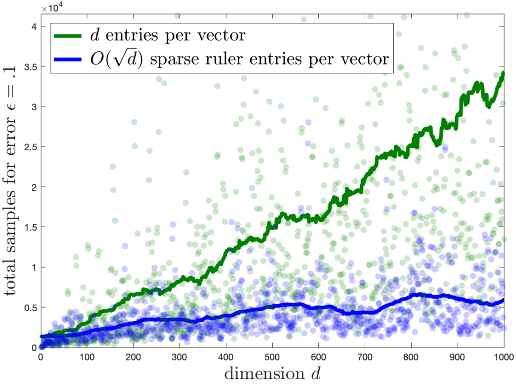
IS THERE ALWAYS A TRADEOFF?

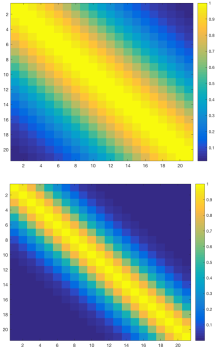




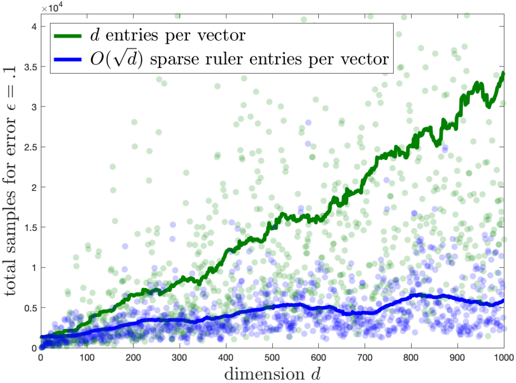


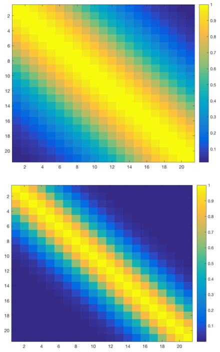
IS THERE ALWAYS A TRADEOFF?



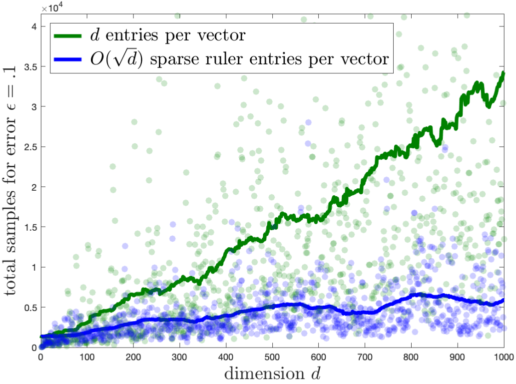


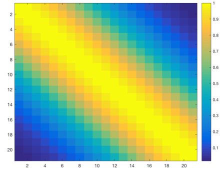
- Total sample complexity is $\tilde{O}(\sqrt{d})$ for sparse ruler estimation vs. $\tilde{O}(d)$ for full sample estimation.

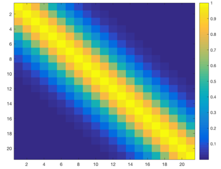




- Total sample complexity is $\tilde{O}(\sqrt{d})$ for sparse ruler estimation vs. $\tilde{O}(d)$ for full sample estimation.
- Sparse rulers give much better total sample complexity when T is (approximately) low-rank.

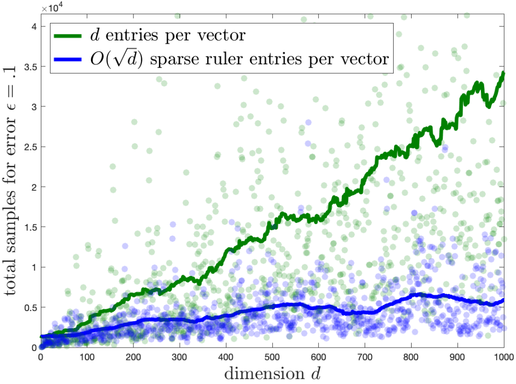


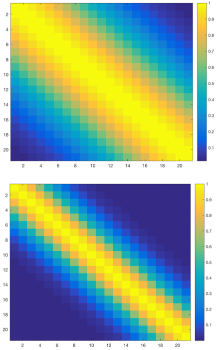




- Total sample complexity is $\tilde{O}(\sqrt{d})$ for sparse ruler estimation vs. $\tilde{O}(d)$ for full sample estimation.
- Sparse rulers give much better total sample complexity when T is (approximately) low-rank.

IS THERE ALWAYS A TRADEOFF?





- Total sample complexity is $\tilde{O}(\sqrt{d})$ for sparse ruler estimation vs. $\tilde{O}(d)$ for full sample estimation.
- Sparse rulers give much better total sample complexity when T is (approximately) low-rank. **Can we explain this?**

Recall that we have with $n = \tilde{O}(1/\varepsilon^2)$ samples:

$$\|T - \tilde{T}\|_2 \leq \varepsilon \|T_R\|_2 \cdot \|M_f\|_F \leq \varepsilon \|T_R\|_2 \sqrt{d} \leq \varepsilon \|T\|_2 \sqrt{d}.$$

.

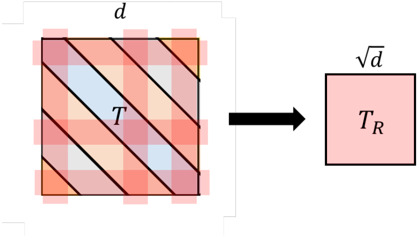
Recall that we have with $n = \tilde{O}(1/\varepsilon^2)$ samples:

$$\|T - \tilde{T}\|_2 \leq \varepsilon \|T_R\|_2 \cdot \|M_f\|_F \leq \varepsilon \|T_R\|_2 \sqrt{d} \leq \varepsilon \|T\|_2 \sqrt{d}.$$

.

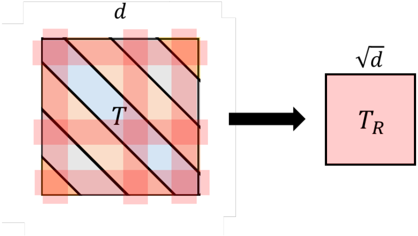
Recall that we have with $n = \tilde{O}(1/\varepsilon^2)$ samples:

$$\|T - \tilde{T}\|_2 \leq \varepsilon \|T_R\|_2 \cdot \|M_f\|_F \leq \varepsilon \|T_R\|_2 \sqrt{d} \leq \varepsilon \|T\|_2 \sqrt{d}.$$



Recall that we have with $n = \tilde{O}(1/\varepsilon^2)$ samples:

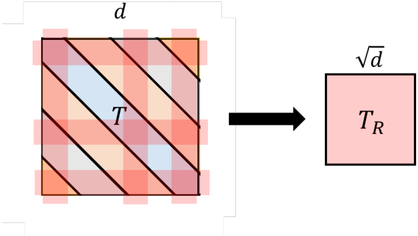
$$\|T - \tilde{T}\|_2 \leq \varepsilon \|T_R\|_2 \cdot \|M_f\|_F \leq \varepsilon \|T_R\|_2 \sqrt{d} \leq \varepsilon \|T\|_2 \sqrt{d}.$$



- If T is the identity, $\|T\|_2 = \|T_R\|_2 = 1$. But this is 'very' full-rank.

Recall that we have with $n = \tilde{O}(1/\varepsilon^2)$ samples:

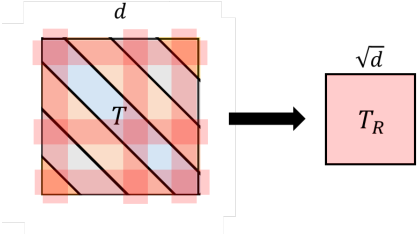
$$\|T - \tilde{T}\|_2 \leq \varepsilon \|T_R\|_2 \cdot \|M_f\|_F \leq \varepsilon \|T_R\|_2 \sqrt{d} \leq \varepsilon \|T\|_2 \sqrt{d}.$$



- If T is the identity, $\|T\|_2 = \|T_R\|_2 = 1$. But this is 'very' full-rank.
- Low-rank matrices cannot look like the identity – have significant off diagonal mass [MMW '19].

Recall that we have with $n = \tilde{O}(1/\varepsilon^2)$ samples:

$$\|T - \tilde{T}\|_2 \leq \varepsilon\|T_R\|_2 \cdot \|M_f\|_F \leq \varepsilon\|T_R\|_2\sqrt{d} \leq \varepsilon\|T\|_2\sqrt{d}.$$



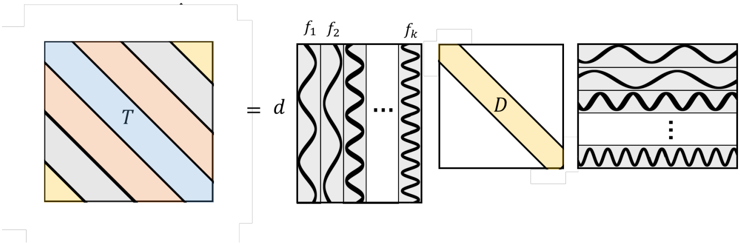
- If T is the identity, $\|T\|_2 = \|T_R\|_2 = 1$. But this is 'very' full-rank.
- Low-rank matrices cannot look like the identity – have significant off diagonal mass [MMW '19].
- **Upshot**: Show $\|T_R\|_2 \leq \frac{k}{\sqrt{d}}\|T\|_2$. Setting $\varepsilon' = \varepsilon/k$ obtain total sample complexity $\tilde{O}\left(\frac{\sqrt{d}k^2}{\varepsilon^2}\right)$.

Remainder of the talk: Will sketch a different approach to low-rank Toeplitz covariance estimation using sparse Fourier transform methods.

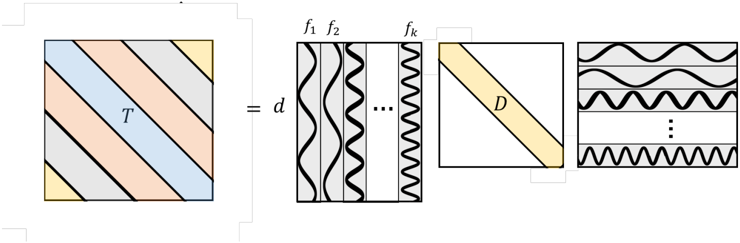
Remainder of the talk: Will sketch a different approach to low-rank Toeplitz covariance estimation using sparse Fourier transform methods.

- Connections between these two approaches.

Vandermonde Decomposition: Any rank-k Toeplitz $T \in R^{d\times d}$ can be written as $F_S D F_S$ where $F_S \in \mathbb{R}^{d\times k}$ is an 'off-grid' Fourier transform matrix with frequencies $f_1, \ldots, f_k$ and D is a positive diagonal matrix.



Vandermonde Decomposition: Any rank-k Toeplitz $T \in R^{d\times d}$ can be written as $F_S D F_S$ where $F_S \in \mathbb{R}^{d\times k}$ is an 'off-grid' Fourier transform matrix with frequencies $f_1, \ldots, f_k$ and D is a positive diagonal matrix.

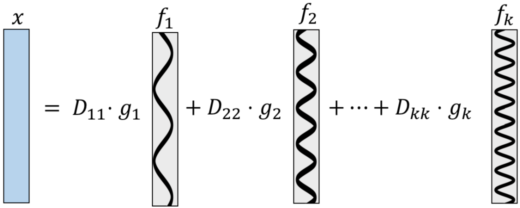


- Any sample $x \sim \mathcal{N}(0, T)$ can be written as $F_S D^{1/2} g$ for $g \sim \mathcal{N}(0, I)$. $\mathbb{E}[xx^T] = F_S D^{1/2} \mathbb{E}[gg^T] D^{1/2} F_S^* = T$.

$x \sim \mathcal{N}(0, T) = F_S D^{1/2} g$ is a Fourier sparse function.

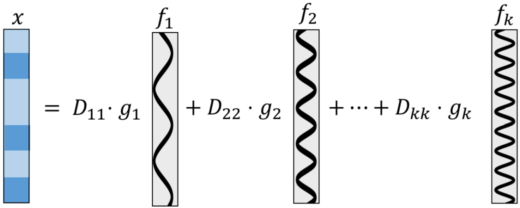
SAMPLE RECOVERY VIA SPARSE FOURIER TRANSFORM

$x \sim \mathcal{N}(0, T) = F_S D^{1/2} g$ is a Fourier sparse function.



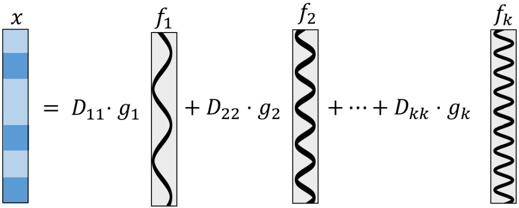
SAMPLE RECOVERY VIA SPARSE FOURIER TRANSFORM

$x \sim \mathcal{N}(0, T) = F_S D^{1/2} g$ is a Fourier sparse function.



- Can recover exactly e.g. via Prony's sparse Fourier transform method by reading any $2k$ entries.

$x \sim \mathcal{N}(0, T) = F_S D^{1/2} g$ is a Fourier sparse function.



- Can recover exactly e.g. via Prony's sparse Fourier transform method by reading any $2k$ entries.
- Take $n = \tilde{O}(1/\varepsilon^2)$ samples, recover each in full by reading $2k$ entries, and then apply our earlier resut for full ruler $R = [d]$. Total sample complexity: $\tilde{O}(k/\varepsilon^2)$.

What about when T is close to, but not exactly rank-k?

What about when T is close to, but not exactly rank-k?

- Prony's method totally fails in this case.

What about when T is close to, but not exactly rank-k?

- Prony's method totally fails in this case.

Step 1: Prove that when T is close to low-rank, there is some set of k frequencies that approximately spans each $x^{(j)} \sim \mathcal{N}(0, T)$.

What about when T is close to, but not exactly rank-k?

- Prony's method totally fails in this case.

Step 1: Prove that when T is close to low-rank, there is some set of k frequencies that approximately spans each $x^{(j)} \sim \mathcal{N}(0, T)$.

- Not as easy as it sounds.

What about when T is close to, but not exactly rank-k?

- Prony's method totally fails in this case.

Step 1: Prove that when T is close to low-rank, there is some set of k frequencies that approximately spans each $x^{(j)} \sim \mathcal{N}(0, T)$.

- Not as easy as it sounds.

Step 2: Use a robust sparse Fourier transform method to approximately recover $x^{(1)}, \ldots, x^{(n)}$ and then estimate T from these samples.

What about when T is close to, but not exactly rank-k?

- Prony's method totally fails in this case.

Step 1: Prove that when T is close to low-rank, there is some set of k frequencies that approximately spans each $x^{(j)} \sim \mathcal{N}(0, T)$.

- Not as easy as it sounds.

Step 2: Use a robust sparse Fourier transform method to approximately recover $x^{(1)}, \ldots, x^{(n)}$ and then estimate T from these samples.

- Well studied in TCS, especially in the case when $f_1, \ldots, f_k$ are 'on grid' integer frequencies.

Step 1: Prove that when T is close to low-rank, there is some set of k frequencies that approximately spans each $x^{(j)} \sim \mathcal{N}(0, T)$.

Step 1: Prove that when T is close to low-rank, there is some set of k frequencies that approximately spans each $x^{(j)} \sim \mathcal{N}(0, T)$.

- We give a proof via a column subset selection result (see e.g., Guruswami Sinop '12):

Step 1: Prove that when T is close to low-rank, there is some set of k frequencies that approximately spans each $x^{(j)} \sim \mathcal{N}(0, T)$.

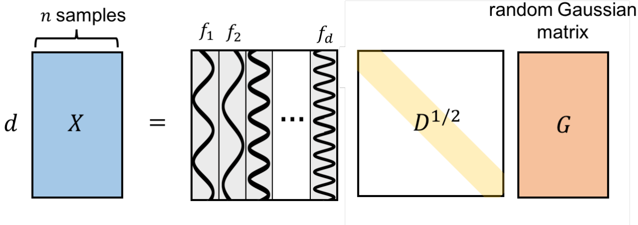
- We give a proof via a column subset selection result (see e.g., Guruswami Sinop '12):

  Theorem: Any $A \in \mathbb{R}^{n \times d}$, contains a subset of $O(k/\varepsilon)$ columns, C such that:

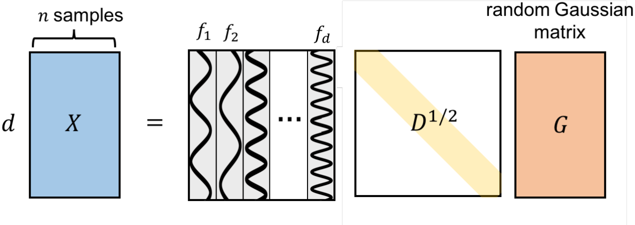
$$\|A - P_C \cdot A\|_F^2 \leq (1+\varepsilon) \min_{\text{rank}-k\ M} \|A - M\|_F^2.$$

$x^{(1)}, \ldots, x^{(n)} \sim \mathcal{N}(0, T)$ can be written as $X = F_S D^{1/2} G$ where columns of G are distributed as $\mathcal{N}(0, I)$.

$x^{(1)}, \ldots, x^{(n)} \sim \mathcal{N}(0, T)$ can be written as $X = F_S D^{1/2} G$ where columns of G are distributed as $\mathcal{N}(0, I)$.

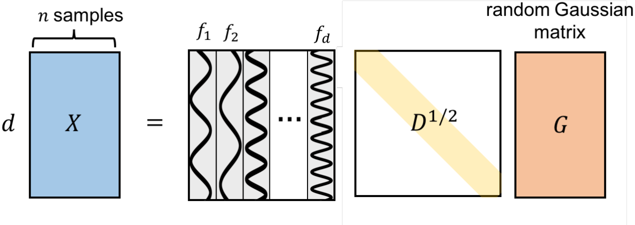


$x^{(1)}, \ldots, x^{(n)} \sim \mathcal{N}(0, T)$ can be written as $X = F_S D^{1/2} G$ where columns of G are distributed as $\mathcal{N}(0, I)$.



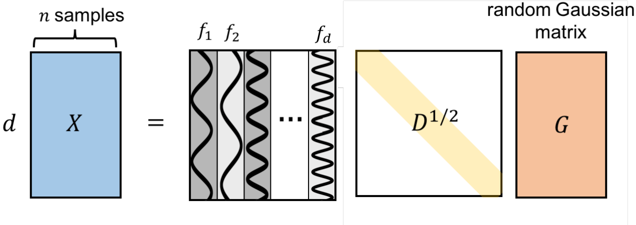
- Think of G as a linear sketch that ensures $F_S D^{1/2} G \approx F_S D^{1/2}$ (formally a projection-cost preserving sketch [CEMMP '15]).

$x^{(1)}, \ldots, x^{(n)} \sim \mathcal{N}(0, T)$ can be written as $X = F_S D^{1/2} G$ where columns of G are distributed as $\mathcal{N}(0, I)$.



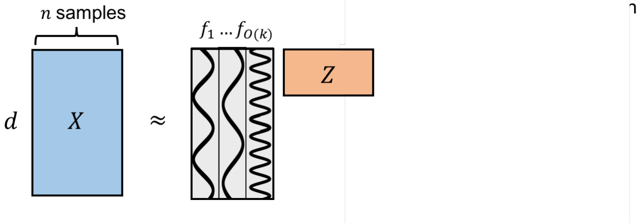
- Think of G as a linear sketch that ensures $F_S D^{1/2} G \approx F_S D^{1/2}$ (formally a projection-cost preserving sketch [CEMMP '15]).
- Apply column subset selection result to $F_S D^{1/2}$.

$x^{(1)}, \ldots, x^{(n)} \sim \mathcal{N}(0, T)$ can be written as $X = F_S D^{1/2} G$ where columns of G are distributed as $\mathcal{N}(0, I)$.



- Think of G as a linear sketch that ensures $F_S D^{1/2} G \approx F_S D^{1/2}$ (formally a projection-cost preserving sketch [CEMMP '15]).
- Apply column subset selection result to $F_S D^{1/2}$.

$x^{(1)}, \ldots, x^{(n)} \sim \mathcal{N}(0, T)$ can be written as $X = F_S D^{1/2} G$ where columns of G are distributed as $\mathcal{N}(0, I)$.



- Think of G as a linear sketch that ensures $F_S D^{1/2} G \approx F_S D^{1/2}$ (formally a projection-cost preserving sketch [CEMMP '15]).
- Apply column subset selection result to $F_S D^{1/2}$.

Step 2: Recover frequencies $f_1, \ldots, f_m$ and $Z \in \mathbb{C}^{m \times n}$ with $X \approx F_M \cdot Z$. Then estimate T using this approximation.

Step 2: Recover frequencies $f_1, \ldots, f_m$ and $Z \in \mathbb{C}^{m \times n}$ with $X \approx F_M \cdot Z$. Then estimate T using this approximation.

- Find frequencies via brute force search over a net.

Step 2: Recover frequencies $f_1, \ldots, f_m$ and $Z \in \mathbb{C}^{m \times n}$ with $X \approx F_M \cdot Z$. Then estimate T using this approximation.

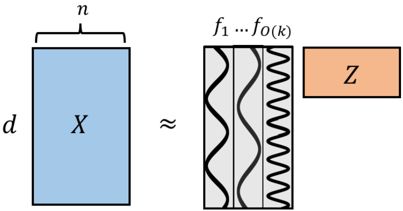
- Find frequencies via brute force search over a net.
- At each step of the search, for a given F_M, we must find Z that reconstructs X as well as possible using these frequencies. How do we do this without reading all of X?

Want to find Z satisfying the approximate regression guarantee:

$$\|X - F_M Z\|_F^2 = O(1) \cdot min_Y \|X - F_M Y\|_F^2.$$

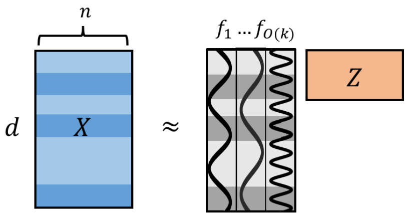
Want to find Z satisfying the approximate regression guarantee:

$$\|X - F_M Z\|_F^2 = O(1) \cdot min_Y \|X - F_M Y\|_F^2.$$



Want to find Z satisfying the approximate regression guarantee:

$$\|X - F_M Z\|_F^2 = O(1) \cdot min_Y \|X - F_M Y\|_F^2.$$

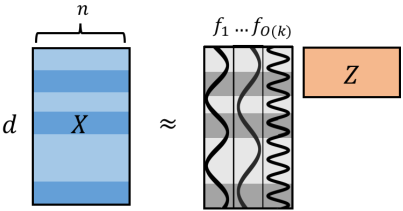


- Suffices to sample $\tilde{O}(k)$ rows by the leverage scores of F_M and solve the regression problem just considering these rows.

APPROXIMATE FREQUENCY REGRESSION

Want to find Z satisfying the approximate regression guarantee:

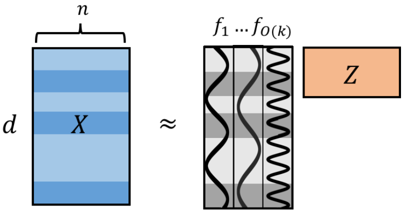
$$\|X - F_M Z\|_F^2 = O(1) \cdot min_Y \|X - F_M Y\|_F^2.$$



- Suffices to sample $\tilde{O}(k)$ rows by the leverage scores of F_M and solve the regression problem just considering these rows.
- **Remark:** If $f_1, \dots, f_m$ are 'on-grid' integers, the columns of F_M are orthonormal and the leverage scores are all k/n

Want to find Z satisfying the approximate regression guarantee:

$$\|X - F_M Z\|_F^2 = O(1) \cdot min_Y \|X - F_M Y\|_F^2.$$



- Suffices to sample $\tilde{O}(k)$ rows by the leverage scores of F_M and solve the regression problem just considering these rows.
- **Remark:** If $f_1, \ldots, f_m$ are 'on-grid' integers, the columns of F_M are orthonormal and the leverage scores are all $k/n \rightarrow$ RIP for subsampled Fourier matrices.

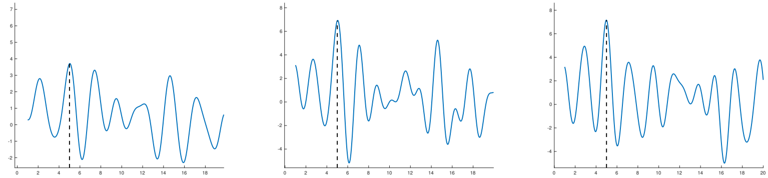
Leverage scores measure much large a function in the column span of F_M can be at index i (i.e., how important that index may be in the regression.)

$$\tau_i(F_M) = \max_y \frac{(F_M y)_i^2}{\|F_M y\|_2^2}.$$

FOURIER LEVERAGE SCORES

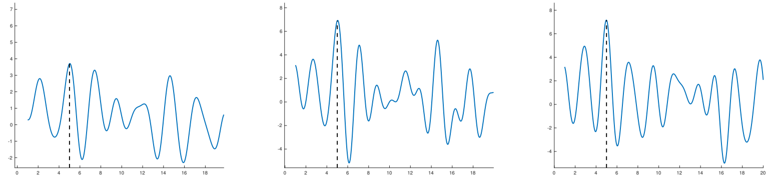
Leverage scores measure much large a function in the column span of F_M can be at index i (i.e., how important that index may be in the regression.)

$$\tau_i(F_M) = \max_y \frac{(F_M y)_i^2}{\|F_M y\|_2^2}.$$



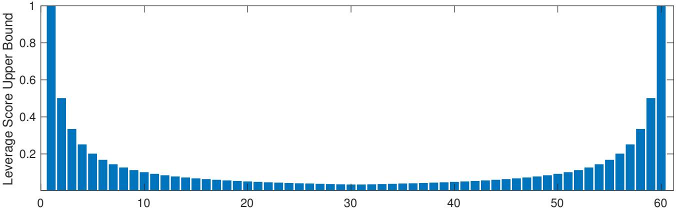
Leverage scores measure much large a function in the column span of F_M can be at index i (i.e., how important that index may be in the regression.)

$$\tau_i(F_M) = \max_y \frac{(F_M y)_i^2}{\|F_M y\|_2^2}.$$

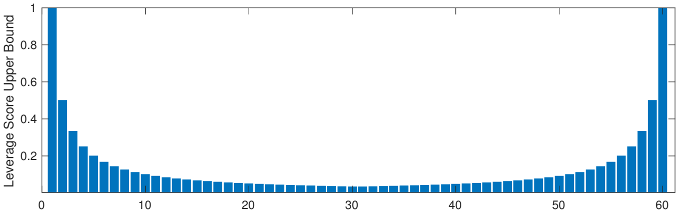


- Using that $F_M y$ is a Fourier sparse function we can bound this quantity a priori, without any dependence on F_M.

Extend bounds of [Chen Kane Price Song '16] to give explicit function upper bounding the leverage scores of any F_M:



Extend bounds of [Chen Kane Price Song '16] to give explicit function upper bounding the leverage scores of any F_M:



Since this distribution is universal, can sample one set of entries by these leverages scores, and find $X \approx F_M \cdot Z$ with high probability for any set of frequencies $f_1, \ldots, f_m$ in net.

1. Sample poly(k/ε) indices $R \subset [d]$ according to the sparse Fourier leverage distribution (a random ‘ultra-sparse’ ruler)

1. Sample poly(k/ε) indices $R \subset [d]$ according to the sparse Fourier leverage distribution (a random 'ultra-sparse' ruler)
2. For all $f_1, \ldots, f_m$ in net $\mathcal{N}$: Compute approximate projection:

$$Z = \underset{Z \in \mathbb{C}^{m \times n}}{\arg\min} \|X_R - (F_M)_R Z\|_F^2.$$

1. Sample poly(k/ε) indices $R \subset [d]$ according to the sparse Fourier leverage distribution (a random 'ultra-sparse' ruler)
2. For all $f_1, \ldots, f_m$ in net $\mathcal{N}$: Compute approximate projection:

$$Z = \arg\min_{Z \in \mathbb{C}^{m \times n}} \|X_R - (F_M)_R Z\|_F^2.$$

3. Set $\tilde{X} = F_M^\star \cdot Z^\star$ to the best frequency-based approximation.

1. Sample poly(k/ε) indices $R \subset [d]$ according to the sparse Fourier leverage distribution (a random 'ultra-sparse' ruler)
2. For all $f_1, \ldots, f_m$ in net $\mathcal{N}$: Compute approximate projection:

$$Z = \arg\min_{Z \in \mathbb{C}^{m \times n}} \|X_R - (F_M)_R Z\|_F^2.$$

3. Set $\tilde{X} = F_M^\star \cdot Z^\star$ to the best frequency-based approximation.
4. Return $\tilde{T} = avg(\tilde{X}\tilde{X}^T)$.

1. Sample poly(k/ε) indices $R \subset [d]$ according to the sparse Fourier leverage distribution (a random 'ultra-sparse' ruler)
2. For all $f_1, \ldots, f_m$ in net $\mathcal{N}$: Compute approximate projection:

$$Z = \arg\min_{Z \in \mathbb{C}^{m \times n}} \|X_R - (F_M)_R Z\|_F^2.$$

3. Set $\tilde{X} = F_M^\star \cdot Z^\star$ to the best frequency-based approximation.
4. Return $\tilde{T} = avg(\tilde{X}\tilde{X}^T)$.

Sample Complexity: Gives $\|T - \tilde{T}\|_2 \leq \varepsilon \|T\|_2 + f(T - T_k)$ when X contains $n = \tilde{O}(\text{poly}(k/\epsilon))$ samples. Entry sample complexity poly(k/ε), total sample complexity $\tilde{O}(\text{poly}(k/\varepsilon))$.

Concrete.

Concrete.

- Runtime efficiency?

Concrete.

- Runtime efficiency?
  - Can likely avoid exponential time net approach using off-grid sparse Fourier transform of [Chen Kane Price Song '16.]
  - Convex optimization-based approaches and 'off-grid' RIP?
  - Matrix sparse Fourier transform $X \approx F_M \cdot Z$. Connections to MUSIC, ESPRIT, etc.
  - In process, maybe improve our sample complexity.

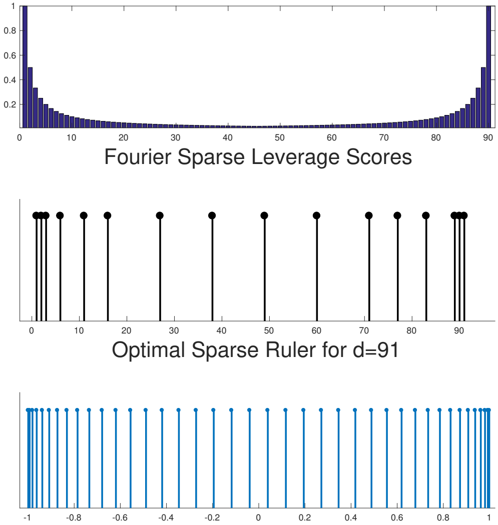
Concrete.

- Runtime efficiency?
  - Can likely avoid exponential time net approach using off-grid sparse Fourier transform of [Chen Kane Price Song '16.]
  - Convex optimization-based approaches and 'off-grid' RIP?
  - Matrix sparse Fourier transform $X \approx F_M \cdot Z$. Connections to MUSIC, ESPRIT, etc.
  - In process, maybe improve our sample complexity.
- 'Continuous' setting with sample access to a arbitrary positions of a signal with stationary covariance. (E.g., $x^{(1)}, \ldots, x^{(n)}$ may be snapshots of this signal.)

Concrete.

- Runtime efficiency?
  - Can likely avoid exponential time net approach using off-grid sparse Fourier transform of [Chen Kane Price Song '16.]
  - Convex optimization-based approaches and 'off-grid' RIP?
  - Matrix sparse Fourier transform $X \approx F_M \cdot Z$. Connections to MUSIC, ESPRIT, etc.
  - In process, maybe improve our sample complexity.
- 'Continuous' setting with sample access to a arbitrary positions of a signal with stationary covariance. (E.g., $x^{(1)}, \ldots, x^{(n)}$ may be snapshots of this signal.)
  - Sample complexity bounds and tradeoffs for applications like direction-of-arrival estimation, Doppler imaging.

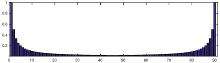
CONNECTIONS BETWEEN SAMPLING SCHEMES

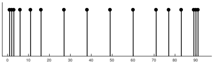


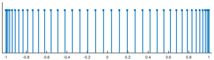
Fourier Sparse Leverage Scores

Optimal Sparse Ruler for d=91

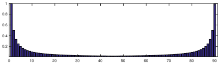
Degree 40 Chebyshev Nodes


Fourier Sparse Leverage Scores

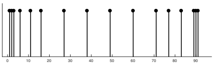

Optimal Sparse Ruler for d=91


Degree 40 Chebyshev Nodes

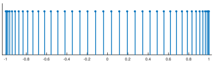
- Some Formal Connections:



Fourier Sparse Leverage Scores

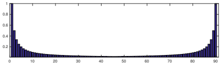


Optimal Sparse Ruler for d=91

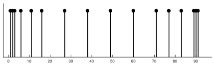


Degree 40 Chebyshev Nodes

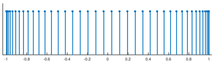
- Some Formal Connections:
  - Limiting density of Chebyshev nodes is the leverage score distribution for k degree polynomials.



Fourier Sparse Leverage Scores

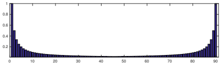


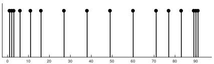
Optimal Sparse Ruler for d=91

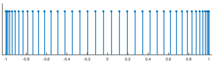


Degree 40 Chebyshev Nodes

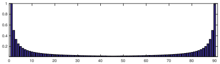
- **Some Formal Connections:**
  - Limiting density of Chebyshev nodes is the leverage score distribution for k degree polynomials.
  - Sampling $O(\sqrt{d})$ indices via Fourier sparse leverage scores gives a sparse ruler with good probability.

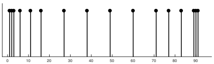

Fourier Sparse Leverage Scores

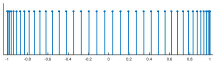

Optimal Sparse Ruler for d=91


Degree 40 Chebyshev Nodes

- Some Formal Connections:
  - Limiting density of Chebyshev nodes is the leverage score distribution for k degree polynomials.
  - Sampling $O(\sqrt{d})$ indices via Fourier sparse leverage scores gives a sparse ruler with good probability.
- Also connected to multi-coset and non-uniform sampling schemes used in signal processing.


Fourier Sparse Leverage Scores


Optimal Sparse Ruler for d=91


Degree 40 Chebyshev Nodes

- **Some Formal Connections:**
  - Limiting density of Chebyshev nodes is the leverage score distribution for k degree polynomials.
  - Sampling $O(\sqrt{d})$ indices via Fourier sparse leverage scores gives a sparse ruler with good probability.
- Also connected to multi-coset and non-uniform sampling schemes used in signal processing.
- Seem to have a lot more to understand.

Thanks! Questions?

Paper draft and slides available at `cameronmusco.com`