# Structured Covariance Estimation

**Christopher Musco** (NYU, Tandon School of Engineering)

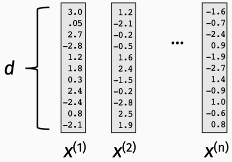
With Yonina Eldar (Weizmann Institute), Jerry Li (Microsoft Research), Cameron Musco (UMass Amherst), Hannah Lawrence (Flatiron Institute).

Basic statistical problem:

- Distribution $\mathcal{D}$ over $d$-dimensional vectors.
- $\mathbb{E}_{x \sim \mathcal{D}}[xx^T] = C$. $C_{j,k}$ is the covariance between $x_j$ and $x_k$.

Basic statistical problem:

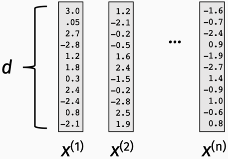
- Distribution $\mathcal{D}$ over $d$-dimensional vectors.
- $\mathbb{E}_{x\sim\mathcal{D}}[xx^T] = C$. $C_{j,k}$ is the covariance between $x_j$ and $x_k$.




How many samples $x^{(1)}, \ldots, x^{(n)} \sim \mathcal{D}$ are required to learn $C$?

**Basic statistical problem:**

- Distribution $\mathcal{D}$ over $d$-dimensional vectors.
- $\mathbb{E}_{x\sim\mathcal{D}}[xx^T] = C$. $C_{j,k}$ is the covariance between $x_j$ and $x_k$.




**How many samples $x^{(1)}, \ldots, x^{(n)} \sim \mathcal{D}$ are required to learn $C$?**

**Reasonable goal:** Find $\tilde{C}$ with $\|C - \tilde{C}\|_2 \leq \epsilon \|C\|_2$.[1]

[1] Lots of other possible metrics.

Assuming $\mathcal{D}$ is high-dimensional Gaussian, subgaussian, subexponential:

Assuming $\mathcal{D}$ is high-dimensional Gaussian, subgaussian, subexponential:

**Known bound:** $\Theta\left(\frac{d}{\epsilon^2}\right)$ samples are necessary and sufficient.

**Estimator:** Simple sample covariance.

$$\tilde{C} = \sum_{i=1}^{n} x^{(i)} x^{(i)T}.$$

**Analysis:** Matrix concentration bounds or JL Lemma + $\epsilon$-net (e.g., Vershynin, "High Dimensional Probability", 2019).

Assuming $\mathcal{D}$ is high-dimensional Gaussian, subgaussian, subexponential:

**Known bound:** $\Theta\left(\frac{d}{\epsilon^2}\right)$ samples are necessary and sufficient.

**Estimator:** Simple sample covariance.

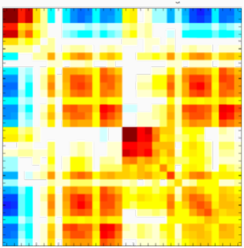
$$\tilde{C} = \sum_{i=1}^{n} x^{(i)} x^{(i)T}.$$

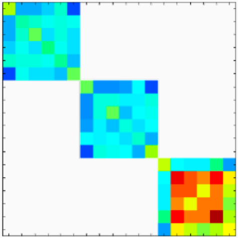
**Analysis:** Matrix concentration bounds or JL Lemma + $\epsilon$-net (e.g., Vershynin, “High Dimensional Probability”, 2019).

Can we improve the dependence on $d$?

## What is we know $C$ has additional structure?

- Block structure.
- Low-rank, low-rank + diagonal.
- Diagonal, banded.
- Many other possibilities.





Some easy improvements over $\Theta\left(\frac{d}{\epsilon^2}\right)$:

Some easy improvements over $\Theta\left(\frac{d}{\epsilon^2}\right)$:

- $C$ is rank-$k$: $\Theta\left(\frac{k}{\epsilon^2}\right)$. Sample covariance.

Some easy improvements over $\Theta\left(\frac{d}{\epsilon^2}\right)$:

- $C$ is rank-$k$: $\Theta\left(\frac{k}{\epsilon^2}\right)$. Sample covariance.
- $C$ is diagonal: $\Theta\left(\frac{\log d}{\epsilon^2}\right)$. Estimate variance $C_{i,i}$ of each index separately. Set $C_{i,j} = 0$.

Some work on more complicated models:

- Sparse graphical models (Meinshausen, Bühlmann, 2006). Dependence on graph sparsity.

But little is known for many natural structures...

But little is known for many natural structures...


**Example:** Spatially structured covariance matrices in ecology.

**This work:** Covariance matrix is Toeplitz.[2]

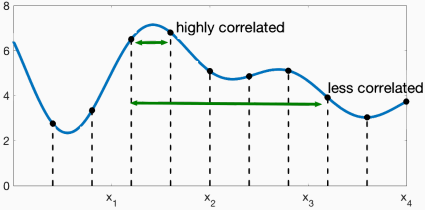
$$T = \begin{bmatrix} a & b & c & d & e \\ b & a & b & c & d \\ c & b & a & b & c \\ d & c & b & a & b \\ e & d & c & b & a \end{bmatrix}$$

**This work:** Covariance matrix is Toeplitz.[2]

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

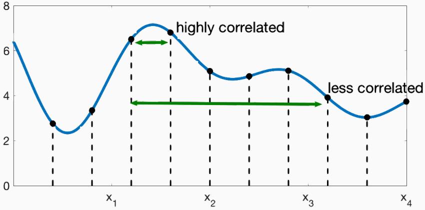
[2] As for any covariance matrix, $T$ must also be positive semidefinite.

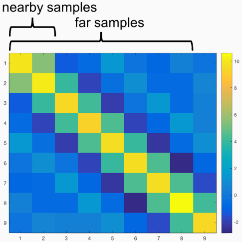
Arises when measurements taken on a spatial or temporal grid.
Covariance depends on distance between them: $\mathbb{E}[x_j \cdot x_k] = f(|j-k|)$.



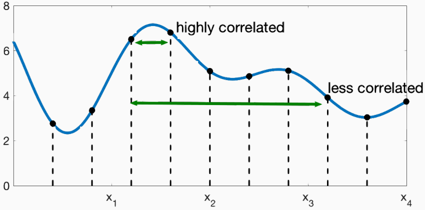
Arises when measurements taken on a spatial or temporal grid.

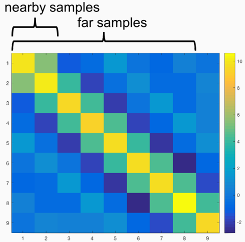
Covariance depends on distance between them: $\mathbb{E}[x_j \cdot x_k] = f(|j-k|)$.

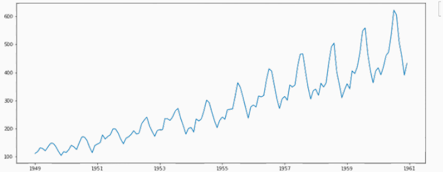


Arises when measurements taken on a spatial or temporal grid.
Covariance depends on distance between them: $\mathbb{E}[x_j \cdot x_k] = f(|j - k|)$.

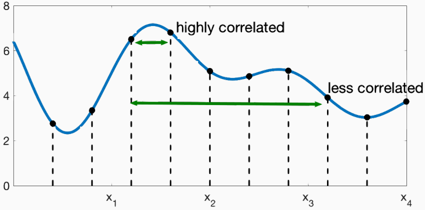


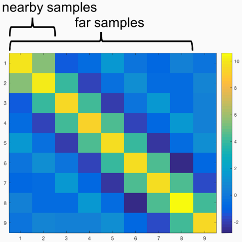



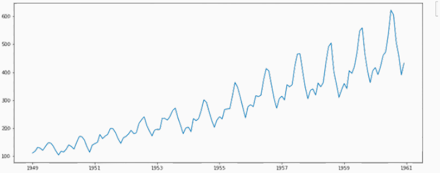



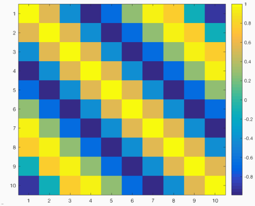
Arises when measurements taken on a spatial or temporal grid.

Covariance depends on distance between them: $\mathbb{E}[x_j \cdot x_k] = f(|j-k|)$.



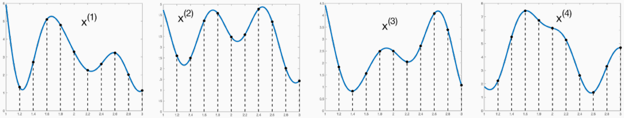





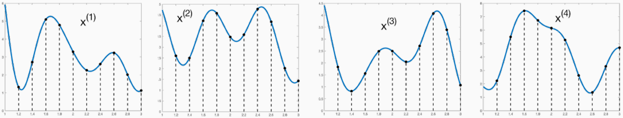
# TOEPLITZ COVARIANCE ESTIMATION

Arises when measurements taken on a spatial or temporal grid.

Covariance depends on distance between them: $\mathbb{E}[x_j \cdot x_k] = f(|j-k|)$.



Arises when measurements taken on a spatial or temporal grid. Covariance depends on distance between them: $\mathbb{E}[x_j \cdot x_k] = f(|j-k|)$.




**Applications in signal processing**: spectrum sensing/cognitive radio, Doppler radar, direction-of-arrival estimation, prediction via Gaussian process regression, etc.

**Note**: Shift-invariant kernel matrices in machine learning are Toeplitz covariance matrices when data points are on a grid.

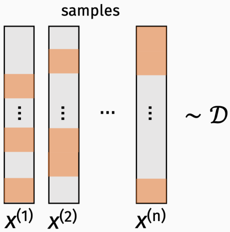
**Goal:** Minimize <u>two types</u> of sample complexity:

**Goal:** Minimize two types of sample complexity:

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

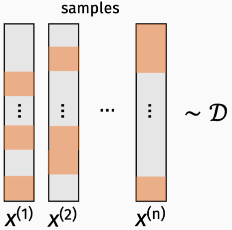
**Goal:** Minimize two types of sample complexity:

- **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 $x^{(1)}, \ldots, x^{(n)}$?



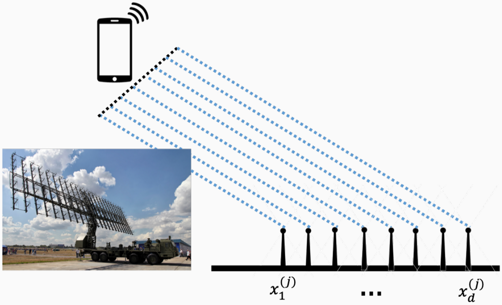
**Goal:** Minimize <u>two types</u> of sample complexity:

- **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 $x^{(1)}, \ldots, x^{(n)}$?

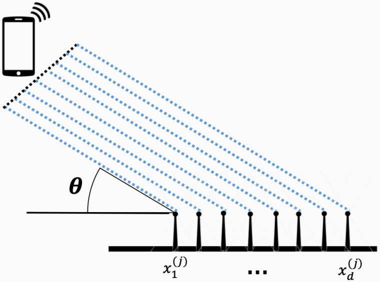



In different applications, these complexities correspond to different costs. <u>Typically there is a tradeoff.</u>

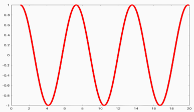
# EXAMPLE: DIRECTION OF ARRIVAL (DOA) ESTIMATION

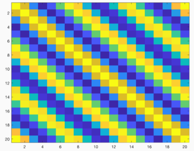


# EXAMPLE: DIRECTION OF ARRIVAL (DOA) ESTIMATION



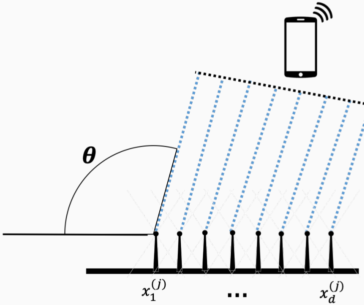

delay between receivers: $\propto \cos(\theta \cdot (k - \ell))$

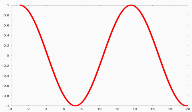


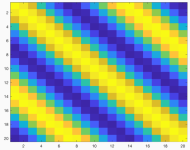


Can back out direction of arrival $\theta$ from covariance structure.

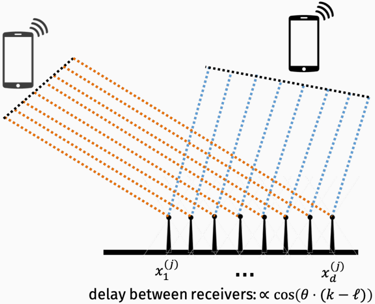
# EXAMPLE: DIRECTION OF ARRIVAL (DOA) ESTIMATION

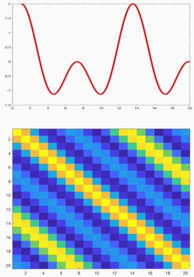




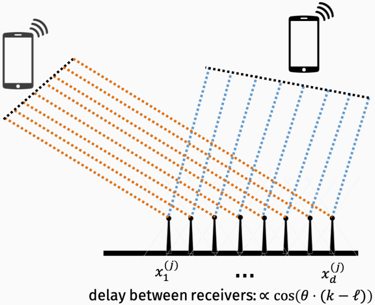
Can back out direction of arrival $\theta$ from covariance structure.

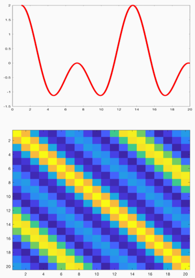


Can back out direction of arrival $\theta$ from covariance structure.

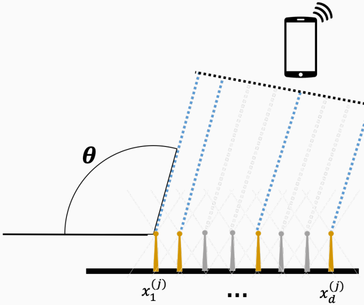
# EXAMPLE: DIRECTION OF ARRIVAL (DOA) ESTIMATION

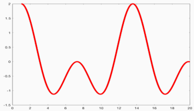


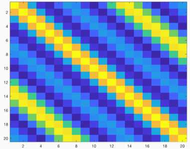



Can back out direction of arrival $\theta$ from covariance structure.

**Vector sample complexity,** $n$ **:** Estimation time (# snapshots).





Can back out direction of arrival $\theta$ from covariance structure.

**Vector sample complexity,** $n$ **:** Estimation time (# snapshots).

**Entry sample complexity,** $s$**:** Number of active receivers.

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

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

- For general covariance matrices, vector sample complexity is $\Theta(d/\epsilon^2)$, entry sample complexity is $d$, so total sample complexity is $\Theta(d^2/\epsilon^2)$.

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

- For general covariance matrices, vector sample complexity is $\Theta(d/\epsilon^2)$, entry sample complexity is $d$, so total sample complexity is $\Theta(d^2/\epsilon^2)$.
- Seems to be interesting even beyond Toeplitz covariance matrices, but not well studied.

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:

# OUR CONTRIBUTIONS

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

**Our contributions:**

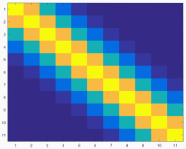
- Non-asymptotic sample complexity bounds by analyzing classic algorithms, including those with sublinear entry sample complexity based on sparse ruler measurements.

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

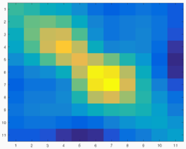
**Our contributions:**

- Non-asymptotic sample complexity bounds by analyzing classic algorithms, including those with sublinear entry sample complexity based on sparse ruler measurements.
- Develop improved algorithms for the case when $T$ is (approximately) low-rank, using techniques from matrix sketching, leverage score-based sampling, and sparse Fourier transform algorithms.

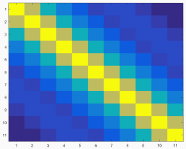
Estimator: $\tilde{T} = \operatorname{avg}\left(\frac{1}{n}\sum x^{(j)} {x^{(j)}}^T\right)$



True covariance $T$

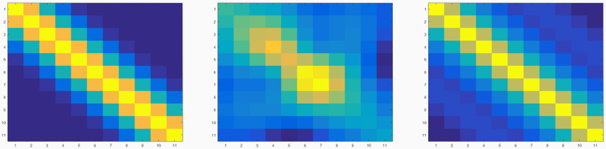


Empirical covariance $\hat{T}$



Improved estimator $avg(\hat{T})$

Estimator: $\tilde{T} = \mathrm{avg}\left(\frac{1}{n}\sum x^{(j)}x^{(j)^T}\right)$

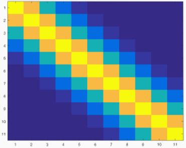


True covariance $T$ Empirical covariance $\hat{T}$ Improved estimator $avg(\hat{T})$

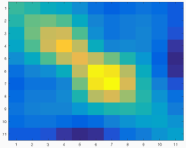
- Vector sample complexity: $O(\log^2 d/\epsilon^2)$

# A FIRST RESULT

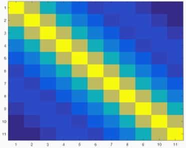
Estimator: $\tilde{T} = \mathrm{avg}\left(\frac{1}{n}\sum x^{(j)}x^{(j)^T}\right)$



True covariance $T$



Empirical covariance $\hat{T}$

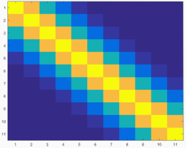


Improved estimator $avg(\hat{T})$

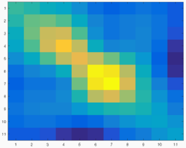
- **Vector sample complexity**: $O(\log^2 d/\epsilon^2)$
- **Entry sample complexity**: $d$.
- **Total sample complexity**: $O(d \log^2 d/\epsilon^2)$.

# A FIRST RESULT

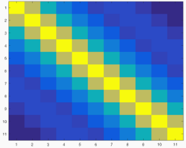
**Estimator:** $\tilde{T} = \mathrm{avg}\left(\frac{1}{n}\sum x^{(j)} x^{(j)^T}\right)$



True covariance $T$



Empirical covariance $\hat{T}$

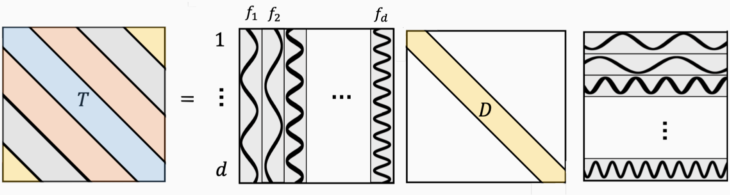


Improved estimator $avg(\hat{T})$

- **Vector sample complexity**: $O(\log^2 d/\epsilon^2)$
- **Entry sample complexity**: $d$.
- **Total sample complexity**: $O(d \log^2 d/\epsilon^2)$.

Improves over $O(d^2/\epsilon^2)$ for generic covariance matrices.

**Vandermonde decomposition:** Any Toeplitz $T$ can be written as $F_S D F_S$ where $F_S$ is an 'off-grid' Fourier matrix with frequencies $f_1, \ldots, f_d \in [0,1]$ and $D$ is a positive diagonal matrix.



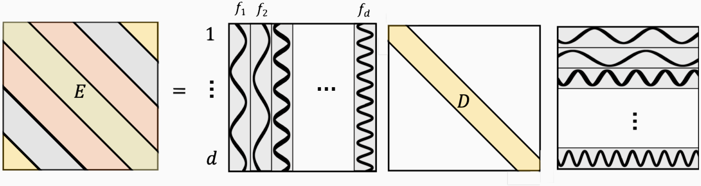

$$F_S(j,k) = \exp\left(-2\pi\sqrt{-1} \cdot j \cdot f_k\right)$$

# VERY ROUGH PROOF IDEA

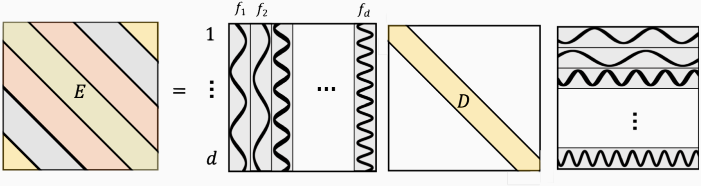
Let $\hat{T} = \frac{1}{n}\sum x^{(j)} {x^{(j)}}^{T}$. $\quad \tilde{T} = \operatorname{avg}\left(\hat{T}\right)$. $\quad E = T - \tilde{T}$.

## VERY ROUGH PROOF IDEA

Let $\hat{T} = \frac{1}{n} \sum x^{(j)} x^{(j)^T}$. $\qquad \tilde{T} = \mathrm{avg}\left(\hat{T}\right)$. $\qquad E = T - \tilde{T}$.

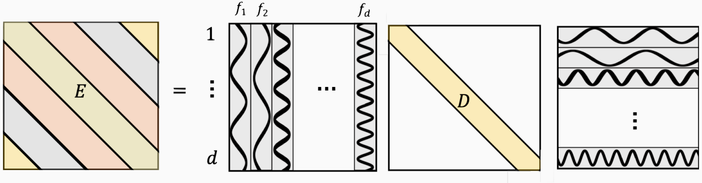


Let $\hat{T} = \frac{1}{n}\sum x^{(j)}x^{(j)^T}$. $\tilde{T} = \mathrm{avg}\left(\hat{T}\right)$. $E = T - \tilde{T}$.




- Roughly, to bound $\|E\|_2 = \max_{\|z\|_2=1} |z^T E z|$, it suffices to bound $|f_j^T E f_j|$. Obvious if $f_1, \ldots, f_d$ where eigenvectors of $E$ (they aren't quite).

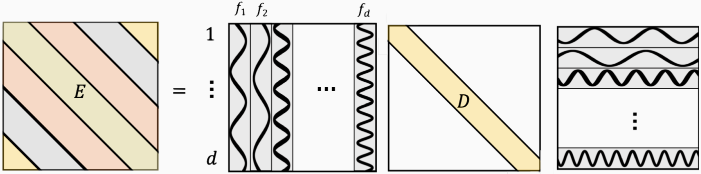
Let $\hat{T} = \frac{1}{n}\sum x^{(j)}{x^{(j)}}^T$. $\quad \tilde{T} = \mathrm{avg}\left(\hat{T}\right)$. $\quad E = T - \tilde{T}$.




- Roughly, to bound $\|E\|_2 = \max_{\|z\|_2=1} |z^T E z|$, it suffices to bound $|f_j^T E f_j|$. Obvious if $f_1, \ldots, f_d$ where eigenvectors of $E$ (they aren't quite).
- Argue that $|f_j^T(T - \tilde{T})f_j| = |f_j^T(T - \hat{T})f_j| \leq \epsilon\|T\|_2$ for all $j$ using standard matrix concentration (Hanson-Wright inequality) + $\epsilon$-net over frequencies in $[0, 1]$ + union bound.

# VERY ROUGH PROOF IDEA

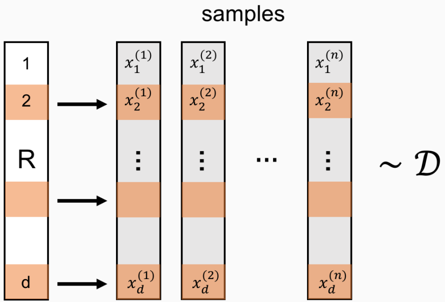
Let $\hat{T} = \frac{1}{n}\sum x^{(j)}x^{(j)^T}$. $\tilde{T} = \text{avg}\left(\hat{T}\right)$. $E = T - \tilde{T}$.




- Roughly, to bound $\|E\|_2 = \max_{\|z\|_2=1} |z^T E z|$, it suffices to bound $|f_j^T E f_j|$. Obvious if $f_1, \ldots, f_d$ where eigenvectors of $E$ (they aren't quite).
- Argue that $|f_j^T(T - \tilde{T})f_j| = |f_j^T(T - \hat{T})f_j| \leq \epsilon\|T\|_2$ for all $j$ using standard matrix concentration (Hanson-Wright inequality) + $\epsilon$-net over frequencies in $[0, 1]$ + union bound.

**Question:** Can $O(\log^2 d)$ samples be improved to $O(\log d)$?

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

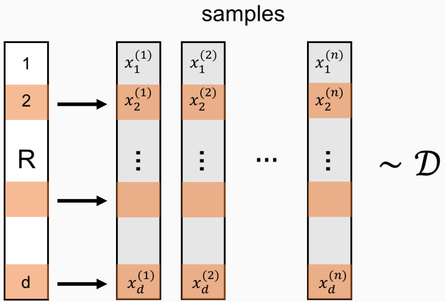



Entry sample complexity: $|R|$. Total sample complexity: $|R| \cdot n$.

# IMPROVING ENTRY SAMPLE COMPLEXITY

Consider algorithms that sample $x^{(1)}, \ldots, x^{(n)} \sim \mathcal{D}$ and read a fixed subset of entries $R \subseteq [d]$ from each $x^{(j)}$.

Approximate $T$ using $x_R^{(1)}, \ldots, x_R^{(n)} \in \mathbb{R}^{|R|}$.




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

Only get information about $\mathrm{cov}(x_j, x_k)$ for subset of pairs $j, k$.

How small can $R$ be if $T$ is Toeplitz?

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

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

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

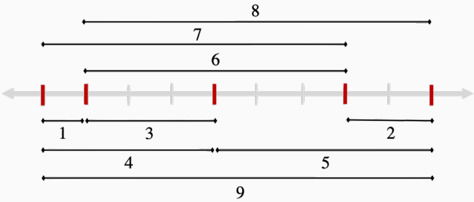
$$T = \begin{bmatrix} a_0 & a_1 & a_2 & \cdots & a_{d-2} & a_{d-1} \\ a_1 & a_0 & a_1 & \cdots & \cdots & a_{d-2} \\ a_2 & a_1 & a_0 & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{d-2} & \cdots & \cdots & \cdots & \cdots & a_1 \\ a_{d-1} & a_{d-2} & \cdots & \cdots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

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

**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.



# SPARSE RULER BASED ESTIMATION

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

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

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

- 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 $x_R^{(j)}$.

# SPARSE RULER BASED ESTIMATION

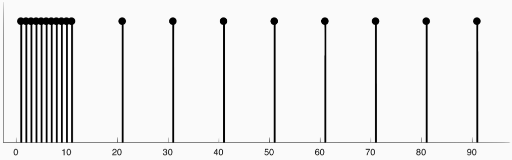
$$T = \begin{bmatrix} a_0 & a_1 & a_2 & \cdots & a_{d-2} & a_{d-1} \\ a_1 & a_0 & a_1 & \cdots & \cdots & a_{d-2} \\ a_2 & a_1 & a_0 & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{d-2} & \cdots & \cdots & \cdots & \cdots & a_1 \\ a_{d-1} & a_{d-2} & \cdots & \cdots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

- 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 from $\mathcal{D}$, can estimate each $a_s$ to high accuracy, and thus get an estimate for $T$.

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

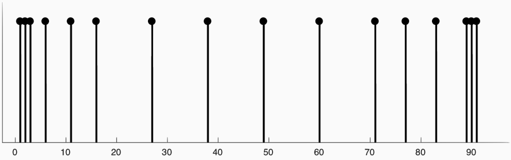
**Claim:** For any $d$ there exists a sparse ruler $R$ with $|R| = 2\sqrt{d}$

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



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

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




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

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?

We prove:

- Upper bound: $\tilde{O}(d)$ vector samples.
- Lower bound: $O(d)$ vector samples.

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

We prove:

- Upper bound: $\tilde{O}(d)$ vector samples.
- Lower bound: $O(d)$ vector samples.

Recall that $O(\log^2 d)$ samples were possible when reading all entries of each sample.

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

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 $|\varepsilon_s| \leq \varepsilon$.

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

## 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, $\|\tilde{T} - T\|_2 = O(\varepsilon d)$.

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, $\|\tilde{T} - T\|_2 = O(\varepsilon d)$.
- Setting $\varepsilon' = \varepsilon / d$, $n = \tilde{O}\left(\frac{d^2}{\varepsilon^2}\right)$ would give

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

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 $|\varepsilon_s| \leq \varepsilon$.

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

- If $\varepsilon_s$ were independent, $\|\tilde{T} - T\|_2 \leq \varepsilon\sqrt{d}$ [Meckes '07].

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 $|\varepsilon_s| \leq \varepsilon$.

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

- If $\varepsilon_s$ 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

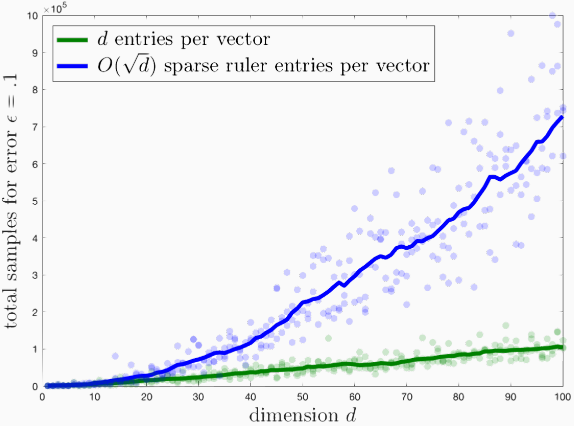
$$\|\tilde{T} - T\|_2 \leq \varepsilon \leq \epsilon\|\tilde{T} - T\|_2.$$

**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 $n = \tilde{O}\left(\frac{d}{\varepsilon^2}\right)$.

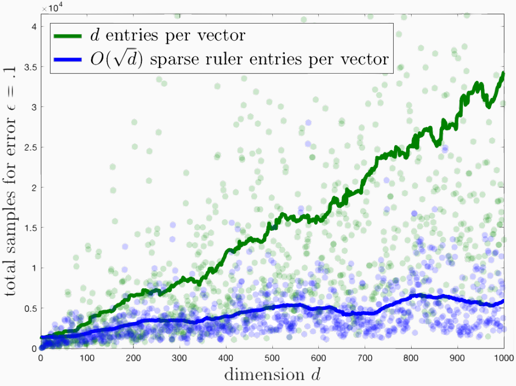
**Theorem.** For any ruler $R \subset [d]$, covariance estimation with $R$ gives $\|\tilde{T} - T\|_2 \leq \varepsilon \|T\|_2$ with entry sample complexity $|R|$ and vector sample complexity $n = \tilde{O}\left(\frac{d}{\varepsilon^2}\right)$.

- Vector sample complexity matches unstructured covariance estimation, but entry sample complexity is $\underline{O(\sqrt{d})\text{ instead of }d.}$

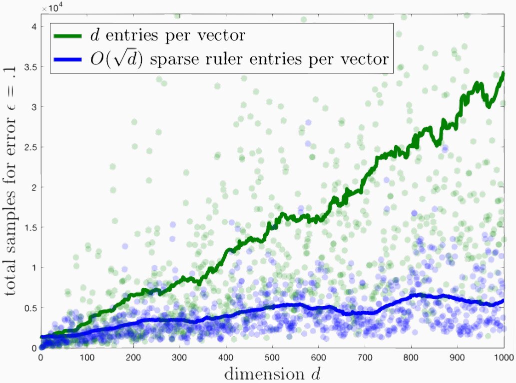
Total sample complexity is $O(\sqrt{d}) \cdot \tilde{O}(d) = \tilde{O}(d^{3/2})$ for sparse ruler vs. $d \cdot \tilde{O}(1) = \tilde{O}(d)$ for full sample estimation.



# NOT WHATS OBSERVED IN PRACTICE...

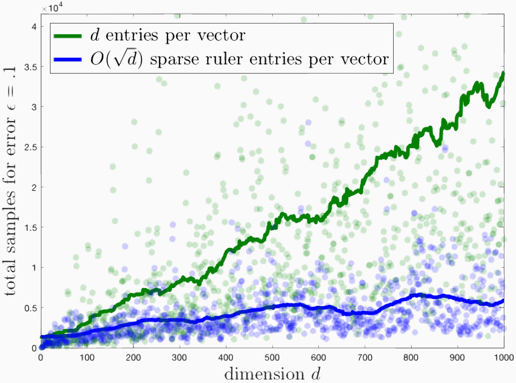


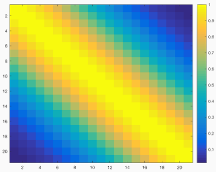
# NOT WHATS OBSERVED IN PRACTICE...

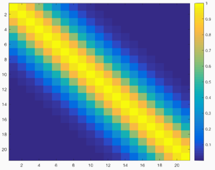



- Total sample complexity appears to be $\tilde{O}(\sqrt{d})$ for sparse rulers vs. $\tilde{O}(d)$ for full samples.

Sparse rulers give much better total sample complexity when $T$ is (approximately) low-rank.





- Total sample complexity appears to be $\tilde{O}(\sqrt{d})$ for sparse rulers vs. $\tilde{O}(d)$ for full samples.

How many vector samples do we need when $T$ is (approximately) rank $k$ and samples are collected with a $O(\sqrt{d})$-sparse ruler?

How many vector samples do we need when $T$ is (approximately) rank $k$ and samples are collected with a $O(\sqrt{d})$-sparse ruler?

We prove:

- Upper bound: $\tilde{O}\left(k^2\right)$ vector samples.
- Lower bound: $O\left(k\right)$ vector samples.

**How many vector samples do we need when $T$ is (approximately) rank $k$ and samples are collected with a $O(\sqrt{d})$-sparse ruler?**

We prove:

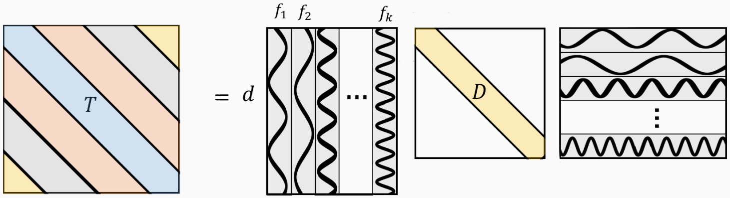
- Upper bound: $\tilde{O}\left(k^2\right)$ vector samples.
- Lower bound: $O\left(k\right)$ vector samples.

**Take-away:** Sublinear total sample complexity $\tilde{O}(k^2\sqrt{d})$ is possible when $T$ is low-rank.

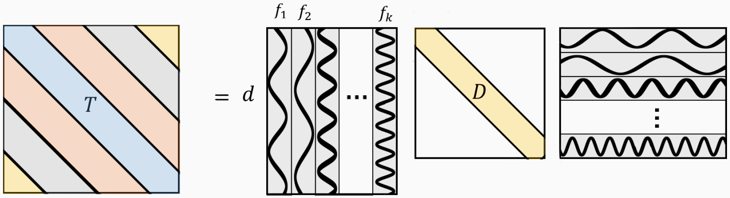
**Question:** Can we reduce the dependence on $d$ even more?

**Remainder of the talk:** Sketch an entirely different approach to low-rank Toeplitz covariance estimation using sparse Fourier transform methods.

**Low-rank Vandermonde decomposition:** Any rank-$k$ Toeplitz $T$ 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 $k \times k$ positive diagonal matrix.



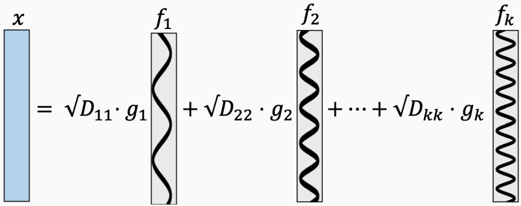
**Low-rank Vandermonde decomposition:** Any rank-$k$ Toeplitz $T$ can be written as $F_S D F_S$ where $F_S \in \mathbb{R}^{d \times k}$ is an 'off-grid' Fourier transform matrix with frequencies $f_1, \ldots, f_k$ and $D$ is a $k \times k$ positive diagonal matrix.



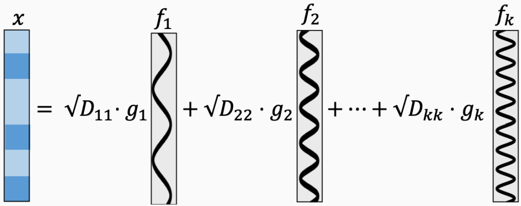

- Any sample $x \sim \mathcal{N}(0, T)$ can be written as $T^{1/2} g = F_S D^{1/2} g$ for $g \sim \mathcal{N}(0, I)$.

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

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

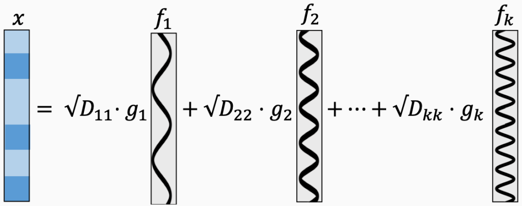


$x \sim \mathcal{N}(0, T) = F_S D^{1/2} g$ is a Fourier sparse function.




- 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 $n = O(\log^2 d/\varepsilon^2)$ samples, recover each in full by reading $2k$ entries, and then apply our earlier result 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 are $k$ frequencies that approximately span 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 are $k$ frequencies that approximately span 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 are $k$ frequencies that approximately span each $x^{(j)} \sim \mathcal{N}(0, T)$.

- Not as easy as it sounds.

**Step 2:** Use a robust sparse Fourier transform method to recover $x^{(1)}, \ldots, x^{(n)}$ and 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 are $k$ frequencies that approximately span each $x^{(j)} \sim \mathcal{N}(0, T)$.

- Not as easy as it sounds.

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

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

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

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

- Use several tools from Randomized Numerical Linear Algebra: Specifically a column subset selection result (see e.g., Guruswami, Sinop '12) + a projection-cost preservation bound (Cohen, Elder, Musco, Musco, Persu, '15).

**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 $X$?

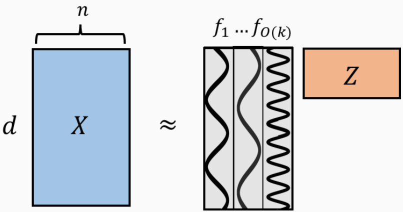
# APPROXIMATE FREQUENCY REGRESSION

Want to find $Z$ satisfying the approximate regression guarantee:

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

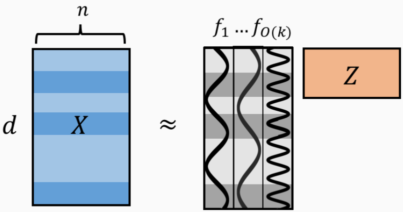
Want to find $Z$ satisfying the approximate regression guarantee:

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



Want to find $Z$ satisfying the approximate regression guarantee:

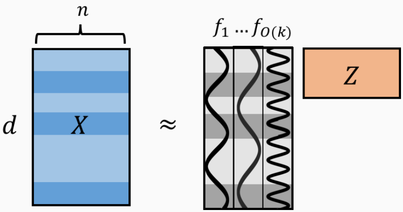
$$\|X - F_M Z\|_F^2 = O(1) \cdot min_Y \|X - F_M Y\|_F^2.$$




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

Want to find $Z$ satisfying the approximate regression guarantee:

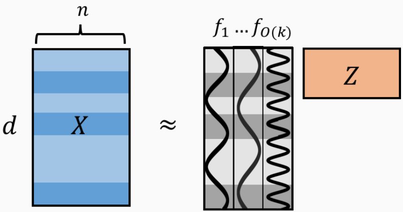
$$\|X - F_M Z\|_F^2 = O(1) \cdot min_Y \|X - F_M Y\|_F^2.$$




- 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 $k/n$

Want to find $Z$ satisfying the approximate regression guarantee:

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



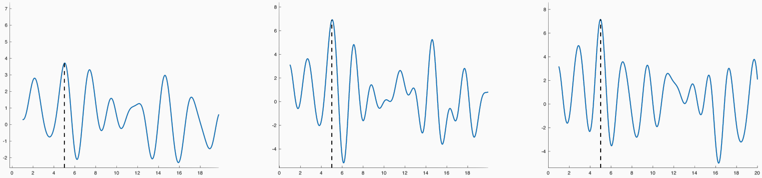

- Suffices to sample $\tilde{O}(k)$ rows by the leverage scores of $F_M$ and solve the regression problem just considering these rows.
- **Remark:** If $f_1, \dots, f_m$ are 'on-grid' integers, the columns of $F_M$ are orthonormal and the leverage scores are all $k/n \rightarrow$ RIP for subsampled Fourier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

Leverage scores measure how 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 how 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.)

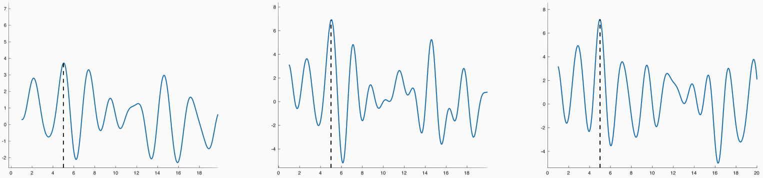
$$\tau_i(F_M) = \max_y \frac{(F_M y)_i^2}{\|F_M y\|_2^2}.$$



# FOURIER LEVERAGE SCORES

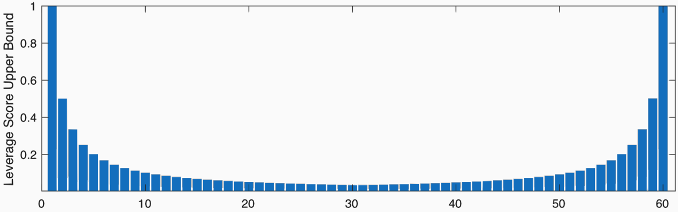
Leverage scores measure how large a function in the column span of $F_M$ can be at index $i$ (i.e., how important that index may be in the regression.)

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

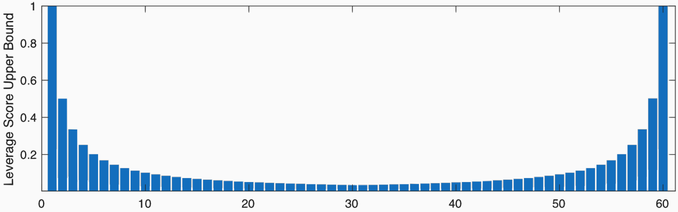


- Using that $F_M y$ is a Fourier sparse function we can bound this quantity a priori, without any dependence on $F_M$.

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

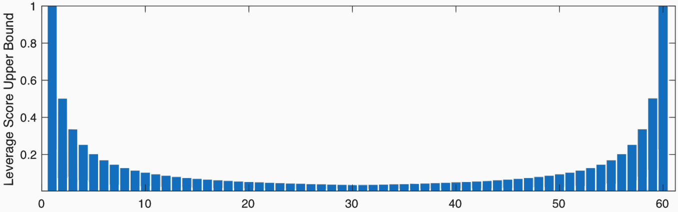


Extend bounds of [Chen Kane Price Song '16] to give explicit function upper bounding the leverage scores of any $F_M$:




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.

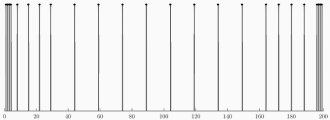
Extend bounds of [Chen Kane Price Song '16] to give explicit function upper bounding the leverage scores of any $F_M$:



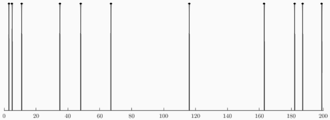

Since this distribution is universal, can sample one set of entries by these leverages scores, and find $X \approx F_M \cdot Z$ with high probability for any set of frequencies $f_1, \ldots, f_m$ in net.

**Note the resemblance to the distribution of marks in an optimal sparse ruler!**

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

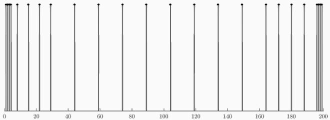


Deterministic sparse ruler pattern.

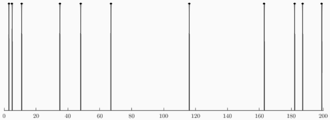


Randomly generated pattern.

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



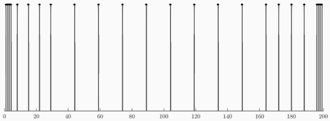
Deterministic sparse ruler pattern.



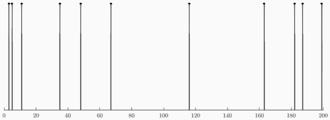
Randomly generated pattern.

2. Solve an exponential number of regression problems to recover $\tilde{X} \approx X$.

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



Deterministic sparse ruler pattern.

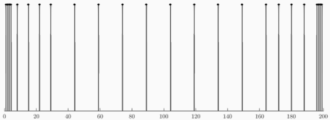


Randomly generated pattern.

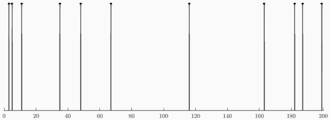
2. Solve an exponential number of regression problems to recover $\tilde{X} \approx X$.
3. Return $\tilde{T} = avg(\tilde{X}\tilde{X}^T)$.

# FINAL ALGORITHM

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



Deterministic sparse ruler pattern.



Randomly generated pattern.

2. Solve an exponential number of regression problems to recover $\tilde{X} \approx X$.
3. Return $\tilde{T} = avg(\tilde{X}\tilde{X}^T)$.

**Vector, entry, total sample complexity:** $O(\text{poly}(k \log d/\epsilon))$.

**Bound:** $\|T - \tilde{T}\|_2 \leq \varepsilon \|T\|_2 + f(T - T_k)$

# OPEN QUESTIONS AND FUTURE DIRECTIONS

Concrete.

Concrete.

- Runtime efficiency.

**Concrete.**

- Runtime efficiency.
  - Can hopefully avoid exponential time net approach using off-grid sparse FFT 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.

**Concrete.**

- Runtime efficiency.
  - Can hopefully avoid exponential time net approach using off-grid sparse FFT 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.
- Improve sample complexity.

Concrete.

- Runtime efficiency.
  - Can hopefully avoid exponential time net approach using off-grid sparse FFT 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.
- Improve sample complexity.
  - We give entry sample complexity of $\tilde{O}(k^2)$ but likely can be improved. **Partial results towards $\tilde{O}(\sqrt{k})$ complexity.**

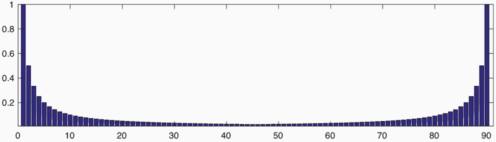
# FUTURE DIRECTIONS

"Low-Rank Toeplitz Matrix Estimation via Random Ultra-Sparse Rulers." Builds on work by [Qiao, Pal, 2017].

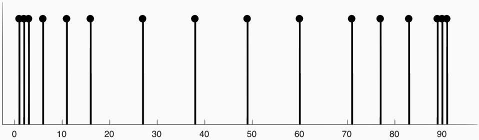
Hannah Lawrence, Jerry Li, Cameron Musco, Christopher Musco.

May 4 - 8th. Registration is now free! Great plenary speakers.

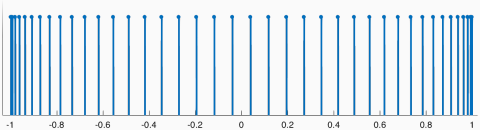
# CONNECTIONS BETWEEN SAMPLING SCHEMES




Fourier Sparse Leverage Scores




Optimal Sparse Ruler for d=91




Degree 40 Chebyshev Nodes

THANKS! QUESTIONS?