THE UNIVERSITY OF RHODE ISLAND

Computing the Spectral Density

Kyle Monette

Department of Mathematics and Applied Mathematical Sciences
University of Rhode Island

Applied Mathematics and Scientific Computing Seminar

March 24, 2025

Introduction

Let $A \in \mathbb{R}^{n \times n}$ be a (large!) **symmetric** matrix and $[a, b] \subset \mathbb{R}$.

Let $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ be the eigenvalues of A.

Question: What is the probability that an eigenvalue of A is in $[a, b]$?

Question: How many eigenvalues of A are in $[a, b]$?

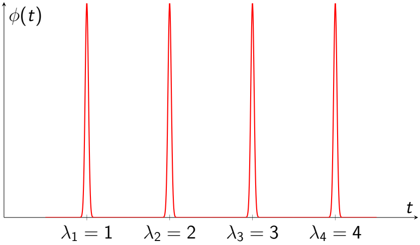
$\longrightarrow$ How many $\underbrace{\textit{singular values}}_{\text{square roots of } \lambda(A^T A)}$ are above a threshold, i.e. in $[a, \sigma_n]$?

Spectral Density

Definition 1 (Spectral Density)

For $A \in \mathbb{R}^{n\times n}$ with eigenvalues $\lambda_1, \ldots, \lambda_n$, the *spectral density* of A is

$$\phi(t) := \frac{1}{n} \cdot \sum_{j=1}^{n} \delta(t - \lambda_j), \qquad \text{where} \qquad \delta(t) := \begin{cases} \infty & t = 0 \\ 0 & t \neq 0 \end{cases}.$$



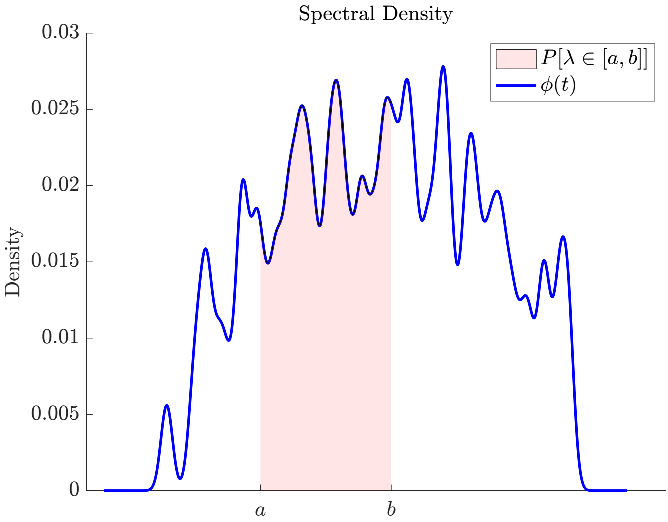
$$\boxed{\mathbb{P}\Big[\lambda \in [a, b]\Big] = \int_a^b \phi(t)\, dt}$$

$$\mathbb{P}\Big[\lambda \in [0, 1.5]\Big] = 1/4$$

$$\mathbb{P}\Big[\lambda \in [0, 3.2]\Big] = 3/4$$

$$\mathbb{P}\Big[\lambda \in [0, 5]\Big] = 1$$

An Illustration



Constructing $\phi(t)$

To approximate $\phi(t) = \frac{1}{n}\sum_{j=1}^{n} \delta(t - \lambda_j)$, we need two components:

1. The eigenvalues of A.

2. A suitable approximation for $\delta(t)$.

Lanczos

Given starting unit vector $v_1 \in \mathbb{R}^n$ and $m \ll n$, $\texttt{Lanczos}(m)$ generates

$$AV_m = V_m T_m + f e_m^T \quad \Leftrightarrow \quad V_m^T A V_m = T_m$$

$$T_m = \begin{bmatrix} \alpha_1 & \beta_1 & & \\ \beta_1 & \alpha_2 & \ddots & \\ & \ddots & \ddots & \beta_{m-1} \\ & & \beta_{m-1} & \alpha_m \end{bmatrix}$$

- $V_m \in \mathbb{R}^{n \times m}$ has orthonormal columns
- V_m is a basis for Krylov subspace span$\{v_1, Av_1, \ldots, A^{m-1}v_1\}$
- $v_i = p_{i-1}(A)v_1 \quad 1 \le i \le m$

Algorithm $\texttt{Lanczos}(A,\ v_1,\ m)$

```
for j = 1 : m do
    f = Av_j    if j > 1 :   f = f − β_{j−1} v_{j−1}
    α_j = f^T v_j
    f = f − α_j v_j
    If j < m :   β_j = ‖f‖,   v_{j+1} = f / β_{j+1}
end for
```

Orthogonalize Av_j against v_{j-1} and v_j (lines 2–4)

Ritz Pairs

$$AV_m = V_m T_m + f e_m^T$$

Suppose (θ, y) is an eigenpair of T_m. Then

$$\begin{aligned} AV_m y &= V_m T_m y + f e_m^T y \\ AV_m y &= \theta V_m y + f e_m^T y \end{aligned}$$

So $(\theta, V_m y)$ is an approximate eigenpair of A, called a *Ritz pair*.

$$A_{n\times n} \xrightarrow{\text{Lanczos}} T_m \xrightarrow{\text{eigenpairs of } T_m} \text{Approx. eigenpairs of } A$$

But ... There are only m Ritz pairs. We need ALL n eigenpairs for ϕ!!

Constructing $\phi(t)$

To approximate $\phi(t) = \frac{1}{n}\sum_{j=1}^{n} \delta(t - \lambda_j)$, we need two components:

1. The eigenvalues of A.

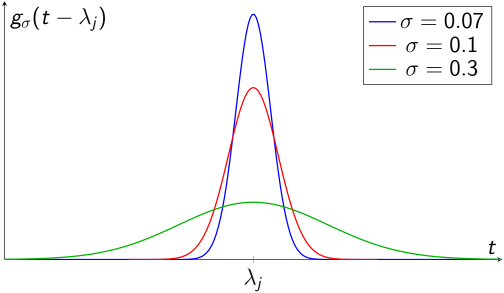
   Lanczos algorithm!

2. A suitable approximation for $\delta(t)$.

Approximating $\delta(t)$

Replace δ in $\phi(t) = \frac{1}{n}\sum_{j=1}^{n} \delta(t-\lambda_j)$ with $g_\sigma(t) := \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-t^2}{2\sigma^2}\right)$

$$\phi_\sigma(t) \quad := \quad \frac{1}{n}\sum_{j=1}^{n} g_\sigma(t-\lambda_j) \qquad \text{("Regularized Density")}$$



Large σ:
Smooth density
Lower resolution

Small σ:
Jagged density
Higher resolution

How do we construct $\phi(t)$?

Our goal: To approximate $\phi(t)$.

1. How can we approximate the eigenvalues of A?

   Lanczos algorithm!

2. How can we approximate $\delta(t)$?

   $g_\sigma(t)$

Now: Use a $\underbrace{\text{Monte–Carlo simulation}}_{\text{Take random samples, find the average}}$ to approximate $\phi_\sigma(t)$.

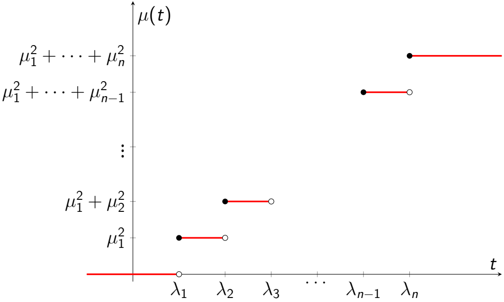
Approximating $\phi_\sigma(t)$

$$g_\sigma(t) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-t^2}{2\sigma^2}\right)$$

If $A = Q\Lambda Q^T$ and v_1 starting Lanczos vector

$$v_1^T g_\sigma(A) v_1 = v_1^T Q\, g_\sigma(\Lambda) Q^T v_1 = \sum_{i=1}^{n} g_\sigma(\lambda_i)\mu_i^2 \qquad \mu_i = [Q^T v_1]_i$$

$$\mu(t) = \begin{cases} 0 & t < \lambda_1 = a \\ \sum_{j=1}^{i-1} \mu_j^2 & \lambda_{i-1} \le t < \lambda_i,\ 2 \le i \le n \\ \sum_{j=1}^{n} \mu_j^2 & t \ge \lambda_n = b \end{cases}$$



$$v_1^T g_\sigma(A) v_1 = \int_a^b g_\sigma(t)\, d\mu(t)$$

(c.f. [1])

Approximating $\int_a^b g_\sigma(t)\, d\mu(t)$

Goal: Approximate $\displaystyle\int_a^b g_\sigma(t)\, d\mu(t)$.

Recall: Each vector v_i, $1 \leq i \leq m$, from Lanczos is given by $v_i = p_{i-1}(A)v_1$.

These polynomials $p_0, p_1, \ldots, p_{m-1}$ are orthogonal w.r.t. $\mu(t)$ [2] via

$$\langle p_k, p_\ell \rangle \quad := \quad \int_a^b p_k(t)p_\ell(t)\, d\mu(t), \qquad a \leq \lambda_1,\ b \geq \lambda_n.$$

The eigenvalues of T_m are the roots of p_m [3].

Result: Eigenpairs (θ_j, y_j) of T_m yield nodes θ_j and weights $w_j = y_{1j}^2$ for a Gaussian Quadrature rule! [1]

$$\int_a^b g_\sigma(t)\, d\mu(t) \quad = \quad v_1^T g_\sigma(A) v_1 \quad \approx \quad \sum_{j=1}^m g_\sigma(\theta_j) w_j$$

Sampling $v^T f(A) v$

Theorem 2 ([4)

] Suppose each component of $v \in \mathbb{R}^n$ is drawn independently from a distribution with mean 0 and variance 1; that is, $\mathbb{E}[v] = 0$ and $\mathbb{E}[vv^T] = I_n$.

Then for any symmetric matrix A and matrix function f,

$$\mathbb{E}\left[v^T f(A) v\right] \quad = \quad \operatorname{trace} f(A).$$

Consider $g_\sigma(tI - A)$ as a matrix function.

Draw $v \sim \mathrm{unif}\{-1, 1\}^n$ (Rademacher vector). (least variance estimator [5])

We'll let $v_1 = \dfrac{v}{\sqrt{n}}$ and start Lanczos with v_1.

Approximation to $\phi_\sigma(t)$

Recall: If $A = Q\Lambda Q^T$, then $f(A) = Qf(\Lambda)Q^T$ so $f(A)$ & $f(\Lambda)$ are similar.

$$
\begin{aligned}
\sum_{j=1}^{n} g_\sigma(t-\lambda_j) &= \operatorname{trace} g_\sigma(tI - A) && \text{(Similarity)} \\
&= \mathbb{E}\left[v^T g_\sigma(tI - A)v\right] && \text{(Theorem 2)} \\
&= n \cdot \mathbb{E}\left[v_1^T g_\sigma(tI - A)v_1\right] \\
&\approx n \cdot \mathbb{E}\left[\sum_{j=1}^{m} g_\sigma(t-\theta_j)w_j\right] \\
&= \frac{n}{n_v}\sum_{i=1}^{n_v}\left[\sum_{j=1}^{m} g_\sigma(t-\theta_j)w_j\right]
\end{aligned}
$$

$$
\widetilde{\phi}_\sigma(t) = \frac{1}{n}\sum_{j=1}^{n} g_\sigma(t-\lambda_j) \approx \frac{1}{n_v}\sum_{i=1}^{n_v}\left[\sum_{j=1}^{m} g_\sigma(t-\theta_j)w_j\right].
$$

Spectral Density Algorithm

Algorithm Spectral Density

Draw n_v vectors $v^{(i)} \sim \mathrm{unif}\{-1, 1\}^n, \quad 1 \leq i \leq n_v$.

for $i = 1$ to n_v **do**

  Call `Lanczos`(m), starting with $v_1^{(i)} = v^{(i)}/\sqrt{n}$.

  Compute eigenpairs $(\theta_j^{(i)}, y_j^{(i)})$ of $T_m^{(i)}$.

  Compute weights $w_j^{(i)}$ from the eigenvectors $y_j^{(i)}$.

  Let $\widetilde{\phi}_\sigma^{(i)}(t) = \sum_{j=1}^{m} w_j^{(i)} g_\sigma(t - \theta_j^{(i)})$

end for

$\widetilde{\phi}_\sigma(t) = \frac{1}{n_v} \sum_{i=1}^{n_v} \widetilde{\phi}_\sigma^{(i)}(t)$.

Example 1

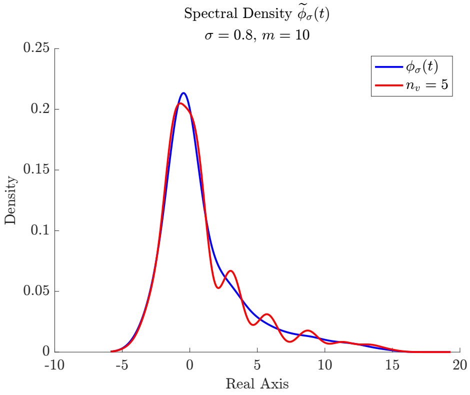
We consider the can_1054 matrix, available from SuiteSparse (online collection).

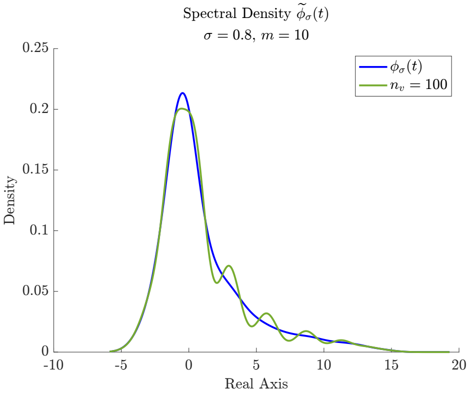
"*A finite-element structure problem in aircraft design*."

Symmetric matrix, size $n = 1054$, $\lambda_1 = -4.51$, $\lambda_n = 14.85$.

Example 1: Varying n_v

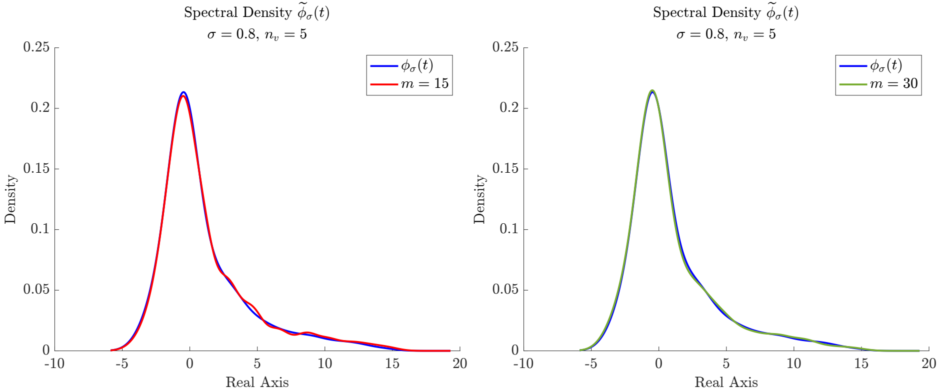
Using can_1054 matrix with $m = 10$ and $n_v = 5, 100$.





Example 1: Varying m

Consider the can_1054 matrix from before, with $n_v = 5$ and $m = 15, 30$.

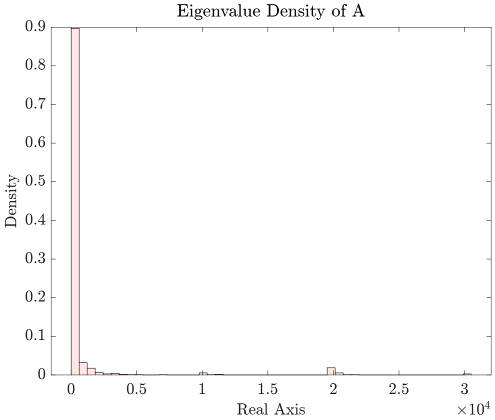


Example 2

We consider the 1138_bus matrix, available from SuiteSparse.

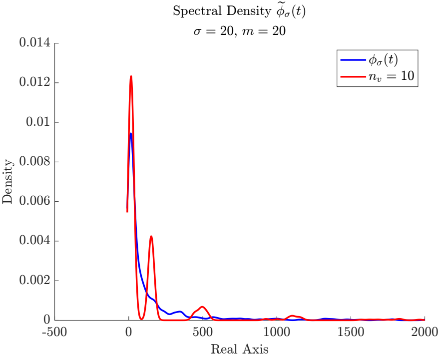
"*Power systems network graph*."

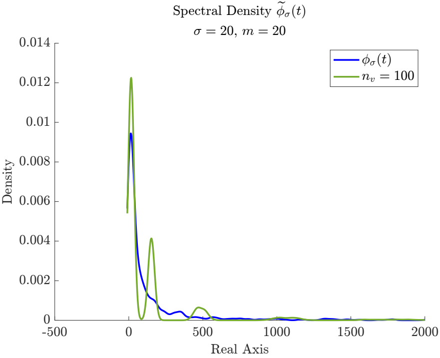
Symmetric matrix, size $n = 1138$, $\lambda_1 = 0.0035$, $\lambda_n = 3.015 \times 10^4$



Example 2: Varying n_v

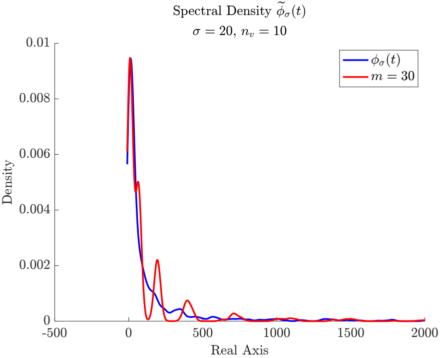
Using 1138_bus matrix with $m = 20$ and $n_v = 10, 100$.

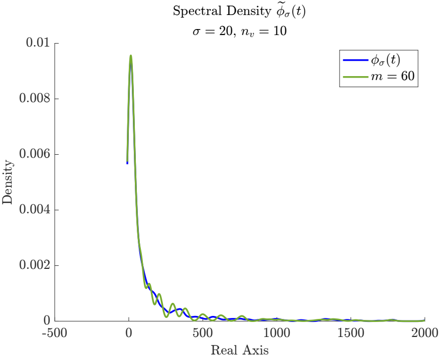




Example 2: Varying m

Using 1138_bus matrix with $n_v = 10$ and $m = 30, 60$.





Quick Note on Errors

In 2017, the authors in [6] proposed an (L_1) error measurement of

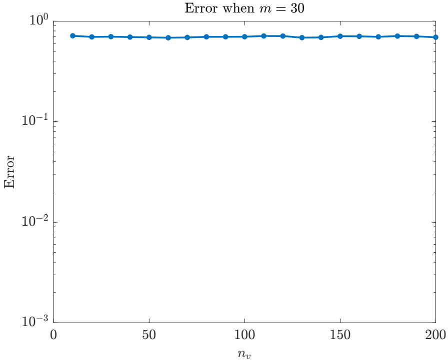
$$\frac{\sum_i \left| \widetilde{\phi}_\sigma(t_i) - \phi_\sigma(t_i) \right|}{\sum_i |\phi_\sigma(t_i)|}$$

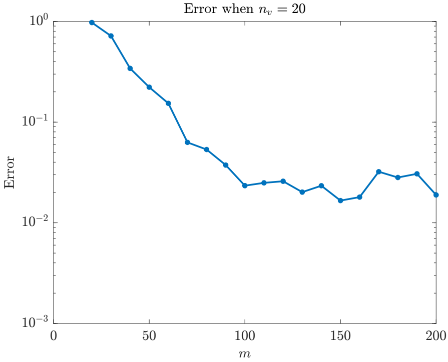
where $\{t_i\}$ are uniformly distributed points.

Error Estimates

Consider the 1138_bus matrix from Example 2.

(Left) Error when $m = 30$, vary n_v. (Right) Error when $n_v = 20$, vary m.





Approximating the Spectral Count

Our approximation $\widetilde{\phi}_\sigma(t)$ for the spectral density gives us the number of eigenvalues in $[a, b]$, the *spectral count*:

$$\mu_{[a,b]} \quad := \quad n \int_a^b \phi(t)\, dt.$$

But do we need the density ϕ to get $\mu_{[a,b]}$? No!

Spectral Count: $\mu_{[a,b]}$

In 2016, the authors in [7] consider the projection matrix

$$P = \sum_{\lambda_i \in [a,b]} u_i u_i^T, \qquad (\lambda_i, u_i) \text{ eigenpairs of } A$$

and interpret P as a step function of A:

$$P = h(A) \quad \text{where} \quad h(t) = \begin{cases} 1 & t \in [a, b] \\ 0 & \text{else} \end{cases}.$$

Then, approximate $h(t)$ as a sum of Chebyshev polynomials T_j:

$$h(t) \quad \approx \quad \sum_{j=0}^{p} \gamma_j T_j(t)$$

Approximating $\mu_{[a,b]}$

$$h(A) \approx \sum_{j=0}^{p} \gamma_j T_j(A)$$

Using our method from before,

$$\begin{aligned}
\mu_{[a,b]} &= \operatorname{trace} h(A) = \mathbb{E}\left[v^T h(A) v\right] \\
&\approx \mathbb{E}\left[\sum_{j=0}^{p} \gamma_j v^T T_j(A) v\right] \\
&= \frac{n}{n_v} \sum_{k=1}^{n_v} \left[\sum_{j=0}^{p} \gamma_j v_k^T T_j(A) v_k\right]
\end{aligned}$$

(*Assuming* $[a, b]$ *and the spectrum of* A *are mapped into* $[-1, 1]$)

Notice: Lanczos is not needed!

An Application to Singular Values

Recall: The eigenvalues of $A^T A$ are the squares of the singular values of A.

Our idea: Apply this spectral count method to $A^T A$ for singular value thresholding, i.e., compute

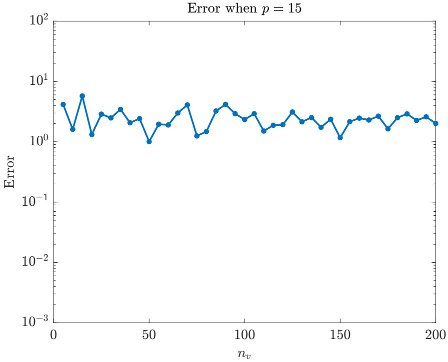
$$\begin{aligned} \eta_a &= \#\ \text{singular values of } A \text{ in } [a, \sigma_n] \\ &\approx \frac{n}{n_v} \sum_{k=1}^{n_v} \left[\sum_{j=0}^{p} \gamma_j v_k^T T_j(A^T A) v_k \right] \end{aligned}$$

Example

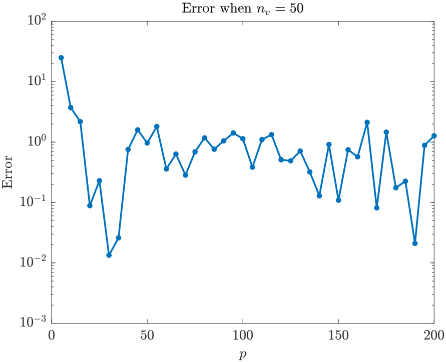
Consider the can_1054 matrix from before.

We estimate the number of singular values in the top 20%: $\eta_{0.8\sigma_n} = 15$

(Left) Error when $p = 15$, vary n_v.



(Right) Error when $n_v = 50$, vary p.



References

[1] S. Ubaru, J. Chen, and Y. Saad, “Fast estimation of tr(f(a)) via stochastic Lanczos quadrature,” *SIAM Journal on Matrix Analysis and Applications*, vol. 38, no. 4, pp. 1075-1099, 2017.

[2] G.H. Golub and G. Meurant, *Matrices, Moments and Quadrature with Applications*. Princeton University Press, 2009.

[3] B.N. Parlett, *The Symmetric Eigenvalue Problem*. SIAM, 1998.

[4] L.Lin, Y.Saad, and C.Yang, “Approximating spectral densities of large matrices,” *SIAM Review*, vol. 58, no. 1, pp. 34-65, 2016.

[5] P.-G. Martinsson and J. A. Tropp, “Randomized numerical linear algebra: Foundations and algorithms,” *Acta Numerica*, vol. 29, pp. 403-572, 2020.

[6] L. Lin, “Randomized estimation of spectral densities of large matrices made accurate,” *Numerische Mathematik*, vol. 136, pp. 183-213, 2017.

[7] E. Di Napoli, E. Polizzi, and Y. Saad, “Efficient estimation of eigenvalue counts in an interval,” *Numerical Linear Algebra with Applications*, vol. 23, no. 4, pp. 674-692, 2016.

Questions?

Proof of Theorem 2.

$$
\begin{aligned}
\mathbb{E}\left[v^T f(A) v\right] &= \mathbb{E}\left[\operatorname{trace}(v^T f(A) v)\right] && (v^T f(A) v \in \mathbb{R}) \\
&= \mathbb{E}\left[\operatorname{trace}(f(A) v v^T)\right] && \text{(cyclic trace property)} \\
&= \operatorname{trace}\left(\mathbb{E}\left[f(A) v v^T\right]\right) && \text{(linearity of } \mathbb{E}\text{)} \\
&= \operatorname{trace}\left(f(A) \cdot \mathbb{E}\left[v v^T\right]\right) && (f(A) \text{ is deterministic}) \\
&= \operatorname{trace} f(A)
\end{aligned}
$$

□