UNIVERSITY
OF MINNESOTA TWIN CITIES

Applications of trace estimation techniques

Yousef Saad
Department of Computer Science and Engineering

University of Minnesota

PMAA-2016, Bordeaux (France)
Jul. 7th, 2016

Introduction

- ➤ Many calculations require estimating the trace of a certain matrix function $B = f(A)$.

- ➤ Related problem: compute diag$(f(A))$.

- ➤ Most methods rely on stochastic methods for this [do not exploit any structure]

- ➤ In this talk: A few specific applications and a few techniques

- ➤ Generally speaking: many new related applications to be discovered

- ➤ Begin with a few well-known examples

Introduction: A few examples

Problem 1: Compute Tr[inv[A]] the trace of the inverse.

➤ Arises in cross validation :

$$\frac{\|(I - A(\theta))g\|_2}{\text{Tr}\,(I - A(\theta))} \quad \text{with} \quad A(\theta) \equiv I - D(D^T D + \theta L L^T)^{-1} D^T,$$

D == blurring operator and L is the regularization operator

➤ In [Huntchinson '90] Tr[Inv[A]] is stochastically estimated

➤ Motivation for the work [Golub & Meurant, “Matrices, Moments, and Quadrature”, 1993, Book with same title in 2009]

Problem 2: Compute Tr [f (A)], f a certain function

Arises in many applications in Physics. Example:

- ➤ Stochastic estimations of Tr (f(A)) extensively used by quantum chemists to estimate Density of States, see

[Ref: H. Röder, R. N. Silver, D. A. Drabold, J. J. Dong, Phys. Rev. B. 55, 15382 (1997)]

- ➤ Will be covered in detail later in this talk.

Problem 3: Compute diag[inv(A)] the diagonal of the inverse

- Harder than just getting the trace
- Arises in Dynamic Mean Field Theory [DMFT, motivation for our work on this topic].
- Related approach: Non Equilibrium Green's Function (NEGF) approach used to model nanoscale transistors.
- In uncertainty quantification, the diagonal of the inverse of a covariance matrix is needed [Bekas, Curioni, Fedulova '09]

Problem 4: Compute diag[f (A)] ; f = a certain function.

➤ Arises in any density matrix approach in quantum modeling - for example Density Functional Theory.

➤ Here, f = Fermi-Dirac operator:

$$f(\epsilon) = \frac{1}{1 + \exp(\frac{\epsilon-\mu}{k_B T})}$$

Note: when $T \rightarrow 0$ then $f \rightarrow$ a step function.

Note: if f is approximated by a rational function then diag[f(A)] ≈ a linear combination of terms like diag[$(A - \sigma_i I)^{-1}$]

➤ Linear-Scaling methods based on approximating $f(H)$ and $\mathbf{Diag}(f(H))$ – avoid ‘diagonalization’ of H

- Rich litterature on ‘linear scaling’ or ’order n’ methods

- The review paper [Benzi, Boito, Razouk, “Decay properties of Specral Projectors with applications to electronic structure”, SIAM review, 2013] provides theoretical foundations

- Several references on approximating Diag$(f(H))$ for this purpose – See e.g., work by L. Lin, C. Yang, E. E [Code: SelInv]

- Also: analysis of network graphs

diag(inv(A)) in Dynamic Mean Field Theory (DMFT)

➤ Quantum mechanical studies of highly correlated particles

➤ Equation to be solved (repeatedly) is Dyson's equation

$$G(\omega) = [(\omega + \mu)I - V - \Sigma(\omega) + T]^{-1}$$

- ω (frequency) and μ (chemical potential) are real
- V = trap potential = real diagonal
- $\Sigma(\omega)$ == local self-energy - a complex diagonal
- T is the hopping matrix (sparse real).

➤ Interested only in diagonal of $G(\omega)$ – in addition, equation must be solved self-consistently and ...

➤ ... must do this for many ω's

DENSITY OF STATES & APPLICATIONS

Density of States

- Formally, the Density Of States (DOS) of a matrix A is

$$\phi(t) = \frac{1}{n}\sum_{j=1}^{n} \delta(t - \lambda_j),$$

where:
- δ is the Dirac δ-function or Dirac distribution
- $\lambda_1 \le \lambda_2 \le \cdots \le \lambda_n$ are the eigenvalues of A

- Note: number of eigenvalues in an interval $[a, b]$ is

$$\mu_{[a,b]} = \int_a^b \sum_j \delta(t - \lambda_j)\, dt \equiv \int_a^b n\phi(t)dt\ .$$

- $\phi(t)$ == a probability distribution function == probability of finding eigenvalues of A in a given infinitesimal interval near t.

- DOS is also referred to as the spectral density

- In Solid-State physics, λ_i's represent single-particle energy levels.

- So the DOS represents # of levels per unit energy.

- Many uses in physics

Issue: How to deal with distributions

- Highly ‘discontinuous’, not easy to handle numerically

- Solution for practical and theoretical purposes: replace ϕ by a regularized (‘blurred’) version ϕ_σ:

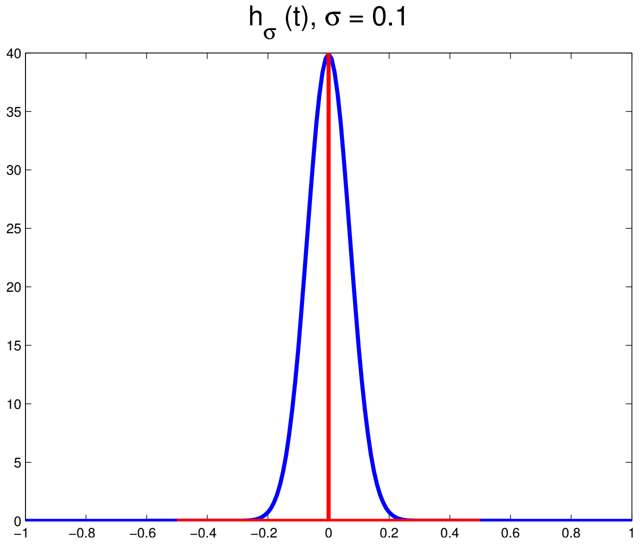
$$\boxed{\phi_\sigma(t) = \frac{1}{n}\sum_{j=1}^{n} h_\sigma(t-\lambda_j),}$$

where $h_\sigma(t)$ = any $\mathcal{C}^\infty$ function s.t.:

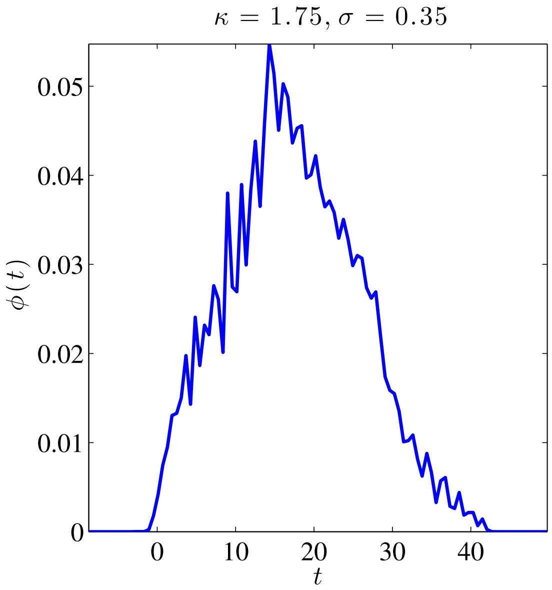
- $\int_{-\infty}^{+\infty} h_\sigma(s)ds = 1$
- h_σ has a peak at zero

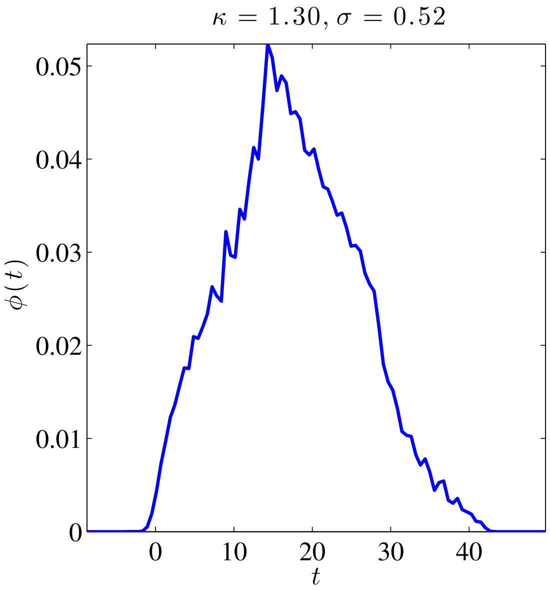
- An example is the Gaussian:

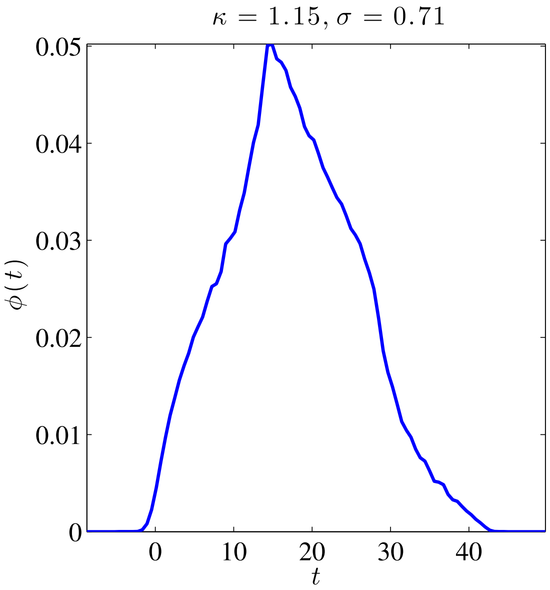
$$h_\sigma(t) = \frac{1}{(2\pi\sigma^2)^{1/2}} e^{-\frac{t^2}{2\sigma^2}}.$$

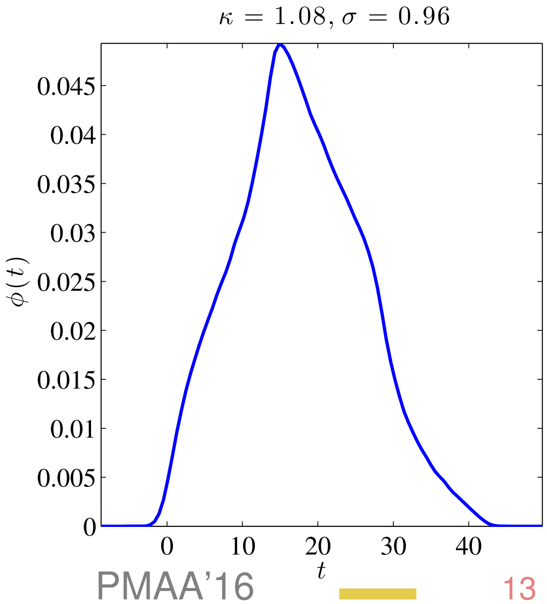


- How to select $\boldsymbol{\sigma}$? Example for $\boldsymbol{Si_2}$









- Higher $\boldsymbol{\sigma} \rightarrow$ smoother curve
- But loss of detail ..
- Compromise: $\boldsymbol{\sigma} = \dfrac{\boldsymbol{h}}{2\sqrt{2\log(\boldsymbol{\kappa})}}$,
- $\boldsymbol{h} =$ resolution, $\boldsymbol{\kappa} =$ parameter > 1

Computing the DOS: The Kernel Polynomial Method

- Used by Chemists to calculate the DOS – see Silver and Röder'94 , Wang '94, Drabold-Sankey'93, + others

- Basic idea: expand DOS into Chebyshev polynomials

- Use trace estimator [discovered independently] to get traces needed in calculations

- Assume change of variable done so eigenvalues lie in $[-1,\ 1]$.

- Include the weight function in the expansion so expand:

$$\hat{\phi}(t) = \sqrt{1-t^2}\phi(t) = \sqrt{1-t^2} \times \frac{1}{n}\sum_{j=1}^{n} \delta(t-\lambda_j).$$

Then, (full) expansion is: $\hat{\phi}(t) = \sum_{k=0}^{\infty} \mu_k T_k(t)$.

- Expansion coefficients μ_k are formally defined by:

$$\begin{aligned}\mu_k &= \frac{2-\delta_{k0}}{\pi}\int_{-1}^{1}\frac{1}{\sqrt{1-t^2}}T_k(t)\hat{\phi}(t)dt\\ &= \frac{2-\delta_{k0}}{\pi}\int_{-1}^{1}\frac{1}{\sqrt{1-t^2}}T_k(t)\sqrt{1-t^2}\phi(t)dt\\ &= \frac{2-\delta_{k0}}{n\pi}\sum_{j=1}^{n}T_k(\lambda_j).\end{aligned}$$

- Here $2-\delta_{k0}$ == 1 when $k=0$ and == 2 otherwise.
- Note: $\sum T_k(\lambda_i) =$ *Trace*$[T_k(A)]$
- Estimate this, e.g., via stochastic estimator
- Generate random vectors $v^{(1)}, v^{(2)}, \cdots, v^{(n_{\text{vec}})}$
- Assume normal distribution with zero mean

- Each vector is normalized so that $\|v^{(l)}\| = 1, l = 1, \ldots, n_{\text{vec}}$.
- Estimate the trace of $T_k(A)$ with stochastisc estimator:

$$\text{Trace}(T_k(A)) \approx \frac{1}{n_{\text{vec}}} \sum_{l=1}^{n_{\text{vec}}} \left(v^{(l)}\right)^T T_k(A) v^{(l)}.$$

- Will lead to the desired estimate:

$$\boxed{\mu_k \approx \frac{2 - \delta_{k0}}{n\pi n_{\text{vec}}} \sum_{l=1}^{n_{\text{vec}}} \left(v^{(l)}\right)^T T_k(A) v^{(l)}.}$$

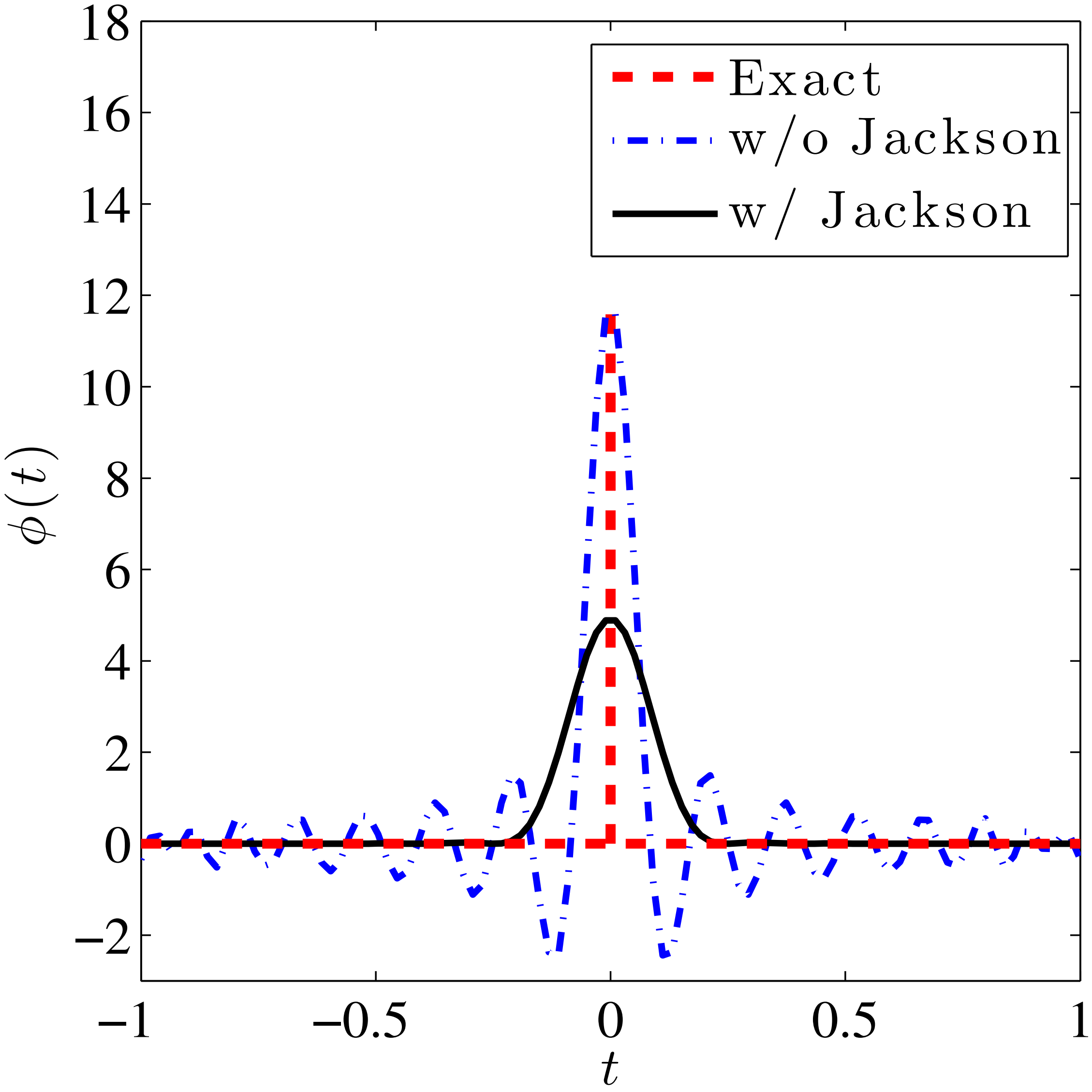
- To compute scalars of the form $v^T T_k(A) v$, exploit 3-term recurrence of the Chebyshev polynomial:

$$T_{k+1}(A)v = 2AT_k(A)v - T_{k-1}(A)v$$

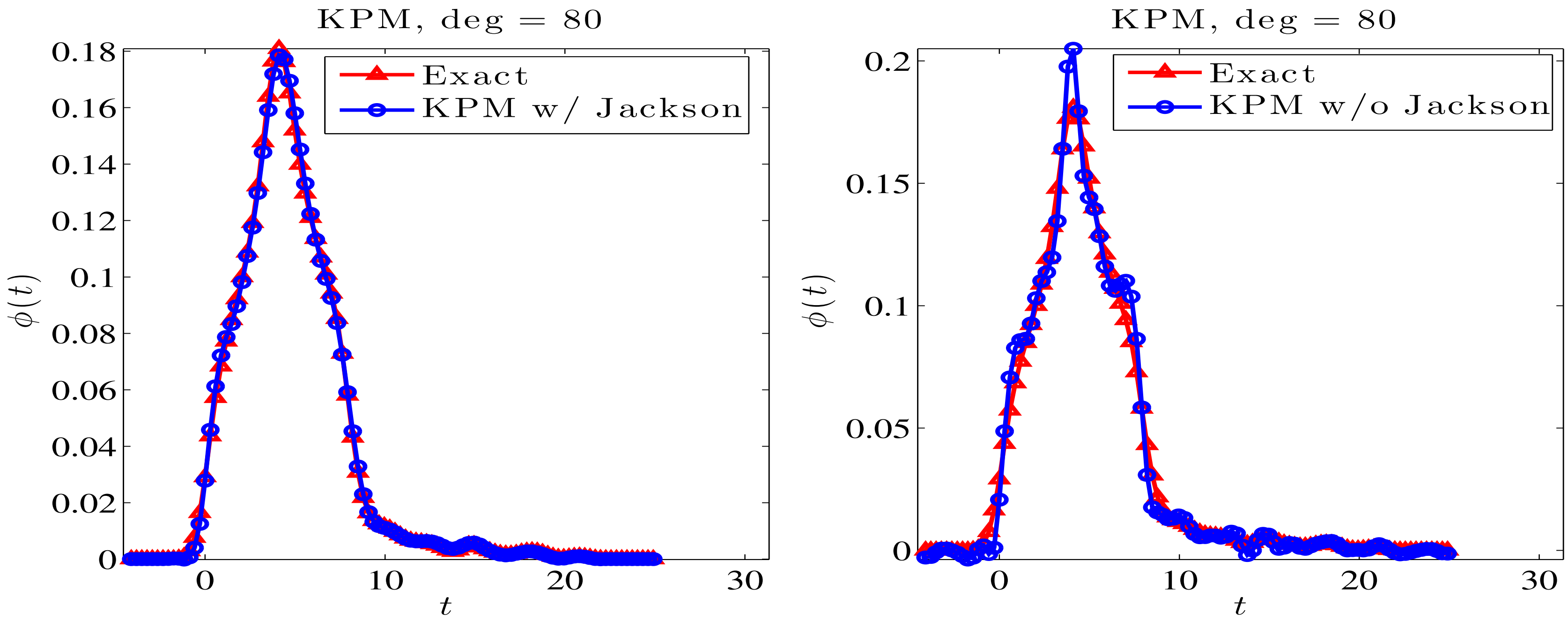
so if we let $v_k \equiv T_k(A)v$, we have

$$\boxed{v_{k+1} = 2Av_k - v_{k-1}}$$

- Jackson smoothing can be used –



An example with degree 80 polynomials



Left: Jackson damping; right: without Jackson damping.

MATLAB

Use of the Lanczos Algorithm

- Background: The Lanczos algorithm generates an orthonormal basis $V_m = [v_1, v_2, \cdots, v_m]$ for the Krylov subspace:

$$\textbf{span}\{v_1, Av_1, \cdots, A^{m-1}v_1\}$$

- ... such that: $V_m^H A V_m = T_m$ - with

$$T_m = \begin{pmatrix} \alpha_1 & \beta_2 & & & & \\ \beta_2 & \alpha_2 & \beta_3 & & & \\ & \beta_3 & \alpha_3 & \beta_4 & & \\ & & . & . & . & \\ & & & . & . & . \\ & & & & \beta_m & \alpha_m \end{pmatrix}$$

- Lanczos process builds orthogonal polynomials wrt to dot product:

$$\int p(t)q(t)dt \equiv (p(A)v_1, q(A)v_1)$$

- In theory v_i's defined by 3-term recurrence are orthogonal.
- Let $\theta_i,\ i = 1 \cdots, m$ be the eigenvalues of T_m [Ritz values]
- y_i's associated eigenvectors; Ritz vectors: $\{V_m y_i\}_{i=1:m}$
- Ritz values approximate eigenvalues
- Could compute θ_i's then get approximate DOS from these
- Problem: θ_i not good enough approximations – especially inside the spectrum.

- ➤ Better idea: exploit relation of Lanczos with (discrete) orthogonal polynomials and related Gaussian quadrature:

$$\int p(t)dt \approx \sum_{i=1}^{m} a_i p(\theta_i) \quad a_i = \left[e_1^T y_i\right]^2$$

- ➤ See, e.g., Golub & Meurant '93, and also Gautschi'81, Golub and Welsch '69.

- ➤ Formula exact when p is a polynomial of degree $\le 2m+1$

- Consider now $\int p(t)dt =< p, 1 >=$ (Stieljes) integral $\equiv$

$$(p(A)v, v) = \sum \beta_i^2 p(\lambda_i) \equiv < \phi_v, p >$$

- Then $\langle \phi_v, p \rangle \approx \sum a_i p(\theta_i) = \sum a_i \langle \delta_{\theta_i}, p \rangle \rightarrow$

$$\phi_v \approx \sum a_i \delta_{\theta_i}$$

- To mimick the effect of $\beta_i = 1, \forall i$, use several vectors v and average the result of the above formula over them..

Experiments

- Goal: to compare errors for similar number of matrix-vector products

- Example: Kohn-Sham Hamiltonian associated with a benzene molecule generated from PARSEC. $\boldsymbol{n = 8,219}$

- In all cases, we use 10 sampling vectors

- General observation: DGL, Lanczos, and KPM are best,

- Spectroscopic method does OK

- Haydock's method [another method based on the Lanczos algorithm] not as good

Method	L^1 error	L^2 error	L^∞ error
KPM w/ Jackson, deg=80	2.592e-02	5.032e-03	2.785e-03
KPM w/o Jackson, deg=80	2.634e-02	4.454e-03	2.002e-03
KPM Legendre, deg=80	2.504e-02	3.788e-03	1.174e-03
Spectroscopic, deg=40	5.589e-02	8.652e-03	2.871e-03
Spectroscopic, deg=100	4.624e-02	7.582e-03	2.447e-03
DGL, deg=80	1.998e-02	3.379e-03	1.149e-03
Lanczos, deg=80	2.755e-02	4.178e-03	1.599e-03
Haydock, deg=40	6.951e-01	1.302e-01	6.176e-02
Haydock, deg=100	2.581e-01	4.653e-02	1.420e-02

L^1, L^2, and L^∞ error compared with the normalized “surrogate” DOS for benzene matrix

➤ Many more experiments in survey paper [L. Lin, YS, C. Yang, SIAM Review, 2015].

Application: Eigenvalue counts

The problem: Given A (Hermitian) with eigenvalues $\lambda_1 \leq \lambda_2 \cdots \leq \lambda_n$ find an estimate of the number of eigenvalues of A in interval $[a, \quad b]$.

Main motivation: Eigensolvers based on splitting the spectrum in intervals and extracting eigenpairs from each interval independently.

- FEAST approach [Polizzi 2011]
- Sakurai-Sigiura method [2002]
- Schofield, Chelikowsky, YS'2011.

Standard method: Use Sylvester inertia theorem. However, this requires two LDL^T factorizations → can be expensive!

Eigenvalue counts: Integrating the DOS

- First alternative: integrate the Spectral Density in $[a,\ b]$.

$$\mu_{[a,b]} \approx n \left(\int_a^b \tilde{\phi}(t) dt \right) = n \sum_{k=0}^{m} \mu_k \left(\int_a^b \frac{T_k(t)}{\sqrt{1-t^2}} dt \right) = \ldots$$

- It turns out: this is equivalent to a method which uses the spectral projector ($u_i =$ eigenvector associated with λ_i) :

$$P = \sum_{\lambda_i \ \in \ [a \ \ b]} u_i u_i^T .$$

- We know that the trace of P is the wanted number $\mu_{[a,b]}$

- Goal: calculate an approximation to :

$$\mu_{[a,b]} = \text{Tr}\,(P) \ .$$

Approximation theory viewpoint (E. Polizzi, E. Di Napoli, YS)

- P is not available ... but can be approximated: Interpret P as a step function of A, namely:

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

- Approximate $h(t)$ by a polynomial $\psi(t)$

- Then $\mu_{[a,b]} \approx \mathrm{Tr}\,(\psi(A))$ approximated by a trace estimator:

$$\mu_{[a,b]} \approx \frac{1}{n_v} \sum_{k=1}^{n_v} v_k^\top \psi(A) v_k$$

where the v_k's are n_v random unit vectors.

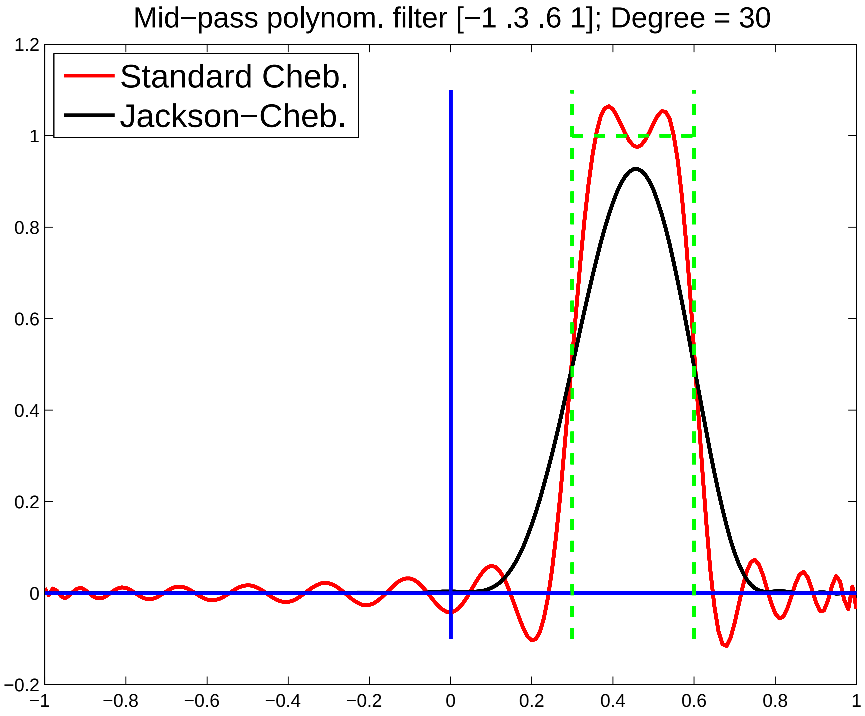
- We use degree p Chebyshev polynomials:

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

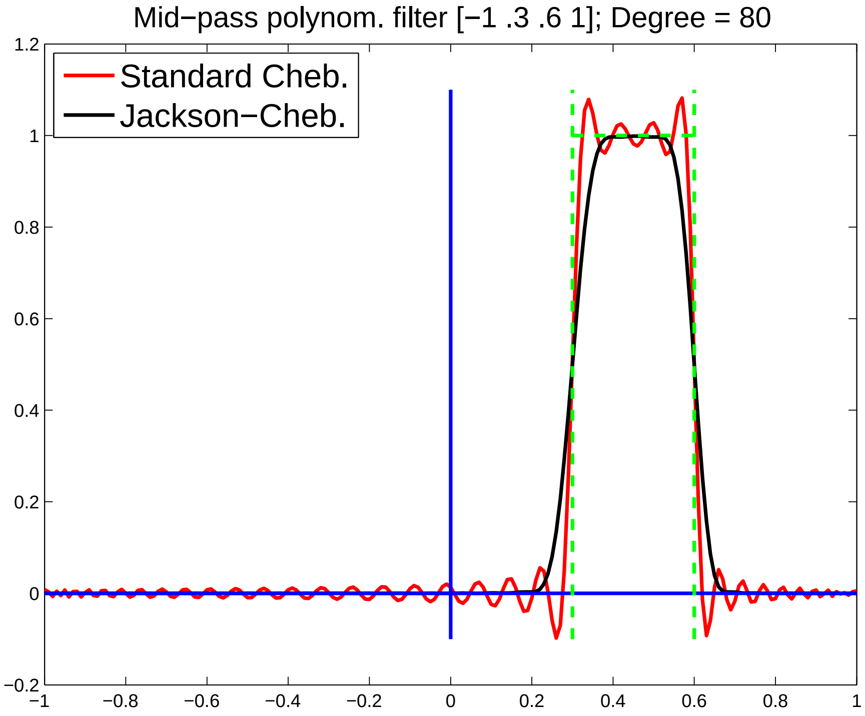
Examples for interval $[a, \ b] = [.3, \ .6]$

- Jackson damping (g_j^p) added to avoid Gibbs oscillations

Degree 30



Degree 80



Recall: $\mu_{[a,b]} = \text{Tr}\,(P) \approx \frac{n}{n_v} \sum_{k=1}^{n_v} \left[\sum_{j=0}^{p} \gamma_j v_k^T T_j(A) v_k \right].$

- To compute $w_j = T_j(A)v_k$, exploit 3-term recurrence of Chebyshev polynomials:

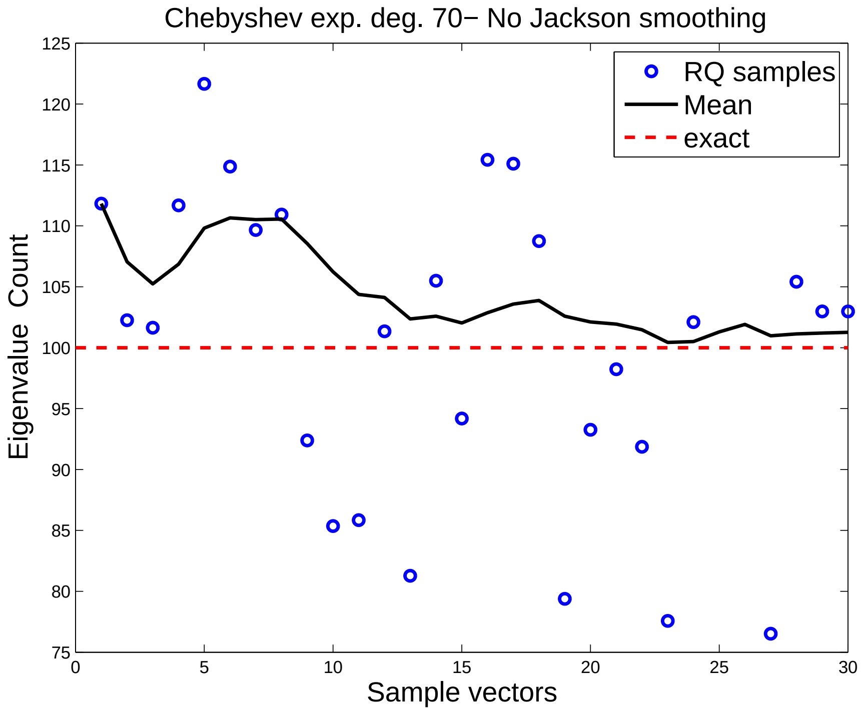
$$w_{j+1} = 2Aw_j - w_{j-1}.$$

(A is transformed so its eigenvalues are in $[-1\ 1]$)

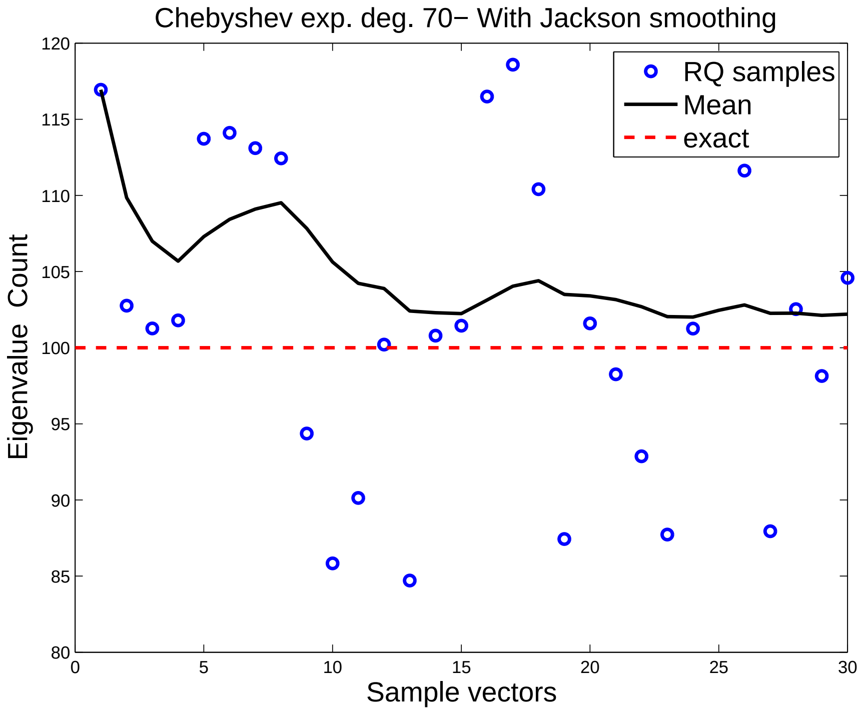
An example: Matrix ‘Na5’ from PARSEC (U Flor. Coll.)

- ➤ $n = 5832$, $nnz = 305630$ nonzero entries.
- ➤ Obtain the eigenvalue count when $a = (\lambda_{100} + \lambda_{101})/2$ and $b = (\lambda_{200} + \lambda_{201})/2$ so $\mu_{[a,b]} = 100$.

Without Jackson Damping



With Jackson Damping



Application: Estimating the rank

- Joint work with S. Ubaru

➤ Very important problem in signal processing applications, machine learning, etc.

➤ Often: a certain rank is selected ad-hoc. Dimension reduction is application with this “guessed” rank.

➤ Can be viewed as a particular case of the eigenvalue count problem - but need a cutoff value..

Approximate rank, Numerical rank

- Notion defined in various ways. A common one:

$$\boldsymbol{r_\epsilon = \min\{rank(B) : B \in \mathbb{R}^{m\times n}, \|A - B\|_2 \le \epsilon\}},$$

$$\boldsymbol{r_\epsilon} = \text{ Number of sing. values } \ge \boldsymbol{\epsilon}$$

- Two distinct problems:

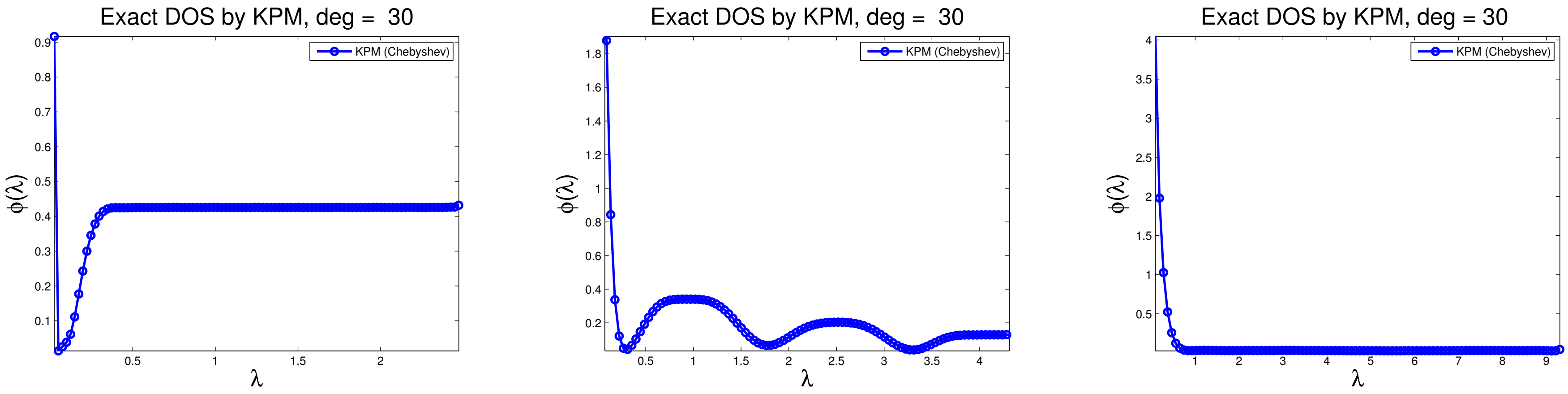
1. Get a good $\boldsymbol{\epsilon}$ 2. Estimate number of sing. values $\ge \boldsymbol{\epsilon}$

- We will need a cut-off value ('threshold') $\boldsymbol{\epsilon}$.

- Could use 'noise level' for $\boldsymbol{\epsilon}$, but not always available

Threshold selection

- How to select a good threshold?
- Answer: Obtain it from the DOS function



(A) (B) (C)

Exact DOS plots for three different types of matrices.

- To find: point immediatly following the initial sharp drop observed.

- Simple idea: use derivative of DOS function ϕ

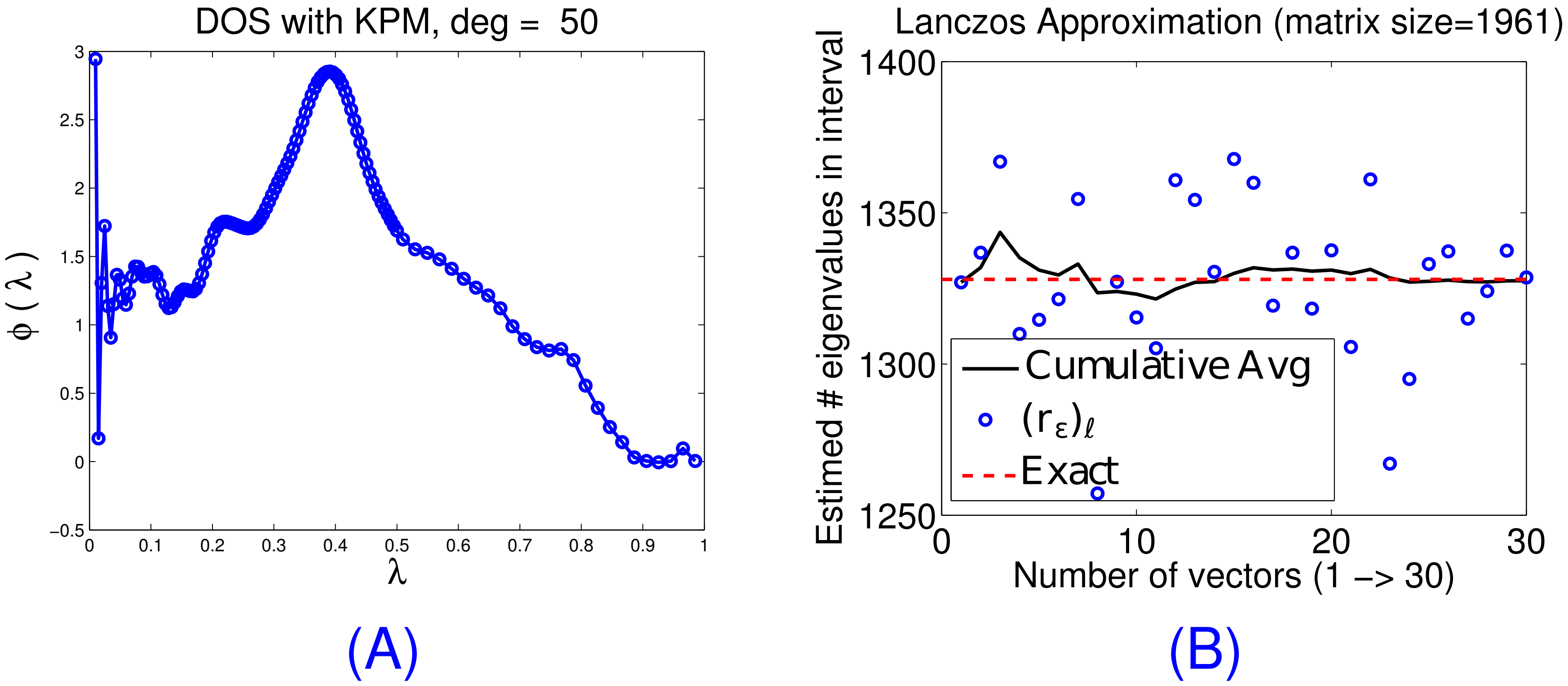
- For an $n \times n$ matrix with eigenvalues $\lambda_n \leq \lambda_{n-1} \leq \cdots \leq \lambda_1$:

$$\epsilon = \min\{t : \lambda_n \leq t \leq \lambda_1, \phi'(t) = 0\}.$$

- In practice replace by

$$\epsilon = \min\{t : \lambda_n \leq t \leq \lambda_1, |\phi'(t)| \geq \text{tol}\}$$

Experiments



(A) The DOS found by KPM.

(B) Approximate rank estimation by The Lanczos method for the example `netz4504`.

Tests with Matérn covariance matrices for grids

- Important in statistical applications

Approximate Rank Estimation of Matérn covariance matrices

Type of Grid (dimension)	Matrix Size	# $\boldsymbol{\lambda_i}$'s $\geq \boldsymbol{\epsilon}$	$\boldsymbol{r_\epsilon}$	
			KPM	Lanczos
1D regular Grid $(\mathbf{2048 \times 1})$	$\mathbf{2048}$	$\mathbf{16}$	$\mathbf{16.75}$	$\mathbf{15.80}$
1D no structure Grid $(\mathbf{2048 \times 1})$	$\mathbf{2048}$	$\mathbf{20}$	$\mathbf{20.10}$	$\mathbf{20.46}$
2D regular Grid $(\mathbf{64 \times 64})$	$\mathbf{4096}$	$\mathbf{72}$	$\mathbf{72.71}$	$\mathbf{72.90}$
2D no structure Grid $(\mathbf{64 \times 64})$	$\mathbf{4096}$	$\mathbf{70}$	$\mathbf{69.20}$	$\mathbf{71.23}$
2D deformed Grid $(\mathbf{64 \times 64})$	$\mathbf{4096}$	$\mathbf{69}$	$\mathbf{68.11}$	$\mathbf{69.45}$

- For all test $\boldsymbol{M(deg) = 50, n_v}$=30

Application: The LogDeterminant

Evaluate the Log-determinant of A:

$$\log\det(A) = \text{Trace}(\log(A)) = \sum_{i=1}^{n} \log(\lambda_i).$$

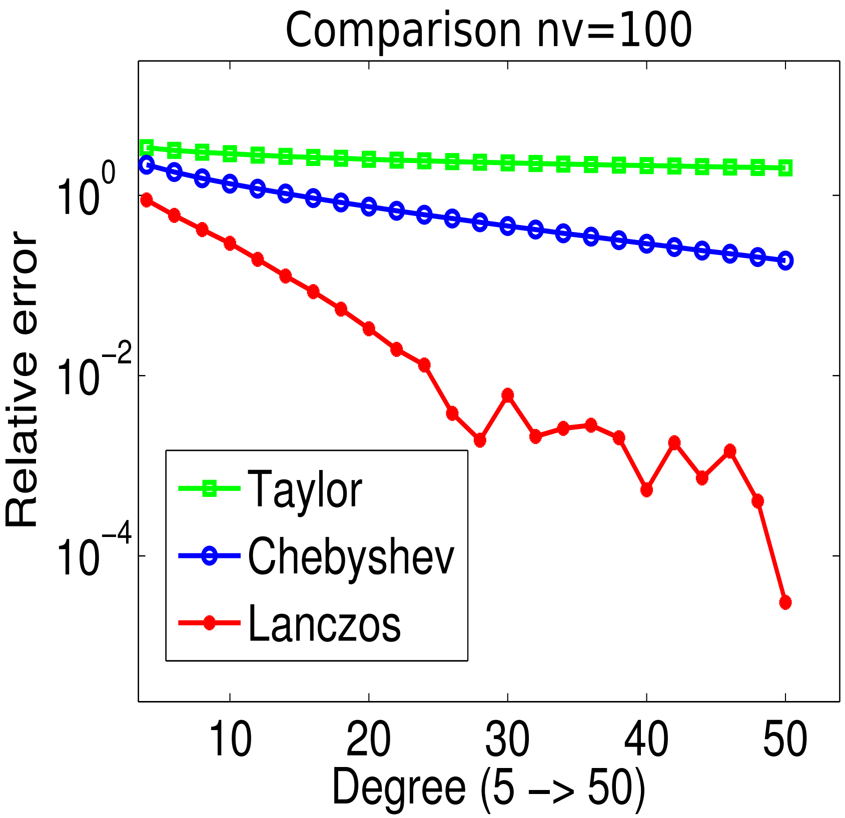
A is SPD.

➤ Estimating the log-determinant of a matrix equivalent to estimating the trace of the matrix function $f(A) = \log(A)$.

➤ Can invoke Stochastic Lanczos Quadrature (SLQ) to estimate this trace.

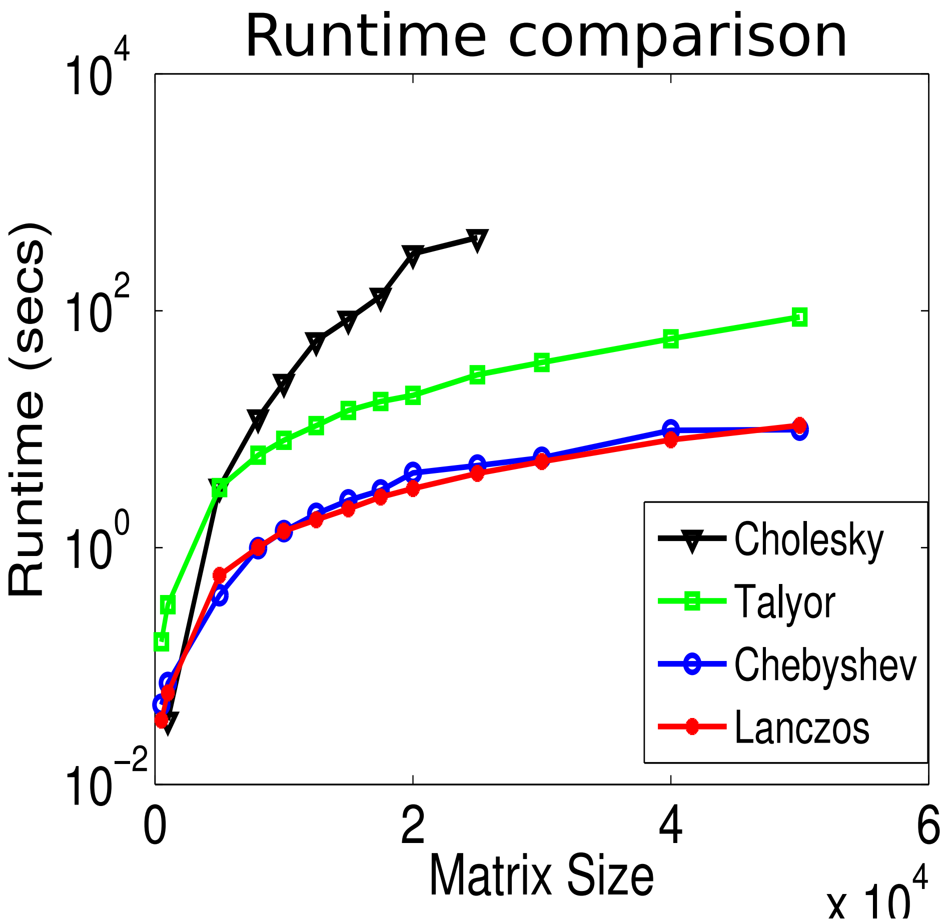
Numerical example: A graph Laplacian `california` of size 9664×9664, $nz \approx 10^5$ from the Univ. of Florida collection.

Rel. error vs degree



- 3 methods: Taylor Series, Chebyshev expansion, SLQ
- # starting vectors $nv = 100$ in all three cases.

Runtime comparisons



Application: Log-likelihood.

Comes from parameter estimation for Gaussian processes

- ➤ Objective is to maximize the log-likelihood function with respect to a 'hyperparameter' vector ξ

$$\log p(z \mid \xi) = -\frac{1}{2}\left[z^\top S(\xi)^{-1} z + log \det S(\xi) + \text{cst}\right]$$

where z = data vector and $S(\xi)$ == covariance matrix parameterized by ξ

- ➤ Can use the same Lanczos runs to estimate $z^\top S(\xi)^{-1} z$ and logDet term simultaneously.

Application: calculating nuclear norm

- $\|\boldsymbol{X}\|_* = \sum \boldsymbol{\sigma}_i(\boldsymbol{X}) = \sum \sqrt{\boldsymbol{\lambda}_i(\boldsymbol{X}^T\boldsymbol{X})}$
- Generalization: Schatten $\boldsymbol{p}$-norms

$$\|\boldsymbol{X}\|_{*,\boldsymbol{p}} = [\sum \boldsymbol{\sigma}_i(\boldsymbol{X})^{\boldsymbol{p}}]^{1/\boldsymbol{p}}$$

- See:

J. Chen, S. Ubaru, YS, “Fast estimation of log-determinant and Schatten norms via stochastic Lanczos quadrature”, (Submitted).

Conclusion

- ➤ Estimating traces is a key ingredient in many algorithms
- ➤ Physics, machine learning, matrix algorithms, ..
- ➤ .. many new problems related to ‘data analysis’ and ’statistics’, and in signal processing,

Q: Can we do better than standard random sampling?