Numerical Matrix Analysis

Notes #26
GMRES

Peter Blomgren
⟨blomgren@sdsu.edu⟩

Department of Mathematics and Statistics
Dynamical Systems Group
Computational Sciences Research Center
San Diego State University
San Diego, CA 92182-7720

http://terminus.sdsu.edu/

Spring 2024
(Revised: April 29, 2024)

Outline

Arnoldi Iteration $\rightsquigarrow A\vec{x} = \vec{b}$

Last time we looked at the Arnoldi Iteration as a procedure for finding eigenvalues. Next, we leverage it to solve $A\vec{x} = \vec{b}$; introducing GMRES, the *"Generalized Minimal RESiduals"* strategy.

Algorithm (Arnoldi Iteration)

$\vec{b} \leftarrow \texttt{random}(\mathbb{R}^{m\times 1})$,
$\vec{q}_1 \leftarrow \vec{b}/\|\vec{b}\|$
for $n \in \{1, 2, \ldots\}$ **do**
  $\vec{v} \leftarrow A\vec{q}_n$
  for $j \in \{1, \ldots, n\}$ **do**
    $h_{j,n} \leftarrow \vec{q}_j^{\,*}\vec{v}$
    $\vec{v} \leftarrow \vec{v} - h_{j,n}\vec{q}_j$
  end for
  $h_{n+1,n} \leftarrow \|\vec{v}\|$ TB-33.2: $h_{n+1,n} = 0$ (Breakdown due to Convergence)
  $\vec{q}_{n+1} \leftarrow \vec{v}/h_{n+1,n}$
end for

Structure, Notation, Idea

Problem Structure and Notation

We consider $A \in \mathbb{C}^{m\times m}$, with $\dim(\mathrm{null}(A)) = 0$; $\vec{b} \in \mathbb{C}^m$; $K(A, \vec{b}, n) = \mathrm{span}\left(\vec{b}, A\vec{b}, \ldots, A^{n-1}\vec{b}\right)$; and $\vec{x}_* = A^{-1}\vec{b}$ (exact solution).

GMRES Idea

At the n^{th} step, $\vec{x}_n \approx \vec{x}_*$ is the vector $\vec{x}_n \in K(A, \vec{b}, n)$ which minimizes $\|\vec{r}_n\|$, where $\vec{r}_n = (\vec{b} - A\vec{x}_n)$; *i.e.* each $\vec{x}_n$ is the solution to a least squares problem over an n-dimensional (Krylov) subspace.

Many iterative optimization methods do something similar (at least in "spirit") — seeking approximately optimal approximations in carefully nested sequences of subspaces. (See [MATH 693A])

GMRES: "Obvious" Strategy

With the Krylov matrix

$$K_n = \left[\begin{array}{c:c:c:c} \vec{b} & A\vec{b} & \cdots & A^{n-1}\vec{b} \end{array} \right],$$

on hand, the "obvious" (ill-conditioned) way is to form

$$AK_n = \left[\begin{array}{c:c:c:c} A\vec{b} & A^2\vec{b} & \cdots & A^n\vec{b} \end{array} \right],$$

which has the column space $\mathrm{range}(AK_n)$. We seek $\vec{c}_n$

$$\vec{c}_n = \underset{\vec{c}\in\mathbb{C}^n}{\arg\min} \|(AK_n)\vec{c} - \vec{b}\|, \quad \text{and } \vec{x}_n = K_n\vec{c}_n.$$

Note: arg min "returns" the argument-that-minimizes the given function (objective).

The "Obvious" Strategy Fails (in Finite Precision)

A Q_nR_n-factorization of AK_n would provide the necessary components of the pseudo-inverse necessary for identification of the solution to the least squares problem.

But, alas, this approach is numerically unstable, and wasteful (the R_n factor is not needed.)

Instead, we use the Arnoldi Iteration to construct Krylov Matrices Q_n, whose columns satisfy

$$\mathrm{span}\,(\vec{q}_1, \vec{q}_2, \ldots, \vec{q}_n) = K(A, \vec{b}, n),$$

thus we can represent $\vec{x}_n = Q_n\vec{y}_n$ rather than $\vec{x}_n = K_n\vec{c}_n$; the associated Least Squares Problem is

$$\vec{y}_n = \underset{\vec{y}\in\mathbb{C}^n}{\arg\min} \|AQ_n\vec{y} - \vec{b}\|.$$

"Shrinking" the Problem 1 of 2

As stated $\vec{y}_n = \arg\min_{\vec{y}\in\mathbb{C}^n} \|AQ_n\vec{y} - \vec{b}\|$ is an $(m \times n)$-dimensional Least Squares Problem, but using the structure of Krylov subspaces, its essential dimension is reduced to $((n+1) \times n)$:

We use the "Arnoldi relation" $AQ_n = Q_{n+1}\tilde{H}_n$ to transform the problem into

$$\vec{y}_n = \underset{\vec{y}\in\mathbb{C}^n}{\arg\min} \|Q_{n+1}\tilde{H}_n\vec{y} - \vec{b}\|,$$

multiplication by Q^*_{n+1} preserves the norm, since both $(Q_{n+1}\tilde{H}_n\vec{y})$ and $\vec{b}$ are — by construction — in the column space of Q_n; we get

$$\vec{y}_n = \underset{\vec{y}\in\mathbb{C}^n}{\arg\min} \|\tilde{H}_n\vec{y} - Q^*_{n+1}\vec{b}\|.$$

"Shrinking" the Problem 2 of 2

Finally, by construction of $Q_n^{\ddagger}$, we get $Q^*_{n+1}\vec{b} = \|\vec{b}\|\vec{e}_1$, so our problem is

$$\vec{y}_n = \underset{\vec{y}\in\mathbb{C}^n}{\arg\min} \|\tilde{H}_n\vec{y} - \beta\vec{e}_1\|, \quad \text{where } \beta = \|\vec{b}\|;$$

and $\vec{x}_n = Q_n\vec{y}_n$.

$\vec{e}_1$ is as usual the first standard basis vector in the appropriate space; it has a single "1" in the first component, and the remaining components are "0".

‡ $\mathrm{span}(Q_1) = \mathrm{span}(\vec{b})$

GMRES Algorithm

Algorithm (GMRES)

$\vec{b} \leftarrow \texttt{random}(\mathbb{R}^{m\times 1})$,
$\beta \leftarrow \|\vec{b}\|$
$\vec{q}_1 \leftarrow \vec{b}/\beta$
for $n \in \{1, 2, \dots\}$ **do**
  $\vec{v} \leftarrow A\vec{q}_n$
  for $j \in \{1, \dots, n\}$ **do**
    $h_{j,n} \leftarrow \vec{q}_j^{\,*}\vec{v}$
    $\vec{v} \leftarrow \vec{v} - h_{j,n}\vec{q}_j$
  end for
  $h_{n+1,n} \leftarrow \|\vec{v}\|$
  $\vec{q}_{n+1} \leftarrow \vec{v}/h_{n+1,n}$
  $\vec{y}_n \leftarrow \arg\min_{\vec{y}\in\mathbb{C}^n} \|\tilde{H}_n\vec{y} - \beta\vec{e}_1\|$
  $\vec{x}_n \leftarrow Q_n\vec{y}_n$
end for

Comments

- In each step we solve an $((n+1)\times n)$ Least Squares Problem with Hessenberg structure; the cost via QR-factorization is $\mathcal{O}(n^2)$ (exploiting the Hessenberg structure).
- It is possible to save work by identifying an updating strategy for the Q_nR_n factorization of $\tilde{H}_n$ from $Q_{n-1}R_{n-1} = \tilde{H}_{n-1}$. The cost is then one *Givens rotation** [T&B PROBLEMS 10.4 & 35.4] and $\mathcal{O}(n)$ work.

* The Givens rotations are the building blocks for a slightly (50%) more expensive alternative to the Householder reflection method for computing the QR-factorization.

Polynomial Approximation

Polynomial Class P_n

$$P_n = \{\ \text{POLYNOMIALS OF DEGREE} \le n,\ \text{WITH } p(0) = 1\ \},$$

i.e. the constant coefficient $c_0 = 1$.

Just as in the Arnoldi Iteration case, we can discuss the GMRES iteration in terms of polynomial approximations:

$$\vec{x}_n = q_n(A)\vec{b}$$

where $q_n(\cdot)$ is a polynomial of degree $(n-1)$ with coefficients from the vector $\vec{c}_n = \arg\min_{\vec{c}\in\mathbb{C}^n} \|AK_n\vec{c} - \vec{b}\|$.

Polynomial Approximation

With $p_n(z) = 1 - zq_n(z)$, we have

$$\vec{r}_n = \vec{b} - A\vec{x}_n = (I - Aq_n(A))\vec{b} = p_n(A)\vec{b},$$

for some $p_n \in P_n$.

GMRES solves the following problem

GMRES Approximation Problem

Find $p_n \in P_n$ such that

$$p_n = \arg\min_{p\in P_n} \|p(A)\vec{b}\|.$$

Invariance Properties

Theorem

Let the GMRES iteration be applied to a matrix $A \in \mathbb{C}^{m \times m}$, then the following holds:

- [Scale-Invariance] *If A is changed to σA for some $\sigma \in \mathbb{C}$, and $\vec{b}$ is changed to $\sigma\vec{b}$, the residuals $\vec{r}_n$ change to $\sigma\vec{r}_n$.*
- [Invariance under Unitary Transformations] *If A is changed to UAU^* for some unitary matrix U, and $\vec{b}$ is changed to $U\vec{b}$, the residuals $\vec{r}_n$ change to $U^*\vec{r}_n$.*

Convergence

Theorem (GMRES Convergence Property#1: Monotonic Convergence)

GMRES converges monotonically,

$$\|\vec{r}_{n+1}\| \leq \|\vec{r}_n\|.$$

This must be the case since we are minimizing over expanding subspaces, *i.e.* $K(A, \vec{b}, n) \subset K(A, \vec{b}, n+1)$.

Theorem (GMRES Converence Property#2: m-step Convergence)

In infinite precision, GMRES converges in at most m steps

$$\|\vec{r}_m\| = 0.$$

This must be the case since $K(A, \vec{b}, m) = \mathbb{C}^m$.

Convergence

The factor that gives us more useful convergence estimates is related to the polynomial p_n:

$$\frac{\|\vec{r}_n\|}{\|\vec{b}\|} \leq \inf_{p_n \in P_n} \|p_n(A)\|,$$

which brings us back to studying matrix polynomials related to Krylov subspaces.

How small can $\|p_n(A)\|$ be?

The standard way to get bounds on the behavior of $\|p_n(A)\|$ is to study polynomials on the spectrum $\lambda(A)$.

Definition

If p is a polynomial and $S \subset \mathbb{C}$, then

$$\|p\|_S := \sup_{z \in S} |p(z)|.$$

In the case where S is a finite set of points in the complex plane, the supremum (sup) is just the maximum (max).

When A is diagonalizable $A = V \Lambda V^{-1}$, then

$$\|p(A)\| \leq \|V\| \, \|p(\Lambda)\| \, \|V^{-1}\| = \kappa(V) \, \|p\|_{\lambda(A)}.$$

$\kappa(V)$ is the conditioning of the Eigenbasis.

How small can $\|p_n(A)\|$ be?

Theorem

At step n of the `GMRES` *iteration, the residual $\vec{r}_n$ satisfies*

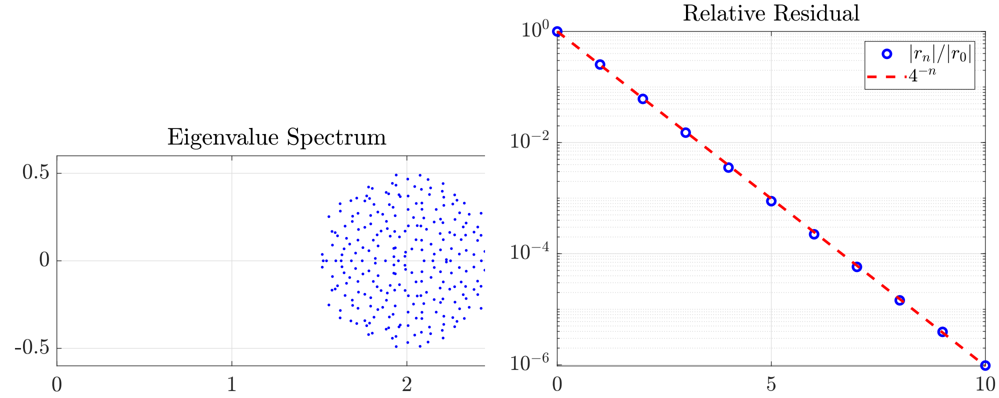
$$\frac{\|\vec{r}_n\|}{\|\vec{b}\|} \leq \inf_{p_n \in P_n} \|p_n(A)\| \leq \kappa(V) \inf_{p_n \in P_n} \|p_n\|_{\lambda(A)},$$

where $\lambda(A)$ is the set of eigenvalues of A, V is a non-singular matrix of eigenvectors (assuming A is diagonalizable), and $\|p_n\|_{\lambda(A)} = \sup_{z\in\lambda(A)} |p_n(z)|$.

As long as $\kappa(V)$ is not too large — *i.e.* the closer A is to being normal (unitarily diagonalizable) — and if polynomials p_n which decrease quickly on $\lambda(A)$ exist, then `GMRES` converges quickly.

T&B-35.1


```
m = 256; b = ones(m,1);
A = 2*eye(m) + 0.5 * randn(m)/sqrt(m);
```



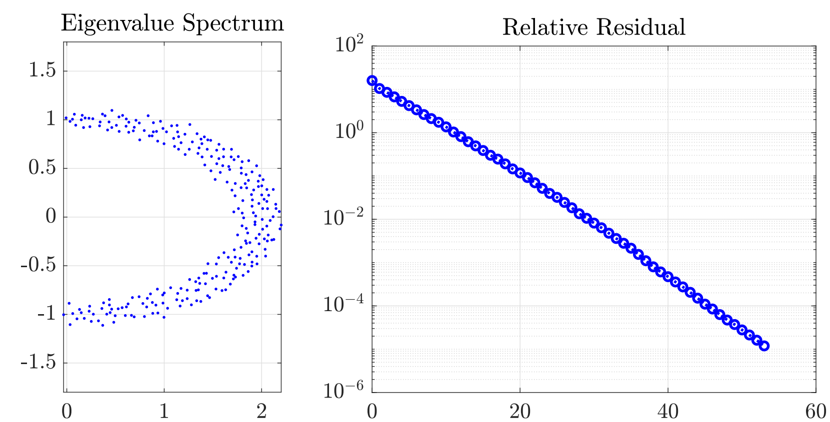
$\kappa(A) = 2.065$ $\kappa(V) = 216.490$

T&B-35.1

- The eigenvalue spectrum of A is roughly contained in the disk of radius $\frac{1}{2}$, centered at $z = 2$.
- $\|p(A)\|$ is approximately minimized by $p(z) = (1 - z/2)^n$;
- $\lambda(I - A/2)$ is roughly contained in the disc of radius $\frac{1}{4}$, centered at $z = 0$, so the convergence rate is $\|p_n(A)\| = \|(I - A/2)^n\| \sim \frac{1}{4^n}$.
- A is quite well-conditioned: $\kappa(A) = 2.065$.
- A is "not too far" from normal: $\kappa(V) = 216.490$.

T&B-35.2


```
m = 256; b = ones(m,1); th = (0:(m-1))*pi / (m-1);
A = 2*eye(m) + 0.5 * randn(m)/sqrt(m) + diag(-2+2*sin(th)+i*cos(th));
```



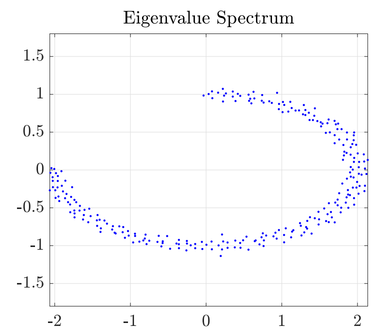
$\kappa(A) = 3.802$ $\kappa(V) = 150.711$

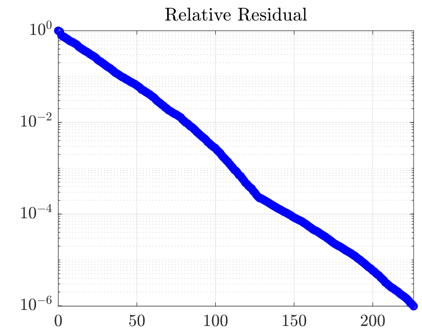
T&B-35.2

- The eigenvalue spectrum of A now "surrounds" the origin.
- A is quite well-conditioned: $\kappa(A) = 3.802$.
- A is not too far from normal: $\kappa(V) = 150.711$.
- The convergence is quite slow in this case (observed $\sim 1.23^{-n}$).
- Note that the slowdown in convergence does not depend on conditioning, but on the location of the eigenvalues.
- Clearly, understanding the impact of the "structure" of the eigenvalue spectrum is a non-trivial topic...

T&B-35.2$^+$

```
m = 256; b = ones(m,1); th = 1.5*(0:(m-1))*pi / (m-1);
A = 2*eye(m) + 0.5 * randn(m)/sqrt(m) + diag(-2+2*sin(th)+i*cos(th));
```

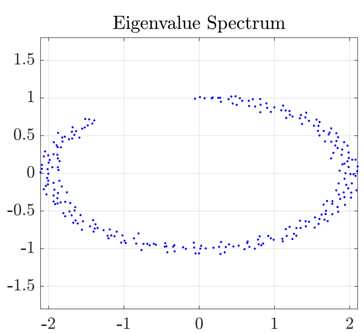


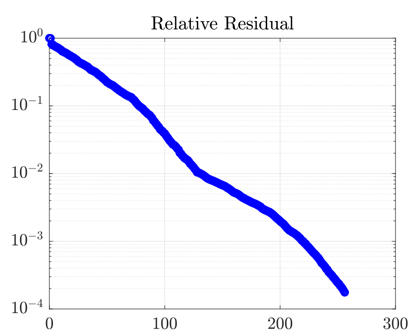


$\kappa(A) = 3.9371$ $\kappa(V) = 73.7831$

T&B-35.2^{++}

```
m = 256; b = ones(m,1); th = 1.75*(0:(m-1))*pi / (m-1);
A = 2*eye(m) + 0.5 * randn(m)/sqrt(m) + diag(-2+2*sin(th)+i*cos(th));
```

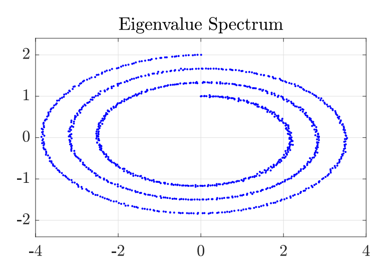


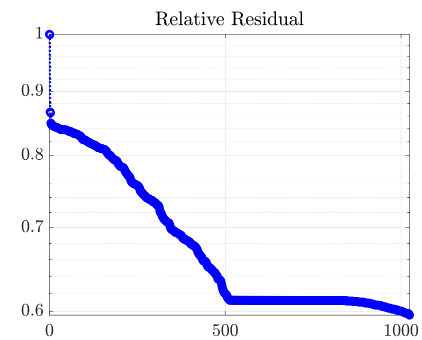


$\kappa(A) = 3.7551$ $\kappa(V) = 58.6277$

T&B-35.2^{+++}

```
m = 1024; b = ones(m,1); th = 6.00*(0:(m-1))*pi / (m-1);
A = 2*eye(m) + 0.5 * randn(m)/sqrt(m) + diag(-2+(1+th/(6*pi)).*(2*sin(th)+i*cos(th)));
```





$\kappa(A) = 4.7704$ $\kappa(V) = 40.2912$