Functions of large-scale matrices

How do we compute $f(A)$ if A is large and sparse? Huge recent research topic.

Most of the time, one wants $f(A)b$ rather than $f(A)$, because $f(A)$ is full (unless f and A are special, e.g., square of a banded matrix). The main techniques are those we have seen in the beginning.

- Replace f with an approximating polynomial (or rational function) on a region U that includes the spectrum of A (how?).
- Contour integration:

$$\frac{1}{2\pi i}\int_\Gamma f(z)(zI-A)^{-1}\mathrm{d}z \approx \sum_{k=1}^{n} w_k f(x_k)(x_k I - A)^{-1}.$$

- Ad-hoc methods, involving e.g. discretization of differential equations: for instance, $\exp(A)b = v(1)$ where $\dot{v}(t) = Av(t)$, $v(0) = b$.

Arnoldi for matrix functions

Another possibility with the "Swiss-army knife algorithm" for large matrices: Arnoldi.

Let us recap Arnoldi (with matrix functions in mind).

Krylov subspace

$$\begin{aligned} K_n(A, b) &= \operatorname{span}(b, Ab, A^2 b, \dots, A^{n-1} b) \\ &= \{p(A)b \colon p \text{ polynomial of degree} < n\}. \end{aligned}$$

Suppose we have computed the vectors $b, Ab, A^2 b, \dots, A^{n-1} b, A^n b$ explicitly.

This gives us a "recipe" to evaluate Av for any $v \in K_n(A, b)$:

$$\begin{aligned} Av &= A(\alpha_0 b + \alpha_1 Ab + \cdots + \alpha_{n-1} A^{n-1} b) = p(A)b \\ &= (\alpha_0 Ab + \alpha_1 A^2 b + \cdots + \alpha_{n-1} A^n b) = Ap(A)b. \end{aligned}$$

Towards Arnoldi

This recipe often is not satisfying: $b, Ab, A^2b, \dots A^{n-1}b$ converge to the leading eigenvector of A (power method), so when n gets large these vectors tend to be parallel (and often also huge/small).

Some tasks, e.g. determining $\alpha_0, \dots, \alpha_{n-1}$ given v, are hopelessly ill-conditioned:

$$V_n = \begin{bmatrix} b & Ab & \dots & A^{n-1}b \end{bmatrix}, \quad a = \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{n-1} \end{bmatrix} = (V_n^* V_n)^{-1} V_n^* v.$$

It would be much better to work with an orthogonal basis of $\operatorname{Im} V_n = K_n(A, b)$, e.g., the Q factor of $V = QR$: then the coordinates are given by $a = V_{n-1}^* v$.

Arnoldi as "recipe"

The "recipe" for Av comes from the columns of

$$A\begin{bmatrix} b & Ab & \dots & A^{n-1}b \end{bmatrix} = \begin{bmatrix} b & Ab & \dots A^{n-1}b & A^n b \end{bmatrix} \begin{bmatrix} 0 & & & \\ 1 & \ddots & & \\ & \ddots & 0 & \\ & & 1 & 0 \\ & & & 1 \end{bmatrix}.$$

With (nested) orthonormal bases of V_n and V_{n+1}, it becomes

$$A \underbrace{\begin{bmatrix} v_1 & v_2 & \dots v_n \end{bmatrix}}_{V_n} = \underbrace{\begin{bmatrix} v_1 & v_2 & \dots & v_n & v_{n+1} \end{bmatrix}}_{v_{n+1}} \underline{H}_n$$

for a certain $\underline{H}_n \in \mathbb{C}^{(n+1)\times n}$ (to be determined).

Arnoldi iteration

We can compute the v_j one after the other: suppose we already have $v_1, \ldots, v_j$. Then,

$$Av_j = v_1\alpha_{1,j} + v_2\alpha_{2,j} + \cdots + v_j\alpha_{j-1,j} + v_{j+1}\alpha_{j+1,j}$$

Thanks to orthogonality, we can compute and subtract $v_i^* A v_j = \alpha_{i,j}$ for $i = 1, \ldots, j$, and we are left with $v_{j+1}\alpha_{j+1,j}$:

```
w = A*V(:,j);
for i = 1:j
       alpha(i,j) = V(:,i)' * w;
       w = w - V(:,i) * alpha(i,j);
end
alpha(j+1,j) = norm(w);
V(:,j+1) = w / alpha(j+1,j);
```

Starting with $v_1 = \frac{b}{\|b\|}$, that is all we need to implement it.

Remarks on Arnoldi

This computed nested bases for the Krylov subspaces:

$$K_j(A, b) = \operatorname{Im}\begin{bmatrix} v_1 & v_2 & \dots & v_j \end{bmatrix}.$$

Why did we start with $w = Av_j$ and not another vector? Because we can prove (inductively) that $v_j = p(A)b$ for a polynomial of degree exactly $j - 1$.

So $Av_j = \tilde{p}(A)b$ for a polynomial $\tilde{p}(z) = zp(z)$ of degree exactly j, and the same holds for $Av_j - \sum_{i=0}^{j} v_i \alpha_{i,j} = v_{j+1}\alpha_{j+1,j}$.

(If we had started instead with a vector w of 'degree' $< j$, we would have obtained $Av_j - \sum_{i=0}^{j} v_i \alpha_{i,j} = 0$.)

Lucky breakdown in Arnoldi

It could still happen that $\alpha_{j+1,j} = 0$ and we get 'breakdown' (division by 0) though! But it happens only when (at some step)

$$A^j b = p(A)b$$

for some polynomial $p(z)$ of degree $< j$, i.e., $d(A)b = 0$ for $d(z) = z^j - p(z)$ of degree j.

When this holds, $K_j(A, b) = \mathrm{span}(b, Ab, \ldots, A^{j-1}b) =$ is an invariant subspace of A.

This could happen as early as in step $j = 1$, if b is an eigenvector of A.

When this breakdown happens, we are lucky, because we know a 'recipe' to compute $n(A)b$ for polynomials $n(z)$ of any degree: from the polynomial division $n(z) = d(z)q(z) + r(z)$, we get $n(A)b = r(A)b$.

Arnoldi: the 'recipe'

Gathering all the relations involving the Av_j in a matrix, we get

$$A \underbrace{\begin{bmatrix} v_1 & \dots & v_n \end{bmatrix}}_{V_n} = \underbrace{\begin{bmatrix} v_1 & \dots & v_{n+1} \end{bmatrix}}_{V_{n+1}} \underbrace{\begin{bmatrix} \alpha_{1,1} & \alpha_{1,2} & \alpha_{1,3} & \dots & \alpha_{1,n} \\ \alpha_{1,2} & \alpha_{2,2} & \alpha_{2,3} & \dots & \alpha_{2,n} \\ 0 & \alpha_{3,2} & \alpha_{3,3} & \dots & \alpha_{3,n} \\ 0 & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \alpha_{n,n} \\ 0 & \dots & \dots & 0 & \alpha_{n+1,n} \end{bmatrix}}_{\underline{H}_n}.$$

When $\alpha_{n+1,n} = 0$ (breakdown), $AV_n = V_n H_n$, where H_n is $\underline{H}_n$ without the last row. This is an invariant subspace relation.

Note that $H_n = V_n^* A V_n$.

Formula for $p(A)b$

Lemma

For all polynomials with $\deg p < n$

$$p(A)b = V_n p(H_n) V_n^* b = V_n p(H_n) e_1 \|b\|$$

Proof: it is sufficient to show that $A^j b = V_n H_n^j V_n^* b$ for $j < n$.

$$V_n H_n^j V_n^* = V_n V_n^* A V_n V_n^* A \cdots V_n V_n^* A V_n V_n^* A V_n V_n^* b$$

Let us start from the right. $V_n V_n^*$ is an orthogonal projection matrix onto the Krylov space. Since $b \in K_n(A, b)$, $V_n V_n^* b = b$. Now the rightmost part reads $V_n V_n^* A b$; but this equals Ab because $Ab \in K_n(A, b)$, and so on.

Arnoldi, matrix functions, and polynomial approximations

Idea: let's compute $f(A)b \approx V_n f(H_n) e_1 \|b\|$. This approximation is exact when f is a polynomial of degree $< n$.

Moreover,

$$V_n f(H_n) e_1 \|b\| = V_n \tilde{p}(H_n) e_1 \|b\| = \tilde{p}(A)b,$$

where $\tilde{p}$ is the interpolating polynomial to f on the spectrum of H_n (not that of A!)

Known behaviour from Arnoldi theory: for many matrices, the eigenvalues of H_n (Ritz values) approximate the extremal eigenvalues of A.

Arnoldi variants [Güttel '13]

We expect good results if (1) enough steps are taken, and (2) the function f takes its larger values in the extremal eigenvalues of A.

What if f takes its larger values at some internal point of the spectrum of A, e.g., $f(x) = \frac{1}{x}$ and A has both positive and negative eigenvalues (or complex values not all in the same half-plane)?

Idea: change the Arnoldi iteration!

- Extended Arnoldi: constructs an orthonormal basis for

  $$\{p(A)b \colon p = \alpha_{-n_1}x^{-n_1} + \alpha_{-n_1+1}x^{-n_1+1} + \cdots + \alpha_{n_2-1}x^{n_2-1}\}$$
  $$= A^{-n_1}K_{n_1+n_2}(A, b) = K_{n_1+n_2}(A, A^{-n_1}b);$$

  (Laurent polynomials)
- Rational Arnoldi: given $q_{n-1}(z)$ of degree $n-1$, o.n. basis for

  $$\{r(A)b \colon r(z) = p(z)/q_{n-1}(z),\ p \text{ any polynomial of degree} < n\}$$
  $$= q_{n-1}(A)^{-1}K_n(A, b) = K_n(A, q_{n-1}(A)^{-1}b);$$

  (rational functions with denominator $q_{n-1}(z)$)

Extended Arnoldi

Idea: as in Arnoldi, start with a continuation vector $v \in \operatorname{Im} V_j$. Compute

- $w = A^{-1}v$ if you want to add a negative power of z to your space $\ell(A)b$
- $w = Av$ if you want to add a positive power of z to your space of Laurent polynomials of the form $\ell(A)b$,

Only detail: the continuation vector must be a vector $v = \ell(A)b$ with a non-zero first or last coefficient (respectively).

A working choice: take the v_k from the last time k that you extended your space with a power of the same kind.

Putting together all orthogonalization relations yields a relation of the form

$$AV_{n+1}\underline{K}_n = V_{n+1}\underline{H}_n$$

This provides a 'recipe' to compute products Ax for every $x \in \operatorname{Im} V_n$.

Rational Arnoldi

Similar idea. At each step j, we have a denominator polynomial $q_{j-1}(z) = (z - \xi_1)(z - \xi_2) \dots (z - \xi_{j-1})$ of degree $j - 1$, and an orthonormal basis for

$$q_{j-1}(A)^{-1} K_j(A, b) = \left\{ r(A)b \colon r(z) = \frac{p(z)}{q_{j-1}(z)}, \deg(p) < j \right\}.$$

We extend the space by adding a new pole ξ_j (possibly repeated).

We start from a suitable continuation vector v (typically one of the v_i), compute $w = (A - \xi_j I)^{-1} v$, and orthogonalize it against all previous basis vectors v_i.

With small changes to the continuation formula ($w = (I - A/\xi_j)^{-1} A v$), one can allow for poles ξ_j equal to ∞ (i.e., "traditional" Arnoldi).

Arnoldi approximation

Once one has computed a suitable approximation space V_n,

$$f(A)b \approx V_n f(A_n) V_n^* b, \quad A_n = V_n^* A V_n.$$

Usually one takes the last pole ξ_n to be ∞ (a traditional Arnoldi step), so the last row of $\underline{K}_n$ is 0 and $A_n = H_n K_n^{-1}$.

Theorem [Güttel '13, Theorem 3.3]

Suppose that $q_{n-1}(A_n)$ is invertible, where $q_{n-1}(z) = (z - \xi_1) \dots (z - \xi_{n-1})$. Then,

$$V_n f(A_n) V_n^* b = \tilde{r}(A) b,$$

where $\tilde{r}(z) = p(z)/q_{n-1}(z)$ is a rational function that interpolates f in the spectrum of A_n.

Proof: similar to the previous lemma, but replacing b with $q_{n-1}(A)^{-1}b$.

Costs and benefits

Computational cost:

- Extended Arnoldi: typically computed with a (sparse) LU factorization of A, once, so that we can reuse it for each product with A^{-1}.
- Rational Arnoldi: typically computed with one sparse direct solver at each step. More degrees of freedom due to choice of poles. (Adaptively? From interpolation theory?)

Both are more expensive than Arnoldi.

Key issues: how much better is rational interpolation (for your f and A) than polynomial interpolation, so that the trade-off is convenient? How to choose good poles ξ_j?

Lots of current research on it. No details here. (I am not an expert myself!)
More detail in the review paper [Güttel '13].

Matlab examples

Using Rktoolbox by Güttel `http://guettel.com/rktoolbox/`.

1. Take $A =$ `randn(100) + 10*eye(100)`
2. Take poles $[-1, -2, \ldots, -10]$: approximates well the leftmost part of the spectrum, but what matters for the exponential is the rightmost part.
3. Take poles $[31, 32, \ldots, 40]$: much better.
4. Try classical Arnoldi (poles $[\infty \times 10]$), and extended Arnoldi (poles $[\infty \times 5, 0 \times 5]$).