The matrix exponential

We will now discuss some specific important matrix functions. First one:

$$\text{expm}(A) = I + A + \frac{1}{2}A^2 + \frac{1}{3!}A^3 + \dots.$$

Useful to recall it: the solution of the ODE initial value problem

$$\frac{d}{dt}v(t) = Av(t), \quad v(0) = v_0$$

is $v(t) = \text{expm}(At)v_0$.

Proof: we can differentiate term-by-term

$$v(t) = v_0 + tAv_0 + \frac{t^2}{2}A^2v_0 + \frac{t^3}{3}A^3v_0 + \dots.$$

How to compute expm(A)?

It is easy to come up with ways that turn out to be unstable.

[Moler, Van Loan, "Nineteen dubious ways to compute the exponential of a matrix", '78 & '03].

Example truncated Taylor series, $I + A + \frac{1}{2}A^2 + \frac{1}{6}A^3 \cdots + \frac{1}{k!}A^k$.

(See example in the previous slide set.)

Growth in matrix powers

The main problem in computing matrix power series: intermediate growth of coefficients.

Example Even on a nilpotent matrix, entries may grow.

$$A = \begin{bmatrix} 0 & 10 & & \\ & 0 & 10 & \\ & & 0 & 10 \\ & & & 0 \end{bmatrix}, A^2 = \begin{bmatrix} 0 & 0 & 100 & \\ & 0 & 0 & 100 \\ & & 0 & 0 \\ & & & 0 \end{bmatrix}, A^3 = \begin{bmatrix} 0 & 0 & 0 & 1000 \\ & 0 & 0 & 0 \\ & & 0 & 0 \\ & & & 0 \end{bmatrix}.$$

Typical behavior for non-normal matrices. Growth $+$ cancellation $=$ trouble.

(For normal matrices, $\|A^k\| = \|A\|^k = |\lambda_{\max}|^k$.)

"Humps"

Similarly, $\exp(tA)$ may grow for small values of t before 'settling down'.

Example

```
>> A = [-0.97 25; 0 -0.3];
>> t = linspace(0,20,100);
>> for i = 1:length(t); y(i) = norm(expm(t(i)*A)); end
>> plot(t, y)
```

For the same reason, it is also a bad idea to use an ODE solver on

$$X'(t) = AX(t), \quad X(0) = I;$$

Nice fact: explicit Euler produces $\exp(At) \approx (I + \frac{t}{n}A)^n$.

Padé approximants

Padé approximants to the exponential (in $x = 0$) are known explicitly.

Padé approximants to $\exp(x)$

$|\exp(x) - N_{pq}(x)/D_{pq}(x)| = O(x^{p+q+1})$, where

$$N_{pq}(x) = \sum_{j=0}^{p} \frac{(p+q-j)!p!}{(p+q)!j!(p-j)!} x^j,$$

$$D_{pq}(x) = \sum_{j=0}^{q} \frac{(p+q-j)!q!}{(p+q)!j!(q-j)!} (-x)^j.$$

$$\exp(A) \approx (D_{pq}(A))^{-1} N_{pq}(A).$$

The main danger comes from $D_{pq}(A)^{-1}$.

For large p, q, $D_{pq}(A) \approx \exp(-\frac{1}{2}A)$. $\kappa(D_{pq}(A)) \approx \frac{e^{-\frac{1}{2}\lambda_{\min}}}{e^{-\frac{1}{2}\lambda_{\max}}}$.

Backward error of Padé approximants

Are Padé approximants reliable when $\|A\|$ is small, at least?

Recall: perfect scalar approximation does not imply good matrix approximation.

Let $H = f(A)$, where $f(x) = \log(e^{-x}\frac{N_{pq}(x)}{D_{pq}(x)})$. H is a matrix function, so it commutes with A.
(Note that $e^{-x}\frac{N_{pq}(x)}{D_{pq}(x)} = 1 + O(x^{p+q+1})$, so the log exists for x sufficiently small).
One has $\exp(H) = \exp(-A)(D_{pq}(A))^{-1}N_{pq}(A)$, so

$$(D_{pq}(A))^{-1}N_{pq}(A) = \exp(A)\exp(H) = \exp(A+H)$$

(since A and H commute).

We can regard H as a sort of 'backward error': the Padé approximant $(D_{pq}(A))^{-1}N_{pq}(A)$ is the exact exponential of a certain perturbed matrix $A + H$.

Can one bound $\frac{\|H\|}{\|A\|}$?

Bounding $\|H\|$

$H = f(A)$, where $f(x) = \log(e^{-x} \frac{N_{pq}(x)}{D_{pq}(x)})$.
f is analytic, so $f(x) = c_1 x^{p+q+1} + c_2 x^{p+q+2} + c_3 x^{p+q+3} + \dots$.

$$H = f(A) = c_1 A^{p+q+1} + c_2 A^{p+q+2} + c_3 x^{p+q+3} + \dots$$

$$\|H\| \leq |c_1| \|A\|^{p+q+1} + |c_2| \|A\|^{p+q+2} + |c_3| \|A\|^{p+q+3} + \dots$$

All these quantities can be computed, explicitly or with Mathematica (but it's a lot of work).
Luckily, someone did it for us. For instance:

[Higham book '08, p. 244]

If $p = q = 13$ and $\|A\| \leq 5.4$, then $\frac{\|H\|}{\|A\|} \leq \mathbf{u}$ (machine precision).

Degree 13 achieves a good ratio between accuracy and number of required operations (with Paterson–Stockmayer + noting that numerator and denominator are of the form $p(x^2) \pm xq(x^2)$.)
Evaluating $N_{13,13}$ and $D_{13,13}$ requires 6 matmuls.

Scaling and squaring

What if $\|A\| > 5.4$? Trick: $\exp(A) = (\exp(\frac{1}{s}A))^s$.

Algorithm (scaling and squaring)

1. Find $s = 2^k$ such that $\|\frac{1}{s}A\| \leq 5.4$.
2. Compute $F = D_{13,13}(B)^{-1}N_{13,13}(B)$, where $D_{13,13}$ and $N_{13,13}$ are given polynomials and $B = \frac{1}{s}A$.
3. Compute F^{2^k} by repeated squaring.

This is Matlab's `expm`, currently (more or less — approximants of degree smaller than 13 are used in some cases).

Is scaling and squaring stable?

Note that 'humps' may still give problems: $\exp(B)$ may be much larger than $\exp(A) = \exp(B)^{2^k}$, leading to cancellation in the squares.

Is scaling and squaring stable for all matrices? Numerically it seems so, but no definitive answer.