The matrix sign function

$$\operatorname{sign}(x) = \begin{cases} 1 & \operatorname{Re} x > 0, \\ -1 & \operatorname{Re} x < 0, \\ \text{undefined} & \operatorname{Re} x = 0. \end{cases}$$

Suppose the Jordan form of A is reblocked as

$$A = \begin{bmatrix} V_1 & V_2 \end{bmatrix} \begin{bmatrix} J_1 & \\ & J_2 \end{bmatrix} \begin{bmatrix} V_1 & V_2 \end{bmatrix}^{-1},$$

where J_1 contains all eigenvalues in the LHP (left half-plane) and J_2 in the RHP. Then,

$$\operatorname{sign}(A) = \begin{bmatrix} V_1 & V_2 \end{bmatrix} \begin{bmatrix} -I & \\ & I \end{bmatrix} \begin{bmatrix} V_1 & V_2 \end{bmatrix}^{-1}.$$

$\operatorname{sign}(A)$ is always diagonalizable with eigenvalues ± 1. $\operatorname{sign}(A) \pm I$ gives the projections on the span of the eigenvectors in the RHP/LHP (unstable/stable invariant subspace).

Sign and square root

Useful formula: $\operatorname{sign}(A) = A(A^2)^{-1/2}$, where $A^{1/2}$ is the principal square root of A (all eigenvalues in the right half-plane), and $A^{-1/2}$ is its inverse.

Proof: consider eigenvalues, $\operatorname{sign}(x) = \frac{x}{(x^2)^{1/2}}$. (Care with signs.)

Theorem

If AB has no eigenvalues on $\mathbb{R}_{\leq 0}$ (hence neither does BA), then

$$\operatorname{sign}\begin{bmatrix} 0 & A \\ B & 0 \end{bmatrix} = \begin{bmatrix} 0 & C \\ C^{-1} & 0 \end{bmatrix}, \quad C = A(BA)^{-1/2}.$$

Proof (sketch) Use $\operatorname{sign}(A) = A(A^2)^{-1/2}$ (and then $\operatorname{sign}(A)^2 = I$). For instance,

$$\operatorname{sign}\begin{bmatrix} 0 & A \\ I & 0 \end{bmatrix} = \begin{bmatrix} 0 & A^{1/2} \\ A^{-1/2} & 0 \end{bmatrix}.$$

Conditioning

From the theorems on the Fréchet derivative, for a diagonalizable A

$$\kappa_{abs}(\text{sign}(A)) \leq \kappa_2(V) \frac{2}{\min_{\text{Re}\,\lambda_i<0,\,\text{Re}\,\lambda_j>0} |\lambda_i - \lambda_j|}$$

This bound tells only part of the truth:

- One one hand, computing $\text{sign}(A)$ is "better" than a full diagonalization: it is not sensitive to close eigenvalues that are far from the imaginary axis.
- On the other hand, when the invariant subspaces are badly separated the bound depends on higher powers of the separation (we will see in the following).

Schur-Parlett method

We can compute $\text{sign}(M)$ with a Schur decomposition. It makes sense to reorder it so that eigenvalues in the LHP come first: $\Lambda(A) \subseteq LHP$, $\Lambda(B) \subseteq RHP$.

$$Q^* M Q = \begin{bmatrix} A & C \\ 0 & B \end{bmatrix}, \quad Q^* f(M) Q = \begin{bmatrix} -I & Z \\ 0 & I \end{bmatrix}$$

where Z solves $AZ - ZB = -f(A)C + Cf(B) = 2C$.

The condition number of this Sylvester equation depends on $\text{sep}(A, B)$.

Schur-Parlett for the sign

1. Compute $M = QTQ^*$.
2. Reorder Schur decomposition so that eigenvalues in the LHP come first.
3. Solve Sylvester equation for Z.
4. $\text{sign}(M) = Q \begin{bmatrix} -I & Z \\ 0 & I \end{bmatrix} Q^*$.

(Matlab example)

Conditioning of the matrix sign

We can use this result to obtain a perturbation bound that explains what happens when $\mathrm{sep}(A, B)$ is small.

Recall our perturbation result for the Sylvester equation: given $M = \begin{bmatrix} A & C \\ 0 & B \end{bmatrix}$ and a perturbation $\tilde{M} = M + E$, there exists X with $\|X\|_F = O(\frac{\|E\|_F}{\mathrm{sep}(A,B)})$ such that

$$\begin{bmatrix} I & 0 \\ -X & I \end{bmatrix} (M + E) \begin{bmatrix} I & 0 \\ X & I \end{bmatrix} = \begin{bmatrix} \tilde{A} + \tilde{C}X & \tilde{C} \\ 0 & \tilde{B} - X\tilde{C} \end{bmatrix}.$$

For a sufficiently small perturbation, $\mathrm{sign}(RHS) = \begin{bmatrix} I & \tilde{Z} \\ 0 & I \end{bmatrix}$, where $\tilde{Z}$ solves

$$(\tilde{A} + \tilde{C}X)\tilde{Z} - \tilde{Z}(\tilde{B} - X\tilde{C}) = 2\tilde{C}.$$

Conditioning of the matrix sign

The coefficients of this equation are a perturbation of size $O(\frac{\|E\|_F}{\mathrm{sep}(A,B)})$ of those of $AZ - ZB = 2C \implies$

$$\|\tilde{Z} - Z\|_F = O(\frac{\|E\|_F}{\mathrm{sep}(A,B)^2})\|Z\|_F = O(\frac{\|E\|_F}{\mathrm{sep}(A,B)^3})\|C\|_F.$$

So

$$\|\mathrm{sign}(M+E) - \mathrm{sign}(M)\|_F = \left\| \begin{bmatrix} I & 0 \\ X & I \end{bmatrix} \begin{bmatrix} I & \tilde{Z} \\ 0 & I \end{bmatrix} \begin{bmatrix} I & 0 \\ -X & I \end{bmatrix} - \begin{bmatrix} I & Z \\ 0 & I \end{bmatrix} \right\|_F$$

is of order $O(\frac{\|E\|_F}{\mathrm{sep}(A,B)^3})$. The sign is extremely sensitive to perturbations, when $\mathrm{sep}(A,B)$ is small.
However, it has a stable invariant subspace $\begin{bmatrix} I \\ -X \end{bmatrix}$ at distance only $O(\frac{\|E\|_F}{\mathrm{sep}(A,B)})$ from that of M (the same as its condition number).
So the (numerically unstable) sign is a numerically stable method to compute the stable invariant subspace.

Newton for the matrix sign

Most popular algorithm:

Newton for the matrix sign

$\text{sign}(A) = \lim_{k\to\infty} X_k$, where

$$X_{k+1} = \frac{1}{2}(X_k + X_k^{-1}), \quad X_0 = A.$$

Suppose A diagonalizable: then we may consider the scalar version of the iteration on each eigenvalue λ:

$$x_{k+1} = \frac{1}{2}\left(x_k + \frac{1}{x_k}\right) = \frac{x_k^2 + 1}{2x_k}, \quad x_0 = \lambda.$$

This is Newton's method on $f(x) = x^2 - 1$, which justifies the name. It has fixed points ± 1, with (locally) quadratic convergence.

Convergence analysis of the scalar iteration

Theorem

The limit of $x_{k+1} = \frac{1}{2}\left(x_k + \frac{1}{x_k}\right)$ is $\operatorname{sign}(x_0)$ (for $\operatorname{Re}(x_0) \neq 0$).

Trick: change of variables (Cayley transform)

$$y = \frac{x-1}{x+1}, \text{ with inverse } x = \frac{y+1}{y-1}.$$

If $x \in$ RHP, then $|x+1| > |x-1| \implies y$ inside the unit disk.

If $x \in$ LHP, then $|x-1| > |x+1| \implies y$ outside the unit disk.
(It's a Padé approximant of $\exp(-2x)$, with the same property.)

If $y_k = \frac{x_k-1}{x_k+1}$ for each k, then $y_{k+1} = y_k^2$ (check).

$$x_0 \in \text{RHP} \implies |y_0| < 1 \implies \lim_{k\to\infty} y_k = 0 \implies \lim_{k\to\infty} x_k = 1;$$

$$x_0 \in \text{LHP} \implies |y_0| > 1 \implies \lim_{k\to\infty} y_k = \infty \implies \lim_{k\to\infty} x_k = -1.$$

Rational approximations of the step function

Let $g(x) = \frac{1}{2}(x + 1/x)$; then its iterates g^k are rational approximations of the step function $\mathrm{sign}(x)$ around -1 and 1.

```
>> syms x
>> g = 1/2*(x + 1/x);
>> g2 = simplify(subs(g, x, g))
>> g3 = subs(g2, x, g)
>> fplot(g, [-2,2])
>> axis([-2 2 -2 2]);
>> hold on
>> fplot(g2, [-2,2])
>> fplot(g3, [-2,2])
```

(They diverge badly around 0, though.)

Can you recognize the coefficients of `g2, g3`?

Theorem

$$g^{(k)}(x) = \frac{((1+x)^{2^k})_{\text{even}}}{((1+x)^{2^k})_{\text{odd}}} = \frac{(1+x)^{2^k} + (1-x)^{2^k}}{(1+x)^{2^k} - (1-x)^{2^k}}.$$

This is not-really-a-Padé approximant, because we are asking for it to be accurate (up to order 2^k) in both $+1$ and -1. However, it can be obtained from

$$\text{sign}(x) = \frac{(x^2)^{1/2}}{x}$$

by taking a Padé approximant of the principal square root $x^{1/2}$.

Convergence analysis of the matrix iteration

A modification of the same proof works in the matrix case.
Assume A has no eigenvalues on the imaginary axis; set

$$Y_k = (X_k - S)(X_k + S)^{-1}, \quad \text{with inverse } X_k = (I - Y_k)^{-1}(I + Y_k)S.$$

All the X_k are rational functions of A, so they commute with it and with S.
Analyzing eigenvalues: the inverse exists and $\rho(Y_k) < 1$.

$$Y_{k+1} = (X_k^{-1}(X_k^2 + I - 2SX_k))X_k(X_k^2 + I + 2SX_k)^{-1} = Y_k^2.$$

$Y_k \to 0$, hence $X_k \to S$.

The algorithm

1. $X_0 = A$.
2. Repeat $X_{k+1} = \frac{1}{2}(X_k + X_k^{-1})$, until convergence.

We really need to compute that matrix inverse (unusual in numerical linear algebra...)

Scaling

If $x_k \gg 1$, then

$$x_{k+1} = \frac{1}{2}\left(x_k + \frac{1}{x_k}\right) \approx \frac{1}{2}x_k,$$

and "the iteration is an expensive way to divide by 2" [Higham].
Same if $x_k \ll 1$: the iteration just multiplies by 2.

If all the eigenvalues of A are very large/small, then the first iterations just increase/reduce them via repeated division/multiplications by 2.

Trick: you can replace A with μA for a scalar $\mu > 0$: they have the same sign. Choose this μ so that eigenvalues ≈ 1.
(Once, or at each step.)

Scaling possibilities

Possibility 1: (determinantal scaling): choose $\mu = (\det A)^{-1/n}$, so that $\det A = 1$. Reduces "mean distance" from 1. Cheap to compute, since we already need to invert A.

Possibility 2: (spectral scaling): choose μ so that $|\lambda_{\min}(\mu A)\lambda_{\max}(\mu A)| = 1$. (We can use the power method to estimate them.)

Possibility 3: (norm scaling): choose μ so that $\sigma_{\min}(\mu A)\sigma_{\max}(\mu A) = 1$. (Again via the power method for $\sigma_{\min}$.)

Surprisingly, on a matrix with real eigenvalues Possibility 2 gives convergence in a finite number of iterations, if done at each step: the first iteration maps $\lambda_{\min}(A)$ and $\lambda_{\max}(A)$ to eigenvalues with the same modulus; then the second iteration adds a third eigenvalue with the same modulus...

Stability of the sign iterations

The (floating point) stability analysis is complicated. [Bai Demmel '98 and Byers Mehrmann He '97]

Even though the algorithm is only sums and inversions, it is difficult to assess and propagate the impact of numerical errors in the first steps (which are the most ill-conditioned ones).

TL;DR The stability analysis reflects the results of our conditioning analysis: while the sign in itself is unstable, it produces invariant subspaces as good (numerically) as those computed via a reordered Schur decomposition.

Inversion-free sign

Suppose that we are given M, N such that $A = M^{-1}N$. Can we compute sign(A) without inverting M? Yes.

$$\begin{aligned} X_1 &= \frac{1}{2}(A + A^{-1}) = \frac{1}{2}(M^{-1}N + N^{-1}M) \\ &= \frac{1}{2}M^{-1}(N + MN^{-1}M) \\ &= \frac{1}{2}M^{-1}(N + \hat{M}^{-1}\hat{N}M) \\ &= \frac{1}{2}M^{-1}\hat{M}^{-1}(\hat{M}N + \hat{N}M) \\ &= (\hat{M}M)\,\frac{1}{2}(\hat{M}N + \hat{N}M) =: M_1^{-1}\,N_1. \end{aligned}$$

assuming we can find $\hat{M}, \hat{N}$ such that $MN^{-1} = \hat{M}^{-1}\hat{N}$.

Then the same computations produce $M_2, N_2, M_3, N_3, \ldots$

Inversion-free sign

How to find $\hat{M}, \hat{N}$ such that $MN^{-1} = \hat{M}^{-1}\hat{N}$?

$\hat{M}M = \hat{N}N$, or $\begin{bmatrix}\hat{M} & \hat{N}\end{bmatrix}\begin{bmatrix}M \\ -N\end{bmatrix} = 0$. We can obtain $\hat{M}, \hat{N}$ from a kernel.

Computing this kernel can be much more accurate than inverting M and/or N, e.g.,

$$\begin{bmatrix}M \\ -N\end{bmatrix} = \begin{bmatrix}1 & 0 \\ 0 & \varepsilon \\ \varepsilon & 0 \\ 0 & 1\end{bmatrix}.$$

All this is a sort of 'linear algebra on pencils': we map $N - xM$ to $N_1 - xM_1$ (one final project on this).