Aitken delta^2 method

Sunday, September 13, 2020 8:33 AM

aka Aitken Extrapolation

TL;DR: If $x_n \to p$ linearly, Aitken extrapolation can accelerate the convergence, though suffers from numerical issues sometimes.

Steffensen's method is a variant/particular case w/ quadratic convergence

Derivation

Let $x_n \to p$ at a linear rate, so $\lim_{n\to\infty} \frac{x_{n+1}-p}{x_n-p} = \lambda < 1$

Ex: $x_n - p = \lambda^n$ converges linearly to 0

Let's assume $x_n - p = \lambda^n$, so $\frac{x_{n+1}-p}{x_n-p} = \frac{\lambda^{n+1}}{\lambda^n} = \frac{\lambda^n}{\lambda^{n-1}} = \frac{x_n-p}{x_{n-1}-p}$

i.e., $\frac{x_{n+1}-p}{x_n-p} = \frac{x_n-p}{x_{n-1}-p}$ or $(x_{n+1}-p)(x_{n-1}-p) = (x_n-p)^2$

Solve for p

$$x_{n+1}\cdot x_{n-1} - (x_{n+1}+x_{n-1})p + \cancel{p^2} = x_n^2 - 2x_n p + \cancel{p^2}$$

$$p = \frac{x_{n+1}\cdot x_{n-1} - x_n^2}{x_{n+1} - 2x_n + x_{n-1}} \quad (*)$$

what's the point?

- If $x_n - p = \lambda^n$ exactly, then from 3 terms in the sequence, we could solve for p exactly (and stop iterating)
- If $x_n - p \approx \lambda^n$, then the estimate for p in $(*)$ is probably more accurate than x_{n+1}

Practical Version

Rewrite and introduce nice notation

ex. $x_{n+1} - 2x_n + x_{n-1}$ is less stable than $(x_{n+1}-x_n) - (x_n - x_{n-1})$

Notation: "Forward difference operator Δ"

$$\Delta x_n = x_{n+1} - x_n$$

Can define

$$\Delta^2 x_n = \Delta\cdot(\Delta x_n) = \Delta(x_{n+1} - x_n)$$
$$= (x_{n+2} - x_{n+1}) - (x_{n+1} - x_n)$$

In this notation, (*) becomes

$$p \approx x_{n-1} - \frac{(\Delta x_{n-1})^2}{\Delta^2 x_{n-1}}.$$

Adjusting indices,

Algo: Aitken Extrapolation

Run iteration (x_n) until at least x_{n+2}

less stable but you can see it depends on x_{n+2}

Define $\hat{x}_n = x_n - \frac{(\Delta x_n)^2}{\Delta^2 x_n} \quad \left(= x_n - \frac{(x_{n+1} - x_n)^2}{x_{n+2} - 2x_{n+1} + x_n} \right)$

Useful but at some point numerical instabilities will stop it from helping

Thm (2.14) If $x_n \to p$ linearly, then $\hat{x}_n \to p$ "faster"

Steffensen

Works if $x_{n+1} = g(x_n)$ (fixed pt. iteration)

Slight modification of Aitken, and has quadratic convergence but still may have numerical issues.

Example of Aitken Acceleration

$p = 0.5839$ is true fixed pt

Generate x_n via $x_{n+1} = g(x_n)$ with $g(x) = .7 \cdot \cos(x)$

(#) = order to execute on a computer

(1) $x_0 = 0$ (arbitrary)

(2) $x_1 = g(0) = 0.7$, (3) $\Delta x_0 = 0.7 - 0 = 0.7$

(4) $x_2 = g(x_1) = 0.54$, (5) $\Delta x_1 = 0.54 - 0.7 = -.16$, (6) $\Delta^2 x_0 = -.16 - 0.7 = -.86$

(7) $\hat{x}_0 = x_0 - \frac{(\Delta x_0)^2}{\Delta^2 x_0} = 0 - \frac{.7^2}{-.86}$

$= \frac{.49}{.86} = \underline{.5697}$

(8) $x_3 = g(x_2) = 0.6$, (9) $\Delta x_2 = .6 - .54 = .06$,

(10) $\Delta^2 x_1 = .06 - (-.16)$
$= .22$

(11) $\hat{x}_1 = x_1 - \frac{(\Delta x_1)^2}{\Delta^2 x_1} = 0.7 - \frac{(-.16)^2}{.22}$

$= .7 - .1163$

$= \underline{.5836}$

Already pretty accurate

x_n	$\hat{x}_n$
0	.5697
.7	.5836
.54	⋮
.6	
⋮	

— postscript =

MOTIVATION Aitken Acceleration is similar to

comparison with a known sequence

(related to variance reduction, e.g. control variates)

Ex: Compute $S = \sum_{j=1}^{\infty} \frac{1}{\sqrt{j^4+1}}$ unlikely to have a closed form.

$$S = \sum_{j=1}^{\infty} a_j$$

Note $\tilde{S} = \sum_{j=1}^{\infty} \underbrace{\frac{1}{j^2}}_{b_j}$ does have a closed form ($\pi^2/6$)

and $b_j \approx a_j$, especially when $j \gg 1$.

i.e., $\lim \frac{a_j}{b_j} = 1$

then $S = S' + (S - S')$

$$= \sum b_j + \sum (a_j - b_j)$$

$$= \pi^2/6 + \underbrace{\sum_{j=1}^{\infty} \frac{1}{\sqrt{j^4+1}} - \frac{1}{j^2}}$$

Converges quickly, so can approximate w/ just a few terms

In this particular case, 4 digits of accuracy w/ just 5 terms.

without this trick, we'd need 20,000 terms to get 4 digits !!

Aitken acceleration is similar, using an exact linearly convergent sequence as a comparison.