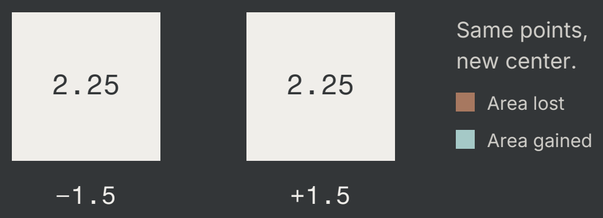
Mathscapes®

Updating variance without storing the data

An online variance calculation retains the count, mean and sum of squared deviations. The diagram connects residual geometry, exact arithmetic and retained state.

Squared deviations before and after an update

Computed example: [4, 7], then 5



AREA ADDED

Old values: −17/36 + 19/36 = 1/18

Add incoming value: (−1/3)² = 1/9

Total correction: 1/18 + 1/9 = 1/6

$$M_2' = M_2 + (x - \mu)(x - \mu')$$

STATE	BEFORE	AFTER
Count n	2	3
Mean μ	5.5	16/3
Total M_2	4.5	14/3
Variance	2.25	14/9

Equal area scale; dashed = previous residual square. Worked example. Iterflow: DOI 10.5281/zenodo.18610143.

CANCELLATION OF CROSS TERMS

Old deviations sum to zero. When the mean shifts by 1/6, their cross terms cancel; only the squared shift remains. The old points add 1/18; the new point adds 1/9. Together these give the residual product, 1/6. The recurrence is exact in real arithmetic, while floating-point rounding remains.

POPULATION AND SAMPLE VARIANCE

Population variance divides M_2 by three, giving 14/9. Sample variance divides by two, giving 7/3 under the usual sampling interpretation. The stored total is unchanged. This is an original computed illustration of an established algorithm, not a new estimator or a claim that all two-pass methods are unstable.

ITERFLOW / IMPLEMENTATION CONTEXT

Streaming implementation

A terminal pulls through filters, numeric stages and limits. Mean, variance, EWMA, covariance, correlation and z-score retain scalar state. Windows need O(k); collection and median need O(n). Array.shift() can make large-window extrema costly.

Reported performance

Limited filtering reports 7,818 versus 26 operations/s against eager arrays at one million inputs. Prefix correlation takes 0.66 versus 518 ms at 10,000 values. Mean and variance report 5.6× and 2.6× gains. EWMA and composed windows trail selected imperative loops. Algorithms and variance denominators differ.

Benchmark limitations

Six versus twelve lines illustrate concise composition. Benchmarks use Node 22, ARM64 Linux and synthetic data; they were not rerun. Hardware and seed details are incomplete. Types do not validate runtime values. Z-score starts with two NaNs. Async streams, quantiles and fusion remain proposed.