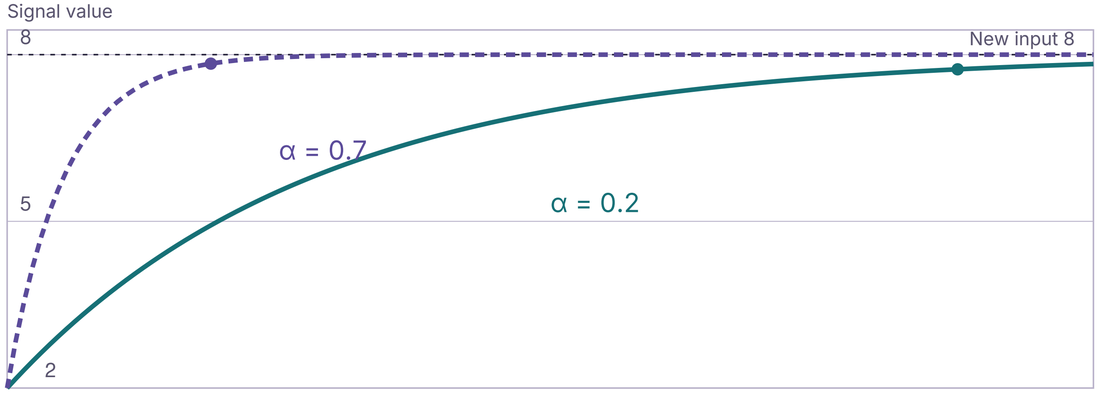
Mathscapes®

EWMA response, memory and noise

The exponentially weighted moving average (EWMA) assigns progressively less weight to older observations. Compare α = 0.2 in teal with α = 0.7 in violet. Both receive the same step from 2 to 8; the parameter controls response time and noise attenuation.

01 Response to a step input

Same input: 2 → 8. Only the new-input weight α changes.

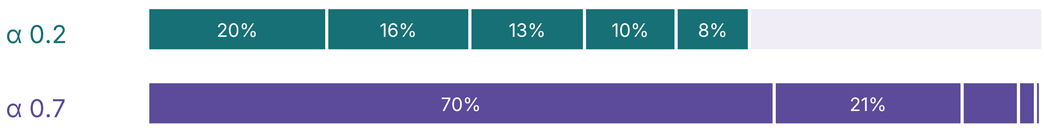


The first update reaches 6.2 at α = 0.7, but only 3.2 at α = 0.2. The faster response also admits more input noise.

Computed · DOI 10.5281/zenodo.18610143

02 Weight assigned to past inputs

Each strip divides total steady-state weight into past inputs.



Current → older inputs; pale tail = lag 5 and beyond.

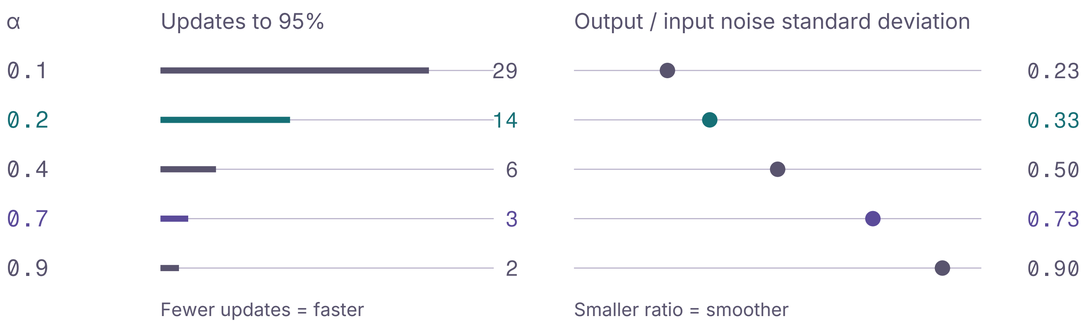
Computed · DOI 10.5281/zenodo.18610143 · Weight at lag k = α(1 − α)^k.

03 Updates required for 95% response

Updates needed to close at least 95% of the original gap.

Computed · DOI 10.5281/zenodo.18610143 · One square = one update; remaining gap = (1 − α)^k.

04 Response time and noise attenuation



Independent steady-state noise only; ratio = √[α / (2 − α)].

Computed · DOI 10.5281/zenodo.18610143

Recurrence and initialization

EWMA uses s(t) = αx(t) + (1 - α)s(t-1), initialized at the first observation. The first new outputs are 3.2 and 6.2. A lag-k input contributes α(1 - α)^k; infinite steady-state weights sum to one.

Response time and noise variance

The remaining step fraction is (1 - α)^k. Reaching 95% takes 14 updates at α = 0.2 and three at 0.7. Independent noise has output variance σ²α/(2 - α). Correlation and warm-up change this analysis.

Implementation benchmarks

Iterflow composes lazy transforms and scalar-state statistics. Manuscript EWMA throughput is 213 vs 951 operations/s for a loop; limited filtering is 7,818 vs 26 for eager arrays. Synthetic Node 22/ARM64 benchmarks, not rerun; these are not noise trials.

Computed signal analysis; reported benchmarks provide separate implementation context.