

jsharpe: Sharpe-Ratio Inference in Practice, and Where the Corrections Do Not Reach

Thomas Schmelzer*

September 30, 2026

Abstract

`jsharpe` implements the standard toolkit for deciding whether a Sharpe ratio is real: its asymptotic variance under non-Gaussian and autocorrelated returns, the Probabilistic Sharpe Ratio, the minimum track record length, critical values and power, and the multiple-testing machinery — Bonferroni, Šidák, Holm and a false-discovery-rate calibration — that a strategy-screening exercise needs. This paper documents what the package computes and checks it against simulation and against the published formulas it implements. Three things hold well and three do not. The variance estimator is accurate to 3.4% at $T = 100$ and 1.4% at $T = 1000$; the FDR calibration is internally exact, reproducing its own target q to five decimals; the FWER adjustments and the effective-rank statistic behave as specified. Against that: the package’s headline non-Gaussian correction is *exactly inert* at its own default null $SR_0 = 0$, because the variance is evaluated under the null, and we measure the cost — a nominally 5% test rejects 8.2% of the time on severely skewed returns. The minimum track record length omits both the leading 1 and the observed-moment terms of the published formula, understating the requirement by a factor of 4.6 at $SR = 1$ with skew -1.5 and kurtosis 8. And the trials parameter K inverts the multiple-testing correction it is named for: raising it from 1 to 1000 lowers the critical Sharpe ratio from 0.164 to 0.058, because two functions in the package model the variance of a maximum of K draws in opposite directions — one correctly (it falls with K), one as a $1 + \log K$ inflation, off by a factor of 39 at $K = 200$. Separately, the cluster-count selector returns $k = n - 1$ — one pair and 58 singletons — on a matrix with ten true blocks, because a singleton silhouette is scored 1 rather than the conventional 0. We give the one-line repairs for each and the simulation harness that would have caught them.

1 Introduction

A Sharpe ratio (Sharpe, 1966, 1994) is a ratio of two estimates, and the literature on how badly that fact is usually ignored is now larger than the literature on the statistic itself. Lo (2002) gave the asymptotic distribution of the estimator, including under serial correlation; Mertens (2002) and Opdyke (2007) extended it to non-Gaussian marginals; Bailey and López de Prado (2012) recast the resulting test as a probability, the Probabilistic Sharpe Ratio, and derived the track record length a claim requires; Bailey and López de Prado (2014) and Harvey and Liu (2015) added the correction for having looked at many strategies before picking one.

`jsharpe` is a small implementation of that toolkit — roughly 1,500 lines over six modules, depending only on NumPy (Harris et al., 2020) and SciPy (Virtanen et al., 2020). It is the kind of package whose value depends entirely on correctness: the functions are short, their inputs are a handful of scalars, and nobody inspects the output for plausibility because the output is the plausibility check. That makes it worth testing against simulation rather than against itself.

*<https://github.com/tschm/jsharpe>

This paper does that. Section 2 states the estimators as the package computes them, in the notation of the sources, so the divergences later are visible rather than asserted. Section 4 tests each against Monte Carlo or against the closed form it claims to implement. The results split cleanly: the core variance estimator and the FDR calibration are right, and three of the corrections that motivate the package do not reach the code paths a user actually calls. Section 5 states each repair as a diff-sized change, because all three are.

2 The estimators

Let $r_1, \dots, r_T$ be returns with mean μ , standard deviation σ , skewness γ_3 , non-excess kurtosis γ_4 and first-order autocorrelation ρ , and let $\widehat{\text{SR}} = \hat{\mu}/\hat{\sigma}$.

2.1 Variance of the estimator

The package computes

$$V(\text{SR}, T) = \frac{a - b\gamma_3\text{SR} + c\frac{\gamma_4-1}{4}\text{SR}^2}{T} \cdot \text{Var}[M_K], \quad (1)$$

where $a = 1 + 2B$, $b = 1 + B + C$, $c = 1 + 2C$ with $B = \rho/(1 - \rho)$ and $C = \rho^2/(1 - \rho^2)$ carrying the autocorrelation adjustment, and $\text{Var}[M_K]$ is the variance of the maximum of K standard normals, computed by Gauss–Hermite quadrature. At $\rho = 0$ and $K = 1$, (1) reduces to $(1 - \gamma_3\text{SR} + \frac{\gamma_4-1}{4}\text{SR}^2)/T$, which is the Bailey and López de Prado (2012) form, and under Gaussian returns ($\gamma_3 = 0$, $\gamma_4 = 3$) to $(1 + \text{SR}^2/2)/T$, the Lo (2002) result. The $\text{Var}[M_K]$ factor has no counterpart in those sources; it is examined in Section 4.4.

2.2 Tests built on it

Every downstream quantity is a normal tail probability with (1) as its variance, evaluated *under the null*:

$$\text{SR}_c = \text{SR}_0 + z_{1-\alpha}\sqrt{V(\text{SR}_0, T)}, \quad (2)$$

$$\text{PSR} = \Phi\left(\frac{\text{SR} - \text{SR}_0}{\sqrt{V(\text{SR}_0, T)}}\right), \quad (3)$$

$$\text{minTRL} = V(\text{SR}_0, 1) \left(\frac{z_{1-\alpha}}{\text{SR} - \text{SR}_0}\right)^2, \quad (4)$$

with power $1 - \Phi((\text{SR}_c - \text{SR}_1)/\sqrt{V(\text{SR}_1, T)})$. The choice to evaluate the variance at SR_0 rather than at the observed SR is the hinge of Section 4.2: it is defensible for (2) and (3), which are null-hypothesis statements, and it is a departure from the published form of (4), whose reference version is

$$\text{minTRL}^* = 1 + \left(1 - \gamma_3\text{SR} + \frac{\gamma_4-1}{4}\text{SR}^2\right) \left(\frac{z_{1-\alpha}}{\text{SR} - \text{SR}_0}\right)^2. \quad (5)$$

2.3 Screening many strategies

For a pool of M candidates the package offers the three standard family-wise adjustments — Bonferroni $\min(1, Mp)$, Šidák $1 - (1 - p)^M$ (Šidák, 1967), and Holm’s step-down procedure (Holm, 1979) in either variant — and a false-discovery-rate calibration (Benjamini and Hochberg, 1995; Storey, 2002) in the two-normal mixture form. In the latter, a strategy is

drawn from $\mathcal{N}(\text{SR}_1, \sigma_1^2)$ with prior probability p_{H_1} and from $\mathcal{N}(\text{SR}_0, \sigma_0^2)$ otherwise, and the critical value solves

$$\mathbb{P}[H_0 \mid X > c] = \left(1 + \frac{\bar{\Phi}((c - \text{SR}_1)/\sigma_1)}{\bar{\Phi}((c - \text{SR}_0)/\sigma_0)} \frac{p_{H_1}}{1 - p_{H_1}} \right)^{-1} = q, \quad (6)$$

with $\sigma_0^2 = V(\text{SR}_0, T)$ and $\sigma_1^2 = V(\text{SR}_1, T)$ and $\bar{\Phi}$ the normal survival function. Separately, the expected best of K trials is the extreme-value approximation of Bailey and López de Prado (2014),

$$\mathbb{E}[\max_K] \approx \sigma \left[(1 - \gamma_e) \Phi^{-1} \left(1 - \frac{1}{K} \right) + \gamma_e \Phi^{-1} \left(1 - \frac{1}{Ke} \right) \right], \quad (7)$$

γ_e the Euler–Mascheroni constant.

2.4 Two utilities

The package also carries a correlation-structure module and a portfolio helper. `effective_rank` is $\exp(H)$ for H the entropy of the normalised eigenvalues of a correlation matrix (Roy and Vetterli, 2007), and `number_of_clusters` selects k by maximising the ratio of the mean silhouette score (Rousseeuw, 1987) to its standard deviation, the criterion of López de Prado (2018). `minimum_variance_weights_for_correlated_assets` solves for the minimum-variance portfolio under the assumption that the correlation matrix is *equicorrelated* at the average of its off-diagonal entries, so that the inverse is available in closed form by Sherman–Morrison — the constant-correlation structure of Elton et al. (1976).

3 Implementation

The package is pure functions over scalars and small arrays, with no state and no configuration. Three details of that surface matter for what follows.

Every moment argument is optional, and defaults to Gaussian. $\gamma_3 = 0$, $\gamma_4 = 3$, $\rho = 0$, $K = 1$. A user who supplies none of them gets the textbook Gaussian iid test, which is the right default. A user who supplies them expects them to matter.

SR_0 defaults to 0 everywhere it appears. It is the natural null — “does this strategy make money” — and it is what the package’s own documentation and worked example use.

The quadrature is shared. $\text{Var}[M_K]$ and $\mathbb{E}[M_K]$ come from a 200-node Gauss–Hermite rule constructed once at import. We checked it against simulation: at $K = 10$ the quadrature gives $\text{Var}[M_{10}] = 0.3443$ against 0.3448 from 4×10^5 draws, and at $K = 200$, 0.1607 against 0.1600. The quadrature is not the problem anywhere below.

4 Results

4.1 The variance estimator is sound

Table 1 confirms (1) as an asymptotic result behaving asymptotically. At $T = 1000$ it is accurate to 1.4%, at $T = 100$ to 3.4%, and at $T = 24$ it understates the true variance by 10.1%. The last figure deserves a note because $T = 24$ is the default in the package’s FDR routine and the order of magnitude of its documented examples: at two years of monthly data, the asymptotic variance is a tenth too small and every test built on it is correspondingly liberal, before any of the issues below.

Table 1: Variance of $\widehat{SR}$ from (1) against 2×10^4 simulations of iid Gaussian returns with true $SR = 0.5$.

T	Simulated	Equation (1)	Error
24	0.052147	0.046875	-10.1%
100	0.011650	0.011250	-3.4%
1000	0.001110	0.001125	+1.4%

Table 2: The moment arguments at $SR_0 = 0$, for $SR = 0.5$ and $T = 100$. All four rows are bit-identical. The last two columns repeat the calculation at $SR_0 = 0.3$, where the arguments do reach the result.

γ_3	γ_4	$V(SR_0, T)$	PSR	minTRL	PSR ($SR_0=0.3$)	minTRL ($SR_0=0.3$)
0.0	3.0	0.01000000	0.9999997133	10.822174	0.974795	70.68
-1.5	8.0	0.01000000	0.9999997133	10.822174	0.942654	108.73
0.0	20.0	0.01000000	0.9999997133	10.822174	—	—
2.0	3.0	0.01000000	0.9999997133	10.822174	—	—

4.2 The non-Gaussian correction is inert under the default null

The package’s first advertised feature is the correction for skewness and excess kurtosis. Substituting $SR = SR_0$ into (1), as (2)–(4) all do, gives

$$V(SR_0, T)|_{SR_0=0} = \frac{a}{T} \cdot \text{Var}[M_K], \quad (8)$$

in which γ_3 and γ_4 do not appear. At the default null they are multiplied by zero. This is not an approximation that degrades; it is exact cancellation.

Table 2 shows the cancellation at work. A user testing a $SR = 0.5$ track record of $T = 100$ observations against $SR_0 = 0$ gets the same PSR to ten decimal places whether the returns are Gaussian or have kurtosis 20. Shifting the null to $SR_0 = 0.3$ restores the correction: the PSR falls from 0.975 to 0.943 and the required track record grows by 54%.

The autocorrelation argument is the exception. It enters through $a = 1 + 2B$, which does not vanish at $SR_0 = 0$, so ρ works as documented at every null: raising it from 0 to 0.5 lifts the minTRL at $SR = 0.5$ from 10.8 to 32.5 and the critical Sharpe ratio at $T = 100$ from 0.164 to 0.285.

What the inertness costs. Whether it matters depends on whether the correction was doing anything. It was.

Table 3 is the practical statement of the problem. Under Gaussian returns the test is calibrated: it rejects 5.04% of the time at a nominal 5%. As the returns acquire the left skew and fat tails that hedge fund returns are notorious for, the test drifts liberal, reaching 8.18% — 64% more false discoveries than advertised — on the generator’s severe setting. Evaluating the same moment polynomial at the observed Sharpe ratio, which is the one-line change of Section 5, recovers about half of the gap, to 6.86%. It does not close it, because at $T = 100$ the sample skewness and kurtosis are themselves badly estimated, and no plug-in correction can fix that.

The documented example. The package’s own worked example is a $T = 36$ series with $\widehat{SR} = 0.8150$, sample skewness -0.581 and kurtosis 2.188. Its PSR is reported as effectively 1, and it is: 0.999999, identically with and without the moment arguments. The published

Table 3: Rejection rate of a nominally 5% one-sided test of $H_0 : \text{SR} = 0$ when H_0 is true, over 2×10^5 series of $T = 100$ from the package’s own non-Gaussian generator. “Package” is (3) with the observed moments supplied; “observed-moment form” evaluates the same moment polynomial at the observed SR instead, and with $T - 1$ in place of T , as Bailey and López de Prado (2012) prescribe. Skewness and kurtosis are the sample averages.

Distribution	Mean skew	Mean kurt.	Package	Observed-moment form
gaussian	−0.001	2.940	5.04%	4.84%
mild	−0.773	5.088	5.71%	4.90%
moderate	−1.410	8.377	7.26%	5.75%
severe	−1.789	11.567	8.18%	6.86%

Table 4: `minimum_track_record_length` against (5), $\text{SR}_0 = 0$, $\alpha = 0.05$. The package’s value does not depend on the moments; the reference value does.

SR	γ_3	γ_4	Package	Equation (5)
0.3	0.0	3.0	30.06	32.41
0.3	−1.5	8.0	30.06	49.32
0.5	0.0	3.0	10.82	13.17
0.5	−1.5	8.0	10.82	24.67
1.0	0.0	3.0	2.71	5.06
1.0	−1.5	8.0	2.71	12.50

form gives 0.999904 — the same conclusion. The `minTRL` is where the example diverges: the package returns 4.07 observations, (5) returns 7.81, and the honest reading of a 36-month record is that the second number is the one to quote.

4.3 The minimum track record length is not the published one

Table 4 separates two divergences from (5). The first is the missing leading 1, which costs one observation and matters only where the answer is small — but the answer *is* small exactly where the function is most likely to be quoted, and at $\text{SR} = 1$ the package’s 2.71 against a reference 5.06 is a factor of 1.9 from that term alone. The second is the inertness of Section 4.2, which under a Gaussian null removes the moment polynomial entirely: at $\text{SR} = 1$ with skew -1.5 and kurtosis 8, the reference requirement is 12.50 observations and the package reports 2.71, understating it by a factor of 4.6.

A minimum track record length is a statement made to somebody else — an allocator, a risk committee — about how much evidence exists. Understating it by a factor of four in exactly the fat-tailed, negatively skewed case that motivates computing it at all is the most consequential of the divergences in this paper, and the cheapest to repair.

4.4 The two models of a maximum of K disagree

The package contains two functions concerned with the maximum of K Sharpe ratios. `moments_Mk` computes $\mathbb{E}[M_K]$ and $\text{Var}[M_K]$ for K standard normals by quadrature, and `variance_of_the_maximum_of_k_S` returns $\sigma^2(1 + \log K)$, documented as a “logarithmic variance inflation factor” that “increases monotonically with K ” because “selection across a larger pool increases the uncertainty of the selected estimate”.

Table 5 settles it. The variance of the maximum of K iid normals *falls* with K — the maximum of many draws is pinned near the upper tail and is therefore less variable, a standard

Table 5: Variance and expectation of the maximum of K standard normals: 4×10^5 simulations against the two package functions. The quadrature is right; the $1 + \log K$ inflation has the wrong sign of dependence on K .

K	Var[M_K]			$\mathbb{E}[M_K]$		
	Simulated	moments_Mk	$1 + \log K$	Simulated	moments_Mk	Eq. (7)
1	1.0028	1.0000	1.0000	0.0002	0.0000	—
2	0.6815	0.6817	1.6931	0.5649	0.5642	0.5198
10	0.3448	0.3443	3.3026	1.5383	1.5388	1.5746
50	0.2159	0.2157	4.9120	2.2489	2.2491	2.2763
200	0.1600	0.1607	6.2983	2.7471	2.7460	2.7655

Table 6: Effect of the trials parameter K on the tests, at $T = 100$, $\text{SR}_0 = 0$, $\alpha = 0.05$. Larger K makes every test easier to pass.

K	Critical SR	Power at $\text{SR}_1 = 0.3$	PSR at $\text{SR} = 0.2$
1	0.164485	0.9075	0.977250
10	0.096521	0.9881	0.999673
100	0.070634	0.9990	0.999998
1000	0.057794	0.9999	1.000000

order-statistic result (David and Nagaraja, 2003) — from 1.00 at $K = 1$ to 0.16 at $K = 200$. The quadrature reproduces this to three decimals. The $1 + \log K$ function returns 6.30 where the truth is 0.16: wrong by a factor of 39, and wrong in direction. The prose in its docstring describes a real phenomenon — selection does make the selected estimate less trustworthy — but that phenomenon lives in $\mathbb{E}[M_K]$, not $\text{Var}[M_K]$, and (7) already captures it, agreeing with simulation to within 0.02 at $K \geq 10$.

Which makes K a discount, not a penalty. The consequence falls on the tests, because (1) — not the $1 + \log K$ function — is what K actually reaches, and it enters as a *multiplier* $\text{Var}[M_K] < 1$.

Table 6 is the trap. A user who screened a thousand strategies, read the documentation for K — “number of strategies for multiple testing adjustment” — and passed $K = 1000$ gets a critical Sharpe ratio of 0.058 instead of 0.164, and a PSR of 1.000000 for an observed 0.2. The parameter named for the multiple-testing problem makes the multiple-testing problem worse. It is not that the correction is too weak; its sign is reversed.

The fix is not to change $\text{Var}[M_K]$, which is correct as a variance. It is that a deflation for selection shifts the null — replace SR_0 by $\text{SR}_0 + \mathbb{E}[\max_K]$, which is what (7) exists for and what Bailey and López de Prado (2014) prescribe — rather than rescaling the variance in place. Used that way, the two ingredients the package already has combine correctly; used as K , they cancel the correction and then reverse it.

4.5 What holds

Table 7 checks the FDR routine two ways: the significance level it reports is recovered exactly from the threshold it returns, and its own re-derivation of q matches the requested q to five decimals across the range. The mixture in (6) is solved by bisection on $[-10, 10]$ with the two degenerate cases — no root in the interval, and no root at all when $\sigma_0 \gg \sigma_1$ with small q — returning $-\infty$ and NaN respectively rather than a spurious number. The trade-off it exposes

Table 7: FDR calibration at $\text{SR}_0 = 0$, $\text{SR}_1 = 0.5$, $p_{H_1} = 0.05$, $T = 24$. α *direct* recomputes $\bar{\Phi}((\text{SR}_c - \text{SR}_0)/\sigma_0)$ from the returned threshold; $\hat{q}$ is the routine’s own re-derivation of the target.

q	SR_c	α	α direct	Power	$\hat{q}$
0.05	0.6563	0.00065	0.00065	0.2351	0.05000
0.10	0.5872	0.00201	0.00201	0.3435	0.10000
0.25	0.4792	0.00944	0.00944	0.5382	0.25000
0.50	0.3597	0.03903	0.03903	0.7415	0.50000
0.75	0.2181	0.14267	0.14267	0.9036	0.75000

Table 8: `minimum_variance_weights_for_correlated_assets` against the exact minimum-variance solution, on one-factor correlations $C_{ij} = b_i b_j$ with loadings drawn uniformly from the stated range and volatilities from $[0.08, 0.30]$. 2000 draws per row; “penalty” is the mean excess of the achieved variance over the minimum.

n	Loadings	Mean ρ	SD(ρ)	Mean $\max_i \Delta w_i $	Variance penalty
3	[0.55, 0.65]	0.360	0.013	0.0076	0.03%
3	[0.30, 0.90]	0.360	0.079	0.0544	1.43%
5	[0.55, 0.65]	0.361	0.018	0.0152	0.14%
5	[0.30, 0.90]	0.360	0.111	0.1018	5.91%
10	[0.55, 0.65]	0.360	0.021	0.0185	0.43%
10	[0.30, 0.90]	0.358	0.132	0.1281	17.80%
25	[0.55, 0.65]	0.360	0.023	0.0139	0.99%
25	[0.30, 0.90]	0.360	0.144	0.1005	38.67%
25	[0.00, 0.95]	0.226	0.190	0.1171	79.77%

is the real one: controlling the false-discovery rate at 5% against a 5% prior leaves the test with only 24% power.

The family-wise adjustments are textbook and behave so. On $p = (0.001, 0.008, 0.02, 0.04, 0.30)$: Bonferroni gives $(0.005, 0.040, 0.100, 0.200, 1.000)$, Šidák the uniformly smaller $(0.005, 0.039, 0.096, 0.185, 0.832)$, and Holm the smaller still $(0.005, 0.032, 0.060, 0.080, 0.300)$, monotone in the sorted order and never below the raw values. The effective rank recovers block structure as it should: on a 60-strategy correlation matrix built from 10 latent factors with noise 0.1 it returns 6.81, and it rises smoothly with the noise (6.51 at noise 0.05, 7.83 at 0.2, 13.9 at 0.5, 28.7 at 1.0, 49.7 at 3.0) towards the full dimension, rather than jumping. It under-counts the latent factors — the blocks are unequal in size and the weakest contribute little entropy — but it is monotone in the right direction and needs no partition to compute.

The autocorrelated generator delivers roughly what it promises. Requesting $\rho = 0.2$ yields a measured 0.171 and $\rho = 0.5$ yields 0.440, the shortfall being the expected consequence of applying a nonlinear marginal transform to a Gaussian AR(1): the copula’s rank dependence survives, the Pearson autocorrelation is compressed by 12 to 15%. The target marginals themselves come through — skewness around -2.4 and kurtosis around 16 for the severe setting, at all three ρ .

4.6 The constant-correlation portfolio, quantified

The equicorrelation assumption is documented, so Table 8 is a calibration rather than a criticism. It is exact where it claims to be: for $n = 2$, where the average off-diagonal correlation *is* the only correlation, and for genuinely equicorrelated matrices at any n , both to machine

Table 9: `number_of_clusters` on correlation matrices generated with a known block structure ($T = 200$ observations, noise 0.1), matrix seed 0 and k -means seed 1. The last two columns are the quality score at the true k and at the near-singleton partition $k = n - 1$.

n	True k	Effective rank	Selected k	Quality at true k	Quality at $k = n - 1$
20	4	2.93	3	1.20	6.61
40	5	4.41	39	2.06	69.00
60	10	6.81	59	1.42	68.51
100	10	9.06	99	1.45	7.31

precision. Away from that, the cost is governed by the dispersion of the correlations and grows with n : at $n = 25$ with correlations spread around 0.36 with a standard deviation of 0.14, the portfolio it returns carries 39% more variance than the minimum. It is a shrinkage estimator without the shrinkage-intensity parameter — Ledoit and Wolf (2004) with λ pinned at 1 — and given how sensitive exact mean-variance solutions are to their inputs (Best and Grauer, 1991; Markowitz, 1952), that is a defensible default and a bad surprise. We note it because the docstring says “assumes” where the results say “costs 39%”. Neither this function nor the exact solution is long-only: both hold short positions in essentially every draw at $n \geq 10$.

4.7 The cluster-count criterion degenerates

Table 9 shows the criterion failing in a specific and diagnosable way. For $n = 60$ with ten true blocks it returns $k = 59$: a partition of one pair and 58 singletons. The quality profile explains it — 3.87 at $k = 2$, 3.34 at $k = 5$, 1.42 at the true $k = 10$, a trough of 0.71 at $k = 20$, then a climb through 4.26 at $k = 58$ to 68.51 at $k = 59$. The selection is not an artefact of one k -means initialisation: $k = 59$ is chosen for all five initialisation seeds we tried. There are two problems compounding.

The first is the criterion. The mean silhouette divided by its standard deviation is unbounded: it diverges as the spread of the silhouette scores goes to zero, whatever the mean is, so a partition in which all points score alike wins regardless of whether the scores are good. The implementation guards against exactly zero spread and nothing more.

The second makes the first reachable. In the silhouette computation, a singleton cluster is given $a(i) = 0$, so its score is $s(i) = (b(i) - 0)/b(i) = 1$ — the maximum possible. The convention since Rousseeuw (1987) is $s(i) = 0$ for a singleton, and it exists precisely to stop a partition into singletons from scoring perfectly. With 58 of 60 points scoring exactly 1, the spread collapses and the ratio explodes. The effective rank on the same matrices, which needs no partition at all, returns 6.81 and 9.06 against a true 10.

Only the smallest case escapes, and not by much: at $n = 20$ the criterion picks $k = 3$ against a true 4, with the degenerate partition the runner-up at 6.61 against 7.02. From $n = 40$ upwards the degenerate partition wins outright. The magnitudes in the last column vary with the k -means seed — a quality of 69 is a near-zero denominator, and near-zero denominators are noisy — but the selected k does not.

5 Repairs

All three substantive findings are small changes.

1. **Evaluate the moment polynomial at the observed Sharpe ratio** in (3) and (4), keeping the null variance for (2) where it belongs. This is a change of argument, from `sharpe_ratio_variance(SR0, ...)` to `sharpe_ratio_variance(SR, ...)`, and it moves the calibration of Table 3 from 8.18% to 6.86%.

2. **Add the leading 1** to (4), making it (5). One character, a factor of 1.9 at $SR = 1$.
3. **Do not route K through the variance.** Either delete the $\text{Var}[M_K]$ factor from (1) and apply selection as a shift of SR_0 by (7), or document K as what it currently is, which is not a multiple-testing correction. Replace the $1 + \log K$ function with `moments_Mk(K)` [2], which the package already computes correctly.
4. **Score singletons at zero** in the silhouette, and bound the cluster-quality criterion — or cap `max_clusters` well below $n - 1$. The first is the standard convention; the second is the cheaper guard.

None of these is hard to test. The simulations in Tables 1, 3 and 5 took seconds to run, and each of them fails loudly against the current code. A package whose entire output is a calibration claim is a package whose test suite should contain a calibration check: draw from a known null, count the rejections, assert the count is near nominal. That test would have caught findings one and three, and its absence is why a documented feature could be exactly inert without anything going red.

6 Limitations

Every number here is a simulation or a closed form, so the usual caveat about sample specificity does not apply — but the choice of null distributions does. The calibration in Table 3 uses the package’s own two-component Gaussian mixture, which reaches skewness -1.8 and kurtosis 11.6 in sample at $T = 100$. Real hedge fund returns are often more extreme and, more importantly, serially dependent in ways an iid mixture is not; we did not test calibration under autocorrelation, where the ρ adjustment is active and may behave differently.

The comparison of (4) against (5) treats the latter as the reference. That is a choice of authority, not a proof: (5) is the form in Bailey and López de Prado (2012), and we have checked its internal consistency with (3) rather than its own calibration.

Table 8 uses one-factor correlations, which are a well-conditioned and optimistic family. Empirical correlation matrices estimated from short samples are neither, and the variance penalties there would be larger and noisier. The clustering results use one generator with equal-sized latent factors; the degeneracy of Section 4 is a property of the criterion rather than of that generator, but the n at which it takes over is not.

Finally, this paper says nothing about the two functions in the package that are conditional on inputs we did not vary systematically — the observed-FDR routine and the power calculation — beyond noting that both are algebraic consequences of (3) and (2) and therefore inherit Section 4.2 exactly.

7 Conclusion

The half of `jsharpe` that computes distributions is right: the asymptotic variance matches simulation to a few percent for $T \geq 100$, the order-statistic quadrature is accurate to three decimals, the extreme-value approximation for the expected best of K trials tracks simulation within 0.02, the FDR calibration is internally exact, and the FWER adjustments and effective rank are textbook.

The half that turns those distributions into a verdict has three defects, and they share a shape: a correction is available, is documented, is passed by the user, and does not reach the answer. Skewness and kurtosis are exactly cancelled by evaluating the variance under a zero null, and the cost is a nominally 5% test rejecting 8.2% of the time on fat-tailed returns. The minimum track record length drops the same terms and the leading constant, understating

a fat-tailed requirement fourfold. And the trials parameter, which exists to punish having searched, rewards it instead, because a variance of a maximum was used where a shift of a mean was needed.

The pattern is worth naming, because it is not specific to this package. Each defect is invisible to a unit test that checks a function returns a float in $[0,1]$, and each is caught immediately by a test that draws from a known null and counts rejections. For a library whose whole purpose is to say how often we should expect to be wrong, that is the test that matters.

A Reproducibility

Everything above runs against the package in this repository with NumPy and SciPy. The inertness of Section 4.2:

```
from jsharpe import probabilistic_sharpe_ratio, minimum_track_record_length
for g3, g4 in [(0.0, 3.0), (-1.5, 8.0), (0.0, 20.0), (2.0, 3.0)]:
    print(g3, g4,
          probabilistic_sharpe_ratio(0.5, 0.0, T=100, gamma3=g3, gamma4=g4),
          minimum_track_record_length(0.5, 0.0, gamma3=g3, gamma4=g4))
```

Table 3, with `np.random.seed(0)` and the package’s own generator:

```
import numpy as np, scipy.stats as st
from jsharpe import probabilistic_sharpe_ratio, generate_non_gaussian_data

def bldp(SR, T, g3, g4):
    return st.norm.cdf(SR * np.sqrt(T - 1) / np.sqrt(1 - g3*SR + (g4-1)/4*SR**2))

for name in ("gaussian", "mild", "moderate", "severe"):
    X = generate_non_gaussian_data(200_000, 100, SR0=0.0, name=name)
    sr = X.mean(axis=1) / X.std(axis=1, ddof=1)
    g3, g4 = st.skew(X, axis=1), st.kurtosis(X, axis=1, fisher=False)
    pkg = [probabilistic_sharpe_ratio(float(a), 0.0, T=100, gamma3=float(b),
                                     gamma4=float(c))
           for a, b, c in zip(sr[:20000], g3[:20000], g4[:20000])]
    print(name, np.mean(np.array(pkg) > 0.95),
          np.mean(bldp(sr[:20000], 100, g3[:20000], g4[:20000]) > 0.95))
```

Table 5:

```
from jsharpe.sharpe.quadrature import moments_Mk
from jsharpe import variance_of_the_maximum_of_k_Sharpe_ratios as vmax
rng = np.random.default_rng(0)
for K in (1, 2, 10, 50, 200):
    m = rng.standard_normal((400_000, K)).max(axis=1)
    print(K, m.var(ddof=1), moments_Mk(K)[2], vmax(K, 1.0), m.mean(), moments_Mk(K)[0])
```

Table 8 draws $C_{ij} = b_i b_j$ with `b = rng.uniform(lo, hi, n)` and unit diagonal, scales by $\sigma \sim U[0.08, 0.30]$, and compares against `np.linalg.solve(V, ones)` normalised to sum to one. Table 9 uses `get_random_correlation_matrix(n, k)` under `np.random.seed(0)` followed by `number_of_clusters(C, retries=10)`, whose second return value is the quality dictionary the table’s last two columns read.

References

- David H. Bailey and Marcos López de Prado. The Sharpe ratio efficient frontier. *Journal of Risk*, 15(2):3–44, 2012.
- David H. Bailey and Marcos López de Prado. The deflated Sharpe ratio: Correcting for selection bias, backtest overfitting, and non-normality. *The Journal of Portfolio Management*, 40(5): 94–107, 2014.
- Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society, Series B*, 57(1):289–300, 1995.
- Michael J. Best and Robert R. Grauer. On the sensitivity of mean-variance-efficient portfolios to changes in asset means: Some analytical and computational results. *The Review of Financial Studies*, 4(2):315–342, 1991.
- Herbert A. David and Haikady N. Nagaraja. *Order Statistics*. Wiley, 3rd edition, 2003.
- Edwin J. Elton, Martin J. Gruber, and Manfred W. Padberg. Simple criteria for optimal portfolio selection. *The Journal of Finance*, 31(5):1341–1357, 1976.
- Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, et al. Array programming with NumPy. *Nature*, 585:357–362, 2020.
- Campbell R. Harvey and Yan Liu. Backtesting. *The Journal of Portfolio Management*, 42(1): 13–28, 2015.
- Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979.
- Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88(2):365–411, 2004.
- Andrew W. Lo. The statistics of sharpe ratios. *Financial Analysts Journal*, 58(4):36–52, 2002.
- Marcos López de Prado. Detection of false investment strategies using unsupervised learning methods. Technical report, SSRN Working Paper 3167017, 2018.
- Harry Markowitz. Portfolio selection. *The Journal of Finance*, 7(1):77–91, 1952.
- Elmar Mertens. Comments on variance of the IID estimator in Lo (2002). Working paper, University of Basel, 2002.
- John Douglas Opdyke. Comparing Sharpe ratios: So where are the p -values? *Journal of Asset Management*, 8(5):308–336, 2007.
- Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.
- Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. In *Proceedings of the 15th European Signal Processing Conference (EUSIPCO)*, pages 606–610, 2007.
- William F. Sharpe. Mutual fund performance. *The Journal of Business*, 39(1):119–138, 1966.
- William F. Sharpe. The sharpe ratio. *The Journal of Portfolio Management*, 21(1):49–58, 1994.

- Zbyněk Šidák. Rectangular confidence regions for the means of multivariate normal distributions. *Journal of the American Statistical Association*, 62(318):626–633, 1967.
- John D. Storey. A direct approach to false discovery rates. *Journal of the Royal Statistical Society, Series B*, 64(3):479–498, 2002.
- Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, et al. SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods*, 17:261–272, 2020.