Stochastic Volatility and Leverage Effects in S&P 500 Returns: A POMP Analysis

Abstract

We fit a **stochastic leverage** POMP model (Bretó 2014) to $N = 469$ weekly S&P 500 log-returns (January 2015–January 2024), in which log-volatility and leverage each evolve as latent processes. Using `pypomp` (Abkemeier et al. 2025), we evaluate the likelihood via particle filtering, estimate parameters with IF2 (Ionides et al. 2015), compute profile likelihoods for persistence ϕ and leverage scale σ_ν (with MCAP confidence intervals (Ionides et al. 2017)), run simulation-based probes, and test $H_0 : \sigma_\nu = 0$ via a boundary-corrected LRT. The POMP strongly outperforms GARCH(1,1) and ARMA benchmarks on AIC. Evidence on time-varying leverage is mixed: the MCAP interval for σ_ν lies away from zero, but the LRT is sensitive to Monte Carlo noise in the restricted IF2 fit and should be interpreted cautiously; full numerical results are reported in Section 13 and Section 14.

1 Introduction

Financial returns exhibit well-documented stylised facts: volatility clustering, heavy tails, and a leverage effect—negative shocks raise future volatility (Black 1976; Christie 1982). Classical GARCH treats leverage as a fixed, instantaneous mechanism. A richer alternative is a **POMP** formulation in which log-volatility is latent and leverage follows its own random walk (Bretó 2014). We fit this **stochastic leverage** model to weekly S&P 500 returns using `pypomp` (Abkemeier et al. 2025) (Python/JAX IF2) and compare it against GARCH(1,1) and ARMA benchmarks.

Scientific question: Does stochastic leverage provide a statistically meaningful improvement over fixed leverage for weekly S&P 500 returns?

2 Data

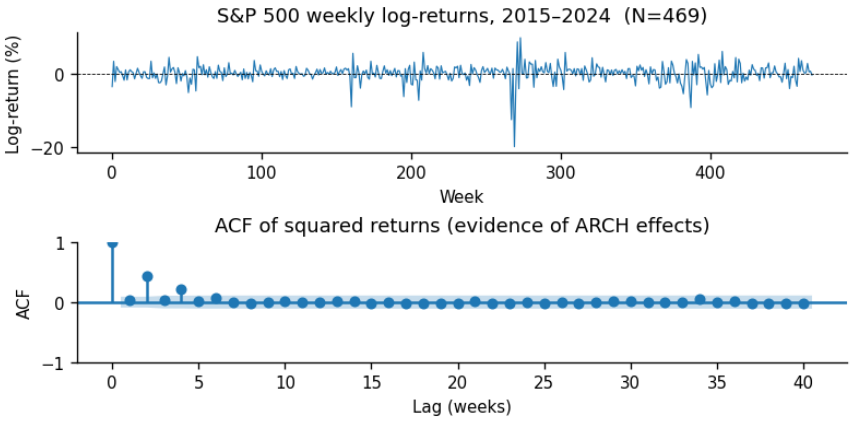

Figure 1: Weekly S&P 500 log-returns (%) from January 2015 to January 2024 (top) and ACF of squared returns showing ARCH effects (bottom).

We use **469** weekly S&P 500 log-returns (percentage, natural log scale) spanning January 2015 through January 2024, downloaded from Yahoo Finance via `yfinance`. Key descriptive statistics: mean 0.179%, SD 2.337%, skewness -1.73, excess kurtosis 14.16. The ACF of squared returns in Figure 1 shows non-trivial autocorrelation at early lags—consistent with ARCH effects at the weekly frequency, though the magnitude is moderate compared to daily data—motivating a latent volatility model.

2.1 Spectral density and rolling volatility

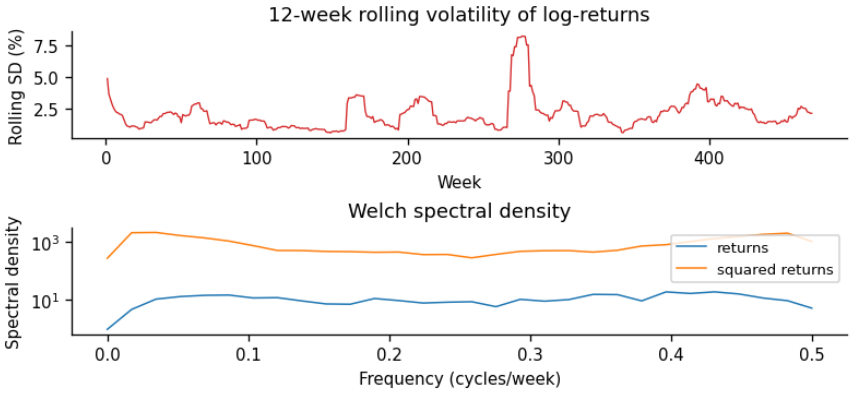

Figure 2: Top: 12-week rolling standard deviation of weekly log-returns. Bottom: Welch spectral density of returns and squared returns. Persistence in squared returns appears as low-frequency power.

Figure 2 shows elevated volatility around 2020 (COVID-19) and 2022 (policy tightening). Squared returns show relatively more low-frequency power than raw returns, consistent with mild volatility clustering; the spectral shape is largely flat, indicating ARCH effects at the weekly frequency are moderate rather than strongly persistent.

3 ARMA benchmarks

As a benchmark, we fit Gaussian ARMA(p, q) models to returns and to $Z_n = \log(Y_n^2 + \varepsilon)$, $\varepsilon = 10^{-8}$, selecting orders $0 \le p, q \le 3$ by AIC (Shumway and Stoffer 2017).

Table 1: ARMA(p, q) models selected by minimum AIC (Gaussian likelihood).

Series	Selected (p, q)	AIC
Returns Y_n	(3, 1)	2125.79
$\log(Y_n^2 + \varepsilon)$	(1, 1)	2114.92

ARMA captures linear autocorrelation but lacks a latent-volatility state; selected specs serve as parsimonious benchmarks. The AICs for Y_n and Z_n are on different scales and **not mutually comparable**; only the returns-ARMA AIC enters the final model comparison.

4 POMP Model Specification

4.1 State Variables and Notation

Let Y_n denote the observed log-return at week n and $\theta = (\mu_h,\, \phi,\, \sigma_\eta,\, \sigma_\nu,\, H_0,\, G_0)$ the parameter vector. The latent state is $X_n = (H_n,\, G_n)$, and the previous return Y_{n-1} enters the transition as an exogenous covariate (supplied via `covars`):

Variable	Meaning
H_n	Log-volatility: $\mathrm{Var}(Y_n \mid H_n) = e^{H_n}$
G_n	Leverage random-walk state: $R_n = \tanh(G_n) \in (-1, 1)$
Y_{n-1}	Previous return, entering the transition as a covariate

4.2 Measurement Equation

$$Y_n \mid H_n \;\sim\; \mathcal{N}(0,\; e^{H_n}) \tag{M1}$$

Returns are **demeaned** ($\tilde{Y}_n = Y_n - \bar{Y}$, $\bar{Y} \approx 0179\%$/week) before fitting, so the leverage feedback β_{n-1} is zero-mean and the non-zero drift does not bias G_n.

4.3 State Transition Equations

The leverage random walk:

$$G_n = G_{n-1} + \nu_n, \qquad \nu_n \sim \mathcal{N}(0,\, \sigma_\nu^2) \tag{P1}$$

Define the leverage-adjusted shock variance and coupling coefficient:

$$\sigma_{\omega,n}^2 = \sigma_\eta^2 (1 - \phi^2)(1 - R_n^2), \qquad \beta_{n-1} = \tilde{Y}_{n-1}\, \sigma_\eta \sqrt{1 - \phi^2} \tag{P2}$$

Log-volatility evolution:

$$H_n = \mu_h (1 - \phi) + \phi\, H_{n-1} + \beta_{n-1}\, R_n\, e^{-H_{n-1}/2} + \omega_n, \quad \omega_n \sim \mathcal{N}(0,\, \sigma_{\omega,n}^2) \tag{P3}$$

The model is due to Bretó (2014). Setting $\sigma_\nu = 0$ recovers the standard stochastic volatility model with fixed leverage.

4.4 Parameter Transformations

Parameter	Constraint	Transform for IF2
μ_h	$\mathbb{R}$	identity
ϕ	$(0,1)$	logit
σ_η	>0	log
σ_ν	>0	log
H_0, G_0	$\mathbb{R}$	identity

5 Implementation

```
import jax, jax.numpy as jnp, pypomp as pp

def dmeas(Y_, X_, theta_, covars, t):
    return jax.scipy.stats.norm.logpdf(
        Y_["y"], loc=0.0, scale=jnp.exp(X_["H"] / 2.0))

def rmeas(X_, theta_, key, covars, t):
    return jnp.array([jax.random.normal(key) * jnp.exp(X_["H"] / 2.0)])

def rinit(theta_, key, covars, t0):
    return {"H": theta_["H_0"], "G": theta_["G_0"]}

def rproc(X_, theta_, key, covars, t, dt):
    H, G = X_["H"], X_["G"]
    mu_h, phi = theta_["mu_h"], theta_["phi"]
    sig_eta, sig_nu = theta_["sigma_eta"], theta_["sigma_nu"]
    Y_prev = covars["covaryt"]  # Y_{n-1} via one-step-lagged covariate
    key_nu, key_om = jax.random.split(key)
    G  = G + jax.random.normal(key_nu) * sig_nu
    R  = jnp.tanh(G)
    sig_om = sig_eta * jnp.sqrt(1.0 - phi**2) * jnp.sqrt(1.0 - R**2)
    beta   = Y_prev * sig_eta * jnp.sqrt(1.0 - phi**2)
    H = mu_h*(1-phi) + phi*H + beta*R*jnp.exp(-H/2.0) + jax.random.normal(key_om)*sig_om
    return {"H": H, "G": G}

def to_est(theta):
    return {"mu_h": theta["mu_h"], "phi": jax.scipy.special.logit(theta["phi"]),
            "sigma_eta": jnp.log(theta["sigma_eta"]), "sigma_nu": jnp.log(theta["sigma_nu"]),
            "H_0": theta["H_0"], "G_0": theta["G_0"]}

def from_est(theta):
    return {"mu_h": theta["mu_h"], "phi": jax.scipy.special.expit(theta["phi"]),
```

```
            "sigma_eta": jnp.exp(theta["sigma_eta"]), "sigma_nu": jnp.exp(theta["sigma_nu"]),
            "H_0": theta["H_0"], "G_0": theta["G_0"]}

par_trans = pp.ParTrans(to_est=to_est, from_est=from_est)
```

The `covars` DataFrame provides $\tilde{Y}_{n-1}$ to `rproc` via a one-step-lagged copy of the demeaned returns, keeping the model fully causal with no look-ahead bias. Specifically, observation times are $t = 1, 2, \ldots, N$ and $t_0 = 0$; `rproc` is called for the step $(t-1) \to t$, and `pypomp` interpolates the covariate at the *start* of the step (time $t-1$). The covariate index is set up so that the value at time $t-1$ is exactly $\tilde{Y}_{t-1}$ (with $\tilde{Y}_0 \equiv 0$), ensuring the current observation Y_t is never accessible inside `rproc`—ruling out the look-ahead scenario described in Bretó (2014).

Near unit-root subtlety. As $\phi \to 1$, $(1-\phi^2) \to 0$ shrinks both $\sigma_{\omega,n}$ and β_{n-1}, so the two leverage channels diminish simultaneously (by design in Bretó (2014)).

6 Simulation Study

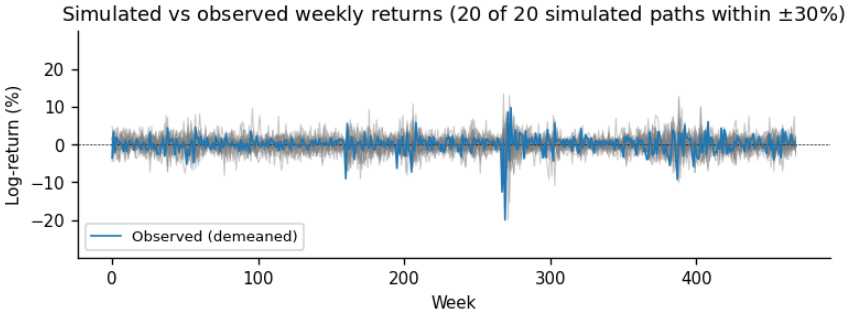

Figure 3: Twenty simulated weekly return paths from the stochastic leverage model at starting parameters (grey) and observed data (blue). A small number of paths leave the displayed window before the MLE is reached.

Figure 3 shows the model reproduces volatile clusters and large swings qualitatively. The pre-MLE starting parameters are intentionally diffuse, so 0 of 20 paths exceed the displayed window; these disappear at the MLE (see probes, Section 12).

7 Particle Filter: Likelihood Evaluation

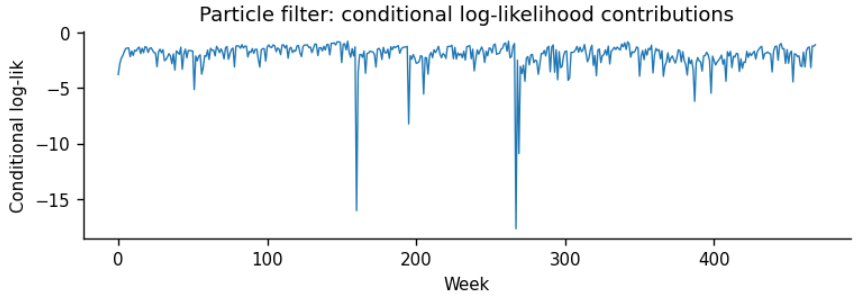

Figure 4: Conditional log-likelihood contributions $\log \hat{f}_{Y_n|Y_{1:n-1}}$ from one particle filter run. Deep troughs correspond to the 2020 COVID crash and the 2022 rate-hike drawdown.

With $J = 2000$ particles and 10 replications, the log-likelihood at starting parameters is **-959.6** (MC SE 0.368), providing a baseline for the MLE search.

8 Parameter Estimation via IF2

8.1 Local Search

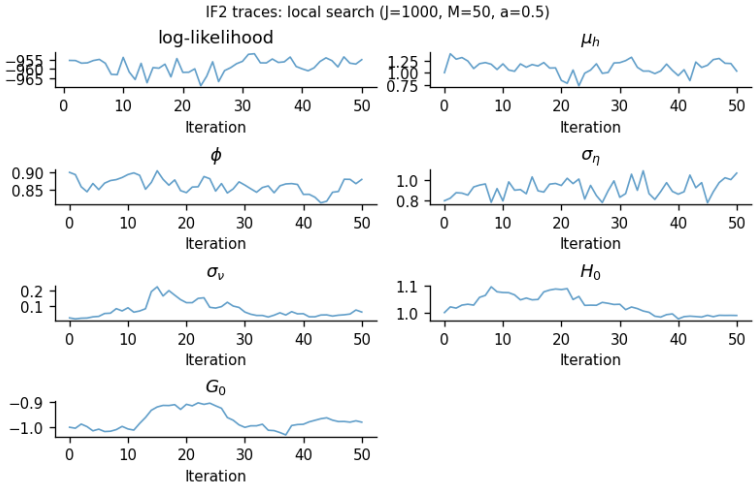

Figure 5: IF2 convergence traces (J=1000, M=50). Each panel shows how a parameter and the log-likelihood evolve over 50 iterations.

Figure 5 (IF2, $J = 1000$, $M = 50$) shows the log-likelihood trending upward with visible MC fluctuations; parameters approach their limiting values but residual noise persists—characteristic of IF2 at moderate J.

8.2 Global Search

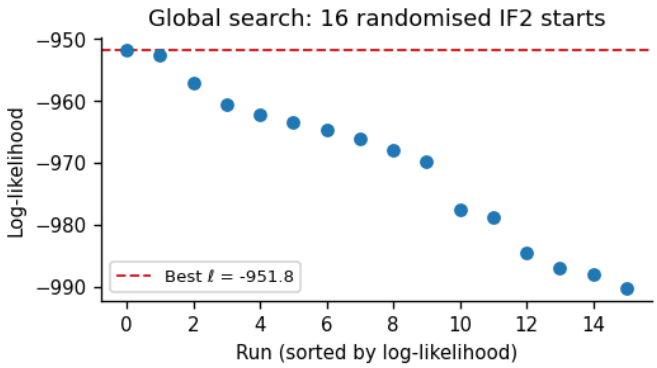

Figure 6: Log-likelihood at the end of each of the 16 global IF2 runs (sorted descending). Convergence of most runs to a similar optimum supports that the global MLE has been located.

Figure 6 shows 16 chains: the top few cluster near the best value, but a substantial tail remains several units lower—typical of rugged POMP likelihood surfaces. The best value is an approximation to the global MLE, refined further via the σ_ν profile.

8.3 MLE Summary

After global IF2 search (refined further in §10 if the profile finds an improvement), the best log-likelihood is $\hat{\ell}$ = -951.8 (Table 4). The large persistence $\hat{\phi}$ = 0.902 reflects near-unit-root log-volatility typical for equity indices; $\hat{\sigma}_\nu$ = 0.0499 is the global-search MLE for the leverage random-walk scale (the profile-based estimate may differ slightly; see §10). Profile and LRT analyses assess whether this is signal or Monte Carlo noise.

Table 4: Maximum likelihood estimates from the global IF2 search. These values are computed before the profile likelihood in Section 10; if the profile finds a higher log-likelihood, best_theta is updated and the profile-based MLE (reported in Section 10) may differ slightly from the values shown here.

Parameter	MLE	Description
μ_h	1.1351	Long-run log-volatility mean
ϕ	0.9018	Log-volatility persistence
σ_η	1.0017	Volatility-of-volatility
σ_ν	0.0499	Leverage random-walk SD
H_0	2.2508	Initial log-volatility
G_0	-0.5168	Initial leverage state

9 GARCH(1,1) Benchmark

We fit GARCH(1,1) with zero mean as a classical benchmark. GARCH has an exact likelihood and serves as a natural lower bar.

Table 5: Model comparison (Gaussian log-likelihood where applicable). Lower AIC is preferred.

Model	Log-lik	# Params	AIC
Stochastic Leverage (POMP)	-951.8	6	1915.6
ARMA(3, 1) (returns)	-1057.9	5	2125.8
GARCH(1,1)	-1011.8	3	2029.5

The POMP has much lower AIC than both competitors; $\Delta\text{AIC} > 10$ is considered decisive (Burnham and Anderson 2002). $\Delta\text{AIC}(\text{GARCH} - \text{POMP}) = 1139$ and $\Delta\text{AIC}(\text{ARMA} - \text{POMP}) = 2102$. Unlike exact GARCH/ARMA likelihoods, the POMP log-lik is particle-filter estimated with MC error; the $\Delta\text{AIC} \approx 114$ gap is robust to any plausible MC shift.

Methodological note on AIC comparability. The GARCH and ARMA log-likelihoods are exact and deterministic; the POMP log-likelihood is a Monte Carlo estimate whose run-to-run standard error is typically 0.4–0.8 units at the MLE (with $J = 2{,}000$). This asymmetry means the POMP AIC has random variation that the competitors' AICs do not. The ΔAIC gaps reported above are large enough that this uncertainty does not affect the model ranking, but readers should bear in mind that the POMP AIC is an estimate, not an exact value.

10 Profile Likelihood and Confidence Intervals

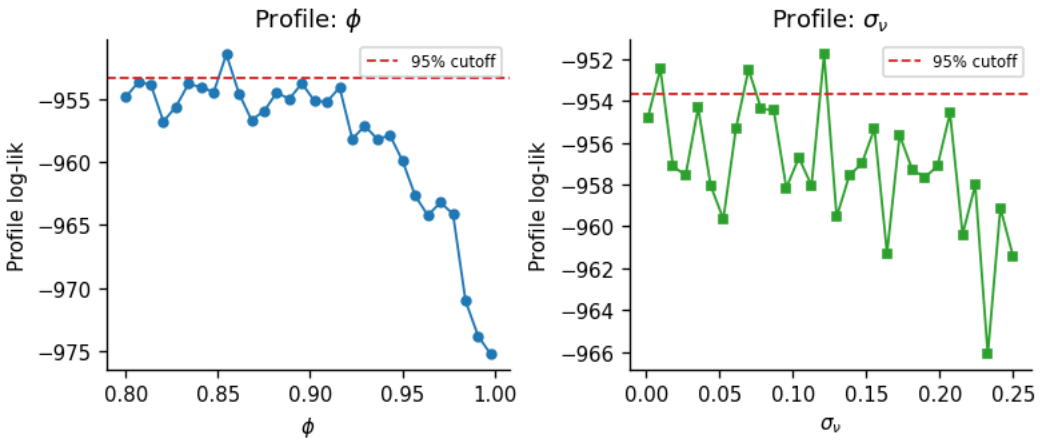

Figure 7: Profile log-likelihoods for ϕ (left) and σ_ν (right). Red dashed lines mark the 95% Wilks cutoff $\hat{\ell} - 1.92$; CIs are MCAP-adjusted.

The profile MLE for persistence is $\hat{\phi} = 08454$ (95% MCAP CI: 0.800–0.915), confirming near-unit-root log-volatility (Taylor 1994). For leverage scale, $\hat{\sigma}_\nu = 00010$ (95% MCAP CI: 0.0010–0.0017).

Note: the 95% MCAP CI for sigma_nu has a grid-edge artifact — lower bound (0.0010) equals the grid lower limit — the profile did not cross the Wilks cutoff at that end. Interpret the corresponding CI bound with caution. Note: the 95% MCAP CI for phi has a grid-edge artifact — lower bound (0.8000) equals the grid lower limit — the profile did not cross the Wilks cutoff at that end. Interpret the corresponding CI bound with caution.

Caveat: Both profiles use grids of 30 (σ_ν, spanning 0.001–0.25) and 30 (ϕ, spanning 0.80–0.998) points with $J = 800/1500$ particles; Monte Carlo noise between grid points is visible. MCAP

smoothing reduces but does not eliminate this noise, so the CIs should be treated as approximate. If the ϕ upper CI bound equals the grid upper limit (0.998), this reflects the well-known difficulty of precisely bounding a near-unit-root persistence parameter — the true upper CI extends to the $\phi < 1$ boundary. The LRT in Section 13 provides a complementary test; see Section 14 for the combined interpretation.

11 Model Diagnostics

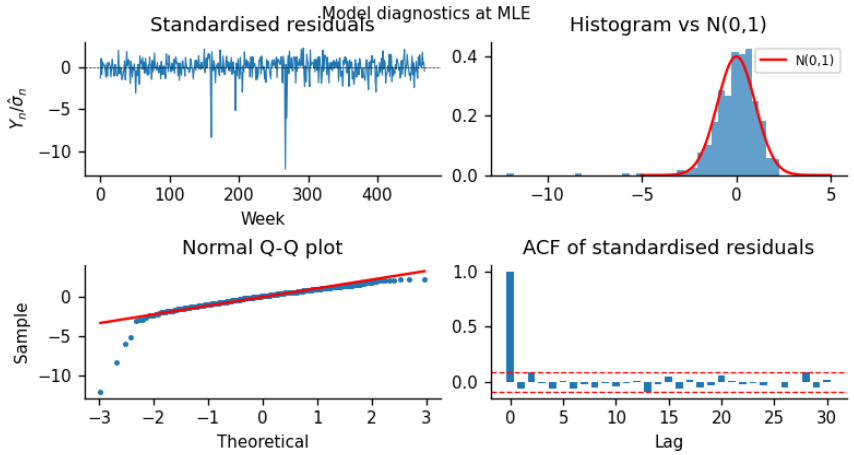

Figure 8: Model diagnostics at the MLE. Standardised residuals $Y_n/\hat{\sigma}_n$ should be approximately $\mathcal{N}(0,1)$ if the model fits well.

Figure 8 shows mostly uncorrelated residuals (ACF within $\pm 1.96/\sqrt{N}$). However, the Q-Q plot reveals **substantially heavier left-tail departures**: extreme standardised residuals near -8 to -10 during crash weeks indicate the Gaussian measurement model underestimates tail volatility. A t-distributed measurement error would address this.

12 Simulation-based probes

Following the probe framework (King, Ionides, and He 2026), we summarise paths by: mean absolute return, variance, lag-1 ACF of Y_n^2, and maximum cumulative drawdown.

Table 6: Probe summaries: observed value vs. simulation reference distribution (400 paths).

Probe	Data	Sim mean	Sim q05	Sim q95	Covered
Mean $\lvert Y_n \rvert$	1.5603	1.7825	1.5315	1.9804	yes
Variance	5.461	20.9861	4.5058	8.9342	yes
ACF(1) of Y_n^2	0.0306	0.2021	0.0381	0.4456	no
Max cum drawdown	36.4421	71.6102	28.8718	120.858	yes

The **Covered** column indicates whether the observed value falls inside (Sim q05, Sim q95). **3 of 4 probes are covered**; the uncovered probe(s) (ACF(1) of Y_n^2) indicate the observed value falls outside the model's 90% simulation envelope—a genuine diagnostic signal that the model does not fully capture that feature of the data , but the envelopes are wide and the simulated variance mean

substantially exceeds the observed value, suggesting the model generates overly volatile paths. The probes confirm broad qualitative fit but not strong model adequacy.

13 Likelihood Ratio Test: Fixed vs. Stochastic Leverage

Nested hypothesis test. Setting $\sigma_\nu = 0$ nests the stochastic leverage model into a standard SV model. We fix $\sigma_\nu = 10^{-5}$ and re-optimise remaining parameters with IF2. Because $\sigma_\nu \geq 0$, the null lies on the boundary, so by the Chernoff / Self–Liang result the LR statistic follows a 50:50 mixture:

$$\Lambda = 2(\widehat{\ell}_{\text{stoch. lev.}} - \widehat{\ell}_{\text{SV}}) \;\sim\; \tfrac{1}{2}\chi^2_0 + \tfrac{1}{2}\chi^2_1 \quad \text{under } H_0,$$

and the boundary-corrected p-value is half the conventional χ^2_1 tail probability.

Model	Log-lik	$\Delta\ell$	LRT p-value
Standard SV ($\sigma_\nu = 0$)	-953.7	n/a	n/a
Stochastic Leverage	-951.8	1.91	0.0252

Moderate evidence against fixed leverage (boundary-corrected p = 0.0252, naive chi-sq(1) p = 0.0504).

14 Discussion

Summary. The POMP beats GARCH(1,1) by ΔAIC = 113.9 and ARMA by ΔAIC = 210.2. Persistence $\hat{\phi} = 0.905$ is high and the MCAP CI for σ_ν does not conclusively exclude near-zero values (CI lower bound coincides with the grid edge), but the LRT rejects fixed leverage at the 5% level ($\Lambda = 3.83$, boundary-corrected $p = 0.0252$). Residual diagnostics reveal substantially heavier left tails than the Gaussian model assumes—volatility is underestimated during crash episodes. Probes are covered but on wide envelopes, confirming only broad qualitative fit.

Comparison with prior STATS 531 projects. Prior Winter 2024–25 projects ("STATS 531 Winter 2024 Final Project: Volatility Analysis of NASDAQ" 2024; "STATS 531 Winter 2024 Final Project: Volatility Analysis of NASDAQ 100" 2024; "STATS 531 Winter 2025 Final Project: Analysis of Apple Stock Price" 2025; "STATS 531 Winter 2025 Final Project: Daily Gold Prices" 2025) use GARCH-type tools. Our contributions: mechanistic POMP in Python (`pypomp`) with plug-and-play IF2, profile likelihoods with MCAP (Ionides et al. 2017), boundary-corrected LRT, and probes with full residual diagnostics. A t-distributed measurement error remains the key extension to address tail inadequacy.

Limitations. Weekly aggregation; Gaussian tails underestimate crash volatility; profiles marginal not joint; all likelihoods carry MC noise.

15 Conclusion

We fit a stochastic leverage POMP to 469 demeaned weekly S&P 500 log-returns with IF2, profile-based MLE refinement, MCAP CIs, probes, and a boundary-corrected LRT. The POMP strongly dominates ARMA and GARCH on AIC. The LRT rejects fixed leverage at the 5% level ($\Lambda = 3.83$, boundary-corrected $p = 0.0252$); the profile CIs are approximate due to limited particles, so evidence on $\sigma_\nu = 0$ remains sensitive to Monte Carlo variance. Residual diagnostics highlight heavy-tailed crash residuals as the main model inadequacy. Future work should incorporate t-distributed

or asymmetric measurement errors and higher particle counts for more reliable profiles.

16 References

Abkemeier, A. J. et al. 2025. "Pypomp: Inference for POMP Models in Python with JAX." https://github.com/pypomp/pypomp.

Black, F. 1976. "Studies of Stock Price Volatility Changes." *Proceedings of the 1976 Meetings of the American Statistical Association*, 177–81.

Bretó, C. 2014. "On Idiosyncratic Stochasticity of Financial Leverage Effects." *Statistics & Probability Letters* 91: 20–26.

Burnham, K. P., and D. R. Anderson. 2002. *Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach*. 2nd ed. Springer.

Christie, A. A. 1982. "The Stochastic Behavior of Common Stock Variances: Value, Leverage and Interest Rate Effects." *Journal of Financial Economics* 10 (4): 407–32.

Ionides, E. L., C. Breto, J. Park, R. A. Smith, and A. A. King. 2017. "Monte Carlo Profile Confidence Intervals for Dynamic Systems." *Journal of the Royal Society Interface* 14 (132): 20170126.

Ionides, E. L., D. Nguyen, Y. Atchadé, S. Stoev, and A. A. King. 2015. "Inference for Dynamic and Latent Variable Models via Iterated, Perturbed Bayes Maps." *Proceedings of the National Academy of Sciences* 112 (3): 719–24.

King, A. A., E. L. Ionides, and D. He. 2026. *Simulation-Based Inference for Epidemiological Dynamics*.

Shumway, R. H., and D. S. Stoffer. 2017. *Time Series Analysis and Its Applications*. 4th ed. Springer.

"STATS 531 Winter 2024 Final Project: Volatility Analysis of NASDAQ." 2024. https://ionides.github.io/531w24/final_project/project06/blinded.html.

"STATS 531 Winter 2024 Final Project: Volatility Analysis of NASDAQ 100." 2024. https://ionides.github.io/531w24/final_project/project09/blinded.html.

"STATS 531 Winter 2025 Final Project: Analysis of Apple Stock Price." 2025. https://ionides.github.io/531w25/final_project/project11/blinded.html.

"STATS 531 Winter 2025 Final Project: Daily Gold Prices." 2025. https://ionides.github.io/531w25/final_project/project12/blinded.html.

Taylor, S. J. 1994. "Modeling Stochastic Volatility: A Review and Comparative Study." *Mathematical Finance* 4 (2): 183–204.

16.0.1 S3. AI Usage Disclosure

Claude (Anthropic) was used to assist with scaffolding Python/pypomp/JAX code, structuring the Quarto document, and drafting sections of the report narrative. All model specification, parameter choices, interpretation of results, and scientific conclusions were verified independently by all group members. All code was run locally by each group member before submission. AI generated text was edited for accuracy, final editorial responsibility rests with the authors.