Value at Risk (VaR) Its Forms

A Practical and Comparative Framework for Estimation and Backtesting

KITOHOU Ambroise

August 22, 2025

Abstract

This document presents a complete, practitioner-oriented framework for estimating and validating Value at Risk (VaR) using Python. We implement multiple VaR methodologies (Historical Simulation, Parametric Normal, Cornish–Fisher, Monte Carlo, and GARCH(1,1)) and evaluate their reliability through Kupiec and Christoffersen backtests, alongside Expected Shortfall (ES). The goal is to compare approaches, discuss trade-offs, and provide actionable recommendations for risk management in real market settings.

1 Context and Objectives

Risk measurement is central to trading, treasury, and portfolio management. VaR has become a standard to summarize potential losses over a given horizon at a chosen confidence level. However, model risk and market regimes can strongly affect VaR accuracy. This project targets:

- A clear comparison of common VaR methodologies.
- Rigorous validation via backtesting (frequency and independence of exceptions).
- Practical guidance on method selection under different market conditions.

2 Data and Preprocessing

Market data (prices) are downloaded via `yfinance`. Log-returns are computed as $r_t = \ln(P_t/P_{t-1})$, with standard cleaning (date alignment, missing values). When reproducibility is required, a fixed time window and a local CSV cache are recommended.

3 Methodology

3.1 Definition

For a loss variable L and level $\alpha \in (0,1)$, the VaR at level α is the lower α-quantile:

$$\mathrm{VaR}_\alpha(L) = \inf\{\ell \in \mathbb{R} : \Pr(L \leq \ell) \geq \alpha\}.$$

We work with (negative) returns so that left-tail quantiles represent extreme losses.

3.2 Implemented Approaches

Historical Simulation Empirical quantile of past returns. Pros: model-free, intuitive. Cons: sensitive to window choice; slow to react to regime shifts.

Parametric Normal Assumes $r_t \sim \mathcal{N}(\mu, \sigma^2)$. Then

$$\mathrm{VaR}_\alpha = \mu + \sigma\,\Phi^{-1}(\alpha),$$

with Φ^{-1} the standard normal quantile. Fast, but may underestimate fat tails.

Cornish–Fisher Expansion Adjusts the normal quantile using skewness (γ_1) and excess kurtosis (γ_2):

$$z_\alpha^{CF} \approx z_\alpha + \frac{1}{6}(z_\alpha^2 - 1)\gamma_1 + \frac{1}{24}(z_\alpha^3 - 3z_\alpha)\gamma_2 - \frac{1}{36}(2z_\alpha^3 - 5z_\alpha)\gamma_1^2,$$

then $\mathrm{VaR}_\alpha \approx \mu + \sigma z_\alpha^{CF}$. Captures asymmetry and heavy tails.

Monte Carlo Simulation Simulates many return paths under a chosen distributional assumption (e.g., normal or Student-t), then takes the empirical quantile of simulated losses. Flexible, but computationally more costly; portfolio correlations should be modeled in multivariate settings.

GARCH(1,1) Models conditional volatility clustering:

$$r_t = \sigma_t \varepsilon_t, \quad \sigma_t^2 = \omega + \alpha r_{t-1}^2 + \beta \sigma_{t-1}^2,$$

with innovations ε_t typically standard normal or Student-t. A one-step-ahead forecast $\hat{\sigma}_{t+1}$ yields a *dynamic* VaR:

$$\mathrm{VaR}_{\alpha,t+1} = \hat{\mu}_{t+1} + \hat{\sigma}_{t+1}\, q_\alpha,$$

where q_α is the innovation quantile.

Expected Shortfall (ES)

$$\mathrm{ES}_\alpha = \mathbb{E}[\, L \mid L \leq \mathrm{VaR}_\alpha \,],$$

measures the *average* loss beyond VaR; more informative for tail risk and favored by regulation.

4 Backtesting Framework

4.1 Kupiec Proportion-of-Failures (POF)

Let $I_t = \mathbf{1}\{r_t < \widehat{\mathrm{VaR}}_{\alpha,t}\}$, $x = \sum I_t$, N the sample size, and $\hat{\pi} = x/N$. The likelihood ratio statistic

$$\mathrm{LR}_{\mathrm{POF}} = -2 \ln \left(\frac{(1-\alpha)^{N-x} \alpha^x}{(1-\hat{\pi})^{N-x} \hat{\pi}^x} \right)$$

is asymptotically χ^2 with 1 d.f.; large values reject correct coverage.

4.2 Christoffersen Independence Test

Let n_{ij} be the number of transitions $I_{t-1} = i \to I_t = j$. Under independence, exception probability does not depend on the previous state. The LR statistic compares the unrestricted two-state Markov model to the restricted i.i.d. model and is χ^2 with 1 d.f.

4.3 Combined Conditional Coverage

$$\mathrm{LR}_{\mathrm{cc}} = \mathrm{LR}_{\mathrm{POF}} + \mathrm{LR}_{\mathrm{IND}} \sim \chi^2(2),$$

jointly testing correct exception rate and independence.

5 Results (Executive Synthesis)

In our experiments:

- Historical and Cornish–Fisher VaR deliver reasonable coverage in calm markets.
- Parametric Normal VaR tends to be optimistic (underestimates tails) during turbulence.
- GARCH-based VaR adapts to volatility clustering and improves conditional coverage in volatile regimes.

- ES complements VaR by quantifying the severity of tail losses.

Illustrative figures are provided below.

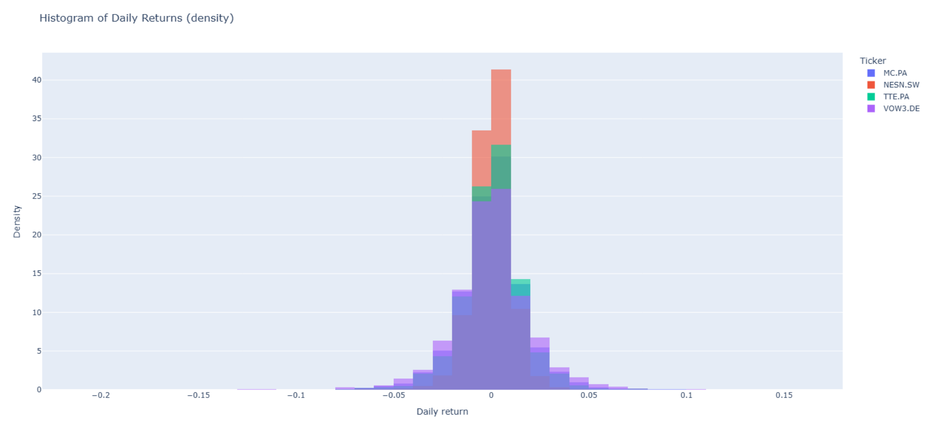

Figure 1: Comparison across VaR methodologies.

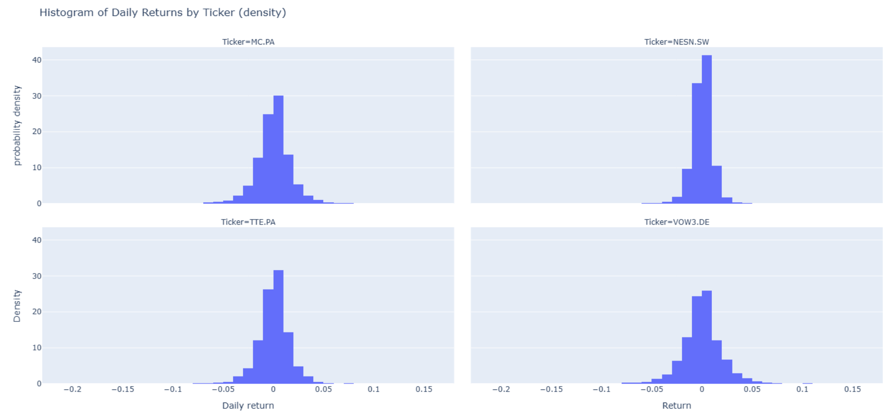

Figure 2: VaR by asset.

6 Recommendations

- Combine **GARCH-based VaR** with **ES** for production risk monitoring.
- Use **Cornish–Fisher** (or Student-t) adjustments when tails are heavy or skewed.
- For multi-asset portfolios, employ **multivariate** simulation or copulas to model dependence.
- Fix data windows and seed values to ensure **reproducibility**; cache raw prices locally.

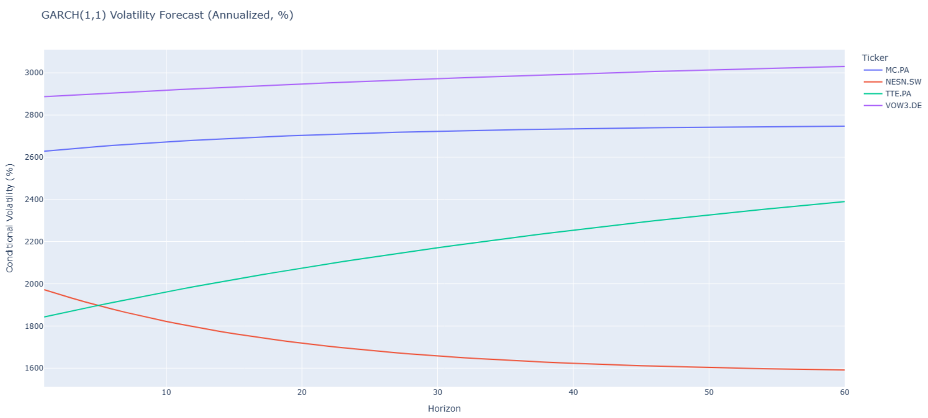

Figure 3: GARCH(1,1) conditional volatility forecast.

7 How to Run

1. Clone the repository: git clone https://github.com/<your_username>/VaR_its_Forms.git
2. Create and activate a virtual environment: python -m venv project_VaR .
   project_VaR
   Scripts
   Activate.ps1 (Windows)
3. Install dependencies: pip install -r requirements.txt
4. Execute: python VaR_Its_Forms.py

8 Repository Structure

```
VaR_its_Forms/
 VaR_Its_Forms.py               # Main Python script
 requirements.txt               # Dependencies
 Project_analyst_report.docx    # Analytical report
 forecasting.png                # GARCH forecast plot
 plot_by_ticket.png             # VaR by asset
 plot_for_all.png               # VaR method comparison
 README.tex / README.pdf        # This document
```

9 Limitations and Future Work

- Normal-based parametrics ignore extreme tails; consider Student-t.
- Monte Carlo examples are univariate; extend to multivariate with correlations.

- Backtesting could include stress scenarios and intraday frequencies.

10 License

This project is released under the MIT License. You may use, modify, and distribute the code with attribution.