



Università della Svizzera Italiana

Faculty of Economics

Programming in Economics and Finance II

pyvar – A Python Package for Risk Management

Alessandro Dodon, Niccolò Lecce, Marco Gasparetti

a.y. 2024/2025

Submission Date: June 13, 2025

Contents

1	Introduction	2
2	Project Plan	2
3	Project Diary	2
4	Theoretical Foundation	3
4.1	Basic VaR Models	3
4.2	Extreme Value Theory	4
4.3	Volatility Models	4
4.4	Time-Varying Correlation Models	5
4.5	Factor Models	5
4.6	Analytic VaR	6
4.7	Simulation Methods	7
4.8	Backtesting	8
5	Results and Applications	9
6	Lessons Learned	9
7	AI Acknowledgement	9
8	Link to GitHub Repository	10
9	References	10
10	Appendix	11

1 Introduction

Risk management is central to finance, yet many practitioners still rely on inefficient tools like spreadsheets. As programming becomes essential in financial workflows, there is a growing need for accessible, efficient solutions to compute Value-at-Risk (VaR) and other risk metrics.

We introduce `pyvar`, a Python package that automates the full VaR estimation pipeline—from portfolio creation to risk metrics, visualization, backtesting, and natural language interpretation using LLMs. Focused on equity portfolios and simple models, the package prioritizes speed, clarity, and ease of use—ideal for both professionals and retail investors seeking intuitive, modern risk tools.

2 Project Plan

The project began with a presentation on April 16, initially focused on a minimal VaR package covering normal parametric and historical VaR, simple volatility models, Monte Carlo simulation, basic backtesting, and an LLM interface. Following our professor’s advice, we added Expected Shortfall (ES) to align with modern risk standards. As development progressed, we expanded the scope to include advanced volatility models, Extreme Value Theory, time-varying correlations, options-inclusive portfolios, factor models, and full portfolio-level risk metrics. Alessandro Dodon implemented volatility models, EVT, backtesting, correlation modeling, and portfolio risk decomposition. Marco Gasparetti developed simulation modules, multi-day forecasting, and the options pricing framework. Niccolò Lecce integrated the LLM interface. Marco and Niccolò jointly handled factor models, the extensive documentation, and the portfolio creation module. All components were reviewed and tested collaboratively.

3 Project Diary

Work on the project began after the idea was approved during the presentation. Most of the initial development took place between April 17 and April 27 during the Easter break, following a kickoff meeting where tasks were divided. We started by implementing the basic VaR models, volatility models, factor models, simulations, and LLM integration. Between April 28 and May 4, we added more sophisticated simulation methods, including bootstrap techniques and options-inclusive portfolios. From May 5 to May 11, we focused on enhancing dynamic correlation models and improving the LLM interface. Between May 12 and May 18, we completed missing components such as Extreme Value Theory (EVT) and the full backtesting suite. Weekly meetings, held either in person or online, helped us coordinate development and resolve issues efficiently. By mid-May, the package was functionally complete. The final weeks of May were dedicated to debugging, refining code style and readability, improving documentation, and testing. Initially, testing was carried out through dedicated notebooks for each function, using simple real-world cases to verify that results matched theoretical expectations. This process also helped identify bugs and improve robustness. In the final week of May, we prepared the complete tutorial notebook and main program, which can be found in the GitHub repository. A detailed development history is available

in our GitHub repository.

4 Theoretical Foundation

This section outlines the mathematical foundations of our risk models. For simplicity, we group them into two categories: univariate and portfolio-level models. We begin with univariate models, which describe the return dynamics of a single asset or, equivalently, a portfolio treated as a single aggregated position. These models typically assume a specific distribution of returns and are used to compute risk measures for individual positions. We then introduce portfolio-level models, which consider the joint distribution of multiple asset returns. These multivariate approaches capture dependencies between assets and allow for a more accurate assessment of portfolio risk. Both analytical and simulation-based methods are discussed in this context. The section concludes with a presentation of the backtesting techniques used to evaluate the performance and reliability of the risk estimates produced by our models.

4.1 Basic VaR Models

Let $c \in (0, 1)$ denote the confidence level, and define $\alpha = 1 - c$ as the corresponding probability of a large loss. Value-at-Risk (VaR) at level α is defined as the worst loss L over a fixed horizon such that the probability of observing a larger loss is at most α :

$$\Pr(L > \text{VaR}_\alpha) \leq \alpha.$$

Intuitively, this means we are confident that with probability c , losses will not exceed VaR_α over the given horizon.

By convention, VaR is expressed as a positive monetary amount. To simplify the notation in single-asset models we omit the scaling by invested wealth W and use percentage-based notation. To simplify the calculations we assume that W is strictly positive and that returns follow a zero-mean process.

We define the loss variable as $\ell_t = -r_t$. We start with models that assume constant volatility over time. In the non-parametric (historical) approach:

$$\text{VaR}_\alpha^{\text{hist}} = Q_\alpha(\ell_t), \quad \text{VaR}_\alpha^{\text{normal}} = z_\alpha \sigma,$$

where $Q_\alpha(\ell_t)$ is the empirical α -quantile of losses, and $z_\alpha = \Phi^{-1}(1 - \alpha)$ is the positive quantile of the standard normal distribution.

Expected Shortfall (ES), also known as Conditional VaR, is computed as:

$$\text{ES}_\alpha^{\text{historical}} = \mathbb{E}[\ell_t \mid \ell_t \geq \text{VaR}_\alpha], \quad \text{ES}_\alpha^{\text{normal}} = \sigma \cdot \frac{\phi(z_\alpha)}{\alpha}, \quad \text{ES}_\alpha^{t_\nu} = \sigma \cdot \frac{f_{t_\nu}(z_\alpha)}{\alpha} \cdot \frac{\nu + z_\alpha^2}{\nu - 1}$$

where ϕ is the standard normal density and f_{t_ν} is the density of the Student- t distribution with ν degrees of freedom.

Normal and Student- t VaR and ES estimates scale with the holding period as \sqrt{h} . For historical VaR,

no scaling is needed—one can simply use weekly or monthly return data to estimate risk over longer horizons.

4.2 Extreme Value Theory

Technically still a parametric model, EVT is more sophisticated than the methods discussed above. We implement it with the Peaks-Over-Threshold (POT) approach: once a high threshold u (e.g., the 99th-percentile loss) is chosen, a Generalized Pareto Distribution (GPD) is fitted to the excesses above u .

VaR and ES are then derived analytically from the fitted GPD parameters. This allows for more accurate estimation of rare, extreme losses. The formulas used are:

$$\text{VaR}_\alpha = u + \frac{\beta}{\xi} \left[\left(\frac{n}{n_u} (1 - \alpha) \right)^{-\xi} - 1 \right], \quad \text{ES}_\alpha = \frac{\text{VaR}_\alpha + \beta - \xi u}{1 - \xi},$$

where ξ and β are the estimated shape and scale parameters, n is the sample size, and n_u is the number of exceedances.

4.3 Volatility Models

We have implemented the most popular univariate volatility models.

The GARCH(1,1) model is specified as:

$$\sigma_t^2 = \omega + a r_{t-1}^2 + b \sigma_{t-1}^2, \quad r_t = \sigma_t z_t,$$

where z_t are i.i.d. standardized innovations. VaR is then computed as:

$$\text{VaR}_t = \sigma_t z_\alpha.$$

Here, $z_\alpha = -Q_\alpha(z_t)$ is the positive α -quantile from the empirical distribution of the standardized innovations.

Future volatility can be forecasted analytically using GARCH(1,1).

The τ -step ahead forecast and cumulative T -step forecast are:

$$\begin{aligned} \mathbb{E}[\sigma_{t+\tau}^2] &= \text{VL} + (a + b)^\tau (\sigma_t^2 - \text{VL}), \\ \mathbb{E}[\sigma_{t,T}^2] &= \text{VL} \left(T - 1 - \frac{(a + b)(1 - (a + b)^{T-1})}{1 - (a + b)} \right) + \sigma_t^2 \frac{1 - (a + b)^T}{1 - (a + b)}, \end{aligned}$$

which naturally allow multi-period VaR estimation.

We also support EGARCH (logs variance, capturing leverage), GJR-GARCH (adds an indicator for negative shocks), and APARCH (introduces a power term for flexible tail dynamics), all with maximum-likelihood estimation under Normal, Student- t , GED, or Skew- t innovations.

For simpler benchmarks we include ARCH(p) and two rolling estimators, Moving Average (MA) and

Exponentially Weighted Moving Average (EWMA):

$$\sigma_t^{2,\text{ARCH}} = \omega + \sum_{i=1}^p \alpha_i r_{t-i}^2, \quad \sigma_t^{2,\text{MA}} = \frac{1}{n} \sum_{i=1}^n r_{t-i}^2, \quad \sigma_t^{2,\text{EWMA}} = \lambda \sigma_{t-1}^2 + (1 - \lambda) r_{t-1}^2.$$

ES for volatility models becomes:

$$\text{ES}_t = \sigma_t \mathbb{E}[-z_t \mid z_t < -z_\alpha].$$

All our volatility models use a semi-empirical approach for VaR computation: we first estimate the empirical distribution of standardized innovations z_t , then compute the required quantile z_α .

4.4 Time-Varying Correlation Models

We now introduce models designed to capture time-varying correlations at the portfolio level, extending beyond single assets. Specifically, we implement two simple approaches: moving average (MA) and exponentially weighted moving average (EWMA), which mirror their univariate volatility counterparts.

Let $x_t = (x_{1,t}, \dots, x_{N,t})^\top$ denote the vector of monetary positions, and let $w_t = x_t / \sum_{i=1}^N x_{i,t}$ be the corresponding vector of portfolio weights. We allow monetary positions in an asset to be negative, representing short positions. However, for simplicity, we assume that the total portfolio value on any day is strictly positive.

In this setting, we adopt both a parametric approach—assuming normally distributed returns—and a semi-empirical method based on standardized innovations.

VaR is computed as:

$$\text{VaR}_t = z_\alpha \cdot \sqrt{w_t^\top \Sigma_t w_t},$$

where Σ_t is the time-varying covariance matrix estimated by the model.

Expected Shortfall is computed either using the analytical formula under the normal distribution or with the semi-empirical method used for univariate volatility models.

More advanced multivariate models, such as DCC-GARCH or VEC(1,1), are not included due to limited support in Python.

4.5 Factor Models

We implement the Sharpe model and Fama-French three factor models to estimate portfolio VaR and ES. These models use asset betas with respect to systematic risk factors, allowing us to infer portfolio-level risk from exposures to common sources of variation.

For the Sharpe Model, we assume that each asset's return r_i follows the linear factor structure:

$$r_i = \alpha_i + \beta_i r_m + \varepsilon_i,$$

where r_m is the return on the market portfolio, β_i is the asset's sensitivity to the market factor, α_i is

the asset's abnormal return, and ε_i is a zero-mean idiosyncratic shock uncorrelated with r_m and other residuals. The market return r_m has variance σ_m^2 , and $\text{Var}(\varepsilon_i) = \sigma_{\varepsilon_i}^2$. In this simplified setting, we assume the market volatility is constant over time.

The total portfolio variance is:

$$\sigma_p^2 = \left(\sum_{i=1}^N w_i \beta_i \right)^2 \sigma_m^2 + \sum_{i=1}^N w_i^2 \sigma_{\varepsilon_i}^2.$$

VaR and ES are computed as follows under the normality assumption:

$$\text{VaR}_\alpha = z_\alpha \cdot \sigma_p, \quad \text{ES}_\alpha = \sigma_p \cdot \frac{\phi(z_\alpha)}{\alpha}.$$

We also extend the previous model by considering three risk factors: market excess return (Mkt-RF), size (SMB), and value (HML). Each asset's excess return is modeled as:

$$r_i - r_f = \alpha_i + \beta_{i,1} \cdot \text{Mkt-RF} + \beta_{i,2} \cdot \text{SMB} + \beta_{i,3} \cdot \text{HML} + \varepsilon_i,$$

where r_f is the risk-free rate, and ε_i is the idiosyncratic error. We estimate factor loadings $\beta_{i,k}$ through OLS regression.

Let B be the $N \times 3$ matrix of estimated factor loadings, Σ_f the 3×3 sample covariance matrix of the factors, and Σ_ε the diagonal matrix of residual variances $\sigma_{\varepsilon_i}^2$. The total portfolio variance is:

$$\sigma_p^2 = w^\top (B \Sigma_f B^\top + \Sigma_\varepsilon) w.$$

Once the portfolio variance is known, we compute VaR and ES as we did for the single factor model.

4.6 Analytic VaR

We compute several Analytic VaR measures using the previously defined monetary position vector $x_t = (x_{1,t}, \dots, x_{N,t})^\top$ and the covariance matrix Σ .

The quantile z_α corresponds to the desired tail probability level and is always derived from the standard normal distribution, as the underlying assumption is that asset returns are normally distributed.

The asset-normal parametric portfolio VaR is defined as:

$$\text{VaR}_t = z_\alpha \cdot \sqrt{x_t^\top \Sigma x_t} \cdot \sqrt{h}.$$

Assuming all asset returns are perfectly correlated ($\rho = 1$) and there are no short positions, the undiversified portfolio VaR is simply the sum of the individual asset VaRs:

$$\text{UVaR}_t = \sum_{i=1}^N \text{VaR}_{i,t}.$$

The marginal VaR, representing the sensitivity of portfolio VaR to a small change in position $x_{i,t}$, is

given by:

$$\Delta \text{VaR}_{i,t} = \text{VaR}_t \cdot \frac{(\Sigma x_t)_i}{x_t^\top \Sigma x_t}.$$

It measures the change in total VaR from adding one extra unit of asset i .

The component VaR is:

$$\text{CVaR}_{i,t} = x_{i,t} \cdot \Delta \text{VaR}_{i,t},$$

which captures the absolute contribution of asset i to total portfolio VaR.

The relative component VaR is:

$$\text{RCVaR}_{i,t} = \frac{\text{CVaR}_{i,t}}{\text{VaR}_t},$$

interpreted as the percentage share of total VaR attributable to asset i .

For a vector a representing changes in the portfolio allocation, the incremental VaR is:

$$\text{IVaR}_t = \Delta \text{VaR}_t^\top \cdot a,$$

which estimates the change in total VaR resulting from shifting the portfolio by a .

Each of these VaR metrics can be complemented with a corresponding Expected Shortfall (ES), computed under the normality assumption.

4.7 Simulation Methods

Simulation methods are powerful tools for computing VaR and ES in portfolios that include complex instruments such as options, which exhibit non-linear payoffs.

We implement a Monte Carlo simulation approach to estimate the one-day ahead distribution of portfolio returns under the assumption of multivariate normality.

Future returns are simulated as:

$$\mu = \mathbb{E}[r_t], \quad \Sigma = \text{Cov}(r_t), \quad L = \text{Cholesky}(\Sigma),$$

$$r^{(i)} = \mu + Lz^{(i)}, \quad z^{(i)} \sim \mathcal{N}(0, I),$$

$$S^{(i)} = S_0 \circ (1 + r^{(i)}),$$

where \circ denotes element-wise multiplication.

The simulated portfolio profit and loss is calculated as:

$$\Delta P^{(i)} = w^\top (S^{(i)} - S_0) + \sum_{j=1}^{N_{\text{opt}}} q_j \left(C(S_j^{(i)}, \tau_j') - C_{j,0} \right),$$

where q_j is the number of option contracts, $C(\cdot)$ is the Black-Scholes option price, and $\tau_j' = \max(T_j - \frac{1}{252}, 0)$ is the adjusted time to maturity.

We define simulated portfolio losses as $\ell^{(i)} = -\Delta P^{(i)}$. From the empirical distribution $\{\ell^{(i)}\}_{i=1}^N$, we

compute:

$$\text{VaR}_\alpha = Q_\alpha(\ell), \quad \text{ES}_\alpha = \mathbb{E}[\ell \mid \ell \geq \text{VaR}_\alpha].$$

We also implement a multi-day version of Monte Carlo, which forecasts VaR and ES over a longer horizon. This version is limited to pure equity portfolios.

As a non-parametric alternative, we implement Historical Simulation and its bootstrap variant (with replacement). Instead of generating returns from a parametric distribution, we simulate alternative return paths by re-applying historical or resampled return vectors to the current portfolio.

Losses $\ell^{(i)}$ are computed as in the Monte Carlo method. VaR and ES are then estimated from their empirical distribution in the same way.

This approach is fully non-parametric and captures fat tails and nonlinear dependence in historical returns. When bootstrap is enabled, repeated resampling improves robustness in small-sample settings.

4.8 Backtesting

Backtesting assesses whether the observed losses are consistent with the VaR predictions by checking the frequency and structure of violations (exceptions). The objective is to test whether the model is correctly calibrated.

We implement three standard tests.

The Kupiec test (unconditional coverage) evaluates if the number of observed exceptions N over T days matches the expected number under the model. If p is the failure probability, then under the null N follows a binomial distribution. The likelihood ratio test statistic is:

$$\text{LR}_{\text{uc}} = -2 \left\{ \ln [(1-p)^{T-N} p^N] - \ln \left[\left(1 - \frac{N}{T}\right)^{T-N} \left(\frac{N}{T}\right)^N \right] \right\}, \quad \text{LR}_{\text{uc}} \sim \chi_1^2.$$

The Christoffersen test (conditional coverage) tests the independence of exceptions by modeling their dynamics as a first-order Markov chain. It compares the likelihoods of transition counts under the null (independence) and alternative. The statistic is:

$$\text{LR}_c = -2(\ln L_0 - \ln L_1), \quad \text{LR}_c \sim \chi_1^2,$$

where L_0 is the likelihood under independence and L_1 under the alternative.

The joint test combines both:

$$\text{LR} = \text{LR}_{\text{uc}} + \text{LR}_c \sim \chi_2^2.$$

Rejecting the null in any of the tests suggests that the VaR model is either miscalibrated (too many or too few violations) or fails to capture time dependence in risk (e.g., volatility clustering).

Our back-testing framework covers every method except the simulation models. Each simulation delivers only a single-day (or multiple day) forecast; to back-test them we would need to rerun a full rolling simulation at every step—a task that is both logically more involved and computationally heavy—so it lies outside the present scope.

5 Results and Applications

All data used in this section are daily returns, downloaded from Yahoo Finance. All VaR and ES metrics reported are computed for a one-day horizon at a 99% confidence level.

We present backtesting results for two VaR models under a simple strategy: investing USD 100 000 in the Vanguard Total World Stock ETF (VT). The analysis covers the period from June 27, 2008, to January 1, 2025.

The Parametric Normal model recorded 77 violations (1.85%), with highly significant test statistics for both Kupiec and Christoffersen tests, indicating poor fit. As seen in Figure 1 in the appendix, this model assumes i.i.d. returns and fails to account for time-varying volatility, which contributes to its inability to pass the independence and conditional coverage tests.

In contrast, the GJR-GARCH with skewed- t innovations model had only 42 violations (1.01%) and passed all backtesting tests with high p-values, indicating better accuracy and reliability. This was one of the best performing models in our evaluation (see Figure 2 in the appendix).

These results highlight that the normality assumption is too strong for daily financial returns, even for relatively stable assets such as the ETF considered.

We also report results for a globally diversified portfolio (value: CHF 105 181.93) including equities and options (value: CHF 1 106.37, priced via the BS model). The analysis covers the period from January 1, 2023, to January 1, 2025. The Monte Carlo simulation estimated a 1-day ahead VaR of CHF 2 481.02 and ES of CHF 2 835.28, showing a nearly normal P&L distribution (Figure 3 in the appendix). In comparison, the Historical Simulation yielded a higher VaR (CHF 2 582.21) and ES (CHF 3 356.69), with a clearly non-normal distribution that better captures heavy tails and asymmetry (Figure 4 in the appendix).

The full detailed analysis and many more results can be found in the notebook `pyvar_example_usage.ipynb`, available in the `examples/` folder of the GitHub repository.

6 Lessons Learned

This project was a valuable exercise in programming style and software design. While the risk models themselves are standard for a graduate-level course in Quantitative Risk Management, the main challenge was building a coherent, user-friendly, and well-structured package. It also allowed us to refresh and deepen our understanding of risk modeling, applied statistics, and new tools like LLMs.

7 AI Acknowledgement

We used ChatGPT-4o extensively for coding support—mainly to generate pseudocode or early drafts, which we refined and corrected. GitHub Copilot also assisted with code completion. For theory extraction (e.g., formulas from books/slides), Notebook LM was particularly helpful. ChatGPT-o3 supported us with theoretical clarifications, while ChatGPT-4o was also used for LaTeX help and documenting our

code with clean docstrings.

8 Link to GitHub Repository

The full codebase, examples, and documentation are available on GitHub: <https://github.com/Alessandro-Dodon/pyvar>

9 References

The following textbooks constitute the main references for this project:

- John C. Hull, *Risk Management and Financial Institutions*, 2023, John Wiley & Sons.
- Peter Christoffersen, *Elements of Financial Risk Management* (2nd Edition), Academic Press, 2011.
- Philippe Jorion, *Value at Risk*, McGraw-Hill, 2006.
- Andrea Resti and Andrea Sironi, *Risk Management and Shareholders' Value in Banking*, John Wiley & Sons, 2007.
- John C. Hull, *Options, Futures, and Other Derivatives*, latest edition, Pearson.

Additional insights and context were drawn from the following MSc in Quantitative Finance courses at USI:

- *Risk Management*, taught by Prof. A. Plazzi
- *Derivatives and Financial Econometrics*, taught by Prof. L. Mancini

In particular, we referred to lecture slides and coursework provided throughout the semester as supplementary material.

10 Appendix

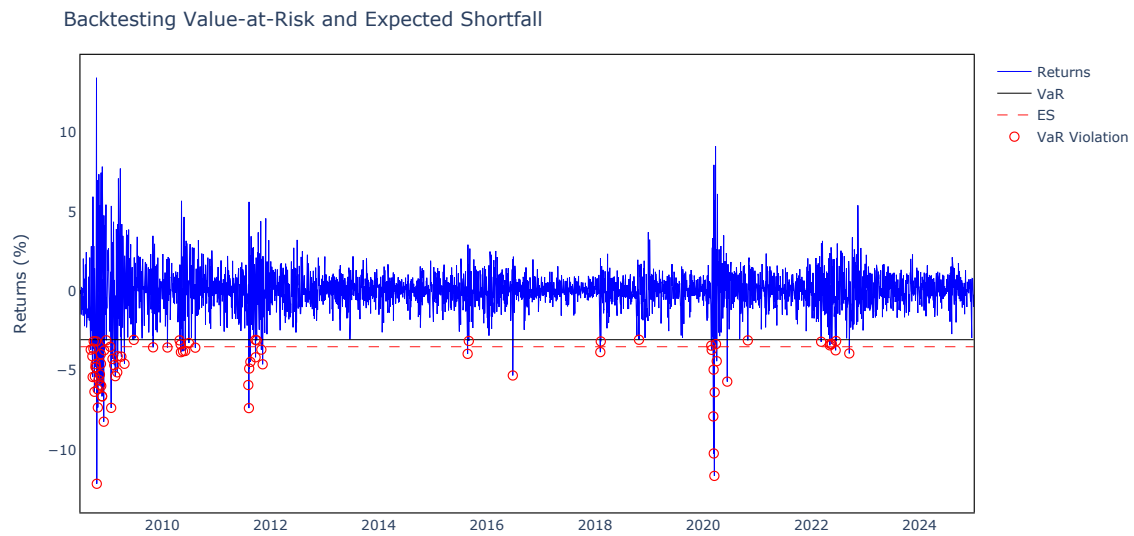


Figure 1: Backtesting results for the Parametric Normal model.

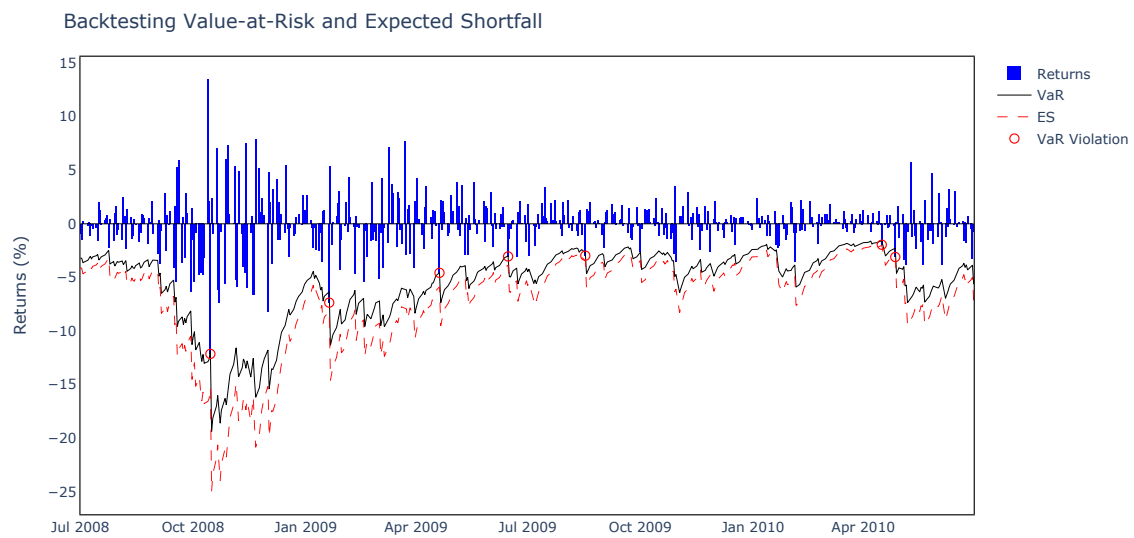


Figure 2: Backtesting results (subset) for the GJR-GARCH model with skewed- t innovations.

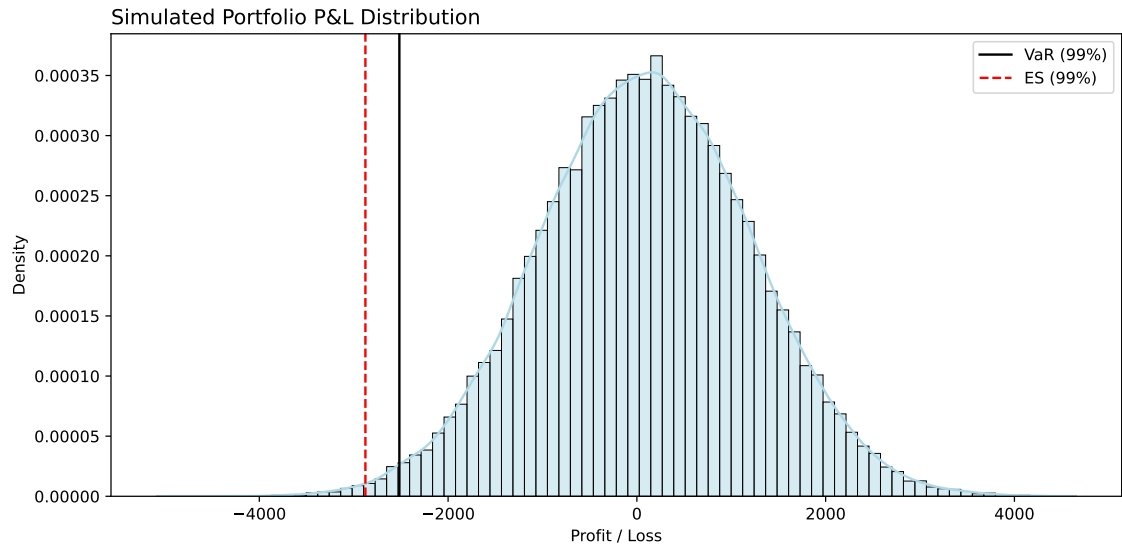


Figure 3: Monte Carlo simulation results for the globally diversified portfolio.

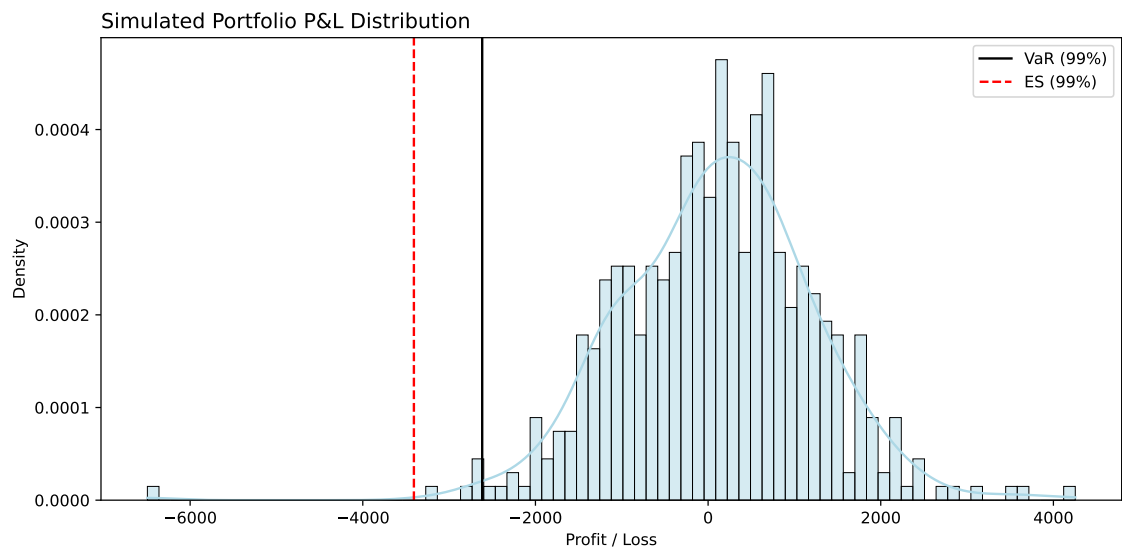


Figure 4: Historical simulation results for the same portfolio.