

1 *MC_VaR_CVaR.py — Portfolio Monte Carlo with Risk Metrics*

Purpose

Simulates the distribution of portfolio returns and values over a year, then calculates **Value at Risk (VaR)** and **Conditional Value at Risk (CVaR)** at a chosen confidence level.

Learning focus

- **VaR**: “With 95% confidence, losses will not exceed X.” This is the 5th percentile of the simulated distribution.
- **CVaR**: “If we end up in the worst 5% of cases, the average loss will be Y.” This captures tail severity.
- The script uses historical mean returns and covariance between assets, preserving correlations via Cholesky decomposition.

Key intuition

VaR gives a threshold; CVaR tells you about the depth of the cliff beyond it. In practice, regulators often require CVaR-type measures (expected shortfall) because they are more robust in fat-tailed markets.

Experiment

Increase the number of simulations (`mc_sims`) and compare how stable your VaR and CVaR become. You’ll see more simulations lead to more reliable estimates.

2 *MC_Stock_Portfolio.py — General Monte Carlo Portfolio Projection*

Purpose

Runs Monte Carlo simulations of a multi-asset stock portfolio over a fixed horizon, returning probability distributions for future portfolio values.

Learning focus

- Allows flexible weight allocation (uniform or random).
- Preserves historical return correlations using Cholesky decomposition.
- Outputs percentile paths (10th, median, 90th) to visualise pessimistic, base, and optimistic scenarios.

Key intuition

The percentile bands act like **fan charts** — traders can visualise likely ranges for portfolio value and plan drawdown tolerances. It is also a way to stress-test the impact of correlations: high correlations in bear markets shrink diversification benefits.

Experiment

Set weights to concentrate heavily in one stock and observe how the percentile spread widens. This demonstrates concentration risk.

Connection between the two scripts

- **MC_Stock_Portfolio.py** focuses on projecting possible portfolio paths and summarising potential outcomes.
- **MC_VaR_CVaR.py** adds a risk-focused lens, extracting specific tail-risk metrics from those distributions.

In a professional trading context, both are complementary:

1. Run simulations to understand the range of possible portfolio values.
2. Extract VaR and CVaR to meet risk policy and regulatory requirements.