Financial Market Uncovered – Article 5

Monte Carlo Simulation for Risk Measurement: Estimating Value at Risk and Expected Shortfall

Kilian Voillaume

April 7, 2025

Summary

1 Introduction

In modern finance, risk is no longer seen as a vague notion of uncertainty but as a quantifiable object—one that must be modelled, measured, priced, and managed. As financial markets grow more complex and interconnected, with increasingly non-linear and path-dependent exposures, the ability to estimate risk precisely has become both a theoretical challenge and a regulatory imperative.

Among the most widely adopted metrics for quantifying market risk are Value at Risk (VaR) and Conditional Value at Risk (CVaR).

- VaR aims to estimate the potential loss of a portfolio over a given time horizon with a specified confidence level.
- In contrast, CVaR—also known as Expected Shortfall—measures the expected loss in the tail, conditional on losses exceeding the VaR threshold.

While VaR has long been the industry standard, CVaR has emerged as a more coherent and robust measure, now recommended under regulatory frameworks such as Basel III and IV.

However, traditional methods for computing VaR—such as the parametric (variance-covariance) approach or historical simulation—rely on strong assumptions about the distribution of returns, and often fail to capture the complexities of modern portfolios. These include:

- Non-linear instruments such as options
- Regime shifts in volatility
- Fat-tailed distributions
- Dynamic correlations

In such settings, Monte Carlo simulation provides a more flexible and theoretically sound alternative. By generating a large number of possible future states of the portfolio through random sampling, it enables the estimation of full empirical loss distributions without restrictive analytical assumptions.

Monte Carlo-based approaches allow for:

- Modelling realistic dynamics (e.g., stochastic volatility, jump processes),
- Incorporating non-linear payoffs and portfolio sensitivities (Greeks),
- Estimating tail risk metrics (like CVaR) with controlled accuracy.

Despite its computational intensity, simulation is increasingly favoured in both institutional and academic settings for its generality, adaptability, and transparency.

2 Foundations of Monte Carlo Simulation

Monte Carlo simulation is one of the most widely used numerical techniques in financial risk analysis. Its core principle is simple: when analytical formulas are unavailable or unreliable, we estimate results by generating a large number of hypothetical outcomes and observing their distribution.

This approach is particularly useful when modelling the behaviour of complex portfolios, where returns may depend on multiple risk factors, exhibit non-linear sensitivities, or deviate from normal distributional assumptions. Rather than relying on closed-form approximations, Monte Carlo allows us to construct a full distribution of potential outcomes by simulating many possible future scenarios.

The method is grounded in the law of large numbers, which guarantees convergence of the simulated average to the true expectation as the number of simulations increases.

Formally, consider a random variable X, representing the future loss (or return) of a portfolio, and let $E[X]$ denote its expected value. If we can simulate N independent realizations $X_1, X_2, \dots, X_N$ from the distribution of X, then the Monte Carlo estimator of the expectation is given by:

$$\hat{\mu}_N = \frac{1}{N}\sum_{i=1}^{N} X_i$$

This estimator is unbiased and, under mild regularity conditions, satisfies a central limit theorem:

$$\sqrt{N}(\hat{\mu}_N - \mu) \overset{d}{\to} N(0, \sigma^2)$$

where σ^2 is the variance of X. The convergence rate of the estimator is $\mathcal{O}(1, \sqrt{N})$, which implies that to reduce the standard error by a factor of 10, the number of simulations must increase by a factor of 100. This quadratic cost underpins the main practical limitation of Monte Carlo methods: computational intensity.

In plain English:

When you take a large number of simulations (or samples), the average result you get $\widehat{(\mu_N})$ gets closer to the true average (μ). But even with lots of simulations, there's still some random fluctuation — and that fluctuation behaves like a normal distribution as you increase N.

The more simulations you run, the smaller the error becomes. Specifically:

- The error shrinks at a rate of $\frac{1}{\sqrt{N}}$.
- This helps us quantify the uncertainty in our estimate.
- It tells us we can trust the simulated average as a good approximation — especially when N is large.

2.1 Simulating portfolio losses

In the context of risk measurement, Monte Carlo simulation proceeds by generating a large number of hypothetical future portfolio returns or losses over a fixed time horizon (e.g., 10 days). These are based on a stochastic model of the underlying risk factors—such as asset prices, interest rates, or volatilities. Each simulation represents one plausible trajectory of market movements.

Once we have simulated many such outcomes, we can construct an empirical distribution of portfolio losses. From this, any risk metric that is a functional of the loss distribution—such as Value at Risk (VaR) or Conditional Value at Risk (CVaR)—can be estimated numerically.

2.2 Key advantages of Monte Carlo methods

The strength of the Monte Carlo approach lies in its flexibility:

- It is model-agnostic—applicable to normal and non-normal return distributions.
- It handles non-linear exposures (such as options) with ease.
- It scales naturally to multi-asset portfolios, accounting for correlation between assets.

This generality makes Monte Carlo simulation particularly well-suited to real-world portfolios where simplifying assumptions (like normality or linearity) break down.

2.3 Computational challenges

The main trade-off for this generality is computational cost. Because the convergence of Monte Carlo estimators is relatively slow (proportional to $\frac{1}{\sqrt{N}}$), high accuracy in the estimation of tail risk metrics like CVaR often requires thousands to millions of simulations.

To address this, practitioners often rely on:

- Variance reduction techniques (e.g., antithetic variates, importance sampling),
- Parallel computation (e.g., GPUs, cloud clusters),
- and efficient sampling methods (e.g., quasi-Monte Carlo, low-discrepancy sequences).

Despite these challenges, Monte Carlo simulation remains the method of choice in many institutional settings due to its robustness, interpretability, and extensibility.

2.4 Monte Carlo simulation in Python

To bring theory into practice, we simulate a portfolio of four technology stocks—Apple (AAPL), Google (GOOGL), Amazon (AMZN), and Microsoft (MSFT)—using a Monte Carlo framework. Our objective is to project the portfolio's potential future value over a one-year horizon (252 trading days) and observe the distribution of outcomes.

Simulation Parameters

- *Initial investment:* $10,000

- *Number of simulations:* 120
- *Time horizon:* 252 trading days (1 year)
- *Assets:* AAPL, GOOGL, AMZN, MSFT
- *Assumptions:* Returns follow a geometric Brownian motion (GBM) with drift and volatility calibrated from historical data.

We assume log-returns are normally distributed and simulate daily portfolio values under these dynamics. While 120 simulations offer an illustrative distribution, practitioners would typically run thousands or millions of paths for more precise tail estimates.

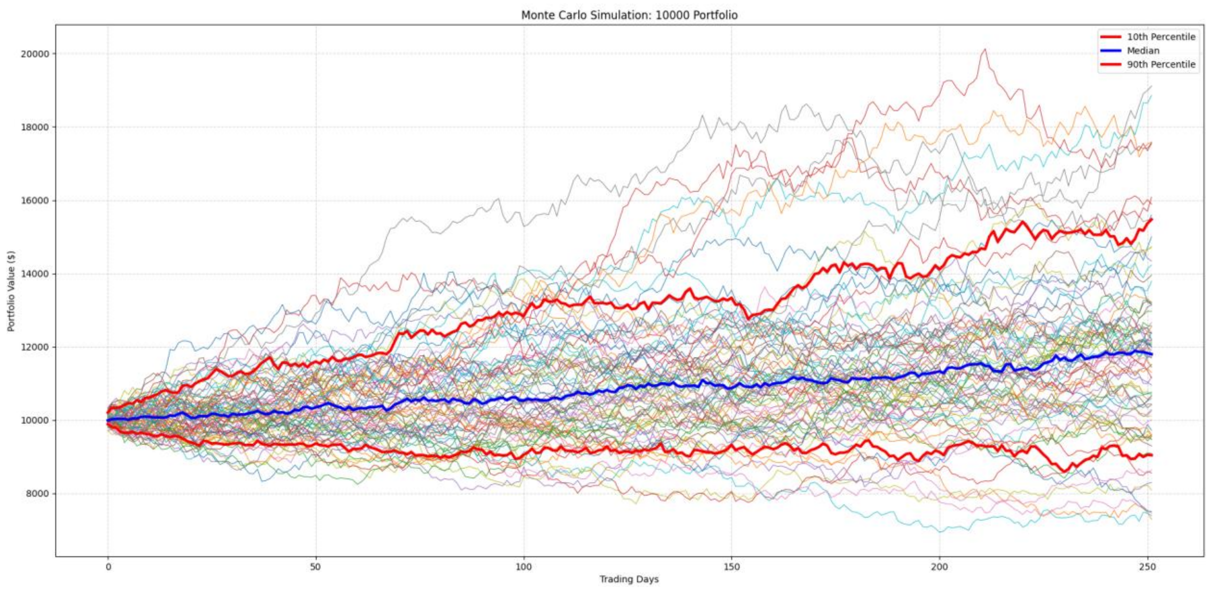

Figure 1: Monte Carlo simulation with percentile

As seen in the graph, the simulated portfolio values diverge over time, reflecting the compounding uncertainty in asset prices. The fan-shaped dispersion illustrates the distribution of possible portfolio outcomes—a central idea in risk modelling. The percentile bands give a natural visual reference for later discussion of Value at Risk (VaR) and Conditional Value at Risk (CVaR).

This simulation specifically gives the following summary:

Ticker	Mean returns	Portfolio wieghts
AAPL	0.001167	43.3%
AMZN	0.000633	22.74%
GOOGL	0.000629	19.96%
MSFT	-0.000171	13.99%

Final Portfolio Value Statistics:	
Minimum:	$7 295.91
Maximum:	$19 115.45
Mean:	$11 913.45
Median:	$11 797.80

3 Value at Risk (VaR): Quantifying the worst

In financial markets, risk is never about what happens *on average*—it's about what happens *at the extremes*. When portfolio managers, regulators, or investors ask how bad things could get, they are rarely interested in small fluctuations. They want to know: *What's the worst I can expect under normal conditions?* And that's exactly the question that Value at Risk (VaR) tries to answer.

Over the past few decades, VaR has become one of the most widely used metrics in risk management. Its appeal lies in its apparent simplicity: VaR expresses the worst expected loss over a given time horizon, with a given level of statistical confidence. For example, a daily 99% VaR of $1 million means that, under normal market conditions, there is only a 1% chance that the portfolio will lose more than $1 million in a single day.

3.1 Formal definition

To define VaR rigorously, we consider a random variable L, representing the potential *loss* on a portfolio over a fixed time horizon. The Value at Risk at confidence level $\alpha \in (0,1)$— typically 95% or 99% — is the smallest number ℓ such that the probability of the loss being less than or equal to ℓ is at least α. In mathematical terms:

$$VaR_\alpha(L) = \inf\{\ell \in \mathbb{R}: \mathbb{P}(L \leq \ell) \geq \alpha\}$$

In plain English:

VaR at confidence level α is the smallest loss level such that the probability of losses being less than or equal to it is at least α. In simpler terms: it's the "worst loss" you expect not to exceed, most of the time.

For instance, if you calculate the 95% VaR to be $1200, you are saying that with 95% confidence, you don't expect to lose more than $1200 over the period. Or inversely, there's a 5% chance you might lose more than $1200."

This definition makes VaR a quantile of the loss distribution. For a 95% confidence level, it corresponds to the 95th percentile of potential losses. It draws a line in the sand: on one side, we have outcomes that are acceptable or expected; on the other side, we have the extreme losses we hope to avoid.

3.2 Estimating VaR with Monte Carlo simulation

In practice, we often do not know the analytical form of the portfolio loss distribution, especially when dealing with complex or nonlinear instruments. This is where Monte Carlo simulation becomes extremely powerful. Instead of relying on simplifying assumptions — such as normally distributed returns or constant volatility—we generate thousands of hypothetical future outcomes for the portfolio based on a stochastic model of market behaviour.

For each scenario, we simulate the evolution of asset prices, compute the portfolio value at the end of the horizon, and calculate the resulting loss. This process yields a large collection of potential losses, from which we can construct an empirical loss distribution. Estimating VaR then becomes a matter of sorting these losses and selecting the appropriate percentile.

Suppose we simulate N portfolio losses $L_1, L_2, \dots, L_N$. The α-quantile of this empirical distribution gives us the estimated Value at Risk:

$$\widehat{VaR}_{\alpha} = Quantile_{\alpha}(\{L_i\}_{i=1}^{N}$$

In plain English:

We simulate a large number of losses, sort them from best to worst, and pick the one that lies right at the boundary—for example, the 95th worst loss out of 100. That's your VaR.

In this approach, the flexibility of the Monte Carlo method allows us to capture the full richness of the underlying portfolio dynamics — including fat tails, skewness, and nonlinear risk exposures—without relying on unrealistic distributional assumptions.

3.3 Reading VaR in practice

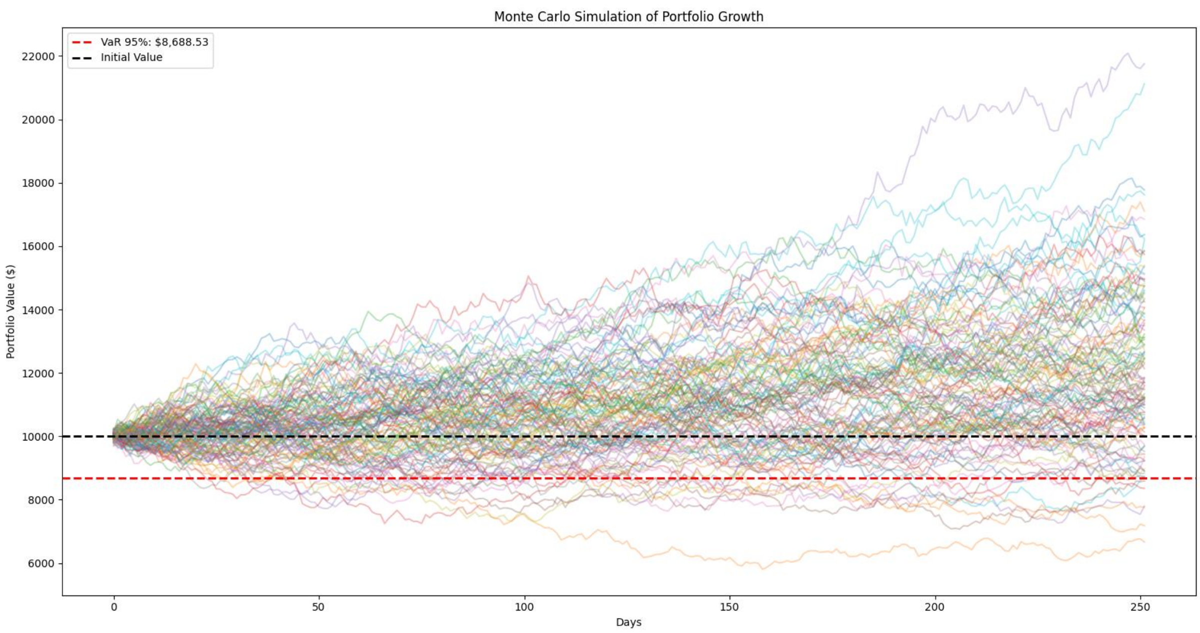

Figure 2: VaR applied to Monte Carlo simulation

This chart shows all 120 simulated daily portfolio paths over the 252-day horizon.

Interpretation:
Each line represents one plausible trajectory for your portfolio, simulated using historical dynamics calibrated to historical data. Most paths stay above the VaR threshold, but a few fall below—those are the worst 5% outcomes we want to guard against.

This figure is powerful because it shows how VaR emerges from a sea of potential futures: it's not an abstract statistical idea, but a real cut-off drawn through a cloud of uncertain outcomes. The lower edge of the distribution, as traced by the red line, represents the boundary of acceptable risk for a given confidence level.

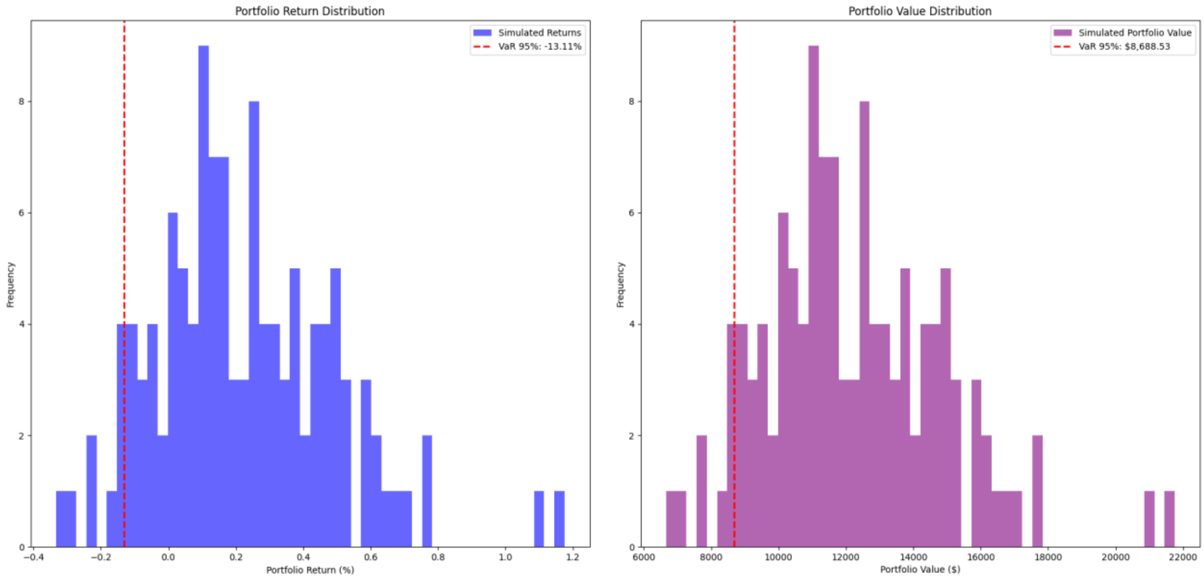

Figure 3: VaR applied to portfolio distribution

Left Panel — Portfolio Return Distribution

This histogram shows the frequency of simulated annual returns across the 50 Monte Carlo paths. Most outcomes are centered around small positive returns, but there's a noticeable left tail—representing unfavourable outcomes. The red dashed line marks the 5th percentile of the distribution, corresponding to a 95% Value at Risk (VaR). In this case, the VaR is approximately –13.11%.

Interpretation:
There is a 5% chance that your portfolio could lose more than 13.11% of its value over the year. Conversely, with 95% confidence, your losses should stay within that bound.

Right Panel — Portfolio Value Distribution

Here, instead of returns, we look at the ending dollar values of the portfolio after 252 trading days. Again, we see the red dashed line marking the 5th percentile. This corresponds to a final value of $8,688.53, meaning a loss of $1,311.47 from the initial $10,000.

Interpretation:
At 95% confidence, your portfolio should not fall below $8,688.53 over the year — this is the dollar VaR estimate. The 5% worst-case scenarios result in values below this threshold.

This dual view — in returns and in actual dollar values — helps make VaR more tangible. While the return percentage gives a normalized risk measure, the dollar loss is more intuitive for investors and managers.

3.4 What VaR tells us

VaR gives a precise answer to a complex question. But it's important to remember what it does—and what it doesn't do.

- VaR tells us about the maximum expected loss under normal conditions, with a specified level of confidence. It's a useful tool for reporting, benchmarking, and setting regulatory capital. It's also relatively easy to communicate: a single number, tied to a specific level of risk tolerance.
- VaR does not say anything about what happens *beyond* the threshold. It tells us how bad things can get with 95% or 99% confidence—but it's silent about that final 5% or 1%. In fact, VaR cuts off the tail of the distribution entirely. In extreme situations—such as financial crises—this is precisely the part of the distribution we most care about.

Moreover, VaR is not always a *coherent* risk measure in the mathematical sense: in particular, it can fail the property of subadditivity, meaning that the VaR of a combined portfolio can exceed the sum of the individual VaRs. This can lead to misleading assessments of diversification benefits.

3.5 Time horizons and risk scaling

VaR is defined for a given time horizon—a day, a week, a year—and its value changes accordingly. In practice, regulatory VaR is often calculated over a 10-day horizon, while trading desks may focus on daily VaR.

Under the assumption of normally distributed returns, the square-root-of-time rule is often used to scale VaR across time:

$$VaR_T \approx VaR_1 * \sqrt{T}$$

However, this approximation relies on strong assumptions: it assumes that returns are independent and identically distributed ($i.i.d.$), which is rarely true in real markets. Monte Carlo simulation offers a more reliable alternative, as it allows us to directly simulate losses over any desired horizon, using realistic dynamics.

3.6 Beyond VaR

Despite its limitations, VaR remains an industry standard—and a natural entry point into the world of quantitative risk management. But as we have seen, it leaves out a critical piece of information: what happens in the worst cases, when losses exceed the VaR threshold.

To address this, we turn to a more refined measure of tail risk: Conditional Value at Risk (CVaR). Rather than drawing a line and stopping, CVaR looks beyond—and asks how bad are the bad outcomes?

In the next section, we explore the definition, properties, and estimation of CVaR, and explain why it has gained increasing prominence as a more coherent and informative measure of risk.

4 Conditional Value at Risk (CVaR): Beyond the threshold

Value at Risk (VaR) gives us a clear summary: it tells us how much we stand to lose, with a given level of confidence, over a specified horizon. But it leaves out a critical question—what happens if things go worse than VaR?

This is where Conditional Value at Risk (CVaR) comes in. Also known as Expected Shortfall, CVaR doesn't stop at the threshold. Instead, it digs deeper into the tail of the distribution and answers a more sobering question:

If a loss worse than VaR happens—how bad might it be, on average?

CVaR is a natural and powerful complement to VaR. Rather than focusing on the limit of acceptable losses, it tells us about the expected magnitude of the unacceptable ones. For this reason, CVaR is widely considered to be a more robust and informative measure of risk—especially in environments with fat tails, skewness, or extreme events.

4.1 *Formal definition*

Let L be a random variable representing the portfolio loss. For a confidence level α, the Conditional Value at Risk at level α, denoted $CVaR_\alpha$, is defined as:

$$CVaR_\alpha(L) = \mathbb{E}[L|L \geq VaR_\alpha(L)]$$

Plainly stated: CVaR is the average loss in the worst $(1-\alpha)\%$ of scenarios. It's what you expect to lose *if* you're already in the tail beyond VaR.

So, if your 95% VaR is \$1,200, the CVaR might be \$2,000. That means while you're 95% confident you won't lose more than \$1,200, if you're unlucky enough to land in the worst 5% of outcomes, you might lose around \$2,000 on average.

4.2 *Estimating CVaR with Monte Carlo simulation*

Monte Carlo simulation gives us a practical and accurate way to estimate CVaR—even in the presence of complex dynamics or non-Gaussian features.

Here's how we do it in practice:

1. Simulate a large number of potential future losses: $L_1, L_2, \dots, L_N$
2. Estimate VaR at confidence level α by taking the α-quantile of the sorted losses.
3. Select the tail: find all simulated losses L_i such that $L_i \geq VaR_\alpha$
4. Take the average of these worst-case losses.

Formally, the Monte Carlo estimator for CVaR becomes:

$$\widehat{CVaR}_\alpha = \frac{1}{M}\sum_{i=1}^{M} L_i, where\ L_i \geq \widehat{VaR}$$

where M is the number of losses that fall in the tail (i.e., beyond the VaR threshold).

Intuition: If VaR tells you that you might lose up to $X, CVaR tells you that if things go that badly, you will probably lose $Y on average.

This makes CVaR especially useful for capital allocation, stress testing, and scenario planning—all situations where tail risk really matters.

4.3 Why CVaR is a better risk measure

In many respects, CVaR improves upon VaR — both theoretically and practically:

- It captures tail risk: Unlike VaR, which ignores what happens after the cutoff, CVaR quantifies it.
- It is coherent: CVaR satisfies all the properties of a good risk measure — including subadditivity, which ensures that diversification always helps.
- It is more sensitive: CVaR reacts to changes in the tail shape, such as fatter or skewed tails — something VaR often overlooks.

In fact, regulatory frameworks like Basel III and Basel IV now recommend the use of CVaR (under the name "expected shortfall") for setting capital requirements, particularly for trading books.

4.4 Reading CVaR in practice

To see how Conditional Value at Risk deepens our understanding of risk, we turn to two visualizations generated from the Monte Carlo simulation.

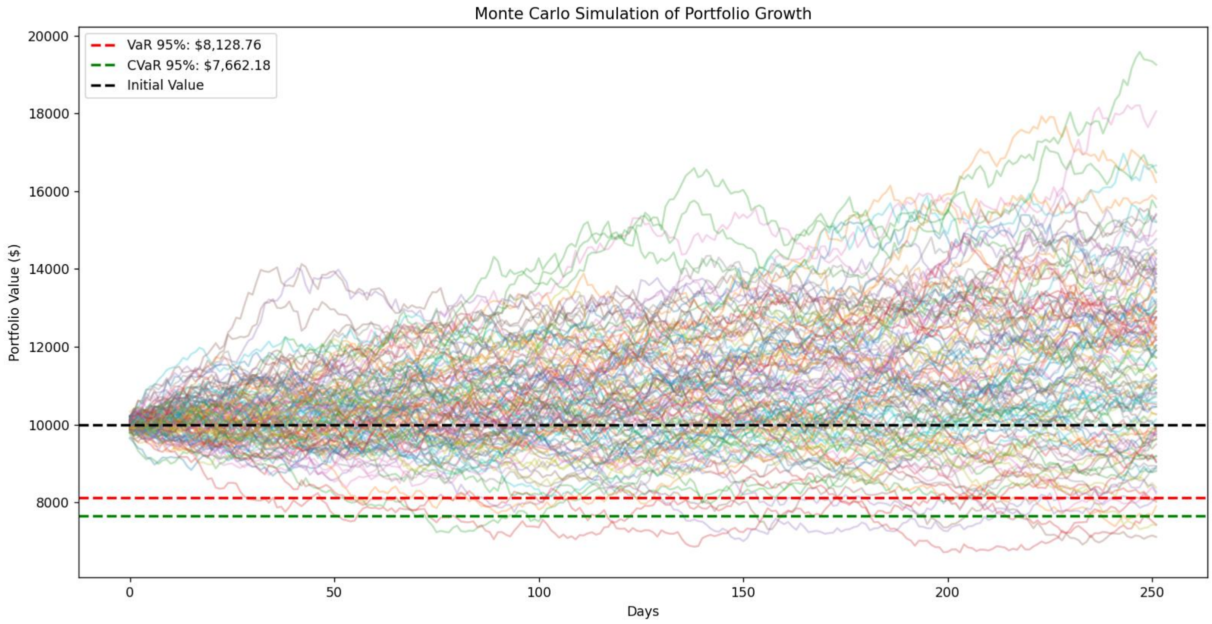

Figure 4: CVaR applied to Monte Carlo simulation

This figure shows 120 simulated trajectories of the portfolio value over a one trading year horizon. The black dashed line marks the initial portfolio value ($10,000), while two horizontal thresholds are superimposed:

- *Red dashed line:* 95% Value at Risk, at $8,128.76
- *Green dashed line:* 95% Conditional Value at Risk, at $7,662.18

We immediately see that the red line marks the boundary below which lie the worst 5% of outcomes. But the green line, slightly lower, goes a step further. It represents the average of those worst outcomes. This is the essence of CVaR: it not only asks where the danger zone begins, but also how deep it goes.

While VaR tells us that 95% of the time, we shouldn't lose more than $1,871, CVaR tells us that if we do fall into the worst 5%, the average loss will be $2,337.82. In crisis scenarios — where tail events drive losses — this distinction is not just technical, it's essential.

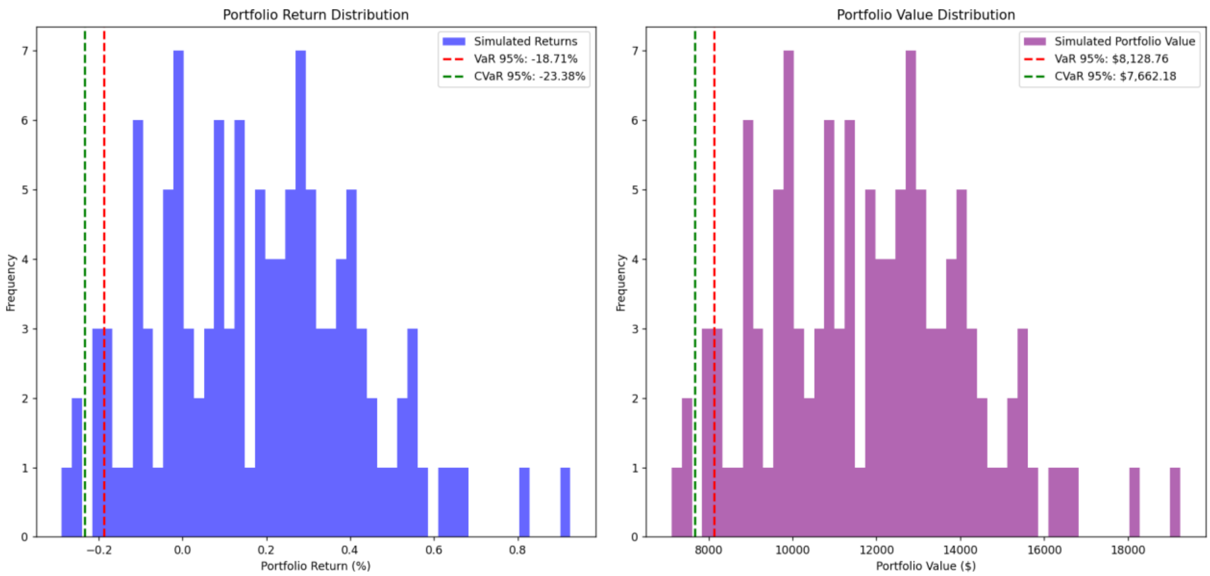

Figure 5: CVaR applied to portfolio distribution

This second figure offers two histograms. On the left, the distribution of simulated returns (in %). On the right, the distribution of final portfolio values (in $)

In both, the red and green dashed lines represent:

- VaR (95%) — the 5th percentile (cutoff point)
- CVaR (95%) — the average of outcomes below that cutoff

The left panel shows that the worst 5% of returns are concentrated below –18.71%, and that the average of those is –23.38%. On the right, the same information is translated into dollar terms. The cutoff (VaR) is $8,128.76—but the mean of the values in the tail (CVaR) is even lower, at $7,662.18.

4.5 Reflections on implementation

While this simulation illustrates the Monte Carlo method clearly, it’s important to remember that real financial markets are messier than Gaussian models (Geometric Brownian Motion or log-normal returns) suggest. Some of the common extensions in professional environments include:

- Non-Gaussian return models (e.g., Student-t, skewed distributions)
- Time-varying volatility (GARCH, stochastic volatility)
- Jump-diffusion or regime-switching models
- Higher path counts (thousands to millions) for stability in CVaR estimation
- Full revaluation of portfolios with derivatives, options, and path dependencies

Nevertheless, even with basic assumptions, Monte Carlo simulation proves to be a powerful, flexible, and transparent way to measure risk — especially when used to estimate tail-sensitive metrics like CVaR.

5 Strengths and limitations of Simulation-based risk estimation

Monte Carlo simulation has emerged as a cornerstone technique in modern financial risk management. As we have seen throughout this article, it provides a highly flexible and intuitive framework for estimating Value at Risk (VaR) and Conditional Value at Risk (CVaR).

But while the method is conceptually elegant and broadly applicable, it is not without challenges. Like any tool, its value depends on the context in which it is used — and the care with which it is implemented.

5.1 Why Monte Carlo matters

The main strength of Monte Carlo simulation lies in its generality. It makes very few assumptions about the underlying structure of returns or asset behaviour. As long as we can model the process that drives portfolio outcomes — even approximately — we can simulate it, observe the results, and compute relevant risk metrics from the empirical distribution.

This is particularly valuable in situations where:

- The portfolio contains options or other derivatives, which have nonlinear payoffs.
- The return distribution is asymmetric or heavy-tailed, violating normality assumptions.
- There are multiple correlated assets, requiring joint simulation of their dynamics.
- The risk horizon is longer than a single day, making path dependency relevant.

In all these cases, standard analytical methods break down — but Monte Carlo remains applicable. For this reason, it is widely used not only in banks and hedge funds, but also in central clearing houses, stress testing, and regulatory capital modelling.

5.2 The importance of CVaR

While VaR continues to be a standard benchmark, it is increasingly recognized that CVaR provides a more complete picture of tail risk. Regulators such as the Basel Committee have adopted expected shortfall (CVaR) as a core risk metric, precisely because it captures the magnitude of losses in rare but devastating scenarios.

Monte Carlo simulation, by generating full empirical distributions, is uniquely suited to estimate both VaR and CVaR in parallel — and to illustrate the difference between them clearly.

5.3 Limitations and practical considerations

Despite its strengths, simulation comes with real-world limitations. First and foremost is its computational intensity. The convergence of Monte Carlo estimators is slow: halving the standard error requires quadrupling the number of paths. In practice, this can translate into hours or even days of runtime for large portfolios — unless one uses variance reduction, parallel processing, or quasi-Monte Carlo techniques.

Moreover, simulation is only as good as the model behind it. If the assumptions used to simulate returns are poorly chosen — for instance, assuming constant volatility in a highly turbulent environment — the resulting VaR and CVaR estimates may be misleading. As always, the output is only as robust as the inputs.

Finally, there is a danger of treating simulation as a black box. Without proper understanding of its structure and assumptions, users may place undue trust in precise-looking numbers that are highly sensitive to model risk and parameter choices.

6 Conclusion

Risk is at the heart of finance — but it is not always visible. It hides in the tails of distributions, in the complexity of portfolios, and in the rare events that standard models often fail to capture. This is why simulation-based risk estimation is so powerful: it gives us the ability to lift the veil of uncertainty and see the range of outcomes a portfolio might face.

Throughout this article, we explored how Monte Carlo simulation provides a flexible and intuitive framework to model uncertainty, especially when analytical formulas fall short. By generating a full empirical distribution of potential portfolio outcomes, we can estimate both Value at Risk (VaR) — the boundary between acceptable and extreme losses — and Conditional Value at Risk (CVaR) — the expected size of those extreme losses when they occur.

We showed that:

- VaR, while widely used, can be misleading if taken alone — because it ignores what lies beyond the threshold.
- CVaR, or expected shortfall, offers a deeper insight into the shape and severity of tail risk.
- Monte Carlo simulation, despite its computational cost, enables the estimation of both risk measures in highly realistic settings — with minimal assumptions and maximal transparency.

We illustrated these ideas with practical examples, visualizations, and Python implementations, showing how even a basic Monte Carlo engine can reveal crucial insights about portfolio behaviour and risk exposure.

As financial markets grow more complex, and as the cost of model failure rises, the ability to simulate uncertainty becomes not just a quantitative exercise, but a core skill in managing capital, designing strategies, and building resilience.

Simulation does not predict the future — but it gives us a way to prepare for it.

7 Python code

```python
# ------- Retrieve the data --------
def get_data(stocks, start, end):
    stockData = yf.download(stocks, start=start, end=end)
    stockData = stockData['Close']
    returns = stockData.pct_change()  # Calculate daily returns
    meanReturns = returns.mean()  # Compute mean daily return for each stock
    covMatrix = returns.cov()  # Compute covariance matrix of returns
    return meanReturns, covMatrix

# ------ Parameters for the simulation ------
stocks = ['AAPL', 'GOOGL', 'AMZN', 'MSFT']
endDate = dt.datetime.now()
startDate = endDate - dt.timedelta(days=365)
mc_sims = 120
T = 252
initialPortfolioValue = 10000
meanReturns, covMatrix = get_data(stocks, startDate, endDate)
```

Figure 6: Sample code to initialize Monte Carlo simulation

```python
# ------ Monte Carlo simulation of portfolio growth ------
for m in range(mc_sims):
      Z = np.random.normal(size=(T, len(weights)))
      dailyReturns = meanM + np.inner(L, Z)
      portfolio_sims[:,m] = np.cumprod(np.inner(weights, dailyReturns.T) + 1) * initialPortfolioValue

# ------ Define VaR and CVaR functions ------
def VaR(returns, alpha=5):
    if isinstance(returns, pd.Series):
        return np.percentile(returns, alpha)
    else:
        raise TypeError("Expected pandas series")

def CVaR(returns, alpha=5):
    if isinstance(returns, pd.Series):
        belowVaR = returns <= VaR(returns, alpha=alpha)
        return returns[belowVaR].mean()
    else:
        raise TypeError("Expected pandas series")

# ------ Convert portfolio values into percentage returns ------
portfolioReturns = (portfolio_sims[-1, :] - initialPortfolioValue) / initialPortfolioValue
VaR_percent = VaR(pd.Series(portfolioReturns), alpha=5)
CVaR_percent = CVaR(pd.Series(portfolioReturns), alpha=5)
VaR_dollar = VaR_percent * initialPortfolioValue
CVaR_dollar = CVaR_percent * initialPortfolioValue
```

Figure 7: Monte Carlo, VaR, and CVaR functions

```python
# ------ SUBPLOT 1: Histogram of Portfolio Returns with VaR & CVaR ------
plt.subplot(1, 2, 1)
plt.hist(portfolioReturns, bins=50, color="blue", alpha=0.6, label="Simulated Returns")
plt.axvline(VaR_percent, color='red', linestyle='dashed', linewidth=2, label=f"VaR 95%: {VaR_percent:.2%}")
plt.axvline(CVaR_percent, color='green', linestyle='dashed', linewidth=2, label=f"CVaR 95%: {CVaR_percent:.2%}")
plt.xlabel("Portfolio Return (%)")
plt.ylabel("Frequency")
plt.title("Portfolio Return Distribution")
plt.legend(prop={'size': 10})

# ------ SUBPLOT 2: Histogram of Portfolio Value with VaR & CVaR ------
plt.subplot(1, 2, 2)
plt.hist(portfolio_sims[-1, :], bins=50, color="purple", alpha=0.6, label="Simulated Portfolio Value")
plt.axvline(VaR_value, color='red', linestyle='dashed', linewidth=2, label=f"VaR 95%: ${VaR_value:,.2f}")
plt.axvline(CVaR_value, color='green', linestyle='dashed', linewidth=2, label=f"CVaR 95%: ${CVaR_value:,.2f}")
plt.xlabel("Portfolio Value ($)")
plt.ylabel("Frequency")
plt.title("Portfolio Value Distribution")
plt.legend(prop={'size': 10})

plt.tight_layout()

# ------ PLOT 3: Monte Carlo Simulations of Portfolio Value with VaR & CVaR ------
plt.figure(figsize=(14, 8))
for i in range(mc_sims):
    plt.plot(portfolio_sims[:, i], alpha=0.3)  # Each simulation path
plt.axhline(VaR_value, color='red', linestyle='dashed', linewidth=2, label=f"VaR 95%: ${VaR_value:,.2f}")
plt.axhline(CVaR_value, color='green', linestyle='dashed', linewidth=2, label=f"CVaR 95%: ${CVaR_value:,.2f}")
plt.axhline(initialPortfolioValue, color='black', linestyle='dashed', linewidth=2, label="Initial Value")
plt.xlabel("Days")
plt.ylabel("Portfolio Value ($)")
plt.title("Monte Carlo Simulation of Portfolio Growth")
plt.legend(prop={'size': 10})
```

Figure 8: Plot example for VaR and CVaR applied to Monte Carlo simulation

8 References

[1] Hull, J. C. (2018). Options, Futures, and Other Derivatives (10th ed.). Pearson.

[2] Cox, J. C., Ross, S. A., & Rubinstein, M. (1979). Option pricing: A simplified approach. Journal of Financial Economics, 7(3), 229-263.

[3] Hull, J. C. (2001). Fundamentals of futures and options markets (4th ed.). Pearson.

[4] Glasserman, P. (2003). Monte Carlo Methods in Financial Engineering. Springer.

[5] Carol, A. (2008). Market Risk Analysis 1. John Wiley & Sons, Ltd.

[6] Carol, A. (2008). Market Risk Analysis 2. John Wiley & Sons, Ltd.