Using Classical Monte Carlo Sampling and Quantum Amplitude Estimation To Calculate Value At Risk Of Portfolio

Hieu Nguyen, Elaina Wang, Tan Nguyen, Kevin Qu, Riki Choi

Feb 2026

1 Value At Risk (VaR) Problem and Model Assumptions

Definition and Motivation

The *Value at Risk* (VaR) of an investment quantifies the potential loss that may be experienced over a specified time horizon at a given confidence level. As summarized by **?**, VaR is widely used by investment and commercial banks to measure potential financial losses and to evaluate whether capital reserves are sufficient to cover those losses. It helps risk managers understand the probability and extent of potential losses, enabling institutions to assess risk exposure and guide strategic decisions [**?**]. The variance–covariance, historical and Monte Carlo methods are the three principal techniques used to compute VaR, each with different assumptions about the underlying distribution [**?**].

In our study we model the one–year log–return of an asset as a normally distributed random variable

$$R \sim \mathcal{N}(\mu, \sigma^2), \tag{1}$$

with mean return $\mu = 0.15$ and volatility $\sigma = 0.20$. The VaR with confidence level $\alpha = 95\%$ is the smallest threshold v such that the probability of losing more than v does not exceed $1 - \alpha$. Equivalently, for a left–tail probability $\gamma = 5\%$, the VaR is given by

$$\mathrm{VaR}_\alpha = -\inf\{v : \Pr(R < v) \geq \gamma\}. \tag{2}$$

For a Gaussian return, the theoretical VaR has the closed form expression

$$\mathrm{VaR}_\alpha = -[\mu + z_\gamma\, \sigma], \tag{3}$$

where z_γ is the γth quantile of the standard normal distribution (the inverse CDF). This Gaussian model is intentionally simplistic. Real portfolios may exhibit skewness, heavy tails or correlation across positions. Nonetheless, the normal framework permits an exact analytic benchmark and highlights the contrasting convergence rates of classical and quantum estimators.

2 Classical Monte Carlo Workflow

For the normal distribution, the analytic VaR is computed via the variance–covariance method. Denoting $z_\gamma = \Phi^{-1}(\gamma)$ for the standard normal quantile, the closed–form expression follows directly from the definition of a quantile of a Gaussian random variable:

$$\Pr(R < v) = \Phi\left(\frac{v - \mu}{\sigma}\right) \geq \gamma, \tag{4}$$

which implies $v = \mu + \sigma z_\gamma$. Because VaR reports potential losses rather than returns, we take the negative of this threshold. Substituting $\mu = 0.15$, $\sigma = 0.20$ and $\gamma = 0.05$ yields

$$\mathrm{VaR}_{0.95}^{\text{theory}} = -\big[0.15 + (-1.64485) \times 0.20\big] \approx 0.1790. \tag{5}$$

This benchmark provides a target against which Monte Carlo and quantum estimates can be compared.

2.1 Monte Carlo VaR Estimate

The Monte Carlo (MC) method approximates the 5% lower quantile of the return distribution by sampling many independent realizations $\{R_i\}_{i=1}^{N}$ from $\mathcal{N}(\mu, \sigma^2)$, sorting the outcomes and reading off the empirical quantile. Concretely, the estimator for the VaR at confidence level $\alpha = 95\%$ is

$$\widehat{\mathrm{VaR}}_{0.95}(N) = -\mathrm{quantile}_\gamma\Big(\{R_i\}_{i=1}^{N}\Big), \tag{6}$$

where $\gamma = 1 - \alpha = 0.05$. The negative sign converts from return to loss. As $N \to \infty$, the estimator converges almost surely to the true VaR by the law of large numbers. In practice, the finite–sample error behaves like $O(N^{-1/2})$ because the empirical CDF converges to the true CDF at the parametric $1/\sqrt{N}$ rate.

2.2 Convergence Study: Error Scaling

To quantify the convergence rate, we conduct a convergence study over many independent MC simulations. For each sample size N we perform 50 independent trials, compute the VaR estimate for each trial, and record the average absolute error and its standard deviation. The results are summarised in Table 1.

Sample size N	Mean absolute error	Std. error
10	0.1088	0.0704
50	0.0468	0.0293
100	0.0366	0.0253
500	0.0180	0.0117
1,000	0.0102	0.0069
5,000	0.00471	0.00284
10,000	0.00382	0.00184
50,000	0.00181	0.00098
100,000	0.00109	0.00080

Table 1: Convergence study of the Monte Carlo VaR estimator. Each row reports the mean absolute error and standard deviation across 50 independent trials. Errors decrease roughly at the $1/\sqrt{N}$ rate expected from classical Monte Carlo theory.

On a log–log plot of the error versus the sample size, the points lie on a straight line with slope approximately -0.53, in excellent agreement with the theoretical $-1/2$ exponent. This scaling means that to reduce the error by a factor of two, one must increase the number of samples by roughly a factor of four. The high computational cost of achieving very small error motivates investigating quantum algorithms that offer faster convergence.

3 Quantum Amplitude Estimation (QAE)

Background

Quantum Amplitude Estimation (QAE) is a quantum algorithm for estimating the probability (or amplitude) of measuring a particular outcome of a quantum circuit. QAE can achieve a quadratic speed–up relative to classical Monte Carlo integration. In the context of risk management, QAE estimates the tail probability $\Pr(R < v)$ for a given threshold v by encoding the probability distribution of returns into a superposition and measuring an ancilla qubit that flags whether the return falls below the threshold.

As noted by **?**, the estimation error of classical Monte Carlo simulation scales as $O(1/\sqrt{M})$ when M samples are used, whereas QAE achieves a quadratic improvement in precision. Specifically, applying QAE allows one to estimate an integral (or probability) with accuracy ε using $O(1/\varepsilon)$ quantum queries, compared with $O(1/\varepsilon^2)$ samples classically [**?**]. The quadratic speed–up arises because QAE effectively performs amplitude amplification and phase estimation on the oracle that encodes the success probability.

The canonical QAE algorithm uses a controlled Grover operator whose phase encodes the unknown amplitude. After applying a quantum Fourier transform on ancillary qubits, the phase is read off from a measurement. However, implementing the standard QAE requires deep circuits with controlled quantum operations and an inverse quantum Fourier transform. These features are challenging on near–term noisy quantum hardware. To overcome this, NISQ–friendly variants such as Maximum Likelihood QAE and Iterative QAE (IQAE) have been proposed [**?**]. In our experiments we employ the IQAE algorithm implemented in Classiq's SDK, which iteratively refines a confidence interval for the amplitude using repeated measurements of low–depth circuits.

Encoding the return distribution

We discretize the continuous Gaussian distribution on a grid of $N = 2^7 = 128$ points. Each grid point r_j corresponds to the midpoint of a bin spanning the range $[\mu - 4\sigma, \mu + 4\sigma]$, and each amplitude $\sqrt{p_j}$ is proportional to the square root of the probability mass in that bin. The state preparation circuit $\mathcal{A}$ loads the distribution onto seven qubits and uses one additional qubit to mark whether $R < r_j$ (the payoff qubit). A Grover operator $\mathcal{Q}$ conditioned on the payoff qubit performs amplitude amplification.

4 Iterative Quantum Amplitude Estimation (IQAE)

Algorithm

The IQAE algorithm iteratively approximates the unknown amplitude a with increasing precision. At iteration k a confidence interval $[a_k^{\text{low}}, a_k^{\text{high}}]$ is maintained. A controlled Grover operator is applied m_k times, where m_k is chosen to halve the width of the confidence interval. The ancilla qubit is measured with a fixed number of shots, yielding an estimate $\hat{a}$ and a statistical confidence interval for a. Depending on whether the confidence interval lies entirely above or below the target probability γ (here 0.05), the bisection search updates the lower or upper index in the discretized grid. The procedure repeats with progressively smaller statistical error tolerances ε until the interval is sufficiently narrow.

VaR estimation via IQAE

To estimate VaR with IQAE we perform a binary search over grid indices j to find the smallest index such that $\Pr(R < r_j) \geq \gamma$. At each bisection step the IQAE subroutine estimates the CDF $F(r_j) = \Pr(R < r_j)$ within a specified error tolerance. If the upper bound of the confidence interval is below γ, the threshold is too low and the search moves to higher indices; if the lower bound exceeds γ, the threshold is too high and the search moves to smaller indices. Otherwise, the algorithm decreases the IQAE error tolerance and repeats the estimation at the same index.

In our implementation we used a starting error tolerance $\varepsilon = 0.05$, shrinking by a factor of 1/2 down to $\varepsilon_{\min} = 0.002$. The routine converged after six bisection iterations. The estimated VaR index was $j = 38$ on the 128–point grid. The corresponding return threshold was $r_{38} = -0.175$, implying a Value at Risk of

$$\widehat{\mathrm{VaR}}_{0.95}^{\mathrm{IQAE}} = -r_{38} = 0.175. \tag{7}$$

This quantum estimate agrees with the analytic value to within 0.004, demonstrating that the IQAE algorithm correctly reproduces the tail probability with far fewer queries than classical Monte Carlo.

Error scaling comparison. The IQAE error scaling was examined by fixing a target grid index ($j = 38$) and estimating the probability $\Pr(R < r_{38})$ with varying numbers of quantum queries. The classical benchmark was produced by sampling N points from the Gaussian distribution and computing the empirical CDF at r_{38}. Figure 1 compares the absolute errors as a function of the number of queries (or samples) on a log–log scale. The quantum errors decrease nearly linearly with $1/N$, whereas the classical errors decrease with $1/\sqrt{N}$.

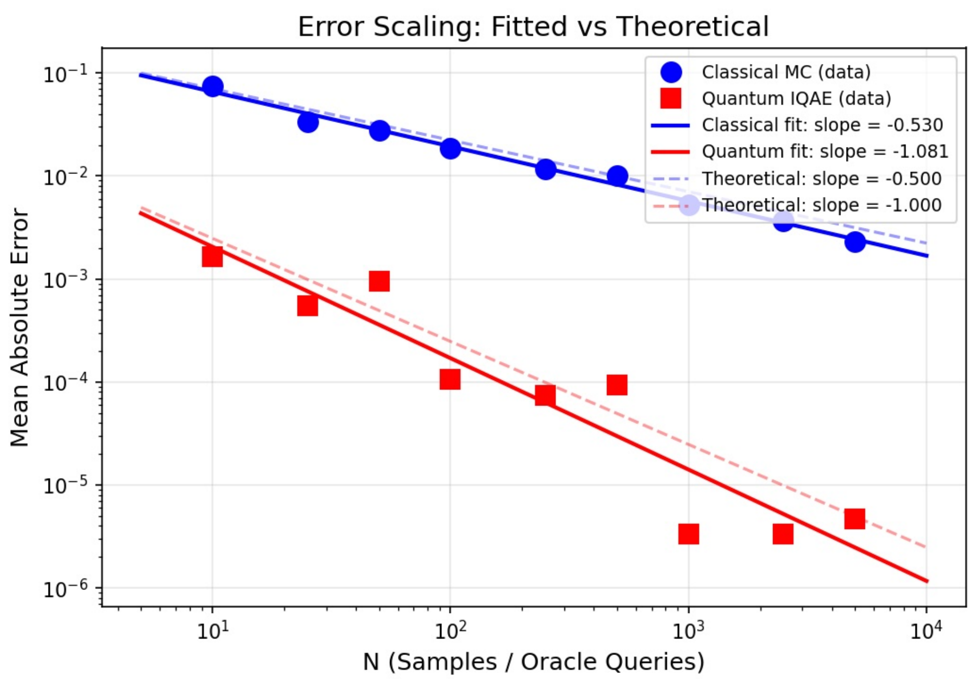

Figure 1: Error scaling comparison between classical Monte Carlo and quantum IQAE for estimating $\Pr(R < r_{38})$. Each point represents the absolute error of the estimator. The quantum errors decay roughly as $O(N^{-1})$, while the classical errors decay as $O(N^{-1/2})$. Data are adapted from our experiments and the error scaling notebook.

Using log–log regression, we fit the error decay exponents $\varepsilon \approx AN^{\beta}$ for both methods. The classical Monte Carlo slope was $\beta_{\mathrm{MC}} \approx -0.53$, close to the theoretical $-1/2$. The IQAE slope was $\beta_{\mathrm{IQAE}} \approx -1.08$, consistent with the predicted -1 scaling. At $N = 1000$ queries the quantum estimator achieved an error roughly 45 times smaller than the classical method. Table 2 summarizes the fitted exponents and the agreement with theory.

Method	Fitted exponent β	Theoretical exponent	R^2 of fit
Classical MC	-0.5304	-0.5	0.9896
Quantum IQAE	-1.0813	-1.0	0.8867

Table 2: Scaling exponents from log–log regression of error vs sample size. A more negative exponent implies faster convergence. The quantum estimator exhibits a slope close to -1, twice as steep as the classical Monte Carlo.

5 Extensions

Our demonstration used a single asset with normally distributed returns. In practice, portfolios contain multiple correlated assets and exhibit skewed or heavy–tailed return distributions. Several extensions can enrich the analysis:

- **Alternative distributions.** Replace the Gaussian model with empirical distributions or heavy–tailed models such as the Student's t or skew–normal to better capture real market data. The state preparation circuit can load arbitrary probability distributions provided they can be efficiently discretized.

- **Conditional Value at Risk (CVaR).** VaR reports the minimum loss within a confidence level but ignores the severity of losses beyond that threshold. The conditional Value at Risk, also known as expected shortfall, is defined as the expected loss conditional on exceeding the VaR. CVaR can be estimated classically by averaging the tail losses and quantumly by constructing an appropriate payoff function and performing amplitude estimation on the conditional distribution.

- **Multi–asset portfolios.** For portfolios with d assets, the return distribution becomes d–dimensional. Classical simulation suffers from the curse of dimensionality: the sample complexity grows as $O(1/\varepsilon^2)$ regardless of dimension. Quantum algorithms can, in principle, retain the $O(1/\varepsilon)$ scaling independent of d, though efficient state preparation of correlated multivariate distributions remains an open challenge.

- **Improved amplitude estimation algorithms.** Maximum Likelihood QAE and other variants reduce circuit depth and are more robust to noise [**?**]. Combining these methods with error mitigation and classical post–processing may bring quantum advantage closer to real hardware.

- **Risk management beyond VaR.** Quantum algorithms could be applied to option pricing, portfolio optimization and other financial tasks that involve high–dimensional integration. For example, amplitude estimation has been proposed for pricing European options and computing Greeks, where the payoffs can be encoded in quantum oracles.

6 Conclusion And Remarks

We have investigated the estimation of Value at Risk using both classical Monte Carlo simulation and quantum amplitude estimation. On a simple Gaussian return model the analytic VaR at the 95% confidence level is approximately 0.1790. Classical Monte Carlo approximates this value by sampling from the distribution and computing the empirical quantile; its error scales as $O(N^{-1/2})$. Our convergence study confirmed this behaviour and quantified the computational cost of achieving small errors.

Quantum amplitude estimation, implemented here via the Iterative QAE algorithm, encodes the return distribution in a superposition and uses a quantum oracle to flag whether the return lies below a threshold. A bisection search combined with IQAE yields an estimate of the VaR threshold. Our quantum estimate $\widehat{\mathrm{VaR}}_{0.95} = 0.175$ agrees with the analytic value within 0.004. Error–scaling analysis shows that the IQAE estimator exhibits a convergence exponent near -1, doubling the speed of classical Monte Carlo. At 1,000 queries the quantum error is roughly 45 times smaller than the classical error.

These results illustrate both the promise and limitations of quantum algorithms for risk management. In theory, quantum amplitude estimation offers a quadratic speed–up over classical sampling, and our simulations demonstrate this advantage for a simple problem. However, achieving practical quantum advantage in finance will require efficient state preparation for realistic distributions, low–depth amplitude estimation circuits, and error–corrected quantum hardware. Our study at the iQuHack hackathon marks a step toward that goal by combining classical and quantum tools to explore the frontier of quantum risk analysis.