Simulating Economic Heavy-Tailed Distributions with the Quantum Boltzmann Machine*

Noorain Noorani[1], Valerio Astuti[2], Giuseppe Bruno[2], Lerby M. Ergun[3], and Vladimir Skavysh[3]

[1]*University of Maryland*
[2]*Bank of Italy*
[3]*Bank of Canada*

Monday 14th April, 2025

Abstract

Accurately estimating tail risk in financial markets is crucial yet difficult, especially when data are scarce. This paper investigates the use of Quantum Boltzmann Machines (QBMs), a class of generative quantum models capable of capturing complex, high-dimensional, and heavy-tailed distributions through quantum superposition and entanglement. Compared to classical Restricted Boltzmann Machines (RBMs), QBMs offer greater expressive power and sampling efficiency. We apply this framework to assess risk for newly listed firms with limited historical data and find that QBM-augmented estimates significantly improve the prediction of long-term risk measures, including Value-at-Risk and expected shortfall, making QBMs a promising tool for financial modeling.

Keywords: Quantum Boltzmann Machines (QBMs), Fat-Tailed Distributions, Economic Modeling, Data Augmentation, Generative Models, Energy-Based Models, Restricted Boltzmann Machines (RBMs), Quantum Computing in Economics, Financial Market Simulation, Heavy-Tailed Distributions

*Corresponding author: skav@bankofcanada.ca

The views expressed in the paper are solely those of the authors and do not necessarily represent the views of the Bank of Italy or the Bank of Canada.

1 Introduction

In economic and financial systems, accurately modeling distributions that exhibit heavy tails, where extreme events occur more frequently than predicted by normal distributions, is crucial for understanding market dynamics and assessing risk. Traditional models often fall short in capturing these heavy-tailed behaviors, leading to underestimation of the likelihood and impact of rare but significant events. A better description and understanding of such distributions has occupied an important role in recent economic, financial, and statistical research (for a review see Gabaix (2009)). However, by their very nature, tail events are infrequent, which limits the number of available observations and complicates reliable statistical inference. This data sparsity makes estimation difficult and often results in high variance in tail-related measures. These challenges are amplified in environments where data is already scarce, making the characterization of the tails even more difficult, for example, for short time series.

Existing approaches to this problem often rely on fitting a parametric distribution to the limited available data, for example through maximum likelihood estimation or by matching the empirical moments of the sample. While these methods are straightforward to implement, they suffer from two important shortcomings. First, they require a priori specification of a functional form, introducing potential model misspecification bias if the true distribution deviates from the assumed family. Second, and more critically for heavy-tailed phenomena, standard sampling-based methods such as bootstrapping systematically underestimate the severity of tail events. This limitation is less problematic in Gaussian settings, where tail behavior is more predictable from the bulk of the distribution. However, in power-law settings, the maximum observed value in a finite sample is heavily influenced by sample size, making it unlikely that resampled data will capture the full extent of future extremes.

The issue of small sample sizes in the estimation of risk measures such as Value-at-Risk (VaR) and Expected Shortfall (ES) has long been recognized in financial econometrics. Foundational studies such as Danielsson and Vries (2005); Duffie and Pan (1997); Embrechts et al. (2005); Hoga (2022); McNeil and Frey (2000) document how tail-based risk measures are prone to high estimation error, especially when sample sizes are limited and the underlying distributions are heavy-tailed. More recent contributions have further explored these limitations in applied settings, showing how small-sample bias affects both unconditional and conditional risk estimates. For instance, Gao et al. (2022) investigate finite-sample distortions in ES estimation using EVT-based methods, while Patton et al. (2019) develop and evaluate non-parametric and semi-parametric estimators of conditional ES in low-data settings. Similarly, Du et al. (2018) and Martins and Ziegel (2021) highlight the challenges in forecasting and back-testing tail risk in volatile or illiquid markets, where data are either short or structurally sparse. Regulatory frameworks such as Basel III and the Fundamental Review of the Trading Book attempt to address these issues through conservative buffers and stressed-scenario measures, which serve as proxies for robustness rather than improving the core estimation process.

To address the modeling challenges posed by heavy-tailed distributions, recent research has turned to generative models that simulate such distributions in a non-parametric way, thereby avoiding rigid structural assumptions. In this paper, we extend this line of work by exploring the use of quantum Boltzmann machines (QBMs), a class of generative quantum machine learning models inspired by statistical physics, for simulating heavy-tailed distributions in low-data environments. QBMs offer a promising approach due to their capacity to model complex,

high-dimensional probability distributions with a compact quantum representation (Tüysüz et al. 2024). We demonstrate how QBM-based approaches can improve the risk assessment of young or early-stage firms, where performance data is limited but accurately capturing downside risk is essential for investment and policy decisions.

Boltzmann Machines (BMs), particularly their restricted variants (RBMs), have been explored as generative models for learning complex probability distributions. RBMs, with their simplified bipartite structure, make learning more tractable compared to general BMs, introducing strong simplification with respect to the original model but still retaining the meaningful predictive capacity (Smolensky et al. 1986; Hinton 2002; Hinton and Salakhutdinov 2006). However, when applied to large-scale problems or high-dimensional data, RBMs face significant scalability challenges. Training these models involves computationally expensive sampling techniques, such as Gibbs sampling or contrastive divergence, which become infeasible as the size of the network grows. Moreover, their reliance on approximations for gradient-based learning often results in suboptimal performance, particularly for capturing complex patterns. These limitations have led to a decline in the use of classical Boltzmann Machines, paving the way for alternative generative approaches, including quantum variants.

The arrival of the new computational paradigm of quantum computing, and with the continued progress achieved in the development of quantum computing hardware, has prompted research in exploring the capabilities of quantum machine learning models and, more specifically, quantum generative models. The exploration of the expressive power of QBMs in comparison to classical RBMs provides an examination of how each model reflects the complexities of quantum and classical data representations. QBMs can be viewed as a natural extension of RBMs into the quantum regime, facilitating the representation of intricate quantum correlations that RBMs do not inherently possess. The fundamental distinction arises from the ability of QBMs to exploit quantum states, represented by Hermitian operators, allowing for the representation of more complex distributions (Anschuetz and Cao 2019; Song et al. 2019).

Research indicates that QBMs can model and represent the ground states of many-body quantum systems more efficiently than RBMs, reflecting the limitations faced by classical models in handling quantum complexities. For instance, studies on the representational capabilities of QBMs in quantum spin-glass systems demonstrate their potential to simulate complex quantum phenomena that are typically intractable for classical systems (Korutcheva et al. 2023). Additionally, QBMs can perform quantum state tomography, a feature that enables the estimation and generation of quantum states, which classical RBMs cannot replicate (Kieferová and Wiebe 2017). This capability is increasingly important as quantum algorithms keep being developed for applications across various domains, involving quantum statistical distributions.

The different training techniques available further highlight the power of these two models. While RBMs depend on methods like Contrastive Divergence (CD) for sampling (Hinton 2010; Adachi and Henderson 2015), which becomes computationally intensive as model complexity increases, QBMs utilize methods based on the Eigenstate Thermalization Hypothesis that enable more efficient training on near-term quantum devices (Anschuetz and Cao 2019). This efficiency arises from the ability of QBMs to sample from quantum thermal distributions, reflecting their increasing advantage in managing larger datasets and complex interactions as quantum technology evolves (Xu and Oates 2021).

The QBM framework adheres to the underlying physics of quantum systems, aligning with

theoretical advancements in quantum computing, which holds promise for applications in generative modeling and fields like computational finance, where quantum models may outperform classical ones in terms of accuracy and representational fidelity (Perot 2023; Coyle et al. 2021). Empirical studies have shown that QBMs significantly outperform classical RBMs in generating synthetic data from complex distributions, underscoring their superior capability in modeling high-dimensional quantum datasets (Coyle et al. 2021; Golubeva and Melko 2021).

The application of quantum generative models in financial applications has largely been researched using gate-based Near Term Intermediate Scale (NISQ) quantum computers and algorithms, such as for example Quantum Circucit Born Machines (Liu 2018) and Quantum Generative Adversarial Networks (Dallaire-Demers and Killoran 2018), employing parameterized quantum circuits to generate data. These generative techniques have been implemented and tested on real hardware (Zhu et al. 2022) garnering attention for the exploration of real applications in finance (Coyle et al. 2021). These applications have been further modified to help enhance financial analysis such as portfolio optimization, risk analysis, time series analysis and anomaly detection. (see for example Orlandi et al. (2024); Ganguly (2023); Zhou et al. (2024); Bhasin et al. (2024); Stein et al. (2024)). Despite the research into these NISQ models, little research has been conducted on using QBM trained on quantum annealing hardware for financial applications.

In this paper, we investigate the efficacy of QBMs in simulating economic scenarios characterized by heavy-tailed distributions. We assess their performance in capturing extreme events and heavy tails, comparing their capabilities with their classical counterpart. Furthermore, we explore the use of these learned distributions for data augmentation, aiming to enhance the robustness of economic models and forecasts. By leveraging the unique properties of quantum computing, this study seeks to contribute to the development of more accurate and reliable tools for economic analysis and decision-making.

To evaluate the practical value of our quantum generative modeling approach, we apply it to a challenging empirical setting: estimating financial risk for newly listed firms with limited historical return data. Using daily return data from 394 U.S. firms obtained through the CRSP database, we focus on measuring risk based on the first year of returns following their initial public offering. This short sample is used to compute standard financial risk metrics—including standard deviation, VaR, ES, skewness, kurtosis, and tail index. These first-year risk measures are then augmented with 400 observations synthesized by a trained QBM, which learns from the observed return distribution. To assess whether the augmented data improve risk estimation, we compare the original one-year estimates and QBM-augmented estimates against the corresponding metrics computed over a full five-year horizon. This extended horizon serves as a more stable benchmark for a firm's risk profile, offering a natural target for evaluating predictive performance. We also benchmark our results against a classical RBM to isolate the contribution of quantum structure in the generative process.

The results show that QBM-generated data significantly enhance the prediction of longer-term risk measures across several dimensions. The added value is clearest for metrics that rely on extreme outcomes, such as VaR, expected shortfall, kurtosis, and the tail index, where small samples typically struggle. The added information by the QBM-augmented one-year VaR, for example, yields a significant coefficient relative to the RBM-augmented one-year VaR. These improvements align with the expectation that tail, sensitive measures benefit most from data augmentation and highlight QBM's ability to capture complex, non-Gaussian structures in the

data. In contrast, standard deviation and skewness, measures that are either less sensitive to the tails or inherently harder to improve, show smaller or no gains. Taken together, these results suggest that QBM offers a promising tool for augmenting financial risk models in settings where traditional approaches are constrained by data availability.

The paper is structured as follows. Section 2 introduces the restricted Boltzmann machine and Section 3 the Quantum counterpart. Section 4 describes the simulation analysis. Subsequently, Section 5 reports the empirical analysis for young firms, followed by the conclusion.

2 Restricted Boltzmann Machine

Developed as a type of generative stochastic artificial neural network, RBMs have garnered attention for their ability to capture intricate patterns and relationships in high-dimensional data. The roots of RBMs trace back to the 1980s (Smolensky et al. 1986), when researchers sought innovative solutions for unsupervised learning, aiming to discover hidden representations and structures within data. Over the years, RBMs have undergone refinement and adaptation, with researchers harnessing their capabilities for diverse applications, ranging from collaborative filtering to feature learning and dimensionality reduction.

This section delves into the theoretical underpinnings of RBMs, exploring their architecture, probabilistic modeling, training procedures, and applications. By unraveling the intricacies of RBMs, we aim to provide a comprehensive understanding of their historical evolution and significance in the broader landscape of machine learning and motivations for quantum machine learning.

2.1 Basic Architecture

RBMs are a type of generative stochastic artificial neural network that consists of two layers: a visible layer and a hidden layer. The visible layer represents the input data, while the hidden layer captures higher-level features or representations learned from the input. These layers are interconnected by weights, and each node in one layer is connected to every node in the other layer. Notably, there are no connections within layers, creating a ”restricted” connectivity pattern, as seen in Figure 1.

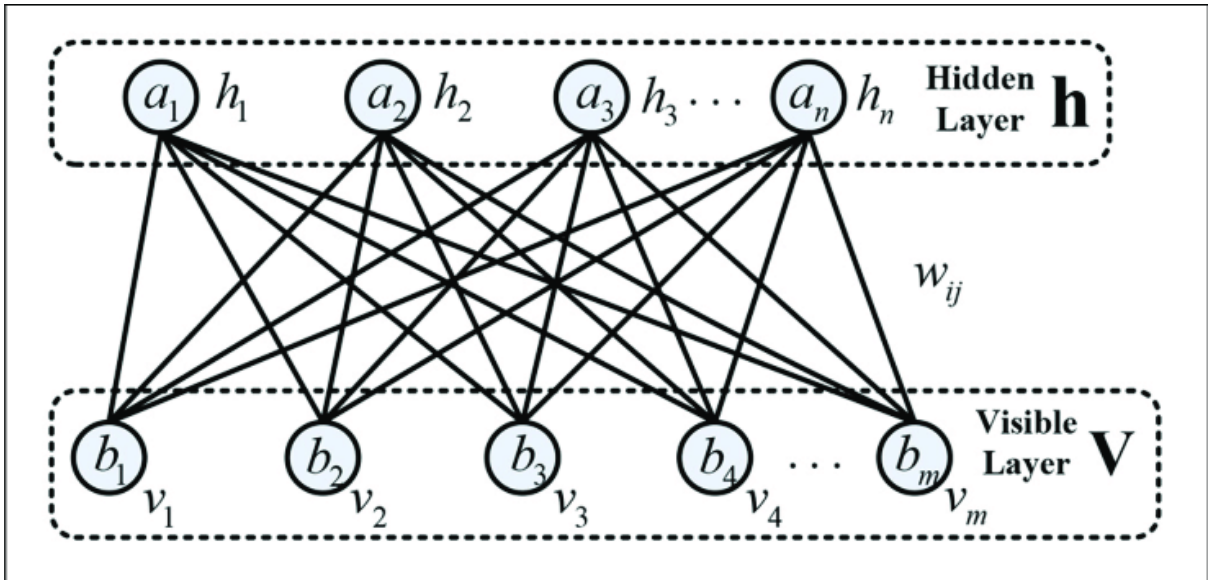

Figure 1: The basic structure of an RBM (Chu et al. 2018), where h_n and v_m represent the hidden and visible units respectively with associated bias vector a_n and b_m. The weights corresponding to the interactions strengths between visible and hidden units are denoted by w_{ij}.

The connectivity pattern in RBMs reflects their bipartite structure, where nodes in the visible layer are only connected to nodes in the hidden layer and vice versa. This restricted connectivity simplifies the learning process and enables more efficient training compared to fully connected networks. The absence of connections within layers reduces the model's complexity, making it computationally more tractable while still capturing intricate patterns in the data.

One key concept in understanding RBMs is the notion of energy. In the context of RBMs, the energy of a particular configuration (assignment of values to visible and hidden units) is a measure of the compatibility of that configuration with the model. The energy is defined in terms of the weights and biases of the RBM. Lower energy configurations are more likely to occur, and the learning process aims to adjust the weights and biases to minimize the energy of observed data while maximizing the energy of unobserved data. This energy-based approach provides a probabilistic interpretation of RBMs, where the model assigns probabilities to different configurations, allowing them to be used for tasks such as sampling and generating new data.

2.2 Probabilistic Model

RBMs are inherently probabilistic models that leverage probability distributions to characterize the relationships between visible and hidden units. The probability of a particular configuration of visible and hidden units is expressed through a joint probability distribution. This distribution encapsulates the likelihood of observing a specific combination of states for both visible and hidden layers.

2.2.1 Energy Function

The energy function in RBMs plays a pivotal role in defining this joint probability distribution. The energy function, denoted as $E(v,h)$, measures the compatibility of a given configuration of visible units (v) and hidden units (h) within the model. Mathematically, the energy function is defined as,

$$E(v,h) = -\sum_{i=1}^{N_v} b_i v_i - \sum_{j=1}^{N_h} a_j h_j - \sum_{i=1}^{N_v}\sum_{j=1}^{N_h} w_{ij} v_i h_j. \tag{1}$$

Here N_v represents the number of visible units, N_h is the number of hidden units and w_{ij} denotes the weight between visible unit v_i and h_j. The values b_i and a_j correspond to the biases of visible v_i and hidden h_j units, respectively.

2.2.2 Boltzmann Distribution

The joint probability distribution over visible and hidden units is derived from the energy function using the Boltzmann Distribution. The probability $P(v,h)$ of a specification configuration is given by

$$P(v,h) = \frac{e^{-E(v,h)}}{Z}.$$

Here Z, the partition function, ensures that the probabilities sum up to 1 over all possible configurations. It is defined as the sum of the exponential of negative energies over all possible visible and hidden unit configurations,

$$Z = \sum_{v}\sum_{h} e^{-E(v,h)}. \tag{2}$$

The Boltzmann distribution reflects the core probabilistic foundation of RBMs, where configurations with lower energy values are assigned higher probabilities, capturing the underlying relationships within the data. This probability framework enables RBMs to model complex distributions and serves as the basis for tasks such as sampling and generation.

2.3 Training Procedure

Training a RBM involves maximizing the log-likelihood of the observed data (or equivalently minimizing the negative log-likelihood). The log-likelihood function ($\mathscr{L}$) for an RBM is given by

$$\mathscr{L}(\Theta) = \sum_{v} \log P(v|\Theta).$$

Here, Θ represents the parameters of the RBM (weights w_{ij}, visible biases a_i, and hidden biases b_j), and the sum is over all possible visible unit configurations v.

The probability of a visible unit configuration is modeled using the RBM's energy function $E(v,h)$ and the Boltzmann distribution:

$$P(v|\Theta) = \frac{e^{-E(v,h)}}{Z},$$

with Z as the partition function, defined as the sum of the exponential of negative energiese over all possible visible and hidden unit configurations, as seen previously in Equation (2).

The gradient of the log-likelihood with respect to the parameters (Θ) is crucial for optimizing the RBM. It provides the direction in which the parameters should be adjusted to increase the log-likelihood. The gradient is obtained by taking the derivative of the log-likelihood function with respect to each parameter. The gradient of the log-likelihood with respect to the model parameters can be expressed as an expectation over the data distribution and the model distribution:

$$\frac{\partial \mathscr{L}}{\partial \Theta} = \left\langle \frac{\partial \log P(v|\Theta)}{\partial \Theta} \right\rangle_{data} - \left\langle \frac{\partial \log P(v|\Theta)}{\partial \Theta} \right\rangle_{model}.$$

The challenge arises in computing these expectations exactly because they involve the partition function Z, which requires summing over all possible configurations of visible and hidden units. This summation is intractable due to the exponential number of configurations, making the exact computation computationally infeasible.

2.3.1 Contrastive Divergence

This is where Contrastive Divergence (CD) comes into play. Instead of computing the exact expectations, CD introduces an efficient approximation. The primary idea is to approximate the gradient of the log-likelihood function by comparing the model's response to real data (positive phase) and a "model" sample generated by Gibbs sampling (negative phase). It initializes the visible layer with a training sample, performs Gibbs sampling for a few steps to create the "model" sample, and then computes the gradient using the positive phase (data) and negative phase (model) expectations. CD exploits the idea that after a few Gibbs sampling steps, the model sample becomes a reasonable approximation of the negative phase of the true data distribution. The algorithm's efficiency comes from the fact that it avoids the need for exact sampling from the RBM's distribution, making it computationally more feasible for training large models. The CD algorithm can be broken down into the following steps:

Initialization
Initialize the visible layer with a training sample (v) and compute the probabilities of the hidden layer being activated:

$$P(h_j = 1|v^{(0)}) = \sigma\left(b_j + \sum_{i=1}^{N_v} v_i w_{ij}\right),$$

where $\sigma(x)$ is the logistic sigmoid function $\sigma(x) = \frac{1}{(1+e^{-x})}$. We then sample binary values for the hidden layer using these probabilities to get your initial hidden values, $h^{(0)}$.

Gibbs Sampling
Perform a Markov Chain Monte Carlo (MCMC) process, specifically Gibbs sampling, to generate a "model" sample (negative phase). This involves alternating between updating the visible and hidden units based on the current state, allowing the system to explore the configuration space.

We perform Gibbs sampling for k steps to create a "model" sample. In each step, alternate between updating the visible and hidden layers using conditional probabilities. This is employed

to approximate the gradient of the log-likelihood.

Negative Phase
Given the sampled hidden states from the initialization step, compute the probabilities of the visible units being activated:

$$P(v_i = 1|h^{(k)}) = \sigma\left(a_i + \sum_{i=1}^{N_h} h_j w_{ij}\right)$$

and sample binary values for the visible layer to get $v^{(k)}$.

Positive Phase
Now given the re-sampled visible states, compute the probabilities of the hidden units being activated again

$$P(h_j = 1|v^{(k)}) = \sigma\left(b_j + \sum_{i=1}^{N_v} v_i w_{ij}\right)$$

and sample binary values for the hidden layer to get $h^{(k+1)}$. Repeat the process, alternating between updating visible and hidden units, for several steps to create a model sample.

Compute Gradients
Using the initial phase ($v^{(0)}$ and $h^{(0)}$) and the final model phase ($v^{(k)}$ and $h^{(k+1)}$), compute the gradients of the log-likelihood with respect to the model parameters (weights and biases),

$$\Delta w_{ij} = \eta(\langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{model}).$$

Here, $\langle . \rangle_{data}$ and $\langle . \rangle_{model}$ denote expectations over the training data and model samples, respectively. Furthermore, η is the learning rate. By iteratively applying Gibbs sampling and computing these differences, CD efficiently approximates the gradient of the log-likelihood, facilitating the training of RBMs on large datasets. Subsequently, adjust the model parameters (weights and biases) in the direction that increases the likelihood of the training data and decreases the likelihood of the model sample.

2.4 Advantages and Challenges

RBMs offer a multifaceted set of advantages, making them adept at modeling intricate relationships and capturing high-dimensional dependencies. One key strength lies in their capacity for representation learning, as RBMs autonomously unveil hierarchical features when trained on unlabeled data. This is particularly valuable in scenarios where feature extraction poses challenges. Moreover, the incorporation of non-linearity through activation functions empowers RBMs to adeptly model complex relationships, a trait crucial for tasks where linear models fall short.

RBMs also excel in dimensionality reduction, effectively distilling essential information from high-dimensional datasets and mitigating the challenges associated with the curse of dimensionality. Their generative modeling capability enables the creation of new samples akin to the training data, proving advantageous in tasks such as image synthesis and recommendation

systems. Notably, RBMs robustly handle missing data during both training and inference, a practical feature for real-world datasets where incomplete information is common.

Despite their merits, RBMs present certain challenges that warrant consideration. Notably, the training of RBMs can be computationally demanding, particularly with large datasets and deep architectures. The iterative nature of the contrastive divergence algorithm, commonly used for training RBMs, can lead to slow convergence, making the training process resource-intensive. Efficiently addressing these computational demands remains a focal point for enhancing the practicality of RBMs in real-world applications.

Another challenge lies in the sensitivity of RBMs to hyperparameters. Selecting appropriate hyperparameter values is a non-trivial task, and suboptimal choices may result in poor model performance or hinder convergence during training. Balancing the learning rate, the number of hidden units, and other hyperparameters is crucial, and a lack of clear guidelines adds complexity to the model development process. This sensitivity underscores the importance of meticulous hyperparameter tuning to unlock the full potential of RBMs.

While RBMs offer significant advantages, acknowledging and addressing challenges such as computational demands and hyperparameter sensitivity is imperative for maximizing their utility in various applications. Efforts directed towards optimizing training efficiency and refining hyperparameter selection will contribute to the continued improvement and practical applicability of RBMs in complex modeling scenarios.

3 Quantum Boltzmann Machines

The motivation for integrating quantum mechanics into machine learning stems from the recognition that certain computational problems, especially those involving complex optimization and probabilistic modeling, can be addressed more efficiently using quantum principles. Quantum mechanics enables the representation of information in superposition and the exploitation of quantum entanglement. In machine learning, tasks such as optimization, matrix operations, and sampling distributions often involve computationally expensive processes. Quantum computing has the potential to provide speedup in these areas.

3.1 QBM Theory

In the classical RBM, our visible layer comprised of classical bits (1s and 0s). If we were to change this to the fundamental unit of quantum information (qubits or quantum bits), we could now govern our mathematics using the tools found in quantum mechanics. Quantum mechanics employs matrices (operators) whose dimensionality corresponds to the total number of potential states (2^N), unlike conventional machine learning methods that utilize vectors with a dimensionality equivalent to the number of variables (N). Now instead of the energy function seen previously in Equation (1), a new formalism creating a $2^N \times 2^N$ diagonal matrix, called the Hamiltonian,

$$H = -\sum_{v} \Gamma_v \sigma_v^x - \sum_{h} b_h \sigma_h^z - \sum_{v,h} w_{vh} \sigma_v^z \sigma_h^z. \tag{3}$$

Here, Γ_i, b_v, w_{vh} represents weights similar to the classical RBM. The binary values of the visible and hidden (σ_i^x and σ_v^z) units are represented as a $2^N \times 2^N$ matrix which corresponds to

$$\sigma_a^z = \overbrace{I \otimes \ldots \otimes I}^{a-1} \otimes \sigma_z \otimes \overbrace{I \otimes \ldots \otimes I}^{N-a}.$$

Here I is the identity matrix, and σ_z is the Pauli-Z matrix.[1]

We represent the eigenstates of our Hamiltonian in Equation (3) as $|v,h\rangle$, where again v and h denote visible and hidden variables, respectively. Like in the RBM we want to define our probability distribution. Given we are using a Hamiltonian we start with by defining a density matrix as

$$\rho = \frac{1}{Z} e^{-H}. \tag{4}$$

We can now define matrix exponentiation through Taylor expansion $e^{-H} = \sum_{k=0}^{\infty} \frac{1}{k!}(-H)^k$. For a diagonal Hamiltonian, e^{-H} is a diagonal matrix with its 2^N diagonal elements corresponding to the energy states e^{-E_z}. The partition function in (4) is given by $Z = Tr[e^{-H}]$.

The diagonal elements of ρ are therefore Boltzmann probabilities of all the 2^N states. Given the visible variables (state $|v\rangle$, we can obtain the marginal Boltzmann probability P_v by tracing over the hidden variables

$$P_v = Tr[\Lambda_v \rho],$$

where Λ_v is a projection operator, linking hidden and visible units. The Λ_v term is a diagonal matrix with diagonal elements being either 0 or 1 dependent on whether the visible states used. Mathematically this is written like:

$$\Lambda_v = |v\rangle\langle v| \otimes I^{\otimes n_h}.$$

Using the probabilities above we can obtain the log-likelihood, which for data set distribution p_{data} and parameters $\theta = (W, a, b)$ is defined as

$$\ell(\theta) = \sum_v p_{data}(v) \log \mathbf{Tr}[\Lambda_v \rho], \tag{5}$$

where $\sum_v$ denotes the sum over all possible configurations of v.

3.2 Training a QBM

To train the Quantum Boltzmann Machine one would have to minimize the log likelihood function, which is achieved by determining the gradient of the log likelihood function given by:

$$\partial_\theta \ell = \sum_v p_{data}(v) \left(\frac{Tr[\Lambda_v \partial_\theta e^{-H}]}{Tr[\Lambda_v e^{-H}]} - \frac{Tr[\partial_\theta e^{-H}]}{Tr[e^{-H}]} \right). \tag{6}$$

Like in the RBM, the goal is to estimate the gradient efficiently via sampling. For the QBM this however is non trivial because, now H and $\partial_\theta H$ are now matrices and they don't commute

[1] The use of the symbol σ_z is established in quantum mechanics, and it is not to be confused with the activation function introduced in the previous section.

meaning $\partial_\theta e^{-H} \neq e^{-H}\partial_\theta H$ like it is in the classical case. As shown in Amin et al. (2018) they show that the second term of Equation (6) can be simplified to:

$$\frac{Tr[\partial_\theta e^{-H}]}{Tr[e^{-H}]} = -\langle \partial_\theta H \rangle,$$

which can be computed by sampling the distribution in Equation (4). However, the first term in Equation (6),

$$\frac{Tr[\Lambda_v \partial_\theta e^{-H}]}{Tr[\Lambda_v e^{-H}]} = \int_0^1 \partial t \frac{Tr[\Lambda_v e^{-tH} \partial_\theta H e^{-(1-t)H}]}{Tr[\Lambda_v e^{-H}]},$$

which cannot be computed via sampling. This makes training a QBM as is inefficient. However, we can use techniques common in classical in machine learning, where we introduce a defined upper-bound to the log-likelihood function what Amin et al. (2018) called the bound-based QBM.

3.2.1 Bound-Based QBM

We can define a lower bound for the probabilities using the Golden-Thompson Inequality,

$$Tr[e^A e^B] \geqslant Tr[e^{A+B}],$$

which holds for any Hermitian matrices A and B, as is the case in this application. We can therefore write our density matrix now as:

$$\rho_v = \frac{Tr[e^{-H} e^{ln\Lambda_v}]}{tr[e^{-H}]} \geqslant \frac{Tr[e^{-H+ln\Lambda_v}]}{Tr[e^{-H}]}.$$

This introduces a new Hamiltonian

$$H_v = H - ln\Lambda_v.$$

This new Hamiltonian is referred to as the clamped Hamiltonian because every visible qubit σ_v^z is clamped to its corresponding classical data value v. With this new Hamiltonian we can rewrite our clamped density matrix as:

$$\rho_v = \frac{1}{Z_v} e^{-H_v}, \tag{7}$$

where $Z_v = Tr(e^{-H_v})$.

Combining Equation (7) and Equation (5) we get our new bounded log-likelihood function

$$\tilde{\ell}(\theta) = \sum_v p_{data}(v) log Tr(\rho_v).$$

Now when calculating the gradient of the clamped log-likelihood function, we get:

$$\partial_\theta \tilde{\ell}(\theta) = \sum p_{data}(v) \left(\frac{Tr[e^{-H_v} \partial_\theta H_v]}{Tr[e^{-H_v}]} - \frac{Tr[e^{-H} \partial_\theta H]}{tr[e^{-H}]} \right)$$

$$\partial_\theta \tilde{\ell}(\theta) = \overline{\langle \partial_\theta H_v \rangle_v} - \langle \partial_\theta H \rangle.$$

Computing the gradients with respect to the parameters of the lower bound QBM are given by:

$$\partial_{w_{ih}} \tilde{\ell}(\sigma) = \langle \sigma_i^z \sigma_j^z \rangle_{data} - \langle \sigma_i^z \sigma_j^z \rangle_{model}$$

and

$$\partial_{b_i}(\sigma) = \langle \sigma_i^z \rangle_{data} - \langle \sigma_i^z \rangle_{model}.$$

Here, $\langle . \rangle_{data}$ and $\langle . \rangle_{model}$ denote expectations over the training data and model samples, respectively.

If connections are restricted within the hidden layer, then the hidden unit probabilities are independent in the positive phase and can be computed easily. The hidden qubits become uncoupled in the positive phase and the calculations can be carried out exactly. Taking into account the restrictions and the boltzmann machine structure we get the clamped Hamiltonian to be reduced to:

$$H_v = -\sum_i^n \left(\Gamma_i \sigma_i^x + b_i'(v) \sigma_i^z \right). \tag{8}$$

As shown in Perot (2023) the final positive phase expectation values are:

$$\langle \sigma_i^z \rangle_{data} = \sum_v p_{data}(v) v_i, i \in I_v,$$

$$\langle \sigma_i^z \rangle_{data} = \sum_v p_{data}(v) \frac{b_i'}{D_i(v)} \tanh(D_i(v)), i \in I_h,$$

$$\langle \sigma_i^z \sigma_j^z \rangle_{data} = \sum_v p_{data}(v) v_i v_j, i, j \in I_v,$$

$$\langle \sigma_i^z \sigma_j^z \rangle_{data} = \sum_v p_{data}(v) v_i \frac{b_i'}{D_i(v)} \tanh(D_i(v)), i \in I_v, j \in I_h,$$

where $b_i'(v) = b_i + (W^T v)_i, D_i(v) = \sqrt{\Gamma_i^2 + b_i'(v)^2}. I_v = \{1, ..., n\}$ represents the indices of the visible qubits and $I_h = \{1, ..., n\}$ represents the indices of the hidden qubits.

3.3 Training on a Quantum Annealer

A QBM is a probabilistic graphical model that extends the classical Boltzmann Machine by incorporating quantum mechanics, particularly in its hidden units, which can exist in superpositions of states. This allows QBMs to model complex probability distributions, leveraging quantum effects such as superposition and entanglement, which are not accessible to classical models. Quantum annealers are specialized quantum computers designed to solve optimization problems by finding the minimum energy state of a system, typically mapped to a transverse-field Ising model. They operate through quantum annealing, a process that leverages quantum phenomena, like superposition, entanglement and tunneling to explore the energy landscape of a Hamiltonian, like that found in (3) and (8).

The topology of the D-Wave devices lends itself nicely for embedding the Restricted Quantum Boltzmann Machine efficiently. Due to the connectivity of the device, we can embed small models without the need for excess connections and chains to fit the model on the physical

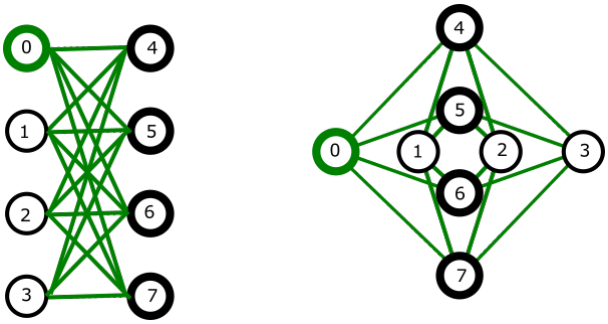

Figure 2: QBM Architecture to D-Wave Chimera Topology, Restricted QBM to Chimera topology embedding

device. As seen in Figure 2, we can see a 4 visible and 4 hidden unit network fits into the D-wave Chimera topology with full connectivity.

The Chimera architecture is an older topology from 2018. Since then D-wave have updated their topologies to the current Zephyr topology which would allow for an 8 visible and 8 hidden unit Boltzmann machine, as seen in Figure 3.

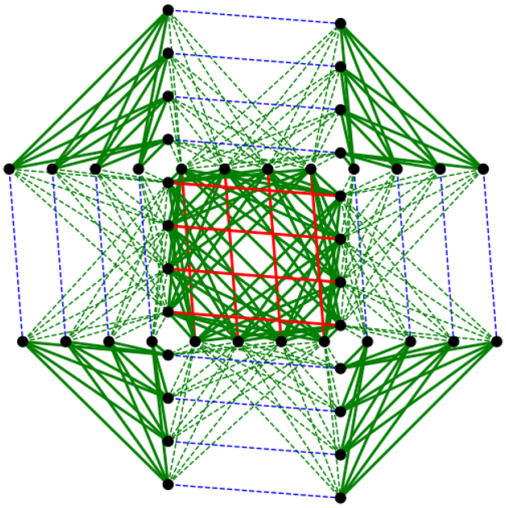

Figure 3: QBM Architecture to D-Wave Zephyr Topology

As mentioned earlier, training a RBM is computationally intensive due to the need to calculate the model-dependent term in gradient-based learning. Typically, RBMs are trained using CD, which, as outlined in the theoretical background, relies on MCMC methods. However, MCMC methods generally require lengthy equilibratio n periods. Additionally, CD does not accurately follow the true gradient of the log-likelihood function and may produce unreliable outcomes. As a result, improved sampling methods could significantly benefit the efficiency and accuracy of RBM training.

QBMs leverage quantum effects like superposition and entanglement to represent and process information in ways that classical models cannot. Training on a quantum annealer allows direct exploitation of these effects, as the annealer can explore the quantum state space more thor-

oughly. This is evidenced by studies on quantum RBMs, which suggest that quantum sampling can lead to better modeling of complex distributions. The training process for QBMs often involves gradient-based optimization, where the gradient is computed as the difference between data and model expectations. Computing model expectations requires sampling, and quantum annealers can accelerate this by providing faster and potentially more accurate samples.

4 Classical Benchmark

As explained in the introduction, suitable applications for demonstrating the value of QBM modeling arise in settings where data are limited, such as the rare events in the tails of a distribution. A key motivation for our focus is the widespread, yet often inappropriate, assumption of normality. If the true data-generating process exhibits heavy tails, such as power law behavior, then relying on normality can lead to a severe underestimation of the likelihood and impact of extreme events. Tail events, however, are rare, hence it is often difficult to have enough data to discard the assumption of normality and to correctly describe the sporadic, large outliers of the distribution. Such situations frequently arise in macroeconomic contexts, where variables like GDP are reported infrequently, and in the case of rare but impactful tail events, both of which lead to high estimation uncertainty. These considerations motivate our empirical focus on heavy-tailed distributions, where QBM-generated synthetic observations can help improve the robustness of inference.

A distribution $\rho_X(x)$ is said to have a (right) heavy tail if it decreases more slowly than any exponential. Defining the complementary cumulative distribution function as:

$$\overline{F}_X(x) = \int_x^\infty \rho(x)dx.$$

We say that the distribution is heavy-tailed if:

$$lim_{x\to\infty}\overline{F}_X(x)\, e^{tx} = \infty \quad \forall t > 0.$$

For a large class of heavy-tailed distributions only a finite number of moments are finite. This is true in particular for all distributions having a right tail decaying as a power law:

$$\rho_X(x) \sim x^{-(\alpha+1)} \quad , \quad \alpha > 0.$$

In this case, the largest integer smaller than or equal to α indicates the maximum finite moment of the distribution. In this section, we directly compare the estimation performance of the QBM and RBM models when applied to a common heavy-tailed distribution. We focus on the Student-t distribution with 5 degrees of freedom, for which only the first four moments exist. For finite sample sizes, this implies that the estimation of higher-order moments becomes unreliable. Moreover, although the fourth moment is theoretically finite, its estimation can exhibit substantial uncertainty in most realistic sample sizes, as we will demonstrate.

The comparison is performed on small Student-t samples of 100 observations. For each sample of 100 observations, we first compute the first four moments of the distribution, denoted in Table 1 as x, x^2, x^3, and x^4. The same sample is then used to train both a RBM and a QBM. Once trained, each model generates a synthetic dataset consisting of 10,000 observations.

These synthetic samples are then used to re-estimate the first four moments of the distribution. The estimates obtained from the QBM- and RBM-generated data are compared against the original sample estimates as well as the true theoretical moments of the Student-t distribution with 5 degrees of freedom. This repeated sampling procedure, performed over 300 independent replications, allows us to evaluate the consistency, bias, and variability of each method in recovering key distributional features from limited data.[2] In particular, the setup highlights how the accuracy of moment estimation deteriorates as the order of the moment increases, and how generative models can help mitigate this issue under small-sample conditions.

The choice of an RBM as classical benchmark is straightforward, being the QBM the immediate quantum extension of the RBM model. As we already mentioned, both the RBM and QBM models need a regularization of the distribution domain: we have to select a finite range of variation of the analyzed data, and discretize the domain. While the latter operation has negligible effects on the resulting distribution, the clipping of the range of variation can have sizable results for a heavy-tailed distribution as a Student-t. To minimize the effects of this choice, we select the range of variation $[-10, 10]$, much larger than the typical maximum for our selected sample size. With this selected range, the probability for a sample point to be fall outside of this interval is less than 2×10^{-4}. Figure 4 shows an example of an augmented sample. With

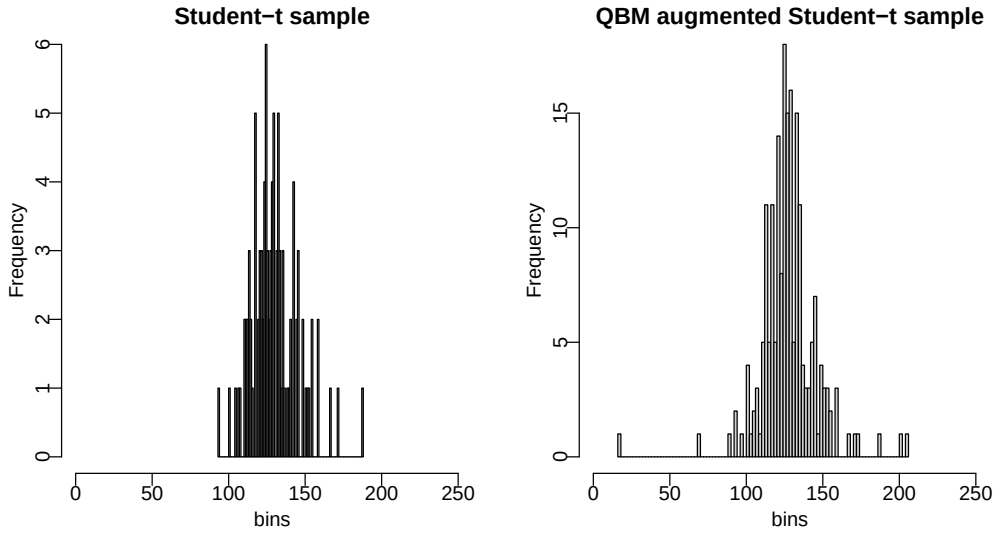

Figure 4: Augmenting Student-t (5) data

This figure shows an example of augmenting data sampled from a Student-t distribution with 5 degrees of freedom. The upper panel shows the histogram of a draw of $n = 100$ observations. We have discretized the data by creating equally sized bins of size $(10+10)/(2^8)$, as the support is limited between -10 and 10. After training the QBM on this sample, we generate a 100 additional QBM simulated observations and add them to the original data, presented in the lower panel.

these cutoffs the Student-t distribution with 5 degrees of freedom has true values of zero for the first and third moments, the true value of the second moment is equal[3] to 1.64 and the fourth

[2] In order to make a meaningful comparison, we first optimize the hyper-parameters set for the RBM. In particular we select the values of the number of hidden neurons, the learning rate, the number of epochs, the batch size and the Gibbs sampling parameter k. We optimize over a subset of samples, using as objective functions the L_1 and L_2 distances between the sample distribution and the synthetic one (both distances result in a similar estimation for the parameters).

[3] These values are obviously lower than for the standard Student-t (5) distribution, given the artificial cutoff

moment equal to 15.96. For samples composed of 10.000 data points, the standard deviation of the second moment is 0.04, while the standard deviation of the fourth moment is 1.52.

The results for sample data, RBM model and QBM model are shown in Table 1. As we anticipated at the beginning of the section, while the first and second moments of the distribution are reasonably stable for the original sample size of 100 points, the oscillations in the fourth moment are of the same order of magnitude of the true value itself. This uncertainty gets amplified when training the generative models. We find, however, a much better estimation using the QBM with respect to the RBM.

Table 1: Simulation results

	Sample		RBM			QBM		
Estimator	Average	St. dev.	Average	St. dev.	Corr	Average	St. dev.	Corr
x	0.01	0.14	-0.21	0.45	0.21	-0.08	0.11	-0.58
x^2	1.62	0.33	13.83	2.51	0.16	3.99	0.44	0.23
x^3	0.19	2.23	-12.54	26.66	0.65	-0.02	1.64	-0.68
x^4	15.63	13.42	745.27	169.23	0.15	188.28	23.07	-0.02

This table display the first four moments of the empirical distribution generated from the DGP, the RBM model and QBM model. We report the mean and standard deviation for the moments over the 300 samples drawn from the Student-t (5) distribution. Correlations between the terms $\varepsilon^{\text{sample}}_{\text{true}}$ and $\varepsilon^{\text{synt}}_{\text{sample}}$ for RBM and QBM data.

As we can see from the table, the estimation performances of the RBM model start degrading already from the second moment. For the first moment we obtain an average value which is of the same order of magnitude of the sample standard deviation, which implies we would not have a significantly worse estimation using the RBM generated data. Howevery, for both the second, third and fourth moment we obtain averages and standard deviations which are orders of magnitudes larger than the sample (and QBM) ones. This implies that, in trying to augment the sample data with synthetic RBM generated ones, we would introduce significant distortions in the data, and alter the statistical properties of the original distribution.

For the QBM-generated data, the results are more encouraging. The first moment exhibits only a small bias, with a standard deviation even lower than that of the sample-based estimate. The second moment shows a slight positive bias; however, both its average value and standard deviation are of the same order of magnitude as those from the sample. The estimation of the third moment also improves relative to the sample data, both in terms of average value and dispersion. This clear enhancement can be attributed to two key factors. First, the synthetic data are generated from a symmetric distribution over a symmetric interval. As a result, with a sufficiently large number of synthetic observations, the average of any odd moment tends to zero. This property would not hold if the QBM model were overfitting to the extreme values in the sample, in contrast to what we observe with the RBM, generated data.

In this regard, the QBM model performs particularly well: it efficiently downweights the influence of outlier observations and more reliably captures the core structure of the distribution.[4]

on the tail values.

[4] As a robustness check, we repeated the same experiment after introducing a random shift in the center of the distribution to remove the advantage conferred by symmetry. The results, presented in Appendix A, confirm the findings reported here.

For the fourth moment, we begin to observe a larger bias, although the standard deviation of the QBM-based estimate remains roughly comparable to that of the sample-based estimate. Overall, this comparison suggests that when synthetic data are used to augment a small sample, the QBM model offers a significantly more accurate replication of the underlying distribution's statistical properties. In contrast, the RBM tends to introduce distortions that may be unacceptable, even when compensated by a much larger number of generated data points.

In addition to the absolute performance of the estimations, it is instructive to consider the correlation between the errors of the sample estimators and the synthetic ones. More precisely, as shown in Appendix B, denoting as $\varepsilon_{\text{true}}^{\text{sample}}$ the errors between the sample estimators and the true values and $\varepsilon_{\text{sample}}^{\text{synt}}$ the errors between the synthetic estimators and the sample ones, the sign of the correlation between $\varepsilon_{\text{true}}^{\text{sample}}$ and $\varepsilon_{\text{sample}}^{\text{synt}}$ directly impacts the quality of the estimations of the true values based on synthetic data.

As already mentioned, one of the reasons the QBM estimators are much closer to the real value than the RBM ones, is that the RBM shows the tendency to learn the outliers in the sample. This translates in a positive correlation between $\varepsilon_{\text{true}}^{\text{sample}}$ and $\varepsilon_{\text{sample}}^{\text{synt}}$. The QBM estimator, on the other hand, manages to neglect the outliers more often, generating a *negative* correlation between the two error terms.

As shown in the last column of Table 1, for both the first and third moment we obtain a large negative correlation between the error terms between sample and true data, and synthetic and sample data. To understand why this happens, it is sufficient to realize that whenever a sample presents, for example, a positive large outlier, the error term $\varepsilon_{\text{true}}^{\text{sample}}$ will be positive. If the generative model does not learn to replicate this outlier, however, the error $\varepsilon_{\text{sample}}^{\text{synt}}$ between the synthetic estimation and the sample one will typically be negative. As we can see from the fifth column of Table 1 the RBM estimation shows positive correlations for the error terms relative to all the distribution moments, pointing to the fact that whenever an outlier is generated in the sample, it is likely replicated also in the synthetic data.

5 Real Data: Risk Assessment of Young Firms

The risk assessment of financial investments forms an important part of investors' capital allocation decision and the financial stability regulatory framework. Value-at-Risk (VaR) has become a cornerstone of both financial risk management and regulatory frameworks. It provides a probabilistic estimate of the potential loss in value of a portfolio over a defined time horizon at a given confidence level, offering a standardized approach to assessing market risk (Jorion 2007). Regulatory bodies such as the Basel Committee on Banking Supervision have incorporated VaR, and later Expected Shortfall (ES) (Artzner et al. 1999), into capital adequacy requirements, requiring banks to hold capital against potential losses implied by their internal risk models (Basel Committee on Banking Supervision 2011).

Estimating these types of measures accurately becomes particularly challenging in the presence of small samples, which are common when working with low-frequency data, short-lived securities, or emerging asset classes. Small samples exacerbate the inherent trade-off between bias and variance in risk estimation and limit the ability to reliably infer tail behavior, which

VaR and ES is especially sensitive to (Danielsson and Vries 1997). Parametric approaches, such as those assuming normality or GARCH-based dynamics, may suffer from model misspecification, while non-parametric methods like historical simulation become unreliable due to insufficient observations in the tail (Pritsker 2006).

To address this, we leverage the QBM to generate synthetic data that enriches the existing limited observations. Specifically, we train the QBM using the first year of returns data from the stock exchange and use it to generate additional synthetic returns. By augmenting the original dataset with these QBM-synthesized observations, we aim to improve the accuracy of various risk measures.

5.1 Stock Return Data

The stock market data used in this study is obtained from the Centre for Research in Security Prices (CRSP). The CRSP database provides individual stock data spanning December 31, 1925, to December 31, 2015, and includes information from the NYSE, AMEX, NASDAQ, and NYSE Arca exchanges. The prices that are adjusted for stock splits and dividend payouts are then used to calculated the returns. We use these returns to calculate various risk metrics from the risk management literature. We exclude stocks with fewer than 60 months of data. Stocks with an average price below \$5 are also excluded as is customary in the finance literature. Futhermore, due to computational limitations we have restricted our sample the 400 stocks with the lowest unique security identifier (PERMNO).

5.2 Risk Assessment

The risk measures we employ characterize the return distribution in terms of both overall uncertainty and exposure to undesirable outcomes. In financial risk management, variance and kurtosis are commonly used to quantify uncertainty, while downside risk is typically assessed using skewness, VaR, ES, and the tail index for the left tail of the return distribution. For formal definitions, see Appendix C; for further discussion, refer to Danielsson (2011). These measures form the basis of our empirical framework, which evaluates whether QBM-generated data improve the ability to explain a firm's longer sample (five year) risk profile. We focus on the five-year horizon because it incorporates more data than short-term estimates, yielding more stable and reliable risk estimate. At the same time, it remains short enough to avoid capturing structural shifts in the firm's fundamentals that could occur over longer horizons. To assess the predictive contribution of QBM, we compare its performance against a RBM. This comparison is particularly relevant from a quantum computing perspective, as it illustrates the potential for quantum generative models to improve risk estimation over classical models, especially in data-constrained environments where traditional methods may underperform.

Figure 5 provides a graphical representation of our analysis for the VaR measured at 5%. The upper panel shows a scatter plot of the VaR of the firms extracted from the first year, VaR_i^{1y}, versus the VaR from the five year sample for the same firm, VaR_i^{1-5y}. The lower panel shows the scatter plot of the same 5 year sample on the y-axis, against the difference between the measure extracted from the 1 year sample augmented with the QBM data and the VaR with the

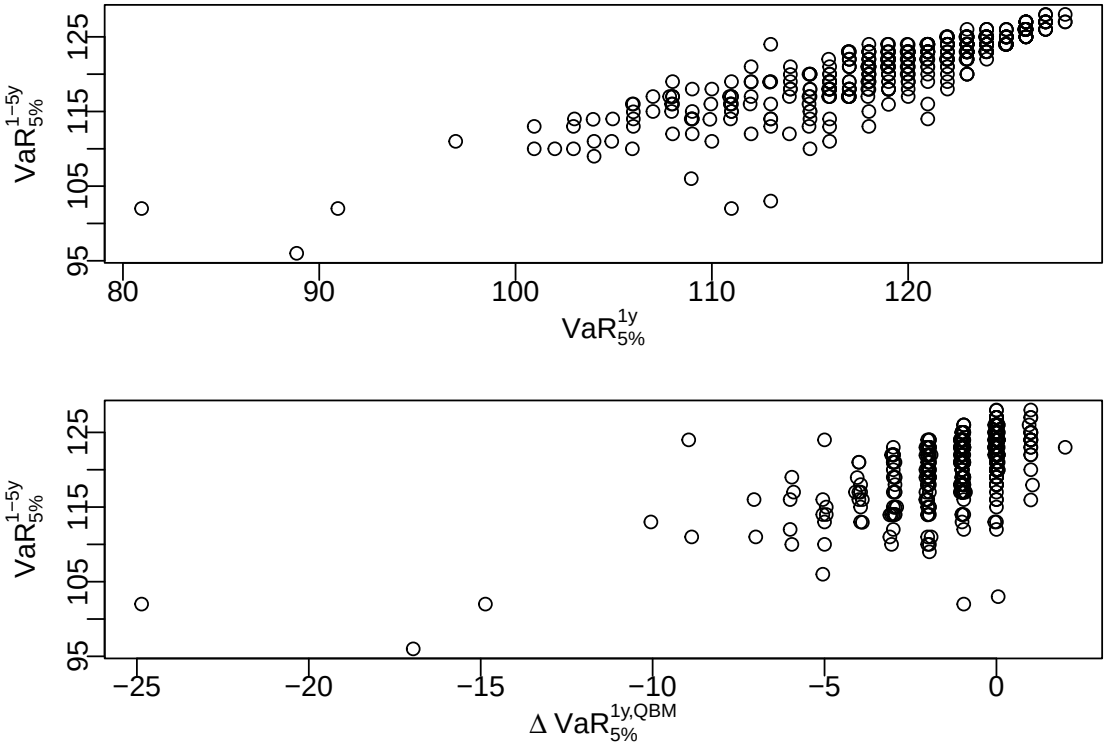

Figure 5: Value-at-Risk prediction - Sample and added value augmentation

The upper panel of this figure shows the scatter plot of the VaR_i^{1y} measurement from the first year sample period against the VaR_i^{1-5y} measures for 394 US stocks. The lower panel shows the scatter plot of $\Delta\text{VaR}_i^{1y,QBM}$ against VaR_i^{1-5y}. We use the CRSP database to obtain time series return data for these firms. To train the QBM model we first need to discretize the data, by creating 256 equally sized buckets between -0.5 and 0.5. Returns beyond this range are winsorized to the lowest or highest bin. We disregard stocks with more than 10 zero return days in the first 200 days (stale prices). For the augmentation we add 400 synthesized observations to the original 200 sample observations.

original one year sample, $\Delta\text{VaR}_i^{1y,QBM}$. The difference captures the added value of the QBM augmented data. The positive relationship in the lower panel shows that the $\Delta\text{VaR}_i^{1y,QBM}$ helps to better predict the larger sample risk measure.

To further analyze this relationship we setup the following regression framework:

$$\text{Risk}_i^{1-5y} = c + \beta_1 \text{Risk}_i^{1y} + \beta_2 \Delta\text{Risk}_i^{1y,QBM} + \varepsilon_i. \tag{9}$$

Here, Risk_i^{1-5y} denotes a given risk measure for firm i, calculated over the first five years of their returns. The term Risk_i^{1y} reflects the same measure based only on the first year of data. Furthermore, $\Delta\text{Risk}_i^{1y,QBM}$ captures the difference between the risk measure estimated from QBM-augmented data and that from the original one-year sample. Our primary interest lies in the estimate for the coefficient β_2, where a positive significant value indicates whether the additional information provided by QBM-generated observations improves the prediction of long sample risk.

Table 2 summarizes the results of the regression analysis, showing that the predictability of risk measures varies slightly across different metrics. The first row highlights that using risk mea-

Table 2: Prediction large sample risk with small sample risk measures

	QBM					
	st. dev.	$VaR_{5\%}$	$ES_{5\%}$	skew	kurtosis	tail index
Risk^{1y}	0.86^{***}	0.72^{***}	0.81^{***}	0.67^{***}	2.86^{***}	0.60^{***}
	(0.02)	(0.03)	(0.03)	(0.08)	(0.28)	(0.03)
$\Delta\text{Risk}^{1y,\text{QBM}}$	0.13^{**}	0.38^{***}	0.07^{**}	-0.04	0.34^{***}	0.05^{**}
	(0.05)	(0.06)	(0.03)	(0.03)	(0.09)	(0.02)
Constant	-0.89	34.48^{***}	24.84^{***}	0.06	-16.74^{***}	0.01^{**}
	(0.55)	(3.52)	(3.07)	(0.07)	(5.43)	(0.002)
R^2	0.77	0.77	0.73	0.17	0.21	0.51
	RBM					
Risk^{1y}	0.82^{***}	0.82^{***}	0.78^{***}	0.69^{***}	2.70^{***}	0.60^{***}
	(0.02)	(0.03)	(0.03)	(0.10)	(0.75)	(0.03)
$\Delta\text{Risk}^{1y,\text{RBM}}$	-0.02	-0.01^{*}	-0.02^{**}	0.0004	0.28	0.01
	(0.01)	(0.005)	(0.01)	(0.09)	(0.63)	(0.01)
Constant	1.01^{**}	21.81^{***}	24.28^{***}	0.11	-1.65	0.01^{***}
	(0.39)	(3.00)	(3.02)	(0.07)	(7.30)	(0.002)
R^2	0.77	0.74	0.73	0.17	0.18	0.50
Observations	394	394	391	394	394	394

This table displays the results of the regression analysis of Equation (9) to asses the increased accuracy of risk measures for young firms (First year around one year after the initial public offering). We use the CRSP database to obtain time series return data for these firms. The independent variables for each column are the risk metrics calculated for the first year of a firms listing (Risk^{1y}). The second risk measure ($\Delta\text{Risk}^{1y,\text{QBM}}$) is the difference between $Risk^{1y}$ and the risk measure extracted from the sample data augmented with the QBM synthesized data. The dependent variable is the same metric on an extended sample of the first 5 years after a firms listing. The second panel displays the results of a similar excersize, with the difference being that they synthesized data comes from the RBM. For the analysis we discretize the data by creating 256 equally sized buckets between -0.5 and 0.5. We disregard stocks with more than 10 zero return days in the first 200 days (stale prices). We add 400 synthesized observations to the original 200 sample observations for the augmented sample. The starts that indicate the significance level are *p<0.1; **p<0.05; ***p<0.01.

sures extracted from the one year ordinary sample significantly predict their five-year counterparts in the expected positive direction. In the second row, we report the coefficient estimates of our variables of interest, the information added by the QBM synthesized data. The two uncertainty measures, standard deviation and kurtosis, benefit from the augmentation of QBM-synthesized data, as indicated by their positive and significant coefficients. The slightly lower significance level for the standard deviation is intuitive; since the standard deviation is less sensitive to tail observations, it is expected to be well estimated even in relatively small samples. In contrast, kurtosis, which captures the fourth moment of the distribution, is highly sensitive to tail observations and is therefore expected to benefit the most from additional data.

The measures focused on characterizing left-tail behavior—excluding skewness—are positive and statistically significant in predicting the large-sample risk measures. While the short-sample estimates themselves are informative of the longer-sample counterparts, the same predictive property does not hold uniformly for the QBM-augmented estimates. However, metrics that rely exclusively on left-tail data do improve with QBM-based augmentation. This is expected, as these measures are particularly sensitive to rare observations and thus benefit the most from an expanded sample.

The variation in significance across the 5% VaR, ES, and the left-tail index can be partly explained by the limitations of Boltzmann Machines. Extending the distribution beyond the range observed in the original data, without imposing strong parametric assumptions, is inherently challenging. Among these three measures, the 5% VaR is the least reliant on data beyond the observed range, making it more robust to synthetic augmentation. In contrast, both ES and the tail index depend more heavily on extrapolated tail behavior, which leads to lower significance levels compared to the 5% VaR.

In the lower panel, we contrast the QBM results with those from the RBM. The RBM struggles to add significant information to the one-year sample, with most coefficients being insignificant. While the VaR and conditional expectation estimates are significant, they have the wrong sign. This aligns with the simulation results, where the QBM consistently outperforms its classical counterpart.

6 Conclusion

This paper explores the potential of Quantum Boltzmann Machines (QBMs) as generative models for improving risk estimation in data-scarce environments. Compared to classical Restricted Boltzmann Machines (RBMs), QBMs offer a more flexible framework that can better approximate complex, heavy-tailed distributions when observations are limited. Our results suggest that even in constrained settings, QBMs can add useful information to small samples, enhancing the estimation of risk measures that are typically difficult to assess with limited data.

We apply this framework to a financial setting, focusing on risk assessment for newly listed firms using return data from the CRSP database. By augmenting short return samples with QBM-generated data, we show that estimates of long-term risk measures, particularly those sensitive to tail behavior, such as VaR, expected shortfall, and kurtosis, improve meaningfully. In contrast, RBM-augmented samples do not provide the same level of predictive enhancement. These findings suggest that QBMs can offer practical benefits in financial applications where

robust risk estimation is needed despite limited historical data.

It is important to note that the current generation of quantum hardware still imposes limits on the training and scalability of QBMs. Nonetheless, as quantum computing technology continues to evolve, the relative advantage of QBMs over classical models is expected to grow. This positions QBMs as a forward-looking tool for financial modeling, particularly in applications where data scarcity is a fundamental constraint. By leveraging quantum mechanical properties to better capture distributional complexity, QBMs represent a promising step toward more robust and accurate risk assessment in modern financial environments.

7 Acknowledgement

We would like to thank Stenio Fernandes, Maryam Haghighi, Miguel Molico and Cameron Perot for their helpful comments. We also thank the seminar participants at the Bank of Canada and Quantum Computing Applications in Economics and Finance Conference 2025.

References

Adachi, S. H. and M. P. Henderson (2015). *Application of Quantum Annealing to Training of Deep Neural Networks*.

Amin, M. H., E. Andriyash, J. Rolfe, B. Kulchytskyy, and R. Melko (May 2018). "Quantum Boltzmann Machine". In: *Physical Review X* 8.2.

Anschuetz, E. R. and Y. Cao (2019). "Realizing quantum boltzmann machines through eigenstate thermalization". In.

Artzner, P., F. Delbaen, J.-M. Eber, and D. Heath (1999). "Coherent measures of risk". In: *Mathematical Finance* 9.3, pp. 203–228.

Basel Committee on Banking Supervision (June 2011). *Basel III: A global regulatory framework for more resilient banks and banking systems*. https://www.bis.org/publ/bcbs189.htm. Revised version.

Bhasin, N. K., S. Kadyan, K. Santosh, R. HP, R. Changala, and B. K. Bala (2024). "Enhancing Quantum Machine Learning Algorithms for Optimized Financial Portfolio Management". In: *2024 Third International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS)*. IEEE, pp. 1–7.

Chu, Y., X. Zhao, Y. Zou, W. Xu, and Y. Z. J Han (2018). "A Decoding Scheme for Incomplete Motor Imagery EEG With Deep Belief Network". In: *Front Neurosci*.

Coyle, B., M. T. Henderson, J. C. J. Le, N. Kumar, M. Paini, and E. Kashefi (2021). "Quantum versus classical generative modelling in finance". In: *Quantum Science and Technology* 6 (2), p. 024013.

Dallaire-Demers, P. and N. Killoran (2018). "Quantum generative adversarial networks". In: *Physical Review A* 98 (1).

Danielsson, J. (2011). *Financial risk forecasting: the theory and practice of forecasting market risk with implementation in R and Matlab*. John Wiley & Sons.

Danielsson, J. and C. G. de Vries (1997). "Tail index and quantile estimation with very high frequency data". In: *Journal of Empirical Finance* 4.2–3, pp. 241–257.

Danielsson, J. and C. G. de Vries (2005). "Using a bootstrap method to choose the sample fraction in tail index estimation". In: *Journal of Multivariate Analysis* 94.2, pp. 295–314.
Du, Z., J. C. Escanciano, and X. Zhu (2018). "Forecasting value-at-risk: The role of conditional skewness and kurtosis". In: *Journal of Financial Econometrics* 16.3, pp. 424–458.
Duffie, D. and J. Pan (1997). "An overview of value at risk". In: *Journal of derivatives* 4.3, pp. 7–49.
Embrechts, P., C. Kluppelberg, and T. Mikosch (2005). *Quantitative Risk Management: Concepts, Techniques and Tools*. Princeton, NJ: Princeton University Press.
Gabaix, X. (2009). "Power laws in economics and finance". In: *Annu. Rev. Econ.* 1.1, pp. 255–294.
Ganguly, S. (2023). "Implementing quantum generative adversarial network (qgan) and qcbm in finance". In: *arXiv preprint arXiv:2308.08448*.
Gao, J., M. J. Kim, and I. Tsiakas (2022). "Bias-corrected expected shortfall estimation and backtesting". In: *Journal of Financial Econometrics* 20.2, pp. 313–341.
Golden, S. (Feb. 1965). "Lower Bounds for the Helmholtz Function". In: *Phys. Rev.* 137 (4B), B1127–B1128.
Golubeva, A. and R. G. Melko (2021). "Pruning a restricted boltzmann machine for quantum state reconstruction". In.
Hill, B. M. (1975). "A simple general approach to inference about the tail of a distribution". In: *The Annals of Statistics* 3.5, pp. 1163–1174.
Hinton, G. (Aug. 2010). "A Practical Guide to Training Restricted Boltzmann Machines (Version 1)". In: *Technical Report UTML TR 2010-003, University of Toronto* 9.
Hinton, G. E. (2002). "Training products of experts by minimizing contrastive divergence". In: *Neural computation* 14.8, pp. 1771–1800.
Hinton, G. E. and R. R. Salakhutdinov (2006). "Reducing the dimensionality of data with neural networks". In: *science* 313.5786, pp. 504–507.
Hoga, Y. (2022). "Limit theory for forecasts of extreme distortion risk measures and expectiles". In: *Journal of Financial Econometrics* 20.1, pp. 18–44.
Jorion, P. (2007). *Value at Risk: The New Benchmark for Managing Financial Risk*. 3rd. McGraw-Hill.
Kieferová, M. and N. Wiebe (2017). "Tomography and generative training with quantum boltzmann machines". In: *Physical Review A* 96 (6).
Korutcheva, E., K. Korutchev, S. N. Santalla, J. Rodríguez-Laguna, and H. Chamati (2023). "The restricted boltzmann machine ansatz through adiabiatic routes". In: *Journal of Physics: Conference Series* 2436 (1), p. 012001.
Liu, J. (2018). "Differentiable learning of quantum circuit born machine". In.
Martins, L. F. and J. F. Ziegel (2021). "Estimating and evaluating conditional expected shortfall with reduced sample sizes". In: *Journal of Business & Economic Statistics* 39.1, pp. 260–273.
McNeil, A. J. and R. Frey (2000). "Estimation of tail-related risk measures for heteroscedastic financial time series: an extreme value approach". In: *Journal of Empirical Finance* 7.3-4, pp. 271–300.
Orlandi, F., E. Barbierato, and A. Gatti (2024). "Enhancing Financial Time Series Prediction with Quantum-Enhanced Synthetic Data Generation: A Case Study on the S&P 500 Using a Quantum Wasserstein Generative Adversarial Network Approach with a Gradient Penalty". In: *Electronics* 13.11, p. 2158.
Patton, A. J., J. F. Ziegel, and R. Chen (2019). "Dynamic semiparametric models for expected shortfall (and Value-at-Risk)". In: *Journal of Econometrics* 211.2, pp. 388–413.

Perot, C. (2023). *Quantum Boltzmann Machines: Applications in Quantitative Finance*.
Pritsker, M. (2006). "The hidden dangers of historical simulation". In: *Journal of Banking & Finance* 30.2, pp. 561–582.
Smolensky, P. et al. (1986). "Information processing in dynamical systems: Foundations of harmony theory". In.
Song, H., T. Song, Q. He, Y. Liu, and D. L. Zhou (2019). "Geometry and symmetry in the quantum boltzmann machine". In: *Physical Review A* 99 (4).
Stein, J., D. Schuman, M. Benkard, T. Holger, W. Sajko, M. Kölle, J. Nüßlein, L. Sünkel, O. Salomon, and C. Linnhoff-Popien (2024). "Exploring Unsupervised Anomaly Detection with Quantum Boltzmann Machines in Fraud Detection". In: *Proceedings of the 16th International Conference on Agents and Artificial Intelligence*. SCITEPRESS - Science and Technology Publications, pp. 177–185.
Thompson, C. J. (1965). "Inequality with Applications in Statistical Mechanics". In: *Journal of Mathematical Physics* 6, pp. 1812–1813.
Tüysüz, C., M. Demidik, L. Coopmans, E. Rinaldi, V. Croft, Y. Haddad, M. Rosenkranz, and K. Jansen (2024). "Learning to generate high-dimensional distributions with low-dimensional quantum Boltzmann machines". In: *arXiv preprint arXiv:2410.16363*.
Xu, G. and W. S. Oates (2021). "Adaptive hyperparameter updating for training restricted boltzmann machines on quantum annealers". In: *Scientific Reports* 11 (1).
Zhou, X., H. Zhao, Y. Cao, X. Fei, G. Liang, and J. Zhao (2024). "Carbon market risk estimation using quantum conditional generative adversarial network and amplitude estimation". In: *Energy Conversion and Economics* 5.4, pp. 193–210.
Zhu, E. Y., S. Johri, D. Bacon, M. Esencan, J. Kim, M. Muir, N. Murgai, J. Nguyen, N. Pisenti, A. Schouela, K. Sosnova, and K. Wright (Nov. 2022). "Generative quantum learning of joint probability distribution functions". In: *Physical Review Research* 4.4.

Appendix A Estimation with non-symmetric distributions

As we described in Section 4, estimating a symmetric distribution on a symmetric interval can introduce spurious negative correlations between the errors made with the sample estimations of the true distributions and the synthetic estimations of the sample distributions. This in turn enhances the estimation of the moments of the true distributions with respect to the fully general setting. For this reason, here we replicate the simulated estimation study of section 4 with a moving true distribution. We keep the estimation interval fixed at $[-10, 10]$, but add a random shift to the distribution center; in other words, in each simulation run the true distribution is given by a Student-t with 5 degrees of freedom for the variable $(x-s)$, with s a random variable uniformly distributed between $[-10, 10]$. For each simulation run, we first extract a value for s, then extract a large sample from a Student-t (5) distribution centered in $(x-s)$, and finally we keep only 100 of the sample points falling in the interval $[-10, 10]$. In this way we are able to generate a sample of true (and sample) distributions which are not centered in the domain of the estimated variables, in order to cancel the enhancing effects of symmetry in the estimation process. A drawback of this process is that the true values of the distribution moments change between the runs, hence the only significant measurements of the estimation performance are the error terms between QBM and RBM estimation on one side, and sample and true values on the other (as opposed to the situation for the fixed true distribution, in which it was meaningful to compare the averages and standard deviations of the estimated moments). In Table 3 we

show the comparison results for the new estimation procedure.

Table 3: Comparison of RBM and QBM Moving Averages

	RBM with moving mean			QBM with moving mean		
	$\varepsilon^{\text{sample}}_{\text{true}}$	$\varepsilon^{\text{synt}}_{\text{true}}$	$\varepsilon^{\text{synt}}_{\text{sample}}$	$\varepsilon^{\text{sample}}_{\text{true}}$	$\varepsilon^{\text{synt}}_{\text{true}}$	$\varepsilon^{\text{synt}}_{\text{sample}}$
x	0.13	0.60	0.57	0.13	0.42	0.41
x^2	1.29	6.83	6.70	1.29	2.66	2.43
x^3	13.33	49.82	47.65	13.33	20.93	17.23
x^4	135.55	582.85	565.61	135.55	243.20	211.18

RBM to QBM comparison for moving true distributions. $\varepsilon^{\text{sample}}_{\text{true}}$ is the root mean squared error between the sample estimation of the moments and the true values (identical in the RBM and QBM cases). $\varepsilon^{\text{synt}}_{\text{true}}$ and $\varepsilon^{\text{synt}}_{\text{sample}}$ are, respectively, the root mean squared error between the synthetic estimations and the true values and the synthetic estimations and the sample ones. As can be seen from the table, most of the error in the estimations of the true values come from the synthetic estimation of the sample distributions.

The results of Section 4 are confirmed by this new procedure: the estimation errors with the QBM model are substantially lower than with the RBM model, and are of the same order of magnitude of the errors between the sample and true distributions. The correlations terms, while playing a smaller role in this setting, are still negative for the QBM estimation errors: they are approximately equal to -8% for the first, second and third moments, and closer to -7% for the fourth moment. The RBM estimation errors, on the other hand, present positive correlation.

Appendix B Relevance of correlation terms

In this appendix we show the role of the correlation terms between the error terms $\varepsilon^{\text{sample}}_{\text{true}}$ and $\varepsilon^{\text{synt}}_{\text{sample}}$ in the performance of the estimation of the true distribution using synthetic data. In the process of estimating a statistics $\hat{\theta}$ we perform a two-step process. First, we obtain a sample - typically of small size - extracted from the true distribution. We use this sample to train a generative model, from which we extract an arbitrarily large number of new data points. Here we are interested on studying the relevant factors influencing the magnitude of the mean square error $\text{MSE}(\hat{\theta}_{\text{aug}})$ between the estimator evaluated on the synthetic data, and the true value. With a simple variance decomposition we obtain:

$$\text{MSE}(\hat{\theta}_{\text{aug}}) = \mathbb{E}[(\varepsilon^{\text{synt}}_{\text{sample}})^2] + \mathbb{E}[(\varepsilon^{\text{sample}}_{\text{true}})^2] + 2\,\mathbb{E}[\varepsilon^{\text{synt}}_{\text{sample}}\,\varepsilon^{\text{sample}}_{\text{true}}] \tag{10}$$

where

$$\varepsilon^{\text{synt}}_{\text{sample}} = (\hat{\theta}_{\text{synt}} - \hat{\theta}_{\text{sample}}),$$

$$\varepsilon^{\text{sample}}_{\text{true}} = (\hat{\theta}_{\text{sample}} - \hat{\theta}_{\text{true}}),$$

and $\hat{\theta}_{\text{true}}$, $\hat{\theta}_{\text{sample}}$, $\hat{\theta}_{\text{synt}}$ are the estimators evaluated with the different types of data.

Whenever the covariance term in equation (10) is null, we see that the best estimation is performed when the error between synthetic data and sample data is minimized, and the quality of the estimation is at best as good as the sample one. If, however, as we saw in section 4 the covariance term is negative, we can obtain an estimation of the true value of $\hat{\theta}$ better than the one available with sample data alone. Of course this is only possible because of the special symmetry of the true distritbution, which is symmetric around zero.

Appendix C Risk measures

In this section we shortly discuss the risk measures we extract from the financial data. Consider a series of return data, $R_1, R_2, ..., R_n$ of random variables. The sorted sample, i.e., order statistics, can be represented as

$$\max(R_1, ..., R_n) = R_{n,n} \geqslant R_{n-1,n} \geqslant ... \geqslant R_{1,n} = \min(R_1, ..., R_n).$$

Take the order-statistics based on the returns, i.e. $R_{(j,n)}$. We can define the following risk measures,

$$\text{Skewness} = \frac{E\left[(R-E(R))^3\right]}{E\left[(R-E(R))^2\right]^{3/2}} = \frac{\sum\left(R_{(j,n)} - \frac{1}{n}\sum R_{(j,n)}\right)^3}{\left(\sum\left(R_{(j,n)} - \frac{1}{n}\sum R_{(j,n)}\right)^2\right)^{3/2}}$$

and

$$\text{Kurtosis} = \frac{E\left[(R-E(R))^4\right]}{E\left[(R-E(R))^2\right]^2} = \frac{\sum\left(R_{(j,n)} - \frac{1}{n}\sum R_{(j,n)}\right)^4}{\left(\sum\left(R_{(j,n)} - \frac{1}{n}\sum R_{(j,n)}\right)^2\right)^2}.$$

In the financial risk literature and in many of the financial institutions regulatory frameworks the Value-at-Risk (VaR) forms an important assessment tool to gauge the level of risk. The VaR estimates the potential loss an investment portfolio may incur over a specific period, given a certain level of confidence.

$$VaR_p = -R_{([np],n)}$$

where $[np]$ is the integer part of np. Due to the need to better characterize the shape of the distribution beyond the VaR quantile. Financial regulators also adopted the conditional tail expectation, Expected shortfall,

$$ES_p = -E[R|R < -VaR] = -\frac{1}{[np]}\sum_{j=1}^{[np]} R_{(j,n)}.$$

As stated in the main text, an important statistic to characterize the risk of a stock is to find how thick the tails of the distribution are. The tail index, α provides such guidance as moments only exist up to α, i.e., $E \mid X_i \mid^p < \infty$ for $p < \alpha$, a decrease in α gives a heavier tail. To estimate the tail index α, the most popular tool is the Hill 1975 estimator

$$\hat{\gamma} = \frac{1}{\hat{\alpha}} = \frac{1}{k}\sum_{i=0}^{k-1}\left(\log\left(R^*_{n-i,n}\right) - \log\left(R^*_{n-k,n}\right)\right),$$

where k is the number of upper-order statistics used in the estimation of α. Furthermore, we are interested in the left tail and therefore $R^*_{(j,n)} = -R_{(j,n)}$.