Research Paper Series

Market Scenario Generation with GenAI

Michele Bonollo
Antonio Menegon
Giuseppe Crupi
Caterina Papetti

March 2026

iason
ESSENTIAL SERVICES FOR FINANCIAL INSTITUTIONS

iason

Research Paper Series

Year 2026 - Issue Number 84

Last published issues are available online:
http://www.iasonltd.com/research

Front Cover: **Umberto Boccioni**, *Elasticità*, 1916.

Executive Summary

Scenario generation is fundamental in financial risk management, enabling institutions to explore possible market evolutions, assess portfolio resilience, and design stress test exercises. Traditional approaches, such as historical simulation and Monte Carlo methods, are the go-to solutions, with practitioners and modellers who found ways to manage the sometimes restrictive assumptions and limitation which used to affect these approaches (e.g., heavy tails, volatility clustering, non-linear cross-asset dependencies, etc.).
Innovations in Machine Learning (ML), especially in the rapidly advancing field of Generative Artificial Intelligence (GenAI), are introducing promising alternatives for market data augmentation and synthetic scenario creation. Deep generative models, including Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), and Gaussian Mixture Models (GMMs), can learn non-linear dependencies and high-dimensional probability distributions directly from empirical data, promising to move beyond traditional resampling methods.
We evaluated academia and industry literature, including the emerging research adapting Transformer-like architectures (the basis of Large Language Models - LLMs) for integrating unstructured data (e.g., news or sentiment) into conditional factor generation. By integrating insights and critically comparing leading methodologies (VAE, GAN, GMM, and Transformers) across core metrics (e.g., interpretability, stability, data efficiency), this article wants to provide a practical overview to assess whether AI (and in particular GenAI) can be really production-ready for financial institutions in complement traditional approaches in market scenario generation and data augmentation.

About the Authors

Michele Bonollo:
Chief of Risk Methodologies Officer

He holds a degree in mathematics, a master degree in mathematical finance at the Paris VII University and a PhD in statistics. He worked as an executive in both large and regional italian banks. He has also collaborated with some consultant companies in the broad area of risk management, asset management, pricing models, software systems and regulatory compliance. Along with his professional activities, he developed applied research in the above fields, with more than 20 papers published in scientific journals and dozens of speeches in international conferences. He currently gives seminars and lessons in some top ranked Italian universities.

Antonio Menegon:
Chief Innovation Office
He holds bachelor and master degrees in mathematics, with a specialization in quantitative finance, complemented by post-degree studies in artificial intelligence and machine learning. He began his career as a risk quant and risk analyst before moving into leadership role advising major banking groups. He guided teams across risk management, front office, and ICT departments, driving model development, risk assessments and monitoring, digital transformation, governance, and AI initiatives. He leads the Group's Innovation Center, focusing on the application of technology and artificial intelligence in financial services.

Giuseppe Crupi:
Quant Analyst
He graduated in Data Physics from the Università Statale di Milano. His thesis, "Development of a Machine Learning framework for Anomaly Detection in Financial Time Series", demonstrates how machine learning and finance can coexist to drive future advancements. In 2023, he joined iason working on projects focused on Market Risk with one of the major Italian banks.

Caterina Papetti:
Quant Analyst
She holds a M.Sc. in Mathematical Engineering, Quantitative Finance, at Politecnico di Milano. She joined iason in 2024 and, as a Quantitative Analyst, she is currently working on various strategic projects related to Market and Credit Risk modelling at major Italian banks.

Table of Content

Scenario generation is a crucial, broad-spectrum component of risk management and strategic decision-making across the financial sector. Depending on the institution — be it a bank, an insurance firm, or an asset manager — and the specific use case, scenario modeling underpins a diverse range of critical functions. Applications span, for example, Market, Credit, and Solvency VaR, future exposure prediction in Counterparty Credit Risk (CCR), and forward-looking simulations of possible events for portfolio optimization. Regardless of the use case, it is always important for financial institutions to rely on reliable and sound data. In this regards, generating plausible future evolutions of key risk factors (e.g., interest rates, stock prices, credit spreads, etc.) to assess portfolio resilience and inform capital allocation decisions is the first goal to be achieved. Secondly, an equally important objective is to have sound data *augmentation techiques* when needed. This includes reconstructing complete historical datasets by filling missing data points (due to non-quotations, trading suspensions, or technical errors) and generating synthetic-but-plausible historical paths. This augmentation is especially critical for stabilizing risk metrics, like long-horizon Historical VaR, where native time series scarcity compromises reliability. The foundational "classic methods" to achieve these goals are based on **historical simulation** and **Monte Carlo** techniques. Widely used, these methods have been long fine-tuned and improved in the years to handle potentially critical reliance on strong parametric assumptions (e.g., *Gaussianity*, *stationarity*). Among the complexities to be handled, we acknowledge critical market stylized facts such as *heavy tails*, *volatility clustering*, and complex, *non-linear cross-asset dependencies*. These methodological challenges highlighted the need for more adaptive and data-driven frameworks. In the subsequent sections, we introduce Generative AI-based approaches as a complementary modern solution for both scenario generation and data augmentation. These methods, leveraging on architectures as Generative Adversial Networks (GANs), Variational AutoEncoders (VAEs), and Gaussian Mixture Models (GMMs), learn market distributions directly from empirical data, trying to overcome some of the constraints of traditional techniques. We will present the methodologies currently proposed in the literature, discussing whether these AI-driven simulations can offer realistic, diverse, adaptive, and financially coherent solutions for risk management.

1. Overview

1.1 Classic Methods and Their Limits

Traditional approaches to scenario generation rely on well-established statistical techniques, primarily categorized into historical and model-driven (Monte Carlo) simulations. Foundational, they have been revised by both the academia and the industry in the past decades not to rely solely on too simplified assumptions and not to fall short to properly fit the modern, complex financial markets.

1.1.1 Historical Simulation

The **historical simulation** generates scenarios by directly resampling from historical time series, often adjusting for recent volatility using techniques like rescaling or normalization. This approach is widely favored (e.g., *classic Market Risk VaR* used in the banking sector) due to its methodological simplicity and its reliance on realized data, sidestepping the need for explicit parametric modeling of risk factor distributions. However, one limitation of the standard historical simulation is its implicit assumption of identically distributed returns, which may fail to capture time-varying volatility and clustering effects observed in financial data. To address this, **Filtered Historical Simulation** (FHS) has been proposed as an enhancement by Barone-Adesi and Giannopoulos (2001) [5]. In FHS, returns are first filtered through a volatility model (typically a *GARCH-type* process), and the standardized residuals are then resampled to generate new scenarios. The simulated returns are reconstructed by reapplying the estimated conditional volatility, allowing the model to account for recent changes in market dynamics. Closely related is the work of Bonollo et al. [7], who develop an enhanced risk management framework based on the FHS model. Their approach extends *FHS-VaR* to produce scenario-based P&L distributions over a one-year horizon, maintaining consistency with underlying risk-factor dynamics and annual realized volatilities via a bootstrap procedure. By linking realized and conditional volatility, the model achieves proportional control over simulated returns, enhancing responsiveness to market shocks while improving backtesting performance and capital efficiency under stress.

1.1.2 Model-Driven (Monte Carlo) Simulation

Monte Carlo simulation remains a foundational quantitative tool, offering significant flexibility and robustness by simulating a large number of potential market paths. This is achieved by randomly sampling from explicit stochastic processes that model the time-evolution of financial risk factors. Once a model (e.g., *Black-Scholes*, *Vasicek*, *GARCH*, etc.) is specified, Monte Carlo is adept at accommodating complex dependencies, non-linear pay-offs, and path-dependent derivative valuations. This is the classic approach used in use case as, for example, *Counterparty Credit Risk metrics* and *Solvency II Internal Model's SCR*.

1.1.3 Systemic Limitations of Classical Modeling

While traditional models are indubitably still the industry best practice, their practical utility can be compromised by their dependence on simplifying assumptions that clash with empirical market data. Among these, we can acknowledge difficulties that have been long studied as:

- **Stylized Facts and Distributional Shape**: in most of their base settings, classical methods impose assumptions as Gaussianity (normal returns) and stationarity. This can lead to a systemic failure ([39]) to capture critical market characteristics that define the empirical distribution shape, including heavy tails and skewness (underestimating extreme events), serial and long-memory effects (autocorrelation), reliance on the i.i.d. bootstrap assumption which is often fundamentally violated in real data. Classical approaches addressing these limitations include the use of heavy-tailed distributions, such as the *Student-t distribution* [31], which better capture the fat tails and excess kurtosis typically observed in financial returns. Another widely adopted solution involves copula models, which decouple marginal distributions from their dependence structure, enabling a flexible modeling of nonlinear correlations and tail dependencies across risk factors [11, 22].

- **Rare Event Modeling**: due to parametrization choices and calibration constraints, rare events (e.g., crises, sudden market shocks) could be poorly represented. The probabilities of such extreme movements can be underestimated, leading to overconfident risk estimates and fragile stress tests. Among the methods used to overcome such a problem, one of the famously proposed solution entails the Extreme Value Theory (EVT) as discussed by Embrechts, Klüppelberg, and Mikosch (1997) [14] or Longin (2000) [29].

FIGURE 1: Summary of the main limitations of traditional models in scenario generation.

- **Dependency and Scalability Challenges**: models struggle with non-linear and time-varying cross-asset dependencies, resulting in oversimplified correlation structures. This issue is compounded by the curse of dimensionality, where modeling high-dimensional, multi-factor dependencies becomes computationally expensive and statistically more complex. A method highly discussed in literature and used in practice, from the banking to the insurance secotr, involves the use of copulas, as it has been proposed in works as Embrechts, McNeil and Straumann (2002) [15] and Demarta, McNeil (2005) [11].

- **Data Integration and Out-of-Sample Performance**: classical statistical approaches are traditionally limited to structured, tabular data, and mostly unequipped to incorporate unstructured inputs like financial news or sentiment. Indeed, integrating financial news into market scenario generation without AI primarily involves leveraging traditional statistical, econometric, and rule-based methodologies. These approaches focus on quantifying the impact of news content and incorporating it into models through historical correlations, sentiment indices, or regression-based techniques [2]. The fundamental challenge lies in systematically transforming unstructured textual information into structured data that can feed into quantitative financial models. Furthermore, calibrating models on historical data and still be able to generalize to new market regimes has always been a genuinely hard task.

1.2 The AI Paradigm: Data-Driven Modeling for Finance

The modeling challenges classical approaches have been facing led both the academia and the industry studying more data-driven and adaptive frameworks. **Advanced Machine**

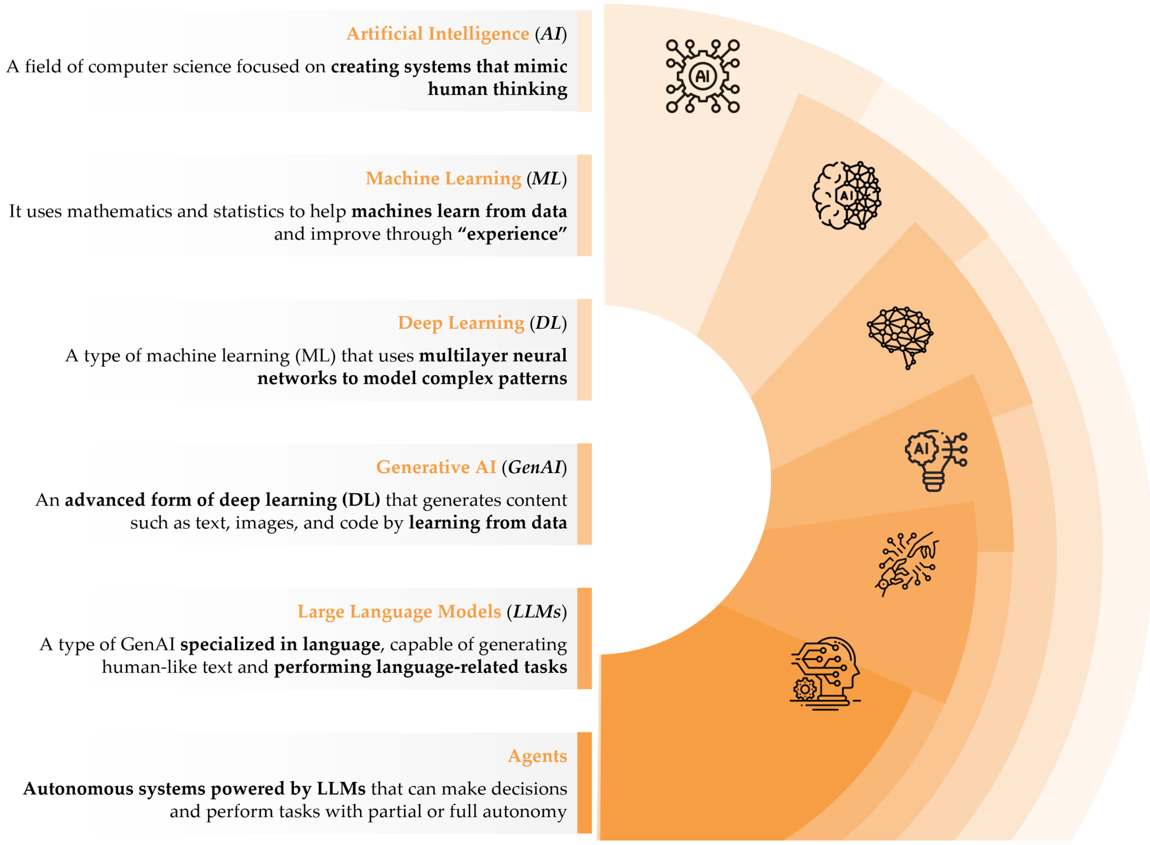

FIGURE 2: *Overview of different modern AI methodologies.*

Learning (ML) and **Deep Learning** (DL) techniques started to proliferate within the financial sector research space, most often for business and managerial risk management applications.

Traditional AI applications, such as credit scoring or fraud detection, typically employ discriminative models, which learn mappings from inputs to outputs by estimating conditional probabilities $P(Y \mid X)$. For instance, a Random Forest classifier distinguishes between "likely to default" and "not likely to default," while models such as LSTMs (Long-Short Term Memories) are used for time-series forecasting. Generative AI, by contrast, aims to model the joint structure of the data itself (i.e., $P(X)$). Rather than predicting labels, it learns to represent and reproduce the statistical regularities of observed financial phenomena. This enables the synthesis of new, realistic data samples and supports applications such as scenario generation and data augmentation. Prominent architectures adapted for such use cases in finance include **Generative Adversarial Networks** (GANs) [19], **Variational Autoencoders** (VAEs) [27], and **Gaussian Mixture Models** (GMMs) [32].

Among these models, **Generative Adversarial Networks** (GANs) learn to produce realistic synthetic data through the competition between a generator and a discriminator [19]. Their conditional variant, cGAN, enables controlled scenario creation by conditioning on specific variables such as volatility regimes or macro factors [34]. Similarly, **Variational Autoencoders** (VAEs) generate data by encoding and decoding through a probabilistic latent space [27]; their conditional form (*CVAE*) and temporal extension (*TimeVAE*) allow for richer, time-dependent representations of financial series.

Alongside these neural models, **Gaussian Mixture Models** (GMMs) provide a classical probabilistic approach [37] and a particularly interpretable and statistically grounded framework. By expressing the data distribution as a weighted sum of Gaussian components, GMMs capture non-linearities, skewness, and multi-modality in empirical financial return distributions. This flexibility makes them valuable for modeling heterogeneous market regimes or mixtures of volatility states. As discussed by Brigo and Mercurio (2006) [8], mixture models naturally extend classical parametric settings by combining multiple Gaussian densities to approximate complex empirical distributions while preserving analytical tractability. Their use in finance ranges from modeling implied volatility surfaces and interest rate dynamics to generating realistic risk-factor scenarios within a probabilistic framework.

To address the sequential nature of market data, **Recurrent Neural Networks** (RNNs) and their improved version, the **Long Short-Term Memory** (LSTM) network, capture temporal dependencies for forecasting applications [23]. More recently, **Transformers**, based on attention mechanisms [42] rather than recurrence, have emerged as efficient architectures to try to model both short- and long-range dependencies in time series, as well as mean to encode automatically unstructured data into scenario generation pipelines [12, 28, 3].

Together, these models form a methodological spectrum: while GANs, VAEs, and GMMs are mainly used for sampling realistic scenarios, LSTMs focus on forecasting temporal dynamics, and Transformers are better used to include unstructured data — a distinction central to the framework developed in this study.

Operating without restrictive parametric assumptions, AI (and particularly GenAI) models promise to emulate historical price dynamics, volatility clustering, and asset interdependencies with high fidelity. By capturing nonlinear relationships and cross-sectional structures, they are proposed to achieve a richer representation of stylized facts, such as heavy tails, skewness, and volatility clustering, while aiming at scaling effectively to high-dimensional settings.

Limits	Description	Classic Solutions	How GenAI addresses it
Stationarity	Models assume constant statistical properties over time.	• Differencing and Return Transformation [Tunnicliffe]; • GARCH and Stochastic Volatility Models [Bollerslev].	Captures evolving market regimes and structural breaks through dynamic, data-driven learning.
Gaussian Assumption	Returns are assumed to follow a normal distribution.	• Heavy-Tailed Distributions (Student-t)[Mandelbrot]; • Copula Models [Haugh]; • Extreme Value Theory (EVT)[Embrechts][Longin].	Learns non-Gaussian, heavy-tailed, and skewed distributions directly from data, improving tail risk representation.
Independence	Ignores autocorrelation and temporal dependencies in returns and volatility.	• ARCH and GARCH Models [Bollerslev]; • Multivariate GARCH (MGARCH, DCC-GARCH) [Engle].	Models temporal and cross-asset dependencies using recurrent or attention-based architectures.
Poor Representation of Rare Events	Underestimates the probability of crises or extreme scenarios.	• Extreme Value Theory (EVT) [Embrechts] [Longin]; • Copula Models for Tail Dependence [Haugh]; • Mixture Models (Gaussian Mixtures)[McLaugh].	Simulates rare and extreme events by learning from both typical and atypical historical patterns.
Markovianity	Assumes future states depend only on current ones, ignoring long-term memory effects.	• ARMA and ARIMA Models (Higher-Order Lags) [Tunnicliff]; • Long-Memory GARCH (FIGARCH, FIEGARCH)[Ballie].	Learns complex, path-dependent relationships.
Curse of Dimensionality	Estimating joint distributions becomes unstable in high-dimensional spaces.	• Principal Component Analysis (PCA) [Jolliffe]; • Copula Decomposition [Embrechts].	Learns high-dimensional dependencies efficiently using latent representations that preserve structure.

TABLE 1: Classical limitation across classical financial modeling approaches and most Generative AI methods.

2. Challenges of Market Scenario Generation

While both academia and industry are assessing whether Generative AI can be a complement (if not in some cases an alternative) to traditional financial modeling, its application to market scenario generation and data augmentation remains far from straightforward. As such models gain attention for risk management and stress testing use cases, it becomes essential to critically assess their reliability, consistency, and theoretical validity within financial contexts.

Three broad areas of challenge emerge at the intersection of finance and AI.

First, there is the challenge of **intra-class heterogeneity**. Risk classes — such as equities, interest rates, etc. — are not monolithic. Each is composed of multiple distinct risk factor types governed by their own dynamics, time scales, and statistical properties. For example, within the single Equity class, a model must simultaneously capture the behavior of spot prices, dividend curves, and implied volatility surfaces. Designing neural architectures capable of jointly capturing these diverse internal components without loss of fidelity is a non-trivial task.

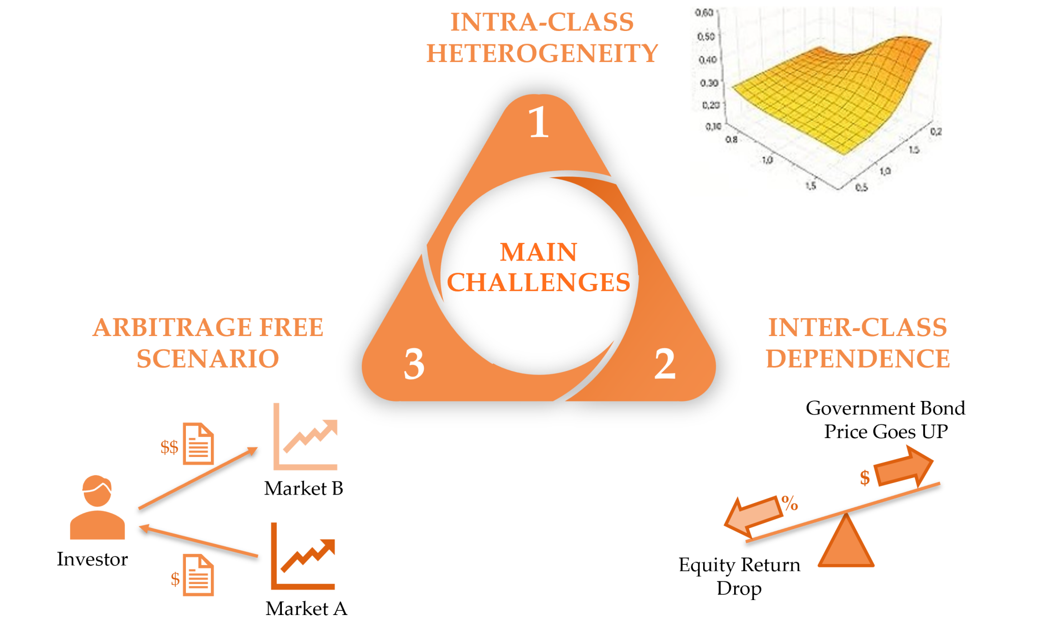

FIGURE 3: The figure illustrates the three key challenges in applying Generative AI to financial scenario generation. The first challenge is the modeling of multidimensional risk classes, such as spots, curves and volatility surfaces. The second challenge concerns inter-class dependencies, such as the example shown between government bond and equity returns during a 'flight-to-quality' period, when prices of the latter goes down, in favor of government bonds. The third and final challenge is the enforcement of financial consistency conditions — e.g., absence of arbitrage opportunities.

Second is the modeling of **inter-class dependencies**. Financial markets are deeply interconnected systems where shocks propagate across assets through non-linear and time-varying channels. A shock to interest rates, for instance, can cascade into equity valuations and credit spreads. Generative models must therefore go beyond marginal accuracy (i.e., modeling each class in isolation) to reproduce the complex web of co-movements and contagion effects that define real-world market behavior — especially under stress conditions.
Finally, the enforcement of financial consistency, most notably the **absence of arbitrage**, represents a foundational constraint often violated by unconstrained neural networks. Ensuring that generated scenarios remain economically valid requires integrating domain knowledge, such as pricing relationships and no-arbitrage principles, directly into model design or training objectives.
In the following sections, we examine these challenges in greater depth. We begin with the problem of modeling multidimensional risk classes, proceed to the representation of inter-class dependencies, and conclude with the mechanisms required to guarantee arbitrage-free scenario generation.

2.1 Intra-Class Heterogeneity

Within each financial risk class lies a multi-dimensional structure that generative models must reproduce accurately. Interest rates, for instance, are represented through curves across maturities; credit spreads vary across tenors and counterparties; equity or FX markets are characterized by both price levels and implied volatilities. Each of these elements embodies a distinct but interrelated risk dimension, driven by different economic mechanisms and data characteristics. Capturing their joint dynamics within a coherent framework is one of the most demanding tasks in AI-based scenario generation.

The challenge arises from the coexistence of heterogeneous data types — scalar, vector, and surface representations — each with different statistical properties, time scales, and sensitivities. Traditional neural architectures are typically optimized for homogeneous inputs, yet financial data within a single risk class can range from dense, high-frequency time series (e.g., spot prices) to sparse, structured surfaces (e.g., yield or volatility curves). Ensuring that models learn consistent patterns across these domains without distorting their internal relationships is non-trivial.
A key risk in this setting is the loss of internal coherence. For example, a generative model trained separately on yield curves and volatility surfaces might reproduce each marginal distribution accurately but fail to maintain the economic relationships between them — such as the link between term-structure shifts and changes in implied volatility. Similarly, naive training may violate smoothness or monotonicity constraints that are fundamental to financial realism.
Different architectures offer complementary strengths, as it will be better explained in the following section; for example:

- *Variational Autoencoder*s (VAEs), which can be viewed as the non-linear and non-parametric extension (or competitor) of Principal Component Analysis (PCA), can encode multi-dimensional structures into latent spaces that preserve relationships among term-structure, volatility, and liquidity factors, provided that the latent design reflects the geometry of the underlying data;

- *Generative Adversarial Networks* (GANs) can simulate complex shapes such as yield or volatility surfaces but require careful regularization to avoid mode collapse or unrealistic discontinuities;

- *Physics-informed or constraint-aware networks* can embed structural priors — such as no-arbitrage or curve-smoothness conditions — directly into the training objective, providing a more financially consistent output.

Ultimately, effective multidimensional modeling depends not only on data abundance but also on accurate architectural alignment with financial structure. Models must learn the internal geometry of each risk class (i.e., how its factors evolve jointly and respect financial constraints) before they can reliably support scenario generation or pricing applications.

2.2 Inter-Class Dependence

Beyond the internal consistency of each risk class lies the broader challenge of capturing dependencies among them. Financial markets operate as interconnected systems; e.g., movements in interest rates affect equity valuations through discount factors; credit spreads respond to both rates and macroeconomic stress; commodities influence inflation expectations, which in turn shape monetary policy. These linkages evolve dynamically and often strengthen under stress, amplifying systemic risk.
A generative model must therefore go beyond marginal (single-asset) accuracy to reproduce this complex web of co-movements, especially under stress conditions where correlations often spike and behave non-linearly. Standard correlation or copula frameworks provide well-established solutions, while deep generative models must infer such dependencies implicitly from data. This creates recurring potential problems as:

- *Hidden vs. explicit correlation modeling*: most generative models, particularly deep learning architectures, do not explicitly model correlation. Instead, they rely on

Asset Class	Statistical Characteristics	Main Driver	Temporal Dynamics
Equities	High volatility, clustering, fat tails.	Investor sentiment, earnings, market cycles.	High-frequency movements.
Rates	Lower volatility, mean-reversion.	Central bank policy, macroeconomic trends.	Slower-moving, medium to long term.
Credit	Sensitive to default risk and liquidity premiums.	Firm fundamentals, systemic risk.	Discontinuous jumps, stress events.
FX	Very high-frequency noise, heavy tails.	Trade flows, interest rate differentials.	Tick-level data, intraday patterns.
Commodities	Seasonality, regime shifts.	Supply/demand shocks, geopolitics.	Mixed: intraday to seasonal cycles.

TABLE 2: Main features of different asset classes.

Asset Class A	Asset Class B	Typical Dependency
Interest Rates	**Government Bonds**	Rising interest rates lead to lower bond prices due to the inverse yield-price relationship.
Interest Rates	**Equities**	Higher rates increase the discount factor for cash flows, generally reducing stock valuations.
Interest Rates	**Credit Spread**	Credit spreads tend to widen when rates rise or in periods of economic uncertainty, reflecting higher default risk.
Commodities (e.g. Oil)	**Inflation**	Energy prices are often positively correlated with inflation expectations, influencing central bank policy.
Equities	**Commodities (Oil)**	Rising oil prices can increase costs for firms, reducing profits and exerting downward pressure on equity markets.
FX (USD)	**Commodities (Oil, Gold)**	Many commodities are priced in USD; a stronger dollar often leads to weaker commodity prices.
Equities	**Credit Markets**	Deterioration in credit conditions or widening spreads often coincides with declining equity valuations.
Volatility index (VIX)	**Equities**	Equities & VIX tends to rise sharply during equity market downturns, reflecting increased risk aversion.

TABLE 3: Examples of cross-asset dependencies observed in financial markets.

the joint training of data to implicitly learn dependencies, which may lead to an under/over-estimation of weak or nonlinear links.

- *Overfitting to marginal structure*: generative models often optimize for marginal fidelity (e.g., minimizing reconstruction loss or adversarial divergence), sometimes at the

No Arbitrage Techniques	
No-Arbitrage Tecniques by Design	Pre- or Post-Fixing Techniques
Embedding economic constraints directly into the model architecture or training objective.	Detecting and correcting arbitrage either during training (as a regularization term) or after training (post-processing).
Examples: - Arbitrage-free neural networks; - Physics-Informed Neural Network; - Generative models trained with economic consistency conditions.	Examples: - Adding no-arbitrage penalties to the loss function during training; - Post-processing scenario outputs to remove arbitrage opportunities.

TABLE 4: Two main paradigms of incorporating no-arbitrage into machine learning models.

Model	Multi Risk Class	Risk Class Dependency	Arbitrage-free
GAN	Needs homogeneous assets.	Capture nonlinear dependencies.	Complex to check.
VAE	Only for multi-latent architectures.	Encode dependencies.	Possible with external constraints.
RNN (e.g. LSTM)	Works with time series of single typology.	Learn time-varying and sequential dependencies	Not easy.
GMM	Need normalization.	Represent dependencies via covariance structures.	Can be implemented.

FIGURE 4: Some AI models and their behavior with respect to the aforementioned challenges in scenario generation.

expense of capturing joint behavior.

- *Insensitivity to systemic events*: rare but crucial market regimes, such as coordinated crashes or liquidity freezes, are underrepresented in training data. Without tailored mechanisms, models may fail to generalize correlation behavior in stress conditions.

2.3 Arbitrage-free Scenario

One of the core requirements in financial modeling is the absence of arbitrage, the impossibility of obtaining a riskless profit with zero investment [45]. This principle underlies derivative pricing, market equilibrium, and risk management. Any realistic scenario generation method must therefore respect fundamental financial constraints such as no-arbitrage, pricing consistency, and risk-neutral valuation. However, the growing use of machine learning, especially deep neural networks, introduces challenges to these principles. Purely data-driven models optimize statistical loss functions without intrinsic awareness of financial structure, and may thus generate economically inconsistent scenarios, embedding artificial arbitrage opportunities. Typical examples include negative swap spreads, unjustified yield-curve inversions, or violations of put–call parity in option pricing outcomes that, while statistically plausible, are financially invalid. To mitigate such issues, the literature distinguishes **two main approaches**. The first *enforces no-arbitrage by design*, embedding financial structure directly into the model or its loss function. The second *detects and corrects violations ex-post*, either through regularization or post-processing adjustments. Both paradigms aim to align data-driven models with the fundamental economic logic required for credible financial scenario generation and data augmentation.

No-arbitrage by design approaches offer the most principled path toward economically valid generative models. One strategy is the development of ad hoc architectures, where the model's output space is constrained to only produce arbitrage-free outputs. A notable example of no-arbitrage by design is provided by Ning et al. (2022) [36], who propose a VAE framework specifically tailored to generate arbitrage-free implied volatility surfaces. An even more rigorous variant of the by-design approach leverages **physics-informed neural networks** (PINNs), both methods will be discussed deeply in the next chapter.

In contrast, pre/post fixing methods offer more flexible but less theoretically elengant solutions. Regularization-based techniques add penalty terms to the training loss that softly discourage arbitrage. For example, a term may penalize deviations from put-call parity, or penalize option price surfaces that violate convexity or monotonicity. These constraints act as guardrails during training, encouraging the model to stay close to the arbitrage-free region of function space without imposing hard constraints that might limit generalization. For instance, Hyndman (2021) adopts a pre-fixing approach, embedding arbitrage constraints directly into the training process through projection-based regularization, thus ensuring yield curve forecasts remain economically consistent [25]. Conversely, Vuletić (2023) proposes **VolGAN**, a post-fixing framework where a generative adversarial network produces implied volatility surfaces that are subsequently corrected via projection procedures to remove butterfly and calendar arbitrage, restoring arbitrage-freeness after generation [43]. Both approaches will be discussed in greater detail in the following section. While these methods cannot eliminate all arbitrage violations, they provide a practical balance when architectural constraints or complex loss formulations are computationally infeasible. Each approach has trade-offs: no-arbitrage architectures may be overly restrictive in high-dimensional settings, PINNs offer theoretical rigor but high computational cost, and regularization or post-processing methods are flexible yet lack formal guarantees. The appropriate choice depends on the application context — strict enforcement for regulatory risk use versus approximate correction for exploratory analyses.

In summary, maintaining arbitrage-freeness remains a central challenge as deep learning advances in finance. To ensure reliability and interpretability, data-driven models must remain anchored in financial theory, integrating no-arbitrage logic — whether through architecture, physics-informed losses, or corrective mechanisms — as a fundamental component of credible AI-based scenario generation consideration.

3. Approaches in Literature

This chapter provides an overview of the analyzed approaches to market scenario generation as found in the academic and practitioner literature. Starting from the concepts recalled on the classical statistical techniques, here we survey emerging machine learning models, highlighting how each method conceptualizes uncertainty, represents dependence structures, and handles regime dynamics. Particular attention is given to the ways in which different approaches address practical requirements such as stress realism, computational feasibility, and the integration of expert judgment.

The reader can find, for each studied approach, a brief description, the main results drawn from the original papers, and a closing summary of the key takeaways. Likewise, the whole

chapter concludes with an overall synthesis of these methods, leaving the reader the choice of where and how to dig deeper.
Overall, through this lens, our aim is to clarify the current state-of-the-art and identify open questions that continue to shape this evolving field.

3.1 Scenario Generation Using GAN by Flaig

A notable contribution to the application of deep learning in scenario generation is the work of Flaig [17], who proposes a framework for generating realistic market scenarios using generative neural networks. The study focuses on the use of GANs as tools to model the joint distribution of financial risk factors, emphasizing their *ability to capture non-Gaussian features, tail behavior, and complex dependencies across multiple assets*. In particular, all simulations evaluated by the author are based on multiple benchmark portfolios, with a different ratio of risk classes, as shown in Figure 5.
One of the main strengths of this approach is that it tries to model multiple risk classes at once, learning directly from historical observations without relying on restrictive parametric assumptions. Figure 7 illustrates the generation of new scenarios using a GAN. The generated samples closely replicate the structure of the historical data, while also extending beyond the original observations, thus creating plausible but novel scenarios. By contrast, a simple resampling approach would be confined within the boundaries of the original data cloud, failing to generate such new variations.
The paper also offers a comparative analysis with traditional techniques, demonstrating that neural approaches can outperform both historical simulation and simple parametric models in capturing the marginal and joint dynamics of financial variables. In particular, Flaig's GAN-based framework is benchmarked against the risk scenarios produced by Economic Scenario Generators (ESGs) of European approved Solvency II internal models, drawing on the results of the MCRCS study (available on EIOPA's homepage [46]). The comparison focuses on five key risk factor categories: corporate credit spreads, sovereign credit spreads, equities, and interest rates (up and down shocks), the results are shown in Figures 8, 6 and 9.
This work exemplifies the growing intersection between machine learning and financial risk management, offering a scalable and adaptive alternative to classical scenario generation methods.

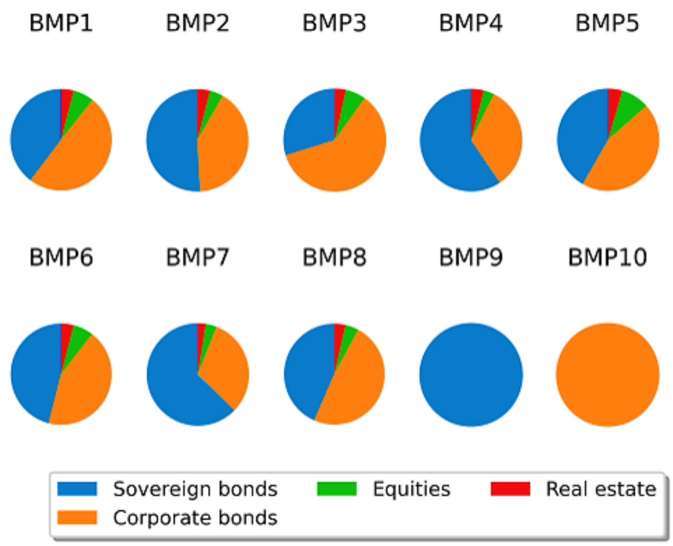

FIGURE 5: Different composition of the benchmark portfolios used for Flaig analysis [17]. Each of the 10 portfolios are made as a ratio of 4 main risk classes: Sovereign bonds (blue), Equities (green), Real Estate (red) and Corporate Bonds (orange).

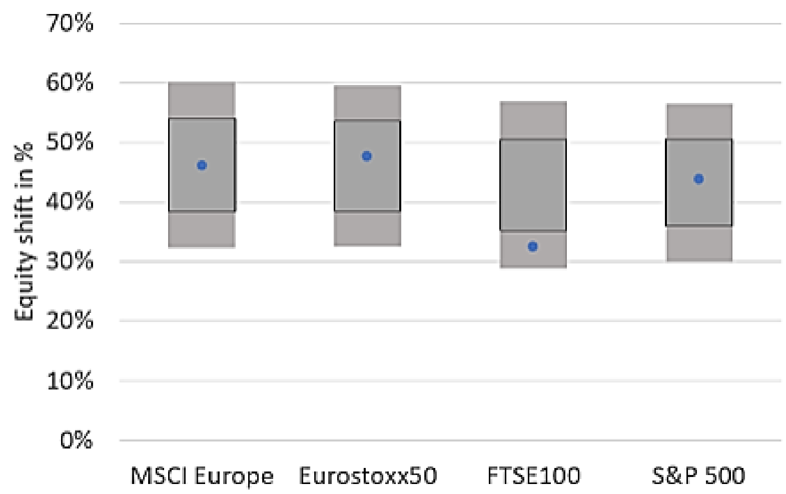

FIGURE 6: Comparison of the simulated shifts for equity risk factors, representation based on own results (blue dots) and EIOPA (gray boxes) [17]. The plot shows that the shifts are broadly consistent across most risk factors. For the FTSE100, the GAN generates less severe shocks compared to the other models. This outcome reflects the characteristics of the training data, as the FTSE100 exhibited lower volatility than the other indices over the period considered. Consequently, the GAN produces results that remain both plausible and data-driven.

3.2 GAN Applied to Synthetic Financial Scenario Generation by Rizzato

Another important contribution to the field of generative modeling for financial applications is provided by Marco Rizzato in his work "Generative Adversarial Networks Applied to Synthetic Financial Scenarios Generation" [38]. The study investigates the use of GANs to simulate synthetic financial time series aiming at *replicating real market dynamics*. The central idea is to train a GAN on historical asset returns so that it learns their distributional properties and can generate new synthetic paths that preserve both marginal features and cross-asset dependencies. The neural network architectures evaluated in the paper are illustrated in Figure 10.

Rizzato's work highlights the flexibility of GANs in capturing nonlinear, high-dimensional

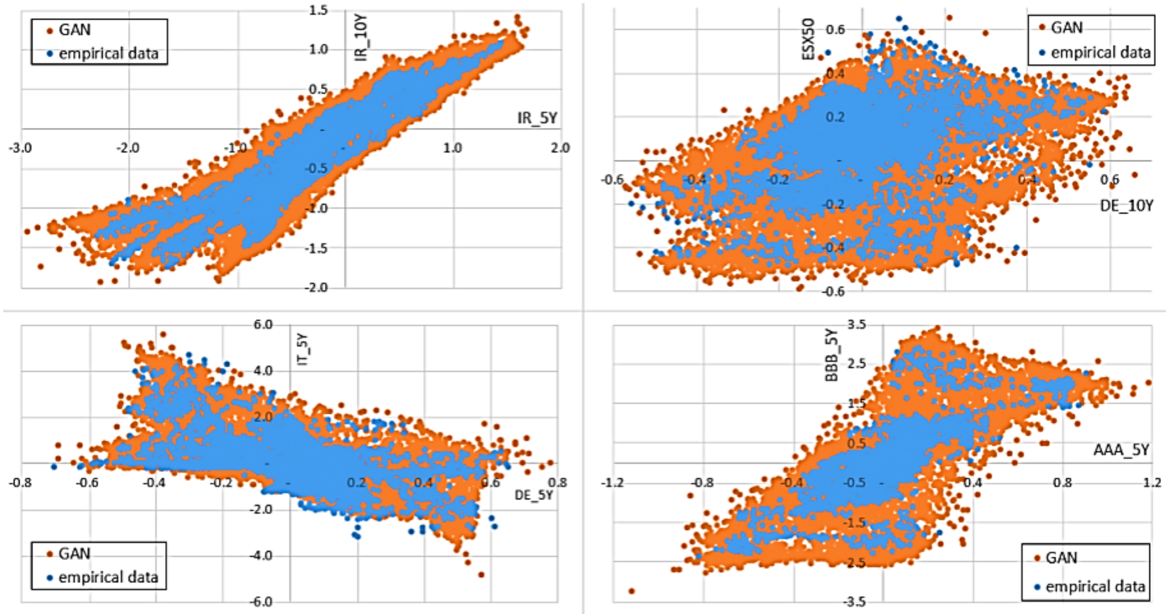

FIGURE 7: Scatterplots of four different risk factor pairs, empirical (blue dots) vs generated data by GAN (orange dots) [17]. The scatterplots display four pairs of risk factors: 5-year vs. 10-year interest rates, Eurostoxx50 vs. German government bond spreads, Italian vs. German government bond spreads, and AAA vs. BBB corporate credit spreads. In each graph, the blue dots correspond to the 4,330 empirical data points used for training, while the orange dots represent 50,000 scenarios generated by the GAN. As shown, the points generated by the GAN exhibit a distribution closely aligned with that of the original empirical data, while also introducing a greater diversity of samples. These newly generated data points lie within the same underlying distribution but were not present in the original training set, demonstrating the model's ability to generalize beyond observed data.

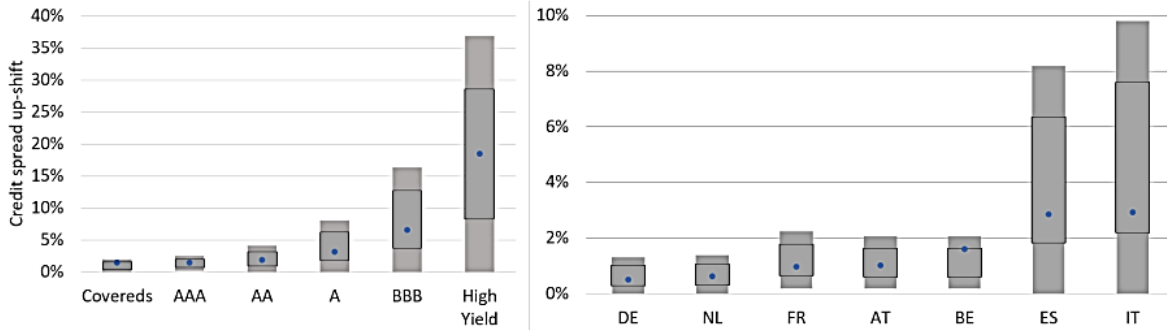

FIGURE 8: Comparison of the simulated shifts for the corporate and sovereign credit spread risk factors, representation based on own results (blue dots) and EIOPA (gray boxes) [17]. As shown, the GAN-based model aligns closely with the approved internal models, especially for corporate and sovereign spreads.

distributions that are difficult to model with traditional statistical approaches. The GANs successfully reproduce the shape of the empirical distributions, as shown in Figures 11 and 12, and this similarity is quantitatively assessed using performance metrics such as the Kolmogorov–Smirnov test and Principal Component Analysis. From a statistical perspective, these tools provide standard methods for evaluating the closeness of two multivariate distributions. Although GANs have shown excellent performance, Rizzato points out that in modeling high-dimensional data distributions across many domains, they are not universally applicable generative models. In fact, GANs do not provide an exact representation of the underlying data distribution and furthermore GANs with Gaussian priors can only generate sub-Gaussian distributions, highlighting a fundamental theoretical limitation.

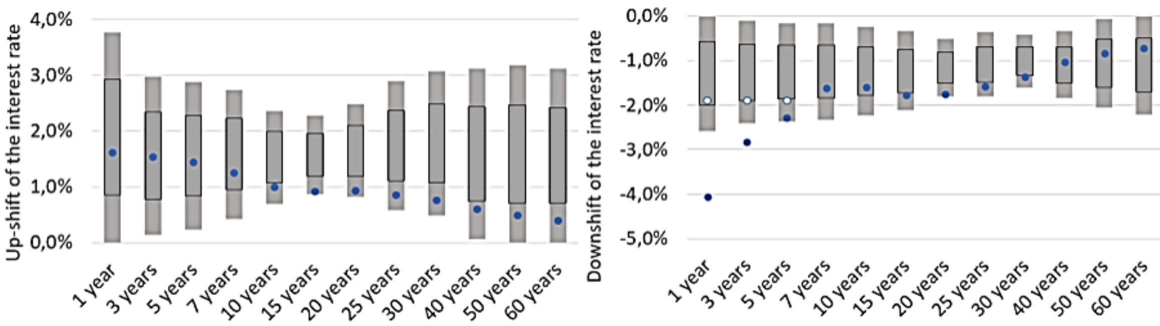

FIGURE 9: *Comparison of the simulated shifts (left: up, right: down) for the interest rate risk factors, representation based on own results (blue dots) and EIOPA (gray boxes). The up-shifts generated by the GAN-based model fall within the reference boxes across all maturities, though for longer tenors they tend to cluster toward the lower bound. This pattern reflects the training period, during which interest rates were predominantly declining. For down-shifts, a different behavior emerges: short-term rates lie below the boxes, while medium- and long-term rates remain within them. This outcome can be explained by the sharp decline in short-term rates observed in the training data, in contrast to the relative stability of longer maturities. The GAN replicates this behavior, whereas traditional ESGs, calibrated over longer horizons and often supplemented with expert judgement, typically impose a floor on how negative rates are allowed to become.*

Flaig (2023)		
Goal of the Paper	Results	About Challenges
To extend GAN-based scenario generation to a full internal market-risk model for insurance (one-year horizon under Solvency II) and compare with regulatory approved internal models.	Generated scenarios replicate empirical structures and extend beyond historical data; strong alignment with EIOPA benchmark models across multiple asset classes.	• The paper works with several risk categories, but they still not cover the full multidimensional space of risk classes; • It does not focus in depth on intra-risk-factor correlation dynamics (the dependencies are learned via GAN); • It does not explicitly treat free-arbitrage scenario generation.

TABLE 5: *Flaig (2023).*

3.3 Generation of Realistic Synthetic Financial Time-Series by Dogariu

The article "Generation of Realistic Synthetic Financial Time-Series" by Dogariu, Stefan, Boteanu, Lamba, Kim, and Ionescu [13] addresses the challenge of *producing synthetic financial datasets* that faithfully reproduce key stylized facts such as heavy-tail distributions, volatility clustering, autocorrelation, and cross-asset correlations, while enabling variable-length outputs and better data augmentation for downstream tasks. Rather than relying on sequence modeling networks alone, the authors explore multiple neural architectures, including fully connected GANs, convolutional GANs, VAEs, and Generative Moment Matching Networks (GMMNs), to generate realistic synthetics from U.S. equity market data, showing how such models handle cross-correlation between different assets and fixed-to-variable length time series. A significant contribution of the paper is its emphasis on evaluation methodology: beyond visual similarity, the authors introduce a portfolio trend prediction framework and quantitative metrics to validate the effectiveness of generated series in portfolio forecasting tasks. They demonstrate that their synthetic data preserves both statistical and predictive characteristics when applied to real trading scenarios.

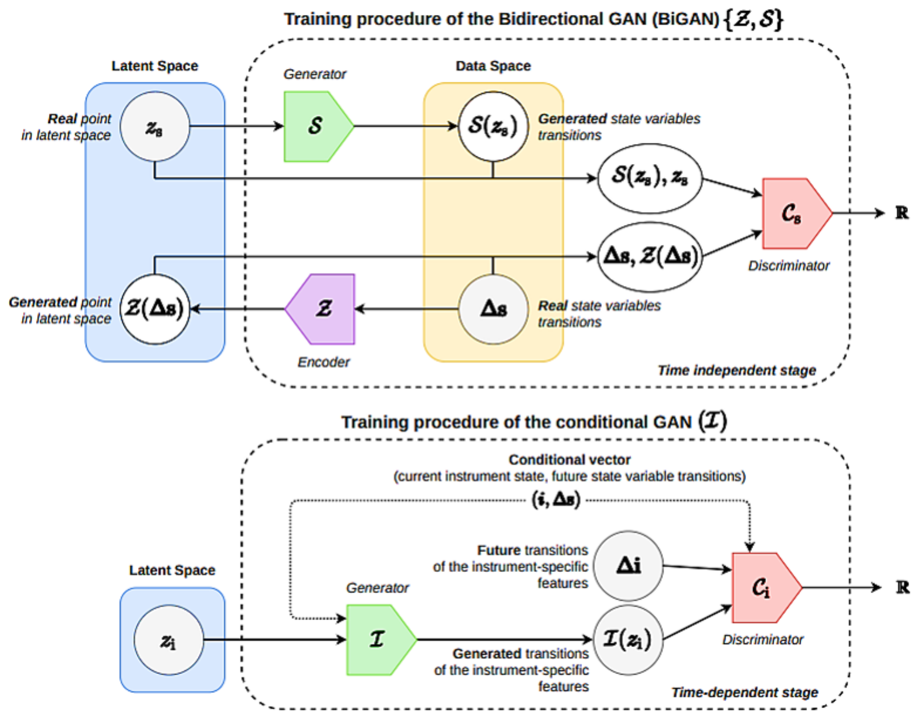

FIGURE 10: Graphical description of the two GAN used in the paper by Rizzato [38]. The BiGAN generator is trained on the historical dataset, and its performance is assessed by comparing the historical test set against an equally sized batch of synthetic data. The Conditional GAN, on the other hand, operates within a conditional probability framework, offering a more nuanced evaluation.

3.4 Time-Casual Variational Autoencoder Approach by Acciaio

A noteworthy recent contribution in the field is the **Time-Causal Variational Autoencoder** (TC-VAE) developed by Acciaio, Eckstein, and Hou [1], which addresses the overlooked issue of temporal causality in synthetic financial time-series generation.

The model's loss function provides an upper bound on the causal Wasserstein distance between true market distributions and generated sequences, offering theoretical guarantees on the closeness of statistical performance in decision-making tasks such as pricing, hedging, and portfolio optimization. Numerical experiments on synthetic models, such as Black-Scholes, are shown in Figure 14.

The same paper also provides additional comparisons with the Black–Scholes model, focusing on log-path and volatility distributions for both real and synthetic paths. A key metric employed is the sliced Wasserstein distance, which offers a principled approach to jointly comparing all one-dimensional projections between two measures, together with *Gaussian-kernel MMD* [20] and *Signature MMD* [10]. Figure 16 illustrates that real and generated paths remain relatively close under all three metrics.

All these tests show that TC-VAE effectively reproduces stylized market features (heavy tails, volatility clustering, skewness, kurtosis, and low autocorrelations) while offering strong performance on downstream tasks evaluated against enhanced statistical distance. This approach constitutes an important step forward in financial scenario generation with AI by combining a principled time-causal structure, rigorous mathematical control via adapted Wasserstein metrics, and empirical validation, making TC-VAE a possible robust and reliable tool for generating synthetic financial data.

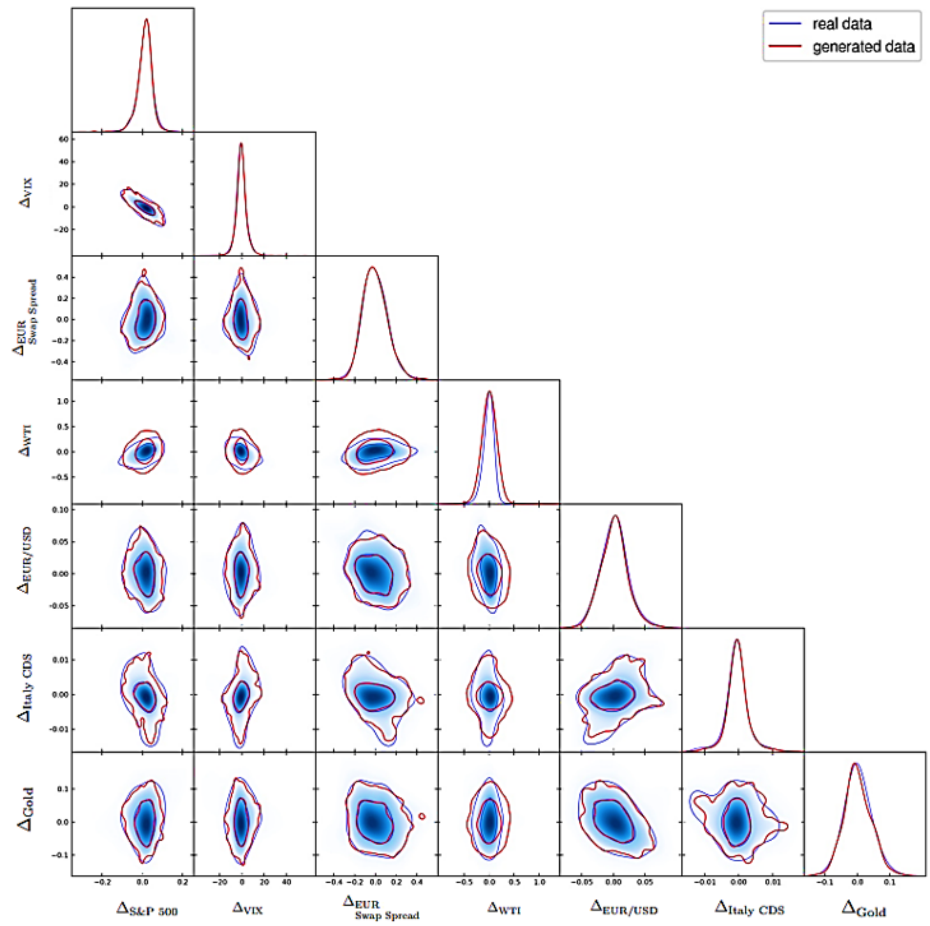

FIGURE 11: Triangle plot for the evaluation of the BIGAN generator S. Comparison between the real (blue) and the generated (red) state variable transitions. This visualization allows to compare one-dimensional (diagonal panels) and two-dimensional marginal distributions (off-diagonal panels) sorting them by couples of variables. In each subplot, the 68% and the 95% confidence intervals are proposed [38]. As shown in the figure, for each of the asset classes considered, the data generated by the BiGAN exhibit a distribution that closely mirrors that of the real data — both in the univariate case and in the bivariate setting, when examining the joint behavior of two classes.

3.5 Synthetic Financial Time Series Generation with LSTM by Schwarz

A recent and highly relevant contribution, if not properly solely generative modeling, is Schwarz contribution (2024) on financial time series with his work titled "Interpretable GenAI: Synthetic Financial Time Series Generation with Probabilistic LSTM" [40]. Despite Generative models which draw samples from the underlying data distribution to produce a range of plausible and coherent financial scenarios, LSTM based approaches are *primarily designed for forecasting, focusing on learning temporal dependencies to predict future values*.
The proposed model (see the neural network design in Figure 17) addresses the dual challenge of producing realistic market scenarios while maintaining interpretability, two attributes that are often at odds in deep learning architectures.
What sets it apart is its incorporation of interpretability constraints, allowing practitioners to understand how scenarios are formed from a blend of deterministic components (e.g., ARMA/GARCH structures) and learned nonlinear transformations. Overall, Schwarz offers a compelling paradigm: combining generative capability with explainability and statistical structure. This balances performance with understandability, and aims at helping bridge the gap between black-box deep models and classical financial time-series approaches.

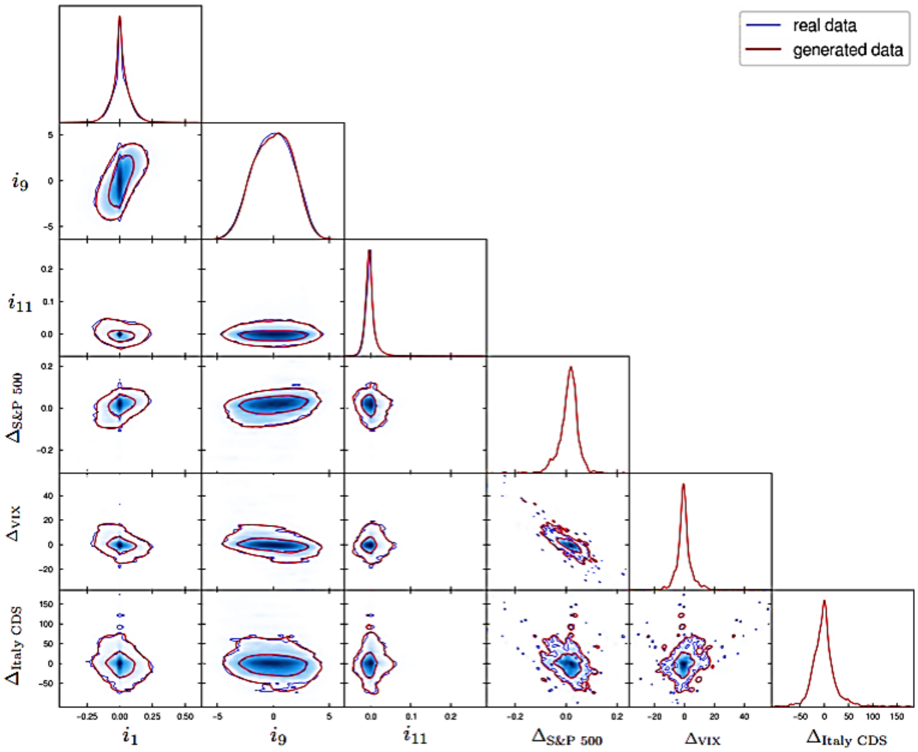

FIGURE 12: Triangle plot for the evaluation of the cGAN generator I. Comparison between the real (blue) and generated (red) data. This visualization allows to compare one-dimensional (diagonal panels) and two-dimensional marginal distributions (off-diagonal panels) sorting them by couples of features. In each subplot, the 68% and the 95% confidence intervals are proposed [38]. As shown in the figure, for each of the asset classes considered, the data generated by the cGAN exhibit a distribution that closely mirrors that of the real data — both in the univariate case and in the bivariate setting, when examining the joint behavior of two classes.

Dogariu et al. (2023)		
Goal of the Paper	Results	About Challenges
The goal of the paper is to develop generative-model techniques (GANs, VAEs) to synthesise realistic financial time-series data for problem of data augmentation.	Synthetic data preserve heavy tails, volatility clustering, and cross-asset correlation.	• The paper focus mainly on multiple stocks, avoiding other type of risk classes; • It addresses cross-correlations between different stocks, but it does not appear to more deeply model intra-risk-factor correlation structures; • The work does not explicitly enforce arbitrage-free constraints.

TABLE 6: Dogariu et al. (2023).

Shwarz's proposal stands out as a practical tool for producing synthetic scenarios that want to be both realistic and interpretable, suitable for risk analysis, stress testing, algorithmic strategy design, or data augmentation.

3.6 Arbitrage-Free Implied Volatility with VAE by Ning

No-arbitrage by-design approaches provide the most principled route toward economically consistent generative models. One effective strategy involves developing ad hoc architectures in which the model's output space is intrinsically restricted to arbitrage-free

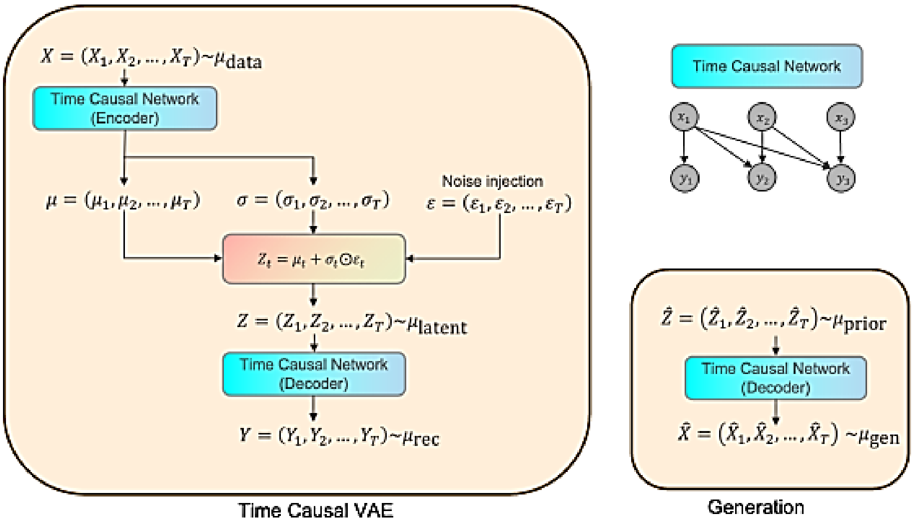

FIGURE 13: *Time-casual variational autoencoder and generation. Unlike standard VAEs, TC-VAE enforces causality constraints on both encoder and decoder, ensuring that each modeled time step depends only on past observations, thus faithfully preserving the chronological structure of financial data.*

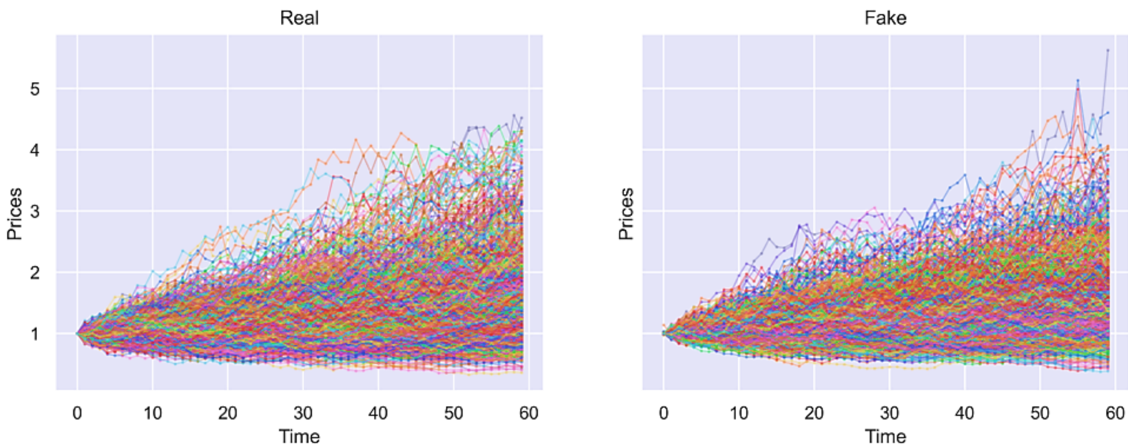

FIGURE 14: *Illustration of real paths from a siscretized Black-Scholes model (left) compared to fake paths generated from TC-VAE model (right) [1]. As shown, the TC-VAE is able to reproduce paths that closely resemble those observed in the training data, successfully capturing a comparable level of depth and variability to that found in real-world cases.*

representations. A notable example is offered by Ning et al. (2022) [36], who propose a Variational Autoencoder (VAE) framework specifically *tailored to generate arbitrage-free implied volatility surfaces*. In their design, the decoder network produces the parameters of the **Stochastic Volatility Inspired** (SVI) functional form — an established arbitrage-free parametrization that ensures the absence of both butterfly and calendar — spread arbitrage by construction.

Rather than learning raw option prices or implied volatilities, the VAE captures a low-dimensional latent representation that is mapped to valid volatility surfaces through this parameterization. This architectural constraint guarantees that every generated surface satisfies no-arbitrage conditions without additional regularization or post-processing. The resulting model combines the generative flexibility of VAEs with financial consistency, making it particularly effective for risk management, model calibration, and scenario generation. The workflow of this approach is illustrated in Figure 18.

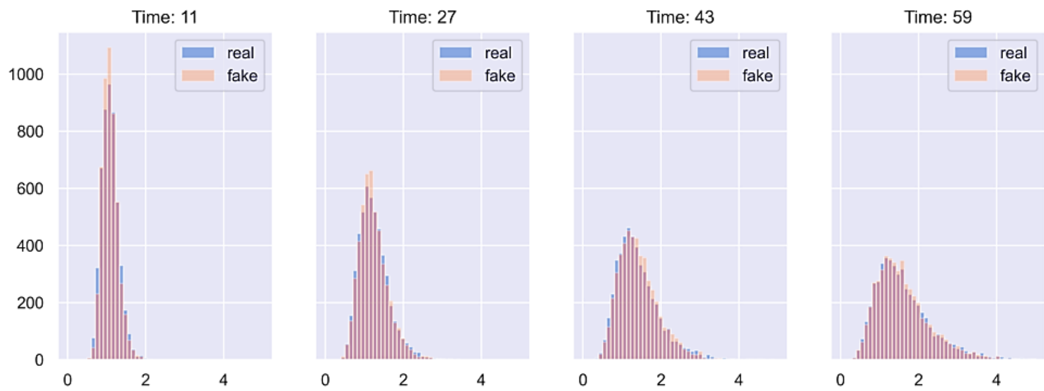

FIGURE 15: Visualization of marginal distributions at different time slices for real paths from a discretized Black-Scholes model (blue) compared to fake paths generated from the TC-VAE model (orange) [1]. As observed, the distributions are nearly overlapping across all the considered time slices, confirming the high reliability and consistency of the TC-VAE model.

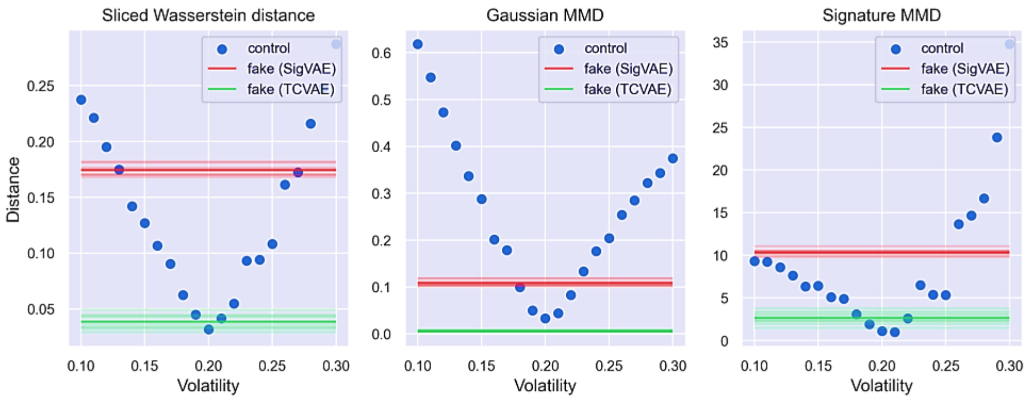

FIGURE 16: From left to right, we visualize the sliced Wasserstein distance, Gaussian MMD and signature MMD. The green (respectively red) lines illustrate distances between real paths of the Black-Scholes model and fake path generated from TC-VAE (respectively Sig-VAE); each line from a different random sees. The blue dots show the distances between real paths and control apths under different volatility levels [1].

Acciaio et al. (2024)		
Goal of the Paper	Results	About Challenges
To develop a TC-VAE for generating synthetic financial time-series data, enforcing a causality constraint on encoder/decoder networks and bounding the causal Wasserstein distance between real and generated distributions.	Generated paths closely match real ones (evaluated via sliced Wasserstein, Gaussian MMD, Signature MMD); preserves causal and temporal consistency.	• Although it handles time-series generation, it does not focus on a fully specified multidimensional risk-factor framework across distinct risk classes; • The treatment of intra-risk-factor correlation is not emphasized; • The work does not explicitly enforce arbitrage-free constraints.

TABLE 7: Acciaio et al. (2024).

3.7 Calibrating Option Price and Volatility Surface Via Physics-informed Neural Network by Hyeon-ok

An even more rigorous form of the free-arbitrage by-design approach involves the use of Physics-Informed Neural Networks (PINNs), which integrate financial partial differential equations (PDEs) directly into the training objective. Unlike purely data-driven models,

Schwarz (2024)		
Goal of the Paper	Results	About Challenges
To introduce a probabilistic LSTM-based generative AI model for synthetic financial market time-series data that balances interpretability and performance.	The experiments show that the proposed model: • replicates the probability distribution of real market data; • handles non-linear relationships, market regime changes and external variables; • outperforms traditional ARMA-GARCH models in non-linear settings.	• It does not explicitly tackle a full multidimensional risk-class framework; • It does not emphasize modelling intra-risk-factor correlation structures; • The paper does not claim enforcement of full arbitrage-free constraints.

TABLE 8: Schwarz (2024).

PINNs learn simultaneously from data and governing equations, ensuring that outputs adhere to the fundamental dynamics of pricing models. In the context of derivative valuation, the loss function can include the residual of the Black–Scholes PDE, compelling the network to approximate solutions that satisfy the equation across both spatial and temporal domains.
A representative application is presented by Hyeong-ok and Nam Sang-yoon (2023) [24], who employ a *PINN framework to jointly calibrate option prices and implied volatility surfaces*. By embedding the Black–Scholes dynamics directly into the loss function, their model learns arbitrage-free pricing relationships consistent with observed market data. This paradigm elegantly combines the data-driven adaptability of deep learning with the theoretical rigor of financial mathematics, and is increasingly used for high-dimensional pricing and calibration tasks. The network architecture employed for model training is illustrated in Figure 19.

3.8 Arbitrage-free Yield Curve and Bond Price Forecasting by Deep Neural Networks by Hyndman

Hyndman (2021) proposes a pre-fixing approach to *enforce arbitrage-free conditions within deep neural network frameworks for yield curve and bond price forecasting*. Rather than correcting arbitrage violations after generation, the model embeds projection-based regularization directly into the training process, ensuring that the outputs remain consistent with the theoretical constraints of fixed-income markets. By incorporating these financial structure conditions into the optimization objective, the network learns to produce yield curves that inherently satisfy no-arbitrage relationships across maturities. This method effectively integrates economic interpretability and machine learning flexibility, offering a robust mechanism for aligning data-driven forecasts with market-consistent dynamics. The resulting model demonstrates improved stability and financial coherence, making it a promising direction for applications in risk management, pricing, and stress testing [25].

3.9 A Generative Model for Arbitrage-Free Implied Volatility Surfaces by Vuletic

Vuletić (2023) introduces *VolGAN*, a generative adversarial network specifically *designed for the synthesis of arbitrage-free implied volatility surfaces*. The framework adopts a post-fixing strategy, in which the model first generates raw volatility surfaces through a data-driven

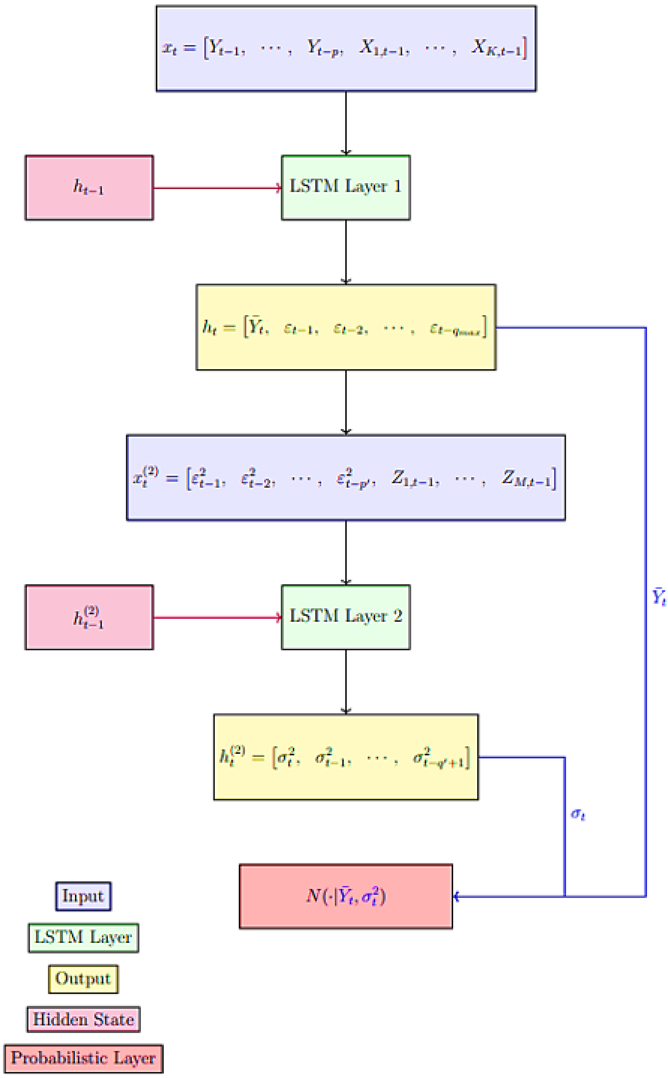

FIGURE 17: Schematic representation of the two LSTM layers feeding into a probabilistic output layer, here illustrated as a conditional Gaussian distribution, but any other distribution could be used [40]. Schwarz's Probabilistic LSTM framework is designed to generate synthetic time-series that capture rich statistical features, such as nonlinear dependencies, regime shifts, and volatility clustering, akin to traditional time series models, but embedded within a generative context.

adversarial process, and subsequently applies a set of projection-based correction procedures to enforce financial consistency. These post-processing steps are aimed at minimizing arbitrage violations by projecting the generated outputs back into the admissible, arbitrage-free manifold. The model effectively combines the expressive capacity of GANs with the structural rigor required in quantitative finance, producing volatility surfaces that remain realistic, dynamically coherent, and consistent with market constraints. Beyond arbitrage elimination, VolGAN demonstrates the potential of generative models to capture complex dependencies within the implied volatility space, offering a flexible tool for applications such as option pricing, risk management, and scenario generation [43].
To conclude this section, Table 12 provides a concise summary of all the methodologies discussed, highlighting their main characteristics, achieved results and comparative advantages in the context of financial scenario generation.

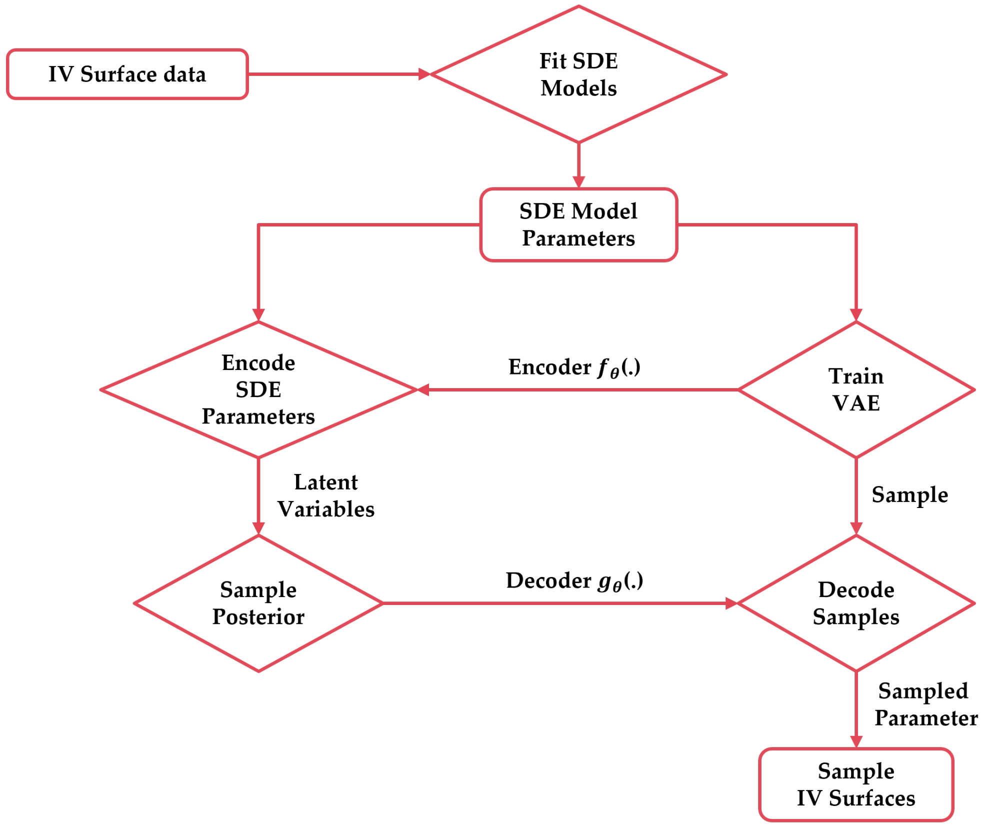

FIGURE 18: Flow chart of algorithm starting from raw surface data to generated surfaces through VAE [36].

Ning et al. (2022)		
Goal of the Paper	Results	About Challenges
To propose a hybrid method combining a variational autoencoder (VAE) with continuous-time SDE-driven models (regime-switching & Lévy additive processes) to generate arbitrage-free implied volatility (IV) surfaces consistent with historical data.	The method demonstrates superior generative out-of-sample performance, producing IV surfaces free of static arbitrage and faithful to historical distributions.	• The paper does not explicitly address multidimensional risk-class scenario generation; • It does not deeply model intra-risk-factor correlation structures; • Free-arbitrage conditions are a core theme.

TABLE 9: Ning et al. (2022).

4. Open Challenges

While *Generative AI* presents a powerful alternative, its application to financial scenario generation is far from straightforward. The transition from academic "proof-of-concept" to a robust, production-grade system introduces several formidable challenges that must be addressed.

First, the "black-box" nature of many generative architectures (GANs, VAEs, Transformers) can be a primary obstacle for their adoption in risk management. For regulatory compliance and internal validation, model interpretability is not optional. It is not trivial to validate why a model produced, for example, a specific tail event. Often this low, if not lack of, transparency can push for the choice of more interpretable alternatives like *Gaussian Mixture Models* (GMMs), which allow their factors (*Gaussian Principal Components*) to be analyzed,

much like traditional **PCA**.
Second, the calibration and training complexity is not to be underestimated. These models are not "plug-and-play." Training deep generative networks can be notoriously unstable (e.g., **mode collapse** in GANs) and computationally expensive. The process requires significant data engineering and hyperparameter tuning, rivaling (and sometimes exceeding) the complexity of calibrating sophisticated classical models. On the other hand, once a model is calibrated, the scenario generation can be extremely fast, hence exploitable also for (at least near-to) real-time simulations.
Third, the scope of existing research is often too narrow for a real-world financial institution. The literature frequently presents models calibrated on a single asset (e.g., S&P 500) or a single, isolated risk class (e.g., interest rate curves). A bank or insurer, however, must model the entire market — the complex, joint distribution of multiple, heterogeneous risk classes (equities, rates, FX, credit, and commodities) simultaneously. Scaling these models to such high dimensions while preserving delicate inter-class correlations is a massive, and largely unsolved, technical leap. To do so, novel literature is trying to blend classical approaches with AI, as in the "Copula Variational LSTM" propsed by Xu and CAo [44].
Fourth, the enforcement of financial consistency, most notably the absence of arbitrage, is not an inherent property of neural networks. Without specialized architectures (like *Autoencoder Market Models* (AEMM)) or complex, "physics-informed" loss functions, a model may produce statistically plausible but economically invalid scenarios, such as violations of put-call parity or arbitrageable yield curves.
Finally, one important and still largely overlooked dimension in the field of AI-driven scenario generation for finance is **conditional generation**, the ability to generate future paths or market states that are explicitly dependent on given initial conditions, macroeconomic assumptions, or market configurations. Despite the increasing sophistication of generative models, surprisingly little attention has been paid in the literature to this conditional setting, especially within the context of multi-asset, risk-sensitive financial forecasting. Yet, **conditional generation** has significant practical relevance: it allows risk managers and decision-makers to ask meaningful "what-if" questions, such as how markets might evolve given a credit spread widening, a spike in oil prices, or a central bank intervention. It enables the production of coherent and targeted scenarios that reflect the distributional implications of specific conditions, rather than relying on unconditional or average-case dynamics.
In practice, conditional generation provides a bridge between supervised inference and generative simulation. It gives institutions the ability to simulate plausible market evolutions under regulatory stress scenarios, or under internally defined stress assumptions, in a data-driven and probabilistically consistent manner. This ability is particularly useful in **forward-looking risk assessments** such as *ICAAP*, *CCAR*, or *Solvency II* projections, where the scenarios must not only reflect plausible tail behaviors but also remain consistent with certain macroeconomic narratives or initial market configurations. Conditional generation thus enhances scenario relevance and realism while retaining statistical diversity.
Different neural network architectures can be adapted to tackle this problem, although none of them has been explicitly designed for this task in standard financial literature. One natural starting point is the **Conditional VAE** (CVAE), where the conditioning variable is passed both to the encoder and the decoder, allowing the latent space to be learned in relation to a specified set of market conditions. This setup enables the generation of scenarios that are statistically consistent with observed distributions, yet anchored to specific assumptions or shocks. CVAEs are particularly well-suited when the goal is to map a structured condition — such as a macroeconomic vector or a set of factor values — to a

wide distribution of possible outcomes.
In the case of GAN, the extension to conditional generation is achieved through **Conditional GANs** (cGANs), where both the generator and the discriminator are fed with conditioning information. This setup forces the generator to learn how to produce samples that are not just realistic, but also coherent with the provided condition. While powerful in theory, training stability and mode collapse can become even more acute in the conditional case, especially when dealing with high-dimensional financial data with sparse conditional examples. Nonetheless, conditional GANs offer a valuable framework for generating scenario distributions that reflect specific stress drivers or forward-looking views.
Recurrent neural networks (RNNs) and particularly **Long Short-Term Memory networks** (LSTMs) can also be adapted for conditional generation by incorporating the conditioning input into the initial state or by concatenating the condition with each input timestep. For example, an LSTM-based model might take as input a time series of historical prices along with a constant conditioning vector (e.g., a shock to interest rates), and generate a sequence of future values that unfold under that assumption. This approach is particularly appealing in time-series applications where temporal coherence is critical. However, most LSTM-based models in finance are trained unconditionally, missing the opportunity to leverage structured scenario guidance.
Another relevant but underutilized approach is the mixture density network (MDN) or conditional **Gaussian Mixture Model** (GMM), where the model outputs the parameters of a distribution conditional on input features. Such models are naturally capable of representing multi-modal outcomes under specific initial states or shocks, and they allow a high degree of interpretability. However, their use in generative financial modeling, at the best of our knowledge, it remains still extremely studied.
The scarcity of literature on conditional generation in financial contexts is perhaps due to the complexity of obtaining consistent, labeled conditioning data, or the difficulty in defining proper training objectives under partial observability. Nonetheless, as the use of machine learning in risk management matures, the need for scenario generation tools that respond to precise assumptions or stress conditions will only grow. Conditional generation offers a principled and flexible solution to this need, making it a critical area for future development.

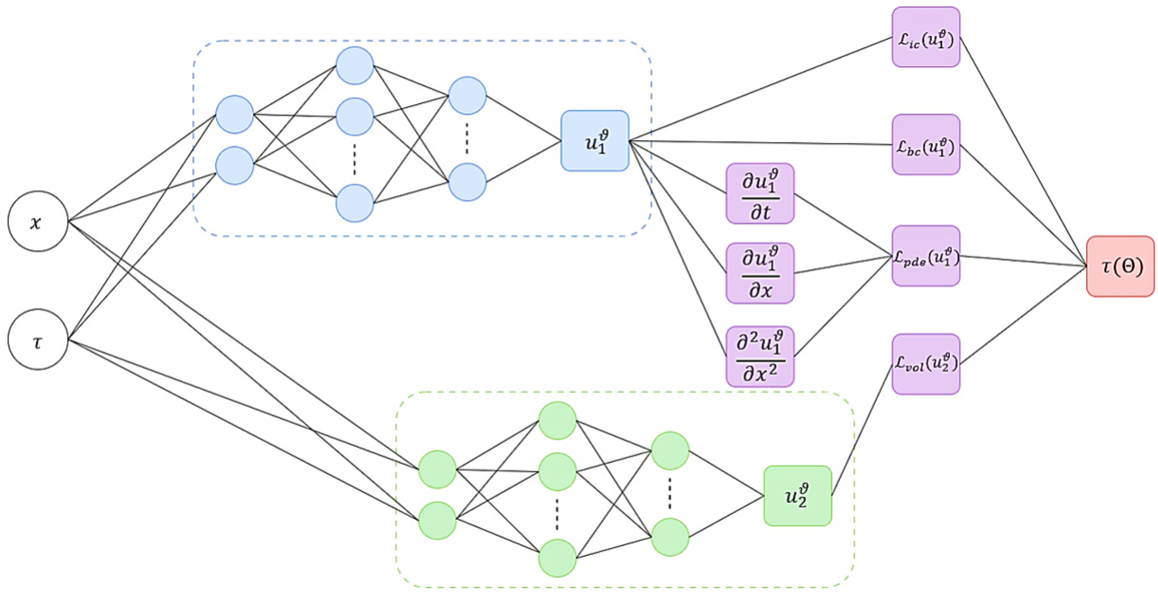

FIGURE 19: *Neural Network schema used by Hyeong-ok and Nam Sang-yoon [24], here x and τ are input data, each are moneyness scaled strike price and time to maturity. u_1^θ and u_2^θ are neural networks parameter. The input used for both the neural network is the same, the second neural network is used as a volatility parameter of the partial differential equation constructed through automatic differentiation for the first one.*

Hyeong-ok B. and Sang-yoon N. (2023)		
Goal of the Paper	Results	About Challenges
To develop a physics-informed neural network (PINN) framework that calibrates option prices and constructs a local volatility surface by embedding PDE constraints. No generative intent.	The model demonstrates successful estimation of option prices and constructs a local volatility surface consistent with market / model data.	• Does not address a full multidimensional risk-class scenario generation framework; • Does not focus in on intra-risk-factor correlation; • PDE constraints reduce arbitrage-type inconsistencies.

TABLE 10: *Hyeong-ok B. and Sang-yoon N. (2023).*

Hyndman (2021)		
Goal of the Paper	Results	About Challenges
To develop a deep-learning framework that forecasts yield curves and coupon-bond prices under the constraint of arbitrage-free regularization	The model shows that introducing an arbitrage-penalty improves forecasting performance.	• The paper does not address multidimensional risk-class scenario generation; • It does not deeply model intra-risk-factor correlation structures; • It introduces arbitrage-free regularization.

TABLE 11: *Hyndman (2021).*

Criteria / Model	GAN	VAE	GMM	RNN
Intra-class Heterogeneity	Has some difficulties on univariate/multivariate time series	Curve/structure natively supported	It can deal with spot, curves, surfaces and tensors	Limited, requires adaptations
Inter-class Dependence	Complex for homogeneous assets	Only with multi-latent architectures	Possible, but requires normalization	Complex for single-type time series
Arbitrage-Free Scenario	It can be applied externally	It can be applied externally	Arbitrage-free approach supported	It can be applied externally
Computational Cost	High	High	Low	High
Interpretability	Deep network, hard to read	Latent space interpretable, but not directly	GPC interpretable, similar to PCA	Black-box

TABLE 12: Comparison of the main models (GAN, VAE, GMM, and RNN) according to key analytical criteria.

Vuletic (2023)		
Goal of the Paper	Results	About Challenges
To introduce a GAN-based generative model (VolGAN) for simulating arbitrage-free implied volatility surfaces (IVS), trained on historical IVS and underlying price data.	Demonstrated that the model learns covariance structure of implied vol co-movements, generates realistic dynamics for the underlying index and VIX.	• The paper does not address multidimensional risk-class scenario generation; • It does not deeply model intra-risk-factor correlation structures; • It targets specifically the theme of no-arbitrage for IV surfaces.

TABLE 13: Vuletic (2023).

Model	Conditional Integration	Advantages	Challenges
CVAE	Conditioning variable passed to encoder and decoder.	Statistical coherence, suitable for structured vactors.	Complex latent space, limited interpretability.
CGAN	Condition provided to both generator and discriminator.	Realistic generation, scenario-consistent outputs.	Training instability, mode collapse.
C-LSTM	Condition integrated in initial state or concatenated at each timestep.	Temporal coherence, useful for time-series data.	Rarely applied in conditional setting.
Conditional GMM	Model output distribution parameters conditional on features.	Multi-modality, high interpretability.	Limited adoption in finance, training complexity.

TABLE 14: Comparison of main conditional generative models for financial scenario generation, highlighting their integration mechanisms, strengths, and challenges.

5. Our Contribution

As presented in the previous chapters, there is no single, universally accepted technique for generating synthetic financial data. The landscape is a complex trade-off between the interpretability and proven effectiveness for most cases of classical models and the high-fidelity, data-driven power promised by AI, especially GenAI. Each approach comes with its own assumptions, strengths, and limitations. For this reason, this paper's primary contribution is not to propose yet another novel architecture, but to provide a methodological blueprint for the selection, hybridization, and calibration of these tools, specifically tailored for a production setting in a financial institution.

The successful transition from theory to a production-ready system is a complex task requiring deep domain expertise. Our contribution is to provide the specialized methodological support to design, implement, test, and govern bespoke solutions at each stage, ensuring a blend of AI innovation and financial-grade robustness.

5.1 Scenario Classification and Data Framework Design

- **The Core Task**: this initial stage involves transforming vast, heterogeneous market data into a curated, model-ready "Scenario Catalog".
- **Our Value**: this is not simple "tagging". We partner with clients to architect the data framework. Our expertise helps:
  - *Define the Problem Space*: we identify the relevant risk factors, historical regimes (e.g., "inflation-shock", "flight-to-quality"), and "semantic contexts" required for a specific institutional need (e.g., Market Risk VaR, CCR, etc.).
  - *Data Augmentation Strategy*: we design and implement the pre-processing pipeline, addressing missing data, different time scales, and heterogeneous data structures (e.g., curves, surfaces).

5.2 Model Selection, Hybrid Design, and Calibration

- **The Core Task**: selecting, building, and training a generative model that is both statistically powerful and financially consistent.
- **Our Value**: this is the primary technical hurdle where generic approaches fail. We design and fine-tune bespoke hybrid models by navigating the critical trade-offs:
  - *Model Selection*: we can guide the choice between different models and architectures, navigating the complexities and trade-offs among high-fidelity, interpretable latent-space, and data-efficient, transparent factor models.
  - *Hybridization*: we can architect solutions that blend classical approaches with AI, such as using a VAE to learn a non-linear representation of a yield curve, and then applying a classical (and auditable) stochastic process to its latent factors.
  - *Constraint Enforcement*: our core expertise in financial modeling can help in injecting financial domain knowledge directly into the model — e.g., arbitrage-free conditions and causality constraints to ensure the model's outputs are not just plausible, but economically valid and robust for dynamic problems).

5.3 Validation, Governance, and Integration

- **The Core Task**: ensuring the calibrated model is trustworthy, maintainable, and usable within the bank's existing infrastructure.
- **Our Value**: a model is only production-ready once it is governed. We provide a complete validation and governance framework:
  - *Rigorous Testing*: we can move beyond standard loss metrics to perform comprehensive statistical validation of the synthetic data, ensuring it correctly reproduces all critical stylized facts (heavy tails, volatility clustering) and, most importantly, preserves complex cross-asset tail dependencies.
  - *Governance & MLOps*: we provide the clear documentation required for internal model validation teams and regulators. We design the MLOps pipeline for efficient retraining, monitoring, and versioning of the generative engine.
  - *Integration*: we support the final integration of the generative model's output into the institution's existing risk engines, ensuring the solution is scalable, auditable, and truly operational.

6. Conclusion and Future Works

This work has reviewed the evolution of financial scenario generation, from the foundational classical techniques to the Generative AI models entering the scene. Traditional approaches, such as historical simulation and Monte Carlo methods, remain widely used and have been consistently improved in the decades to overcome constraints on possibly reductive assumptions. To tackle differently critical market stylized facts like non-linear dependencies, heavy tails, and volatility clustering, and blend-in unstructured data, the academia and the industry started to look for more flexible, data-driven frameworks.
Our analysis detailed leading GenAI alternatives — GANs, VAEs, and GMMs with their variants — along with LSTMs, and surveyed key literature presenting their potential. These models promise to offer a powerful new toolkit for data augmentation and for generating high-fidelity scenarios by learning complex market structures directly from data. The primary advantage of these AI-driven approaches lies in their *flexibility* and *potential for customization*. For instance, an Autoencoder Market Model (AEMM) can provide a parsimonious, data-driven representation of yield curves, while a GMM offers an interpretable factor model via Gaussian Principal Components (GPCs).
However, our review concludes that these advanced models, even if worthy to be studied deeper and tested in use cases with financial institutions, are not yet "production-ready" off-the-shelf. Critical and practical challenges remain. The "black-box" nature of many deep learning models poses a significant hurdle for internal validation and regulatory approval; we acknowledge indeed that these approaches can be used mainly, if not only, for managerial purposes. Furthermore, the calibration and training complexity of these models is substantial, often rivaling classical Monte Carlo in its demand for expert tuning and computational power.
Most importantly, two key gaps seems to persist:

1. **Scalability**: the majority of existing research remains narrowly focused on single assets or risk classes. A production-ready system must be capable of modeling the joint distribution of all heterogeneous, multi-asset risk classes in a bank's portfolio.

2. **Financial Consistency**: ensuring that generated scenarios are, for example, arbitrage-free is a non-trivial requirement that demands specialized architectures or complex loss functions.

Looking ahead, we see the future not in a pure AI-only solution, but in *hybrid approaches* that combine the statistical rigor of classical models with the adaptive power of GenAI. In this context, Large Language Models (LLMs) are poised to play a crucial, dual role. First, we see their primary strength not in generating the quantitative scenarios themselves, but in managing unstructured data, processing news, sentiment, and geopolitical reports to create the rich, qualitative inputs for conditional scenario generation. Second, LLMs can serve as a powerful natural-language interface, providing a "chat-like" entry point that allows risk managers and business people to translate financial intuition into robust, calibrated simulations. Ultimately, the path to adoption relies on a practical framework for selection, hybridization, and governance.
By carefully integrating these new technologies, financial institutions can leverage the power of GenAI to build scenario generation tools that are not only data-driven and realistic but also transparent, operationally feasible, and trusted for real-world risk management and business applications.

References

[1] **Acciaio B., Eckstein S. and Hou S.** *Time-Causal VAE: Robust Financial Time Series Generator.* November 2024.

[2] **Atkins A., Niranjan M. and Gerding E.** *Financial news predicts stock market volatility better than close price*, The Journal of Finance and Data Science, Vol. 4, Issue 2, pp. 120-137, June 2018.

[3] **Bagattini G. et al.** *Leverage Large Language Models in Finance: Pathways to Responsible Adoption*. ESMA ILB, The Alan Turing Institute, 2025.

[4] **Baillie R. T., Bollorslev T. and Mikkelsen H. O.** *Fractionally integrated generalized autoregressive conditional heteroskedasticity*. Journal of Econometrics, Vol. 74, pp. 3-30, September 1996.

[5] **Barone-Adesi G. and Giannopoulos K.** *Non parametric VaR Techniques. Myths and Realities*. July 2001.

[6] **Bollerslev T.** *Generalized autoregressive conditional heteroskedasticity*. Journal of Econometrics, Vol. 31, pp. 307-327, April 1986.

[7] **Bonollo M., Damato V. and Luce F.** *FHS-VaR and Financial Risks*, MAF, 2025.

[8] **Brigo D. and Mercurio F.** *Interest Rate Models — Theory and Practice*. 2006.

[9] **Cheng Z., Sjarif N.N.A. and Roslina I.** *Deep learning models for price forecasting of financial time series: A review of recent advancements: 2020–2022*. September 2023.

[10] **Chevyrev I. and Oberhauser H.** *Signature moments to characterize laws of stochastic processes*. Journal of Machine Learning Research, Vol. 23, pp. 1-42, October 2018.

[11] **Demarta S. and McNeil A.** *The t copula and related copulas*. May 2004.

[12] **Divate M. et al.** *Harnessing LLMs for Financial Forecasting: A Systematic Review of Advances in Stock Market*. International Journal for Research in Applied Science and Engineering Technology, Vol. 12, pp. 1101-1105, November 2024.

[13] **Dogariu M., Stefan L. and Boteanu B. A.** *Generation of Realistic Synthetic Financial Time-series*. Association for Computing Machinery, March 2022.

[14] **Embrechts P., Kluppelberg C. and Mikosch T.** *Modelling Extremal Events for Insurance and Finance*. February 1997.

[15] **Embrechts P., McNeil A. and Straumann D.** *Correlation and Dependence in Risk Management: Properties and Pitfalls. In Risk Management: Value at Risk and Beyond*. 2002.

[16] **Engle R. F.** *Dynamic Conditional Correlation: A Simple Class of Multivariate GARCH Models*. Journal of Business and Economic Statistics, Vol. 20, pp. 339-350, January 2012.

[17] **Flaig S. and Junike G.** *Scenario generation for market risk models using generative neural networks*. Risks, August 2023.

[18] **Fritzsh S. and Timphus M.** *Marginals versus copulas: Which account for more model risk in multivariate risk forecasting?*. Journal of Banking & Finance, Vol. 158, January 2024.

[19] **Goodfellow I. J. et al.** *Generative Adversarial Networks*. June 2014.

[20] **Gretton A. et al.** *A kernel two-sample test*. The Journal of Machine Learning Research, Vol. 13, pp. 723-773, December 2012.

[21] **Harrison J. and Pliska S.** *Martingales and stochastic integrals in the theory of continuous trading*. Stochastic Processes and their Applications, Vol. 11, pp. 215-260, August 1981.

[22] **Haugh M.** *An introduction to copulas*. 2016.

[23] **Hochreiter S.** *Long Short-Term Memory*. Neural Computation, Vol. 9, pp. 1735-1780, 1997.

[24] **Hyeong-ok B. and Sang-yoon N.** *Calibrating option price and volatility surface via physics-informed neural network*. The Graduate School, Ajou University, February 2023.

[25] **Hyndman C.** *Arbitrage-free yield curve and bond price forecasting by deep neural network*. September 2021.

[26] **Jolliffe I. T.** *Principal Component Analysis*. 2002.

[27] **Kingma D. P. and Welling M.** *An Introduction to Variational Autoencoders*. Foundations and Trends in Machine Learning, Vol. 12, pp. 307-392, December 2019.

[28] **Liu Y. and Yun L.** *The Development of Large Language Models in the Financial Field*. Proceedings of Business and Economic Studies Vol. 8, pp. 49-54, April 2025.

[29] **Longin F.** *From Value at Risk to Stress Testing: The Extreme Value Approach*. Journal of Banking & Finance, Vol. 24, pp. 1097-1130, July 2000.

[30] **Lu C. and Sester J.** *Generative model for financial time series trained with MMD using a signature kernel*. July 2024.

[31] **Mandelbrot B.** *The variation of certain speculative prices*. The Journal of Business, Vol. 36, pp. 394-419, December 2009.

[32] **McLachlan G. J.** *Finite Mixture Models*. June 2019.

[33] **McNeil A. and Frey R.** *Estimation of tail-related risk measures for heteroscedastic financial time series: an EVT approach.* June 2000.

[34] **Mirza M. and Osindero S.** *Conditional Generative Adversarial Nets*. November 2014.

[35] **Mohammadabadi S. et al.** *A Survey of Large Language Models: Evolution, Architectures, Adaptation, Benchmarking, Applications, Challenges, and Societal Implications*. Electronics, Vol. 14, September 2025.

[36] **Niang B., Jaimunga S. and Bergeron M.** *Arbitrage-Free Implied Volatility Surface Generation with Variational Autoencoders*. January 2022.

[37] **Reynolds D.** *Gaussian Mixture Models*. Encyclopedia of Biometrics, pp. 659–663, 2009.

[38] **Rizzato M. et al.** *Generative Adversarial Networks applied to synthetic financial scenarios generation*. Physica A: Statistical Mechanics and its Applications, May 2024.

[39] **Schmitt T. A. et al.** *Non-stationarity in financial time series: Generic features and tail behavior*. May 2013.

[40] **Schwarz C.** *Interpretable GenAI: Synthetic Financial Time Series Generation with Probabilistic LSTM*. June 2024.

[41] **Tunnicliffe G.** *Time Series Analysis: Forecasting and Control*. Journal of Time Series Analysis, Vol. 37, March 2016.

[42] **Vaswani, A. et al.** *Attention Is All You Need*. August 2023.

[43] **Vuletić M. and Cont R.** *VolGAN: A Generative Model for Arbitrage-Free Implied Volatility Surfaces*. Applied Mathematical Finance, Vol. 31, pp. 203–238, December 2023.

[44] **Xu J. and Cao L.** *Copula Variational LSTM for High-dimensional Cross-market Multivariate Dependence Modeling*. May 2023.

Sitography

[45] **Arbitrage.** *Website*.

[46] **EIOPA. YE2019 Comparative Study on Market & Credit Risk Modelling, 2021a.** *Website*.

A. Annex

This chapter presents a structured overview of these methods, highlighting both classical techniques and more recent data-driven frameworks. From traditional parametric models to deep generative networks, the landscape of scenario generation has grown increasingly rich and diverse — reflecting the complex, nonlinear, and high-dimensional nature of financial markets. Understanding the underlying principles, assumptions, and trade-offs of each approach is essential for selecting the most appropriate tool for a given risk management or forecasting application.

A.1 Generative Adversarial Network

Generative Adversarial Networks (GANs), introduced by Ian Goodfellow et al. in 2014 [19], represent a groundbreaking framework for training generative models. GANs leverage game theory to model a system in which two neural networks — a generator and a discriminator — are trained simultaneously in a minimax game. This approach enables the generator to learn the underlying data distribution and produce highly realistic synthetic data, without requiring explicit probability models or likelihood functions.
At the core of GANs is the interaction between two models:

- Generator (G): Takes as input a random noise vector $z \sim p_z(z)$ (typically sampled from a uniform or Gaussian distribution) and transforms it into a sample $G(z)$ that mimics the data distribution;
- Discriminator (D): Receives either a real data sample $x \sim p_{data}(x)$ or a generetad sample $G(z)$ and outputs the probability that the input is real (i.e., from the true data distribution).

The generator and discriminator are trained in opposition: the generator aims to fool the discriminator, while the discriminator aims to correctly classify real versus generated data, in Figure 20 can be seen an example structure of a GAN. This dynamic is formalized as a two-player minimax game:

$$\min_G \max_D V(D,G) = \mathbf{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbf{E}_{z\sim p_z(z)}[\log(1-D(G(z)))].$$

Here, $D(x)$ is the probability assigned by the discriminator that x is a real sample, and $G(z)$ is the synthetic sample produced by the generator from noise z. Goodfellow et al. demonstrated that under optimal conditions, the generator recovers the true data distribution p_{data}. Specifically, for a fixed generator G, the optimal discriminato r $D*(x)$ is:

$$D*(x) = \frac{p_{data}(x)}{p_{data}}, \tag{1}$$

where p_g is the model distribution induced by the generator G. Plugging $D*$ back into the value function yields:

$$\begin{aligned} C(G) &= \max_D V(D,G) = \mathbb{E}_{x\sim p_{\text{data}}}\big[\log D^*(x)\big] + \mathbb{E}_{z\sim p_z(z)}\big[\log(1-D^*(G(z)))\big] \\ &= -\log 4 + 2\cdot \text{JSD}(p_{\text{data}} \parallel p_g). \end{aligned} \tag{2}$$

This shows that minimizing the GAN objective is equivalent to minimizing the Jensen–Shannon divergence (JSD) between the real and generated data distributions — a symmetric, bounded divergence measure.

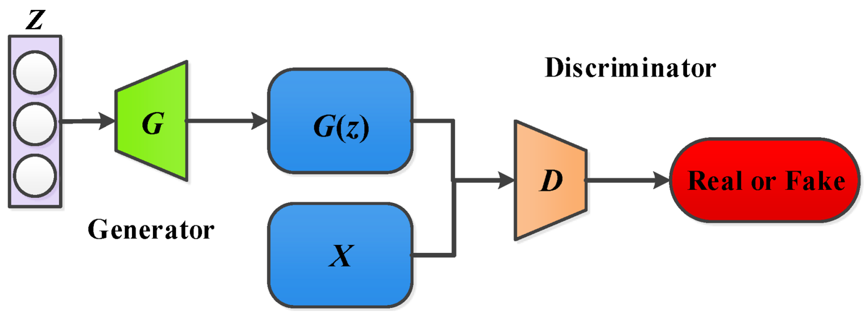

FIGURE 20: In the figure above there is an example of the architecture of a GAN. At the beginning a noised input is fed to the generator to produce fake sample $G(z)$. Both the real data X and the generated sample are then given to the discriminator D, which learns to distinguish real from fake.

A.2 GAN Extension: Conditional GAN

CGANs extend traditional GANs by introducing a conditioning vector y that influences both the generator and the discriminator. This allows data x to be generated in a conditional way, leading to more targeted and controllable outputs. The objective function becomes:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{\text{data}}(x)}\left[\log D(x,y)\right] + \mathbb{E}_{z\sim p_z(z), y\sim p_{\text{data}}}\left[\log(1 - D(G(z,y),y))\right]. \quad (3)$$

- The generator G receives both noise z and condition y as input, typically concatenated before the upsampling stages;
- The discriminator D evaluates pairs (x,y) versus $(G(z,y),y)$.

A.3 Gan Extension: Time GAN

TimeGAN (Time-series Generative Adversarial Network) is a powerful model designed to generate realistic synthetic time series data while preserving both temporal dynamics and realistic feature distributions. Traditional GANs struggle with time series because they do not model temporal dependencies explicitly. TimeGAN addresses this by combining supervised and unsupervised learning in a unified framework that captures both: step-wise temporal dynamics and latent representations of multivariate data.
TimeGAN blends components of a GAN, an autoencoder and a recurrent network (usually LSTM). Its architecture consists of:

- Embedding Network: learns a latent representation h of the original data x, typically via a RNN encoder;
- Recovery network: reconstruncts the input from embedding $\hat{x} = R(h)$;
- Generator G: generates latent sequences from random noise and conditions;
- Discriminator D: tries to distinguish real vs synthetic latent sequances;
- Supervisor S: predict the next latent state from the current one, helping model temporal transitions.

And the training includes three losses, Reconstruction, Supervised and Adversaral Loss as follow:

$$\begin{aligned}\mathcal{L}_{tot} = \mathcal{L}_{rec} + \mathcal{L}_{sup} + \mathcal{L}_{adv} = \mathbf{E}[||x - \hat{x}||^2] + \mathbf{E}[||h_{t+1} - \hat{h}_{t+1}||^2] + \min_G \max_D \mathbf{E}[\log D(h)] \\ + \mathbf{E}[\log(1 - D(\hat{h}))].\end{aligned} \quad (4)$$

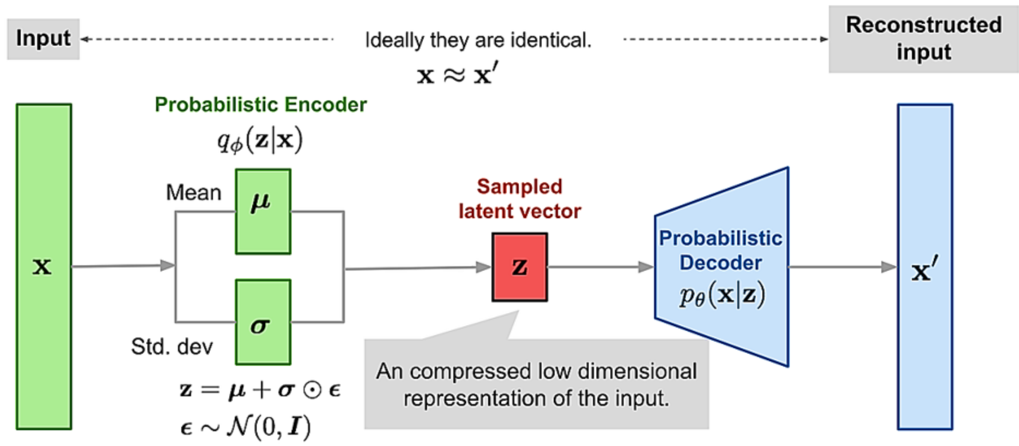

FIGURE 21: Example of a Variational Autoencoder (VAE). The input x is encoded into a mean μ and a standard deviation σ, from which a latent vector z is sampled as a compressed low-dimensional representation of x. A probabilistic decoder then reconstructs the data, generating x'.

A.4 Variational Autoencoder

A Variational Autoencoder (VAE) is a type of generative model in machine learning that learns a compressed representation of data while also being able to generate new, synthetic data samples. A VAE consists of two neural networks:

- Encoder/Inference model $q_\phi(z|x)$: encodes input x to a latent distribution;
- Decoder/Generative model $p_\theta(x|z)$: reconstructs x given latent z.

This setup implements amortized inference by sharing encoder parameters across data points — far more efficient than traditional per-sample optimization. In Figure 21 is possible to see an example of the architecture of a VAE.
Training maximizes the ELBO, which serves as a tractable lower bound on the data log-likelihood $\ln p_\theta(x)$:

$$\mathcal{L}(\theta, \phi; x) = \mathbf{E}_{z \sim q_{phi}(z|x)}[\ln p_\theta(x|z)] - D_{KL}(q_\phi(z|x)||p(z)). \tag{5}$$

- Reconstruction term encourages $p_\theta(x|z)$ to match true x;
- KL divergence regularizes latent z to stay close to the prior $p(z)$, typically $\mathcal{N}(0, I)$.

Maximizing ELBO achieves a trade-off between accurate reconstructions and latent space regularization.
To backpropagate through stochastic sampling, the reparameterization trick is used:

$$z = \mu_\phi(x) + \sigma_\phi(x) \odot \epsilon, \epsilon \sim \mathcal{N}(0, I). \tag{6}$$

This transforms z-sampling into a differentiable operation, allowing efficient gradient estimation with encoder parameters.

A.5 VAE Extension: Conditionale VAE

Conditional Variational Autoencoders (CVAEs) are powerful models for learning conditional generative processes. They extend standard VAEs by introducing a conditioning variable

y, allowing generation of data x conditioned on labels, attributes, or context. While the idea is conceptually simple, practical challenges arise in training CVAEs effectively. In this chapter, we revisit CVAEs through the lens of Christopher Beckham's analysis, offering a theoretically sound and empirically motivated perspective on their structure, training, and trade-offs.
At the core of CVAEs lies the Evidence Lower Bound (ELBO), which becomes:

$$\log p_\theta(x|y) \geq \mathbf{E}_{q_\phi(z|x,y)}[\log p_\theta(x|z,y)] - D_{KL}(q_\theta(z|x,y)||p(z|y)). \tag{7}$$

A key modeling decision in CVAEs is whether to use a conditional prior $p(z|y)$ or a fixed, indipendent prior $p(z)$. The dependent prior allows the latent space to shift in a way that reflects the semantics of y, potentially enriching expressiveness.

A.6 VAE Extension: Time VAE

The Time-Variational Autoencoder (TimeVAE) framework combines probabilistic latent variable modeling with temporal encoding mechanisms, enabling expressive and interpretable time-series generation. TimeVAE can be viewed as an extension of the Variational Autoencoder (VAE) that integrates temporal models like RNNs, GRUs, or transformers into the encoder–decoder architecture. This allows the model to learn not only latent structure but also temporal dynamics of sequences.
The model is trained to maximize the ELBO:

$$\mathcal{L}_{ELBO} = \sum_{t=1}^{T} \mathbf{E}_{q_\phi(z_t|x_{\leq t}, z_{<t})}[\log p_\theta(x_t|z_t)] - D_{KL}(q_\phi(z_t|x_{\leq t}, z_{\leq t})||p_\theta(z_t|z_{<t}). \tag{8}$$

A.7 Recurrent Neural Network

In many real-world applications such as natural language processing, time-series forecasting, and speech recognition, data is inherently sequential. Traditional feedforward neural networks are inadequate for modeling such sequences, as they assume all inputs are independent of each other. Recurrent Neural Networks (RNNs) address this limitation by introducing cycles within the network architecture, enabling information to persist across time steps.
An RNN processes input sequences one element at a time, maintaining a hidden state that captures information from previous inputs. Given a sequence $x = (x_1, x_2, ..., x_T)$, the RNN updates its hidden state h_t and outputs y_t at each time step using the following equations:

$$h_t = \phi(W_{hh}h_{t-1} + W_{xh}x_t + b_h); \tag{9}$$

$$y_t = W_{hy}h_t + b_y. \tag{10}$$

Here, ϕ is typically a non-linear activation function such as tanh or ReLU. The key idea is that the hidden state h_t carries temporal information across the sequence.
To address the shortcomings of vanilla RNNs, Hochreiter and Schmidhuber (1997) proposed the Long Short-Term Memory network. LSTMs introduce a memory cell and gating mechanisms that regulate the flow of information, enabling the network to selectively retain or discard information over long periods.
An LSTM unit consists of the following gates:

- Forget gate f_t: Decides what information to discard from the cell state;

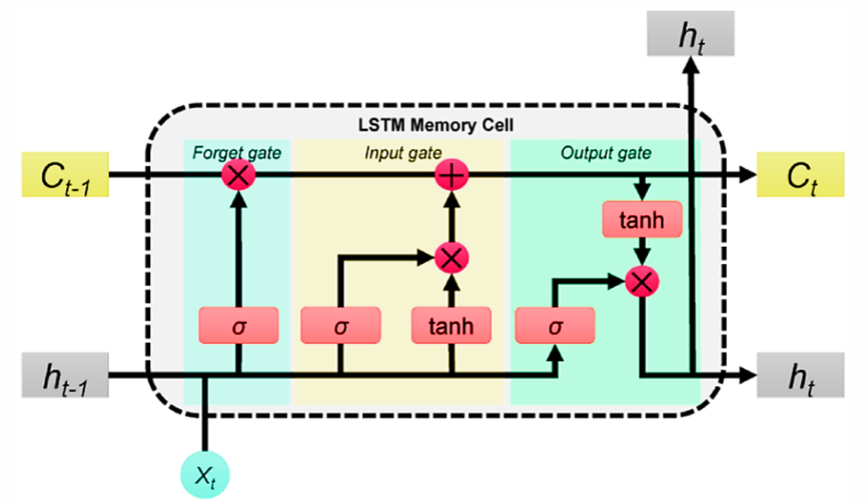

FIGURE 22: Illustration of a Long Short-Term Memory (LSTM) unit. It contains three main gates: the forget gate, the input gate, and the output gate, which regulate the flow of information and help capture long-term dependencies in sequences.

- Input gate i_t: Decides what new information to store;
- Cell candidate $\tilde{C}_t$: Potential values added to the cell state;
- Output gate o_t: Determines the output and hidden state.

In Figure 22 is showed an example architecture of a LSTM.
The computations are as follows:

$$f_t = \sigma(W_f[h_{t+1}, x_t] + b_f; \tag{11}$$

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i); \tag{12}$$

$$\tilde{C}_t = \tanh(W_C[h_{t-1}, x_t] + b_C); \tag{13}$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t; \tag{14}$$

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o; \tag{15}$$

$$h_t = o_t \odot \tanh C_t, \tag{16}$$

where σ is the sigmoid activation function, $\odot$ denotes element-wise multiplication, C_t is the cell state at time t and h_t is the hidden state output at time t.
LSTM networks are capable of learning dependencies over long time horizons by preserving gradients and maintaining stable learning dynamics. They have become the de facto standard in sequence modeling tasks prior to the rise of Transformer-based architectures.

A.8 Gaussian Mixture Models

Gaussian Mixture Models (GMMs) provide a flexible, unsupervised method to model such heterogeneity by assuming that data are generated from a mixture of multiple Gaussian distributions. Unlike k-means clustering, GMMs offer a soft clustering approach, assigning probabilities for a data point to belong to each cluster.
A Gaussian Mixture Model defines the probability density function (pdf) of a data point $\mathbf{x} \in \mathbf{R}^D$ as a convex combination of K Gaussian components:

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\mu_k, \mathbf{\Sigma_k}), \tag{17}$$

where:

- $\pi_k \in [0, 1]$ are the mixing coefficients, such that $\sum_{k=1}^{K} \pi_k = 1$;
- $\mu_k \in \mathbf{R}^D$: mean of the $\mathbf{k}^{th}$ Gaussian;
- $\Sigma_k \in \mathbf{R}^{DxD}$: covariance matrix of the $\mathbf{k}^{th}$ Gaussian;
- $\mathcal{N}$;($\mathbf{x}$ | μ_k,$\mathbf{\Sigma_k}$): multivariate Gaussian PDF.

GMM can be viewed ad a latent variable model where each data point $\mathbf{x}_i$ is associated with a latent cluster label $z_i \in \{1, ..., K\}$. The generative process firstly sample a cluster index $z_i \sim Categorical(\pi_1, ..., \pi_K)$ and the sample the data point $\mathbf{x}_i \sim \mathcal{N}(^-_{z_i}, \mathbf{\Sigma}_{z_i})$.

Iason is an international firm that consults Financial Institutions on Risk Management. Iason is a leader in quantitative analysis and advanced risk methodology, offering a unique mix of know-how and expertise on the pricing of complex financial products and the management of financial, credit and liquidity risks. In addition Iason provides a suite of essential solutions to meet the fundamental needs of Financial Institutions.