Machine Learning or Modelling?

From fitting historical data to constructing future scenarios

Nikolaos Halidias

Actuarial-Financial Mathematics Lab

Department of Statistics and Actuarial - Financial Mathematics

June 22, 2026

The central confusion

Two different questions

- **Fitting question:** Which function or stochastic mechanism explains past observations?
- **Decision question:** Which future scenarios should we test our decision against?

Main claim

A good fit to historical data is not automatically a good scenario generator for the future.

Why this matters in finance and actuarial science

- Many tasks are not pure prediction tasks.
- They are **decision problems under uncertainty**:
  - portfolio construction,
  - hedging,
  - capital allocation,
  - reserving,
  - reinsurance selection,
  - stress testing.
- The final output is not only a forecast; it is an **action**.

Key point

The relevant question is not only "Does the model fit?" but also "Does it produce decision-relevant scenarios?"

A motivating example: a stock price

Suppose that we observe a stock price process

$$S_{t_0}, S_{t_1}, \ldots, S_{t_n}.$$

We may want to answer very different questions:

1. Can we fit the observed historical dynamics?
2. Can we generate plausible future paths for risk management?
3. Can we choose a portfolio or hedge that performs well under uncertainty?

Important distinction

The same data can support many different future narratives.

A stochastic differential equation with jumps

A flexible model may be written informally as

$$\frac{dS_t}{S_{t-}} = a(t, S_{t-})\, dt + b(t, S_{t-})\, dW_t + \int_E g(t, S_{t-}, z)\, \widetilde{N}(dt, dz).$$

- a controls the drift.
- b controls the local volatility.
- g controls jump sizes and jump responses.
- $\widetilde{N}$ is a compensated jump measure.

The modelling freedom

Different choices of a, b, g generate different future path properties.

Machine learning view: fit the coefficients

One approach is to approximate the unknown functions a, b, g by flexible parametrisations. For example, using sigmoidal functions,

$$a_\theta(t,s) = \alpha_0 + \sum_{j=1}^{m} \alpha_j \sigma(u_j t + v_j s + c_j), \qquad \sigma(x) = \frac{1}{1+e^{-x}}.$$

Similarly,

$$b_\theta(t,s) = \phi\left(\beta_0 + \sum_{j=1}^{m} \beta_j \sigma(r_j t + q_j s + d_j)\right),$$

where ϕ may enforce positivity, for example $\phi(x) = e^x$ or a softplus link.

ML task

Find parameters θ that make the model fit the historical sample.

The fitting problem

The fitted model is obtained through an empirical criterion such as

$$\widehat{\theta} = \arg\min_{\theta \in \Theta} \mathcal{L}\big(\theta; S_{t_0}, \ldots, S_{t_n}\big).$$

Typical choices of $\mathcal{L}$ include:

- prediction error for increments or returns,
- negative log-likelihood,
- moment matching,
- calibration error to liquid market quotes,
- hybrid statistical and financial losses.

What this solves

This solves an estimation problem on observed data. It does not, by itself, solve a future decision problem.

If we simulate the fitted SDE, what are we assuming?

After fitting, one may simulate future paths from

$$dS_t = S_{t-}\, a_{\widehat{\theta}}(t, S_{t-})dt + S_{t-}\, b_{\widehat{\theta}}(t, S_{t-})dW_t + S_{t-} \int_E g_{\widehat{\theta}}(t, S_{t-}, z)\widetilde{N}(dt, dz).$$

This is legitimate only if we accept additional assumptions:

- the historical regime is informative for the future;
- the estimated coefficients remain stable;
- jump intensity and tail behaviour are adequately learned;
- the decision-relevant losses are captured by the fitted criterion.

When the ML route is appropriate

The fitted-model route is useful when:

- the environment is sufficiently stable;
- the sample is rich enough to identify the relevant dynamics;
- the loss function used for fitting is close to the loss function used for decisions;
- extrapolation outside the observed region is limited;
- the objective is short-horizon forecasting or statistical replication.

Summary

Machine learning is excellent for approximation. The problem starts when approximation is confused with scenario design.

Why fitted dynamics may be insufficient

A fitted historical model can fail for decision purposes because:

- many different dynamics can fit the same historical sample;
- tail events are rarely identified by ordinary fitting criteria;
- regime shifts change the meaning of historical estimates;
- path-dependent decisions care about trajectories, not only one-step errors;
- optimization can exploit small modelling errors and produce fragile actions.

Decision risk

A model with low historical error can still produce the wrong hedge, wrong capital number, or wrong portfolio.

Modelling view: construct future scenarios

In the modelling view, a model is a **scenario generator**:

$$M_k \longrightarrow \left\{ S_{0:T}^{(k,1)}, \ldots, S_{0:T}^{(k,N_k)} \right\}.$$

A full scenario universe may be

$$\mathcal{S} = \bigcup_{k=1}^{K} \left\{ S_{0:T}^{(k,i)} : i = 1, \ldots, N_k \right\}.$$

Interpretation

Each model M_k represents a structured view of the future: normal diffusion, jump risk, volatility clustering, crisis regime, liquidity stress, or other behaviour.

Examples of scenario-generating models

For a stock price, the scenario library may include:

- geometric Brownian motion as a baseline;
- local-volatility or stochastic-volatility models;
- Merton-type jump-diffusion models;
- regime-switching models;
- stress models with large negative jumps;
- historical block-bootstrap paths;
- market-implied scenarios from options;
- expert-designed non-historical paths.

Point

The model set is chosen for coverage of relevant behaviours, not only for best historical fit.

Desired properties of scenarios

A scenario universe should be designed to have properties relevant to the decision:

Financial properties

- no-arbitrage consistency,
- positivity of prices,
- realistic volatility and jumps,
- dependence and correlation structure,
- liquidity and transaction-cost stress.

Risk properties

- tail behaviour,
- path dependence,
- drawdowns,
- stress consistency,
- interpretability and auditability.

Two pipelines

Fitting pipeline

Historical data → ML fitting → Fitted dynamics → Simulated future → Decision

Scenario-design pipeline

Data, markets, expert views → Model library → Scenario universe → Optimization engine → Decision + sensitivity

How should we choose the models?

A useful model library should balance several criteria:

- **Empirical relevance:** the model should not contradict important data features.
- **Market consistency:** when applicable, it should respect liquid market prices and no-arbitrage constraints.
- **Decision relevance:** it should stress the payoff, hedge, reserve, or loss function we care about.
- **Diversity:** models should not all generate the same type of path.
- **Admissibility:** scenarios must respect hard constraints such as positivity or boundedness.
- **Governance:** the assumptions should be explainable and auditable.

How many scenarios from each model?

Let N_k be the number of scenarios generated from model M_k. This is part of the design.

A practical allocation principle

Allocate more scenarios to models that have higher plausibility, higher loss variance, or stronger tail relevance.

A simple variance-oriented rule is

$$N_k \propto q_k \sqrt{\mathrm{Var}_{M_k}\left(\ell(u, S)\right)},$$

where q_k is a model weight and $\ell(u, S)$ is the decision loss.

- impose a minimum scenario quota for every credible model;
- oversample stress models for tail-risk decisions;
- use scenario weights if some models represent more plausible futures;
- test sensitivity to q_k and N_k.

The decision problem after scenario construction

Suppose $u \in \mathcal{U}$ is a portfolio, hedge, reserve, or reinsurance decision. For scenario paths $S^{(k,i)}$, define the loss $\ell(u, S^{(k,i)})$.

A weighted scenario objective may be

$$u^\star = \arg\min_{u \in \mathcal{U}} \left\{ \sum_{k=1}^{K} q_k \frac{1}{N_k} \sum_{i=1}^{N_k} \ell(u, S^{(k,i)}) + \lambda \, \widehat{\mathrm{CVaR}}_\alpha(\ell(u, S)) \right\}.$$

A robust alternative is

$$u^\star = \arg\min_{u \in \mathcal{U}} \max_{k=1,\ldots,K} \widehat{\mathrm{CVaR}}_{\alpha, M_k}\big(\ell(u, S)\big).$$

Example: choosing a hedge

Suppose the decision is a static hedge u for a derivative or a risky position.

- ML fitting may produce a single estimated path distribution.
- Scenario modelling may test the hedge under:
  - normal diffusion paths,
  - volatility spikes,
  - negative jumps,
  - gap risk,
  - liquidity-stressed rebalancing conditions.
- The selected hedge should not only perform under the fitted past, but also under plausible adverse futures.

Engineering view

A hedge is not validated by fit. It is validated by performance under a well-designed scenario set.

A useful comparison

	Machine learning for fitting	**Modelling for decisions**
Primary question	What explains the observed data?	What futures should the decision face?
Output	fitted function or fitted dynamics	scenario universe and weights
Criterion	empirical loss, likelihood, prediction error	admissibility, relevance, robustness, stress coverage
Main risk	overfitting and poor extrapolation	missing important future regimes
Validation	out-of-sample statistical performance	decision performance and sensitivity
Role in workflow	approximation layer	scenario-generation and governance layer

Actuarial analogue

The same distinction appears in actuarial applications.

- A sigmoidal map can fit survival probabilities, reserves, ruin probabilities, or risk measures.
- Links and constraints impose admissibility: probability values, positivity, monotonicity, boundedness.
- For reinsurance selection, scenarios of aggregate losses can be used to minimize premium plus residual tail risk, for example empirical CVaR.

Connection

Actuarial modelling also separates fitted maps from decision-relevant scenario construction.

Practical workflow

1. Define the decision u and the loss $\ell(u, S)$.
2. Identify the path features that can make the decision fail.
3. Fit useful statistical components with machine learning where appropriate.
4. Build a library of scenario-generating models.
5. Assign model weights and scenario budgets.
6. Solve the optimization problem.
7. Stress test the selected decision under alternative scenario weights and model sets.

Discipline

Do not start with the model. Start with the decision and its failure modes.

A compact algorithm

Scenario-based decision construction

1. Input: historical data, market quotes, expert views, constraints, decision set $\mathcal{U}$.
2. Fit components if useful: $a_\theta, b_\theta, g_\theta$ or other predictive maps.
3. Choose model library $\{M_1, \ldots, M_K\}$.
4. For each M_k, choose weight q_k and scenario count N_k.
5. Simulate or construct paths $S_{0:T}^{(k,i)}$.
6. Solve
$$\min_{u \in \mathcal{U}} \text{scenario-based objective}(u).$$
7. Report the decision, the active scenarios, and sensitivity to model weights.

Who Chooses the Future Scenarios?

The selection of future scenarios is itself a forecasting problem.

Assume candidate models

$$\mathcal{M}_1, \ldots, \mathcal{M}_K.$$

A predictive mechanism should determine:

- which models are relevant for the future,
- the participation weight of each model,
- the number of scenarios generated from each model.

Example: a three-month decision

Suppose the decision horizon is the next three months and one year of historical data is available.

- Fit the best ML model on the historical data.
- If no relevant events suggest a behavioural change, keep the ML output.
- Otherwise, we must assess how these events may reshape the future: for example, whether they are likely to lower the stock value, increase volatility, generate jumps, trigger regime shifts, or introduce other structural changes. Based on this assessment, we should then select models capable of producing future trajectories with the properties we expect the market to exhibit.

If the future is expected to behave essentially like the past, then

$$\mathcal{M}_{\text{future}} = \mathcal{M}_{\text{ML}}, \qquad w_{\text{ML}} = 1.$$

Fitting the past is not the same as constructing the future.

- Machine learning estimates functions from data.
- Modelling constructs scenario worlds with desired properties.
- Financial and actuarial decisions require both approximation and scenario design.
- The future is not given by the best-fitting past model; it must be represented, stressed, and tested.

Final sentence

Machine learning answers: “What did the data say?” Modelling asks: “What futures must our decision survive?”

A new model should not be justified by showing that it fits a data set. It should be justified theoretically, by proving that it has the properties for which it was constructed. Then, when we need future trajectories with specific features, we know which model to choose. If the objective is merely to fit past data, machine learning is sufficient.

References

Reading guide

The last two references show why a decision taken today requires the construction of possible future paths: the hedging or valuation rule is evaluated on historical, simulated, or stressed trajectories before the final action is selected.

[1] N. Halidias.
Financial Engineering with Python.
World Scientific, forthcoming 2026.
World Scientific book page.

[2] N. Halidias.
"Bounded Data-Driven Actuarial Maps and CVaR-Based Reinsurance Selection."
2026.
ResearchGate.

[3] N. Halidias.
"Dynamic Trading and Path Dependent Options."
ResearchGate preprint, March 20, 2026.

[4] N. Halidias.
"Feasible Dynamic Hedging and Data-Driven Valuation of Options in Discrete Time."
ResearchGate preprint, June 3, 2026.