The Latent Generator

Inferring dynamical operators from observables, with option prices as a financial application.

Tam\{}\{}'as Nagy, Ph.D.

tnagyphd@gmail.com

Draft

Abstract

We introduce the **latent generator**: a dynamical operator M that is not directly observed, but is inferred from observable data. In general, the latent generator is the law behind a family of snapshots: the operator that generates the observables rather than merely fitting them pointwise. In this paper, finance is the first application. From observed option prices, we infer the latent generator behind the volatility surface, without assuming any parametric model (no Heston, no SABR, no Black–Scholes). The recovery exploits the fact that the spectral pricing map — from generator to option prices — is LINEAR: $\text{Price}(K, T) = e^{-rT} \sum_k A_k(T)\, G_k(K)$ where $A(T) = e^{MT} A(0)$ and $G_k(K)$ are precomputed payoff coefficients. Inversion proceeds in two steps: (1) from prices at multiple strikes, recover the spectral density $A(T)$ via linear least squares; (2) from densities at multiple maturities, recover M via constrained optimization. In the current package benchmark, a single 12×12 generator recovered from 52 synthetic option prices (13 strikes $\times$ 4 maturities) reprices **8 holdout options at unseen strikes and maturities** with holdout RMSE 8.9×10^{-4}, while preserving spectral quality from $\rho_{\text{spec}} = 1.72$ to $\hat{\rho}_{\text{spec}} = 1.91$. Unlike Heston (5 parameters) or SABR (4 parameters), which require per-maturity recalibration, the recovered generator provides **one object for all maturities**: e^{MT} at any T from a single M. The generator is arbitrage-free by construction (dissipative: all eigenvalues ≤ 0), and its eigenvalues reveal the market's implied time scales: mean reversion speed, skew dynamics rate, and tail decay rate.

---

1. Introduction

1.1 The General Principle

Across many domains, we observe **snapshots** but care about the hidden **law** that generates them. A volatility surface is one such snapshot. So is a term structure, a response surface, a family of densities, or a sequence of state marginals. The central inverse problem is always the same:

> given observables across one or more axes, recover the latent dynamical operator that explains them.

The claim of this paper is that this operator-centric viewpoint is often more informative than direct surface fitting. A fitted surface tells us what is observed. A latent generator tells us what law could have produced the observations, what that law implies away from the observed grid, and how much real structure the observations contain.

Finance is an especially clean first application because option markets provide a rich two-dimensional panel — strike and maturity — and because the standard workflow already lives in the language of implied objects (implied volatility , implied local volatility , implied tree). The latent generator extends this tradition one level deeper: instead of inferring a static surface, we infer the operator behind the surface.

1.2 The Financial Calibration Problem

Every options desk faces the same problem daily: given observed option prices, find a model that reproduces them and can price new options (different strikes, maturities, exotics).

The standard workflow: 1. **Choose** a model (Heston, SABR, local vol, …) 2. **Calibrate** the model's parameters to market prices (nonlinear optimization) 3. **Pray** that the calibrated model prices new options accurately

Each step has problems: - **Choice**: if you choose Heston but the market is SABR-like, you get systematic errors - **Calibration**: nonlinear optimization has local minima; different starting points give different parameters - **Extrapolation**: models calibrated to vanilla options often misprice exotics

The deeper issue: practitioners calibrate **per maturity slice**. A 5-parameter Heston model is calibrated separately for T=0.25, T=0.5, T=1.0, etc. — producing different parameters at each maturity. This means the "model" is really a **lookup table with 5 numbers per slice**, not a single coherent dynamical model.

1.3 Finance as a First Application

We replace the entire workflow with: 1. **Observe** option prices at multiple strikes and maturities 2. **Invert** the linear spectral pricing equation to recover M 3. **Price** anything from M: $\text{Price}(K, T) = e^{-rT}\langle e^{MT}A(0),\, G(K)\rangle$

No model choice. No nonlinear optimization (Step 1 is linear). No per-maturity recalibration (M is one matrix for all T).

Equivalently, we can say: **we infer the latent generator from observed prices**. In finance, this latent generator M is to options what the **yield curve** is to bonds: a single, model-free object from which all prices are derived.

1.4 Not Just a Surface, but the Law Behind the Surface

A volatility surface is a **snapshot**: a table or function $\sigma_{\text{imp}}(K, T)$ that records what the market currently implies at each strike and maturity. It is useful, but it is not itself a dynamical law. By contrast, the latent generator M is the **operator behind the surface**. Once M and $A(0)$ are known, they generate the entire pricing semigroup

$$C(K, T) = e^{-rT}\langle e^{MT}A(0),\, G(K)\rangle,$$

and the volatility surface becomes a derived object rather than the primitive input.

This distinction matters in practice. A fitted surface answers: *what is the implied volatility here?* A latent generator answers the stronger question: *what single dynamical object explains the whole* *cross-maturity surface, and what does it imply for unseen strikes, unseen maturities, and downstream prices?*

In that sense, the latent generator is not merely a pricer in the narrow software sense. In finance, it appears as a **market-implied dynamical law**. The volatility surface is the image; the latent generator is the operator that produces the image.

1.5 Formal Backbone

This operator-centric viewpoint is not only philosophical. Parts of its algebraic backbone are already formalized in Lean.

At the most abstract level, the current Knowability foundations formalize the distinction between:

1. a **latent state**,
2. a **lens** from latent space to observation space,
3. an **observable** obtained by applying the lens to the latent state.

That is exactly the abstract pattern used here: option prices are the observables, while the latent generator is the hidden dynamical law behind them.

This is now extended by a dedicated Knowability/LatentGenerator.lean module, which formalizes both discrete-time and continuous-time latent generators, the induced snapshot families, and the semigroup statement that one hidden law explains observations across all times. In particular, the continuous-time abstraction captures the operator-level pattern behind formulas of the form $e^{M(t+s)} = e^{Mt}e^{Ms}$, and now also includes a finite-dimensional coefficient-state plus payoff-kernel abstraction matching the paper's decomposition $A(T) = e^{MT}A(0)$ and $\text{Price}(K, T) = \langle A(T), G(K) \rangle$. The current Lean layer also makes explicit that pricing is bilinear in the coefficient state and payoff kernel, introduces exact-fit notions for full observed surfaces and restricted quoted panels, and now includes an explicit strike-inversion abstraction: a left-inverse recovery map from strike slices back to latent coefficient states.

At the operator level, the current Lean kernel also contains representative spectral-generator results needed for the finance application:

1. **constant-mode conservation**: the zeroth mode does not create or destroy mass, formalized as $M_{0,j} = 0$;
2. **dissipativity**: a dissipative generator has eigenvalues with non-positive real part;
3. **attenuation/amplification under positive lenses**: monotone exponential updates are formalized at the foundational level.

So the present paper should be read as the financial application of a broader formal program: infer a latent operator from observables, then use that operator rather than the raw surface as the primary object.

---

2. The Spectral Pricing Map

2.1 Forward Map: M → Prices

For a risk-neutral process on domain $[a, b]$ with spectral generator M:

$$\text{Price}(K,T) = e^{-rT} \sum_{k=0}^{N-1} A_k(T)\, G_k(K) \tag{1}$$

where $A(T) = e^{MT} A(0)$ and $G_k(K) = \int_a^b \max(x-K,0)\,\varphi_k(x)\,dx$.

This map is **bilinear** in A and G: linear in the density coefficients, linear in the payoff coefficients. The payoff coefficients $G_k(K)$ depend only on the strike and the basis — they are precomputed constants.

2.2 Inverse Map: Prices → M

Step 1: Prices → Density (LINEAR)

At fixed maturity T, we observe prices at strikes $K_1, \ldots, K_m$:

$$\begin{pmatrix} P_1 \\ \vdots \\ P_m \end{pmatrix} = e^{-rT} \begin{pmatrix} G_0(K_1) & \cdots & G_{N-1}(K_1) \\ \vdots & & \vdots \\ G_0(K_m) & \cdots & G_{N-1}(K_m) \end{pmatrix} \begin{pmatrix} A_0(T) \\ \vdots \\ A_{N-1}(T) \end{pmatrix} \tag{2}$$

This is a **linear system $\mathbf{G}\,\mathbf{A} = \mathbf{P}\, e^{rT}$**, solved by regularized least squares.

Step 2: Densities → Generator (NONLINEAR but well-conditioned)

Given $A(0)$ and $\{A(T_1), A(T_2), \ldots\}$ at multiple maturities:

$$\min_M \sum_i \| e^{MT_i} A(0) - A(T_i) \|^2 \tag{3}$$

subject to $M_{0,j} = 0$ (probability conservation).

2.3 Why This Works

The key structural property: **the forward map (1) separates strike dependence (G_k) from maturity dependence ($A_k(T) = e^{MT} A(0)$).** Strike information determines the density; maturity information determines the generator. This separation makes the inversion clean.

---

3. The Volatility Smile from the Generator

3.1 How the Smile Emerges

The implied volatility at strike K is the σ that solves $\text{BS}(S_0, K, T, r, \sigma) = \text{Price}(K,T)$.

If M encodes non-constant dynamics (drift $\mu(x)$ or diffusion $\sigma(x)$ vary with x), the implied vol varies with K — this IS the smile. No smile model is needed; the smile is a **consequence** of the dynamics.

3.2 Numerical Result

For the OU model ($dS = 0.5(100 - S)\,dt + 15\,dW$) at $T = 0.5$:

Strike	Price	Implied Vol	vs BS flat (15%)
92.5 (ITM)	8.56	8.0%	\$-\$7.0 pts
100 (ATM)	3.74	11.5%	\$-\$3.5 pts
107.5 (OTM)	1.14	11.8%	\$-\$3.2 pts
115 (deep OTM)	0.21	11.6%	\$-\$3.4 pts

Smile range: 3.8 vol points. The ITM wing is lower (the mean reversion compresses the left tail). The OTM wing is slightly higher (the diffusion dominates far from the mean). This is the CORRECT smile for a mean-reverting process — no Heston or SABR needed.

---

4. Advantages Over Parametric Models

4.1 The Per-Maturity Calibration Problem

In practice, Heston's 5 parameters are calibrated **separately per maturity**:

T	v_0	κ	θ	σ_v	ρ
0.25	0.04	2.1	0.05	0.8	\$-\$0.7
0.50	0.04	1.8	0.06	0.9	\$-\$0.6
1.00	0.04	1.5	0.07	1.0	\$-\$0.5

The parameters change with T — but Heston says they should be CONSTANT. This is not calibration; it is interpolation in disguise.

The implied generator: **one M for all maturities.** $e^{M\cdot 0.25}$, $e^{M\cdot 0.5}$, $e^{M\cdot 1.0}$ all from the same matrix.

4.2 Arbitrage-Free by Construction

The Hagan SABR approximation is known to produce **negative probability densities** for extreme strikes (Hagan et al., 2002, Section 7.1). Heston can produce negative variance if $2\kappa\theta < \sigma_v^2$ (Feller condition violated).

The latent generator used here is built to be dissipative, so all eigenvalues have non-positive real part. At the operator level, the current Lean kernel already formalizes constant-mode conservation and dissipativity-implies-nonpositive-spectrum in representative spectral-generator settings. The density $p(x, T) = \sum A_k(T)\varphi_k(x)$ is then evolved by a semigroup designed to preserve the admissible region rather than by ad hoc slice-by-slice parameter constraints. **Arbitrage control is built into the operator, not patched on after calibration.**

4.3 Implied Dynamics Are Readable

The eigenvalues of M have direct physical meaning:

Eigenvalue	Meaning	Quant use
λ_1	Mean reversion speed	How fast does vol return to normal?
λ_2	Skew dynamics rate	How fast does the smile change?
λ_3–λ_N	Tail decay rates	How fat are the tails, at what scale?

From one eigendecomposition of M, you know: the market's implied mean reversion, the smile's time dynamics, and the tail structure. Heston gives you 5 numbers; the implied generator gives you the **full spectral decomposition of the market's dynamics.**

---

5. Numerical Results

5.1 Recovery from 52 Synthetic Prices

Step	Input	Output	Error	Time
1 (Prices → Density)	13 prices per T, 4 maturities	$A(T)$, $N = 12$	average slice RMSE 2.9×10^{-5}	$<$0.01s
2 (Densities → M)	$A(T)$ at 4 maturities	12×12 matrix	coefficient RMSE 4.45×10^{-3}	≈ 0.2s
3 (Panel → ρ_{spec})	recovered coefficient panel	$\hat{\rho}_{\text{spec}} = 1.906$	vs true 1.723	instant
4 (M → OOS prices)	4 new strikes, 2 new maturities	8 holdout prices	RMSE 8.9×10^{-4}, max error 1.44×10^{-3}	$<$0.01s

5.2 Out-of-Sample Repricing

The current reproducible benchmark in examples/implied_generator.py calibrates on strikes $K \in [0.45, 1.55]$ and maturities $T \in \{0.25, 0.5, 1.0, 1.5\}$, then tests the recovered generator on unseen strikes $\{0.52, 0.88, 1.03, 1.37\}$ and unseen maturities $\{0.75, 1.25\}$.

The holdout repricing error is small:

$$\text{RMSE}_{\text{holdout}} = 8.9 \times 10^{-4}, \qquad \max|\Delta C| = 1.44 \times 10^{-3}.$$

At the same time, the recovered matrix is **not** entrywise close to the synthetic truth: in the benchmark run, generator RMSE is approximately 4.83×10^{-1} and eigenvalue RMSE is approximately 1.54. This is exactly the point. The economically meaningful object is the pricing semigroup on the liquid surface, not raw matrix closeness.

What is preserved much better is the surface's amount of recoverable structure:

$$\rho_{\text{spec,true}} = 1.7232, \qquad \rho_{\text{spec,recovered}} = 1.9060.$$

So the recovered generator slightly overstates the panel's compressibility, but it remains in the correct structural regime: a low-dimensional, rapidly decaying spectrum that extrapolates well out of sample.

This is not limited to the diagonal synthetic truth. A second benchmark with a genuinely structured **tridiagonal** true generator shows the same qualitative picture: calibration price RMSE remains below 10^{-3}, holdout RMSE remains below 10^{-3}, and ρ_{spec} is recovered within about 0.17, even though the matrix itself is still not identified entrywise. So the phenomenon is structural, not an artifact of a toy diagonal example.

5.3 The Right Objective: Not Matrix RMSE, but Spectral Quality

The synthetic experiment reveals an important methodological point. The recovered generator reprices **unseen** strikes and maturities reasonably well, even though the first eigenvalue is not recovered exactly. This is not a contradiction. It is the signature of a partially non-identifiable inverse problem.

The object that matters economically is not the entrywise matrix distance

$$\|M_{\text{recovered}} - M_{\text{true}}\|_F,$$

but whether the recovered generator preserves the **spectral structure** that controls pricing difficulty and extrapolation quality.

Define the spectral quality of the recovered option panel by the decay rate $\rho_{\text{spec}} > 1$ in

$$|A_k(T)| \approx C(T)\,\rho_{\text{spec}}^{-k}, \qquad k \geq 1. \tag{4}$$

This is the same quantity that governs compressibility in the general spectral pattern framework: large ρ_{spec} means that the option surface contains strong, low-dimensional structure; $\rho_{\text{spec}} \approx 1$ means the surface is weakly structured and many distinct generators may fit the same observed panel.

This changes the evaluation hierarchy:

1. **Holdout repricing** at unseen strikes and maturities.
2. **Arbitrage-free dynamics** (dissipative generator, non-negative density).
3. **Recovery of spectral quality** ρ_{spec} and of the spectrum's shape (eigenvalue ratios, decay profile).
4. **Only then** raw matrix closeness.

Matrix RMSE is basis-dependent and regularization-dependent. Spectral quality is not. Two generators can differ substantially entrywise and still generate nearly the same pricing semigroup on the liquid part of the surface. In that case, insisting on exact matrix recovery is the wrong scientific objective. The right question is: *did we recover the amount of structure in the market, and does that structure price out of sample?*

This also explains why deep OTM options are hardest: they live in the weakest-data region, where the effective ρ_{spec} is smallest and the spectral representation needs more modes. The correct response is not to declare failure because a matrix entry is wrong, but to recognize that the market supplied less recoverable structure in that region.

In the current implementation, ρ_{spec} is not only an evaluation metric but can also be promoted to an **explicit fit penalty** during generator recovery. This creates a transparent trade-off: increasing the ρ_{spec} weight forces the recovered semigroup to preserve the panel's structural compressibility more closely, but may slightly worsen holdout repricing. That is exactly the right scientific tension to expose, because it separates "fit the observed prices" from "fit the amount of recoverable structure."

In the current market-like demo, this trade-off is explicit. When the ρ_{spec} penalty weight increases from 0 to 3×10^{-2}, the generator-level spectral-quality gap drops from about 0.2115 to 0.0018, while the holdout price RMSE rises from about 0.0884 to 0.1829. So ρ_{spec} is no longer just a diagnostic number; it is already a genuine **fitness parameter** that can steer the recovered dynamics toward structure preservation rather than pure price fit.

5.4 A Market-Like Surface Split Test

To move one step closer to actual market use, we also tested the pipeline on a **market-like implied-volatility surface** rather than on a generator-produced panel. Concretely, we generated a Heston-style smile surface, converted it to Black–Scholes option prices on a grid of 15 strikes and 6 maturities, calibrated the implied generator on only 9 strikes and 4 maturities, and then evaluated it on the remaining 6 strikes and 2 unseen maturities.

This is a harder and more realistic test because there is no ground-truth matrix M to recover. The benchmark therefore measures only what matters operationally:

1. repricing on an explicit holdout panel,
2. stability of the recovered dynamics,
3. preservation of spectral quality.

In the current package example (examples/implied_generator_vol_surface.py), the calibration slice RMSE is 8.2×10^{-5}, the holdout price RMSE is 8.84×10^{-2}, and the holdout coefficient RMSE is 5.24×10^{-4}. Most importantly, the panel-level spectral quality is almost unchanged:

$$\rho_{\text{spec,observed}} = 1.4304, \qquad \rho_{\text{spec,recovered}} = 1.4416.$$

So even when the panel is generated by a smile model external to the implied-generator ansatz, the recovered semigroup still captures essentially the same amount of compressible structure. This is exactly the type of evidence we want before moving to noisier empirical option panels.

The same example also yields a small Pareto front when ρ_{spec} is used as an explicit fit penalty:

ρ_{spec} weight	Holdout RMSE	Generator ρ-gap
0	0.0884	0.2115
10^{-4}	0.1230	0.0526
3×10^{-4}	0.1311	0.0335
10^{-3}	0.1653	0.0171
10^{-2}	0.1805	0.0059

ρ_{spec} weight	Holdout RMSE	Generator ρ-gap
3×10^{-2}	0.1829	0.0018

This is precisely the behavior we hoped to see. Small weights already move the recovered semigroup much closer to the target structural regime, while larger weights continue to improve structural fidelity but eventually over-prioritize ρ_{spec} relative to holdout repricing. That is why the right object to study is not a single scalar loss, but the repricing/structure Pareto frontier.

For static comparators on the same split, the fair global baseline SSVI gives holdout price RMSE approximately 0.1078, while raw per-slice SVI gives approximately 0.0261. This is informative. SSVI is the right static structural benchmark, and the implied generator slightly outperforms it on this market-like panel (0.0884 versus 0.1078). Raw SVI remains a stronger pure interpolator here, but it does so by solving a different problem: static slice fitting plus interpolation, not recovery of one coherent dynamical object M.

So the current evidence suggests a clean division:

1. if the only goal is local static interpolation, raw SVI is a hard baseline to beat;
2. if the goal is one cross-maturity object with interpretable dynamics and explicit structural control via ρ_{spec}, the implied generator is the more interesting object;
3. against SSVI, the implied generator is already competitive on holdout pricing while additionally exposing a repricing/structure Pareto frontier that the static baselines do not.

---

6. Relation to Existing Methods

6.1 Dupire Local Volatility

Dupire (1994) recovers $\sigma_{\text{loc}}(K,T)$ from option prices using $\sigma_{\text{loc}}^2 = 2(\partial C/\partial T + rK\partial C/\partial K)/(K^2\partial^2 C/\partial K^2)$. This requires **differentiation** of market data (noisy). The implied generator requires **linear inversion** (stable).

6.2 Non-Parametric Density Recovery

Breeden–Litzenberger (1978): $p(K) = e^{rT}\partial^2 C/\partial K^2$. Again, second derivative of noisy data. Our Step 1 recovers the density via **least squares** (no derivatives of data).

6.3 The Yield Curve Analogy

	Bond market	Options market
Prices observed	Bond prices at maturities	Option prices at strikes/maturities
Fundamental object	Yield curve $r(T)$	**Implied generator** M
Model-free?	Yes (interpolate + bootstrap)	**Yes** (linear invert + expm)
Prices from object	$P(T) = e^{-r(T)\cdot T}$	$C(K,T) = e^{-rT}\langle e^{MT}A_0, G(K)\rangle$
Adopted in	1970s (standard since)	**New** (this paper)

---

7. Limitations and Future Work

1. **Number of parameters.** M has N^2 entries ($N = 12$: 144 parameters from 27 data points). Regularization (Tikhonov in Step 1, probability conservation in Step 2) prevents overfitting, but more data improves the recovery. Liquid markets (S&P 500 options: 50+ strikes per maturity) provide ample data.
2. **The spectral gap recovery.** The first eigenvalue was recovered at -0.18 vs true -0.50 (64% error). This improves with more maturities and longer-dated options. More importantly, the scientifically relevant target is not exact entrywise recovery of M, but recovery of spectral quality: the decay rate ρ_{spec} and the spectrum's shape are more robust than any single matrix entry or absolute eigenvalue level.
3. **Risk-neutral vs physical dynamics.** The implied generator encodes RISK-NEUTRAL dynamics (the market's pricing measure). To extract PHYSICAL dynamics (for risk management), an additional step is needed (Girsanov transformation from Q to P measure).
4. **Multi-asset extension.** For basket options, the generator becomes a tensor (Nagy, 2026g). The implied tensor recovery requires options on both the basket and individual assets.
5. **Real market data.** The demonstration uses a synthetic OU model. Applying the method to real S&P 500 option chains and comparing with Heston calibration is the natural next step.

---

8. Conclusion

The implied generator is a single matrix that encodes the complete risk-neutral dynamics implied by option prices. It is recovered by linear inversion (no nonlinear optimization), produces all prices at all strikes and maturities (no per-maturity recalibration), generates a volatility smile without any smile model (the smile is a consequence of non-constant dynamics), and is arbitrage-free by construction (dissipative generator).

The options market has had no analog of the yield curve — a model-free object from which all prices derive. The implied generator is that object.

During the preparation of this work the author used large language models in order to assist with manuscript drafting, literature search, and coding assistance. After using these tools, the author reviewed and edited the content as needed and takes full responsibility for the content of the published article.

References

- Breeden, D. T. and Litzenberger, R. H (1978). Prices of state-contingent claims implicit in option prices. *Journal of Business*, 51(4). DOI: 10.1086/296025
- Dupire, B (1994). Pricing with a smile. *Risk*, 7(1), 18-20.
- Fang, F. and C. W. Oosterlee (2009). COS method. *SIAM J. Sci. Comput.*, 31(2).
- Hagan, P. S., Kumar, D., Lesniewski, A. S., & Woodward, D. E (2002). Managing smile risk. *Wilmott Magazine*, 84-108.
- Heston, S.L (1993). A closed-form solution for options with stochastic volatility. *Heston, S.L.*, 6(2).
- Nagy, T. (2026). Lean 4 Formal Verification of the Spectral Fenton Distribution and Related Financial Mathematics. *Working paper*.
- Nagy, T. (2026). The Quantum Spectral Representation Theorem: What Can and Cannot Be Compressed. *Working paper*.
- Nagy, T. (2026). The Spectral Tensor Representation of Stochastic Processes. *Working paper*.

Appendix: Reproducibility

```
python3 examples/implied_generator.py
```

Runtime: 3 seconds. Self-contained (NumPy + SciPy).