Nonparametric Options Pricing: Reconciling $\mathbb{P}$ and $\mathbb{Q}$ with Bootstrapping and Scaling

Fred Viole

2025-03-23

Nonparametric Options Pricing: Reconciling $\mathbb{P}$ and $\mathbb{Q}$ with Bootstrapping and Scaling

The reconciliation of real-world $\mathbb{P}$ and risk-neutral $\mathbb{Q}$ measures lies at the heart of arbitrage-free pricing, tracing back to Harrison and Kreps (1979) and Harrison and Pliska (1981), who established $\mathbb{E}^{\mathbb{Q}}[S_T] = S_0 e^{rT}$ as the cornerstone of risk-neutral valuation. Traditional methods, like Girsanov's theorem for measure changes, minimal-entropy adjustments, or weighted Monte Carlo, often impose multiple constraints or require calibration to replicate option market surfaces, trading simplicity for precision. In contrast, this paper presents a nonparametric extension of mean-matching that stands apart by leveraging empirical distributions directly, scaling them multiplicatively to enforce risk-neutrality without altering their intrinsic shape or relying on probabilistic reweighting. Enhanced by bootstrapping tools (e.g., `NNS.meboot`, `NNS.MC`) and integrated into `NNS.rescale`, this approach seamlessly adapts to any price distribution (empirical, synthetic, or AI-generated), offering a robust, intuitive alternative that prioritizes data-driven flexibility over parametric rigidity.

1. Understanding the Difference Between $\mathbb{P}$ and $\mathbb{Q}$

- **Real-world measure $\mathbb{P}$:**
  In the real world, asset prices include a **risk premium**, so the expected return is:

  $$\mathbb{E}^{\mathbb{P}}[S_T] = S_0 e^{\mu T},$$

  where μ is the real-world expected return (typically higher than the risk-free rate r).

- **Risk-neutral measure $\mathbb{Q}$:**
  Under the risk-neutral measure, asset prices grow at the risk-free rate:

  $$\mathbb{E}^{\mathbb{Q}}[S_T] = S_0 e^{rT}.$$

  This ensures **arbitrage-free pricing**, with the key distinction being:

  $$\mu \neq r.$$

This fundamental gap drives pricing theory, but traditional reconciliations often lean on complex probabilistic tools. My method separates itself by sidestepping such reweighting, instead using a simple scalar to bridge $\mathbb{P}$ and $\mathbb{Q}$, preserving empirical fidelity while ensuring universality across data sources, a practical departure from the parametric constraints of models like Black-Scholes or Heston.

2. The Transformation: Adjusting Any Distribution Multiplicatively

To convert a distribution X_T (e.g., terminal prices) from $\mathbb{P}$ to $\mathbb{Q}$, we scale outcomes by a constant factor. This applies to any distribution (empirical, synthetic, or AI-generated):

$$X_T^* = X_T \times e^{\theta},$$

where e^{θ} aligns the adjusted mean with the risk-neutral expectation:

$$e^{\theta} = \frac{S_0 e^{rT}}{\mathbb{E}^{\mathbb{P}}[X_T]},$$

ensuring:

$$\mathbb{E}^{\mathbb{Q}}[X_T^*] = \mathbb{E}^{\mathbb{P}}[X_T] \cdot e^{\theta} = S_0 e^{rT}.$$

For example, if $\mathbb{E}^{\mathbb{P}}[X_T] = S_0 e^{\mu T}$, then $e^{\theta} = e^{(r-\mu)T}$.

Unlike Girsanov's theorem, which reweights probabilities under a specific process, this deterministic adjustment avoids stochastic assumptions, offering nonparametric freedom. Its simplicity, adjusting the mean without reshaping probabilities, sets it apart, delivering risk-neutrality with clarity and empirical integrity across diverse inputs.

3. How This Resolves $\mathbb{P}$ and $\mathbb{Q}$

Before Adjustment (Under $\mathbb{P}$):

- The distribution reflects an expected return of μ due to the risk premium, exceeding the risk-free rate.

After Adjustment (Under $\mathbb{Q}$):

- Scaling by e^{θ} adjusts the mean to $S_0 e^{rT}$.
- All moments scale (e.g., variance by $(e^{\theta})^2$), but no specific shape is imposed.
- This deterministic adjustment ensures risk-neutrality without probabilistic reweighting (e.g., Girsanov's theorem).

This resolution stands out by maintaining the distribution's intrinsic features up to a scalar, contrasting with methods like minimal-entropy measures that overhaul probabilities to fit market data. Its practical simplicity preserves empirical shape, making it a robust, intuitive bridge that does not demand the computational heft of traditional approaches.

4. Universality Across Distributions

This approach works for any distribution of terminal prices X_T:
- **Empirical**: Historical price data with observed risk premia.
- **Synthetic**: Modeled price data from simulations.
- **AI-Generated**: Price outputs from generative models.

Since X_T represents asset prices ($S_T > 0$), the mean $\mathbb{E}^{\mathbb{P}}[X_T]$ is inherently positive, ensuring e^{θ} is well-defined. The bootstrapping step can adapt the shape, enhancing flexibility (e.g., for scenario analysis), while scaling guarantees risk-neutrality. This universal compatibility with any data source, free from the parametric shackles of models like Heston or the data-hungry nature of neural SDEs, truly sets it apart, making it a versatile tool for modern pricing challenges.

5. A New Procedure: Bootstrapping and Scaling

Bootstrapping is a cornerstone of this method because it liberates pricing from the restrictive assumptions of parametric models, such as lognormality or constant volatility, which often fail to capture the complex dynamics of real-world markets (e.g., fat tails, volatility clustering). By generating synthetic replicates of price distributions, `NNS.meboot` and `NNS.MC` preserve empirical features while allowing for controlled variation, enabling scenario analysis (e.g., stress testing extreme market conditions) and ensuring the method remains data-driven. This nonparametric approach not only aligns with the method's goal of empirical fidelity but also provides the flexibility to adapt the distribution's shape, making it ideal for pricing under uncertainty without imposing a rigid stochastic framework.

This method leverages nonparametric bootstrapping:

1. Start with an empirical distribution (e.g., historical prices).
2. Bootstrap it using `NNS.meboot` or `NNS.MC` from the NNS package (NNS Sampling Vignette) to generate X_T under $\mathbb{P}$. These methods can preserve or adapt the original shape, depending on configuration.
3. Scale X_T to X_T^* using e^θ (e.g., via `NNS.rescale` with `method = "riskneutral"`, `type = "Terminal"` for mean $S_0 e^{rT}$, or `type = "Discounted"` for mean S_0) for risk-neutrality.
4. Price options with partial moments.

The shape adaptability of NNS bootstrapping separates this procedure from static mean-matching or weighted Monte Carlo, allowing controlled exploration of distributional variations. Combined with the universal scaling of `NNS.rescale`, it offers a flexible, data-driven workflow that traditional methods, tied to fixed processes or heavy calibration, cannot replicate.

6. Application to Derivative Pricing with Partial Moments

Partial moments are particularly well-suited for option pricing in this nonparametric framework because they directly address the asymmetry inherent in financial price distributions, which often exhibit skewness and fat tails that standard moments (e.g., variance) fail to capture adequately. The Upper Partial Moment (UPM) focuses on outcomes above the strike price K, corresponding to the positive payoffs of a call option, while the Lower Partial Moment (LPM) targets outcomes below K, aligning with put option payoffs. This separation allows for a more precise evaluation of upside potential and downside risk, reflecting the asymmetric nature of option payoffs without assuming a symmetric distribution like normality. Moreover, UPM and LPM are computationally straightforward, requiring only the empirical distribution of X_T^*, which fits seamlessly with the method's data-driven ethos and avoids the need for parametric assumptions about the underlying process.

Using X_T^* (e.g., from `NNS.rescale` with `type = "Terminal"`, where $\mathbb{E}^{\mathbb{Q}}[X_T^*] = S_0 e^{rT}$), compute present-value option prices by discounting at the risk-free rate:
- **Call Option Price**:

$$C = e^{-rT} \cdot \mathbb{E}^{\mathbb{Q}}[(X_T^* - K)^+] = e^{-rT} \cdot \mathrm{UPM}(1, K, X_T^*).$$

- **Put Option Price**:

$$P = e^{-rT} \cdot \mathbb{E}^{\mathbb{Q}}[(K - X_T^*)^+] = e^{-rT} \cdot \mathrm{LPM}(1, K, X_T^*).$$

The discounting ensures the present value of $\mathbb{E}^{\mathbb{Q}}[X_T^*]$ is S_0, since $e^{-rT} \cdot S_0 e^{rT} = S_0$. Alternatively, using `NNS.rescale` with `type = "Discounted"` yields X_T^* with mean S_0, requiring no further discounting for present-value expectations. Pricing with a discounted distribution ($\mathbb{E}^{\mathbb{Q}}[X_T^*] = S_0$) eliminates the need for explicit discounting, offering an alternative workflow for present-value calculations. This flexible pricing duality, terminal or discounted, distinguishes it from rigid, terminal-only frameworks, enhancing its practical utility.

7. Comparison to Other Techniques

- **Heston with Jumps**: Combines stochastic volatility and jumps, capturing volatility smiles. Parametric and calibration-heavy, unlike your nonparametric, adaptable approach that avoids such complexity.

- **Rough Volatility**: Models volatility with fractional Brownian motion (Hurst $H < 0.5$), computationally intensive. Your method is simpler, with flexible shape adaptation via bootstrapping.

- **Neural SDEs**: AI-driven, learns complex dynamics but lacks transparency. Your method is interpretable, efficient, and works with any bootstrapped X_T, prioritizing clarity over opacity.

- **Advantages**: Nonparametric, universal, computationally light. **Limitation**: Variance scales with $(e^\theta)^2$, no explicit volatility dynamics.

This method separates itself by balancing empirical fidelity with simplicity, avoiding the calibration burdens or process-specific assumptions of competitors, while offering a transparent, adaptable alternative suited to diverse pricing needs.

8. Extending to Path-Dependent Options

The replicates from `NNS.meboot` or `NNS.MC` can be treated as full price paths to price path-dependent options. Generate n paths of prices S_t from $t = 0$ to T, scale each timestep to ensure $\mathbb{E}^{\mathbb{Q}}[S_t] = S_0 e^{rt}$, and compute payoffs (e.g., average price for Asian options, barrier crossings for knock-out options). This extends the method's nonparametric framework to handle intermediate timesteps, preserving empirical path dynamics while ensuring risk-neutrality.

9. Conclusion: A Versatile Bridge Between $\mathbb{P}$ and $\mathbb{Q}$

This procedure, bootstrapping with NNS (adaptable shape) and scaling with $e^\theta = S_0 e^{rT}/\mathbb{E}^{\mathbb{P}}[X_T]$ or $S_0/\mathbb{E}^{\mathbb{P}}[X_T]$ via `NNS.rescale`, provides a robust, nonparametric bridge from $\mathbb{P}$ to $\mathbb{Q}$. It applies to any price distribution, ensures arbitrage-free pricing with flexible terminal or discounted outputs, and stands apart as a practical alternative to parametric models (e.g., Heston+jumps), rough volatility, and neural SDEs. By emphasizing data-driven flexibility, shape adaptability, and simplicity over probabilistic reweighting or calibration, it excels in empirical applicability, offering a fresh, intuitive path for options pricing in a complex financial landscape.

10. References

- Harrison, J. M., & Kreps, D. M. (1979). Martingales and arbitrage in multiperiod securities markets. *Journal of Economic Theory*, 20(3), 381-408.
- Harrison, J. M., & Pliska, S. R. (1981). Martingales and stochastic integrals in the theory of continuous trading. *Stochastic Processes and their Applications*, 11(3), 215-260.