Simulated Method of Embeddings: Econometrics Without Likelihoods or Moments via Contrastive Learning

Steven Otis

April 7, 2025

Motivation: Estimating Models in Finance is Hard

- Finance relies heavily on structural models: option pricing, term structures, stochastic volatility, portfolio choice.
- These models are grounded in economic theory, but their estimation is notoriously difficult.
- Why? The likelihood is often intractable, and method-of-moments approaches require hand-crafted moments.
- As models become more realistic, estimation becomes less feasible.

Limitations of Traditional Approaches

Maximum Likelihood Estimation (MLE):

- Requires explicit form of $P(Y|\theta)$, which is usually unavailable.

General Method of Moments (GMM):

- Relies on analytical moment conditions $\mathbb{E}[m(Y,\theta)] = 0$.
- Intractable in complex models.

Bottom Line: Standard methods don't scale to modern structural models.

Simulation-Based Alternatives

MSM (McFadden 1987) and Indirect Inference (Gourieroux, Monfort, and Renault 1993) offer workarounds:

- Replace analytical expressions with simulations.
- Use moment matching or auxiliary models as proxies for the data-generating process.
- But: These rely on heuristic summaries and subjective choices.

Limitation: Hand-crafted features limit generality and robustness.

Simulated Method of Embeddings (SME)

- I propose a new simulation-based method using contrastive learning.
- Leverages recent advances in representation learning to estimate θ directly.
- No need for likelihoods, moments, or auxiliary models.
- Replaces heuristic compression with learned embeddings.
- **Idea:** Learn an implicit likelihood using ML.
- Is an extension of E&E (Jiang, Lu, and Willett 2024)
  - Recover entire posterior.
  - My method is more focused, made for economic inference (and is more efficient).

Overview

In this presentation, I will:

- Introduce the Simulated Method of Embeddings (SME).
- Explain the theoretical foundation behind SME.
- Demonstrate how SME recovers the shape of the likelihood from simulation.
- Evaluate SME on benchmark models to assess accuracy and robustness.
- Apply SME to a real-world financial model:
  - Chan–Karolyi–Longstaff–Sanders (CKLS; 1992) process for interest rate modeling.
  - Compare performance with standard methods in pyMLE (Kirkby et al. 2024), showing SME's superiority.

Contrastive Learning Setup

Simulation:

- Sample parameters: $\theta_i \sim P(\theta)$.
- Generate observations: $Y_i \sim P(Y|\theta_i)$.

Scoring Function:

$$s_\nu(Y, \theta) : \mathcal{Y} \times \Theta \to \mathbb{R}$$

- Implemented as a neural network with parameters ν.

Finite Sample InfoNCE Loss

For a batch $\{(Y_i, \theta_i)\}_{i=1}^{M}$:

$$\mathcal{L}_{\text{InfoNCE}}^{(M)}(\nu) = -\frac{1}{M} \sum_{i=1}^{M} \log P_{\nu}^{(M)}(Y_i|\theta_i)$$

where

$$P_{\nu}^{(M)}(Y_i|\theta_i) = \frac{\exp(s_{\nu}(Y_i, \theta_i))}{\sum_{j=1}^{M} \exp(s_{\nu}(Y_i, \theta_j))}$$

Finite Sample InfoNCE Loss

For a batch $\{(Y_i, \theta_i)\}_{i=1}^{M}$:

$$\mathcal{L}^{(M)}_{\text{InfoNCE}}(\nu) = -\frac{1}{M}\sum_{i=1}^{M} \log \frac{\exp\left(s_\nu(Y_i, \theta_i)\right)}{\sum_{j=1}^{M} \exp\left(s_\nu(Y_i, \theta_j)\right)}$$

To remove the trivial scaling with M, define the *adjusted* loss:

$$\bar{\mathcal{L}}^{(M)}_{\text{InfoNCE}}(\nu) \equiv \mathcal{L}^{(M)}_{\text{InfoNCE}}(\nu) - \log M = -\frac{1}{M}\sum_{i=1}^{M} \log \frac{\exp\left(s_\nu(Y_i, \theta_i)\right)}{\frac{1}{M}\sum_{j=1}^{M} \exp\left(s_\nu(Y_i, \theta_j)\right)}$$

Asymptotic Behavior and Implicit Likelihood ($M \to \infty$)

As $M \to \infty$, by the Law of Large Numbers:

$$\frac{1}{M}\sum_{j=1}^{M} \exp\left(s_\nu(Y,\theta_j)\right) \xrightarrow{a.s.} \mathbb{E}_{P(\theta)}\Big[\exp\left(s_\nu(Y,\theta)\right)\Big]$$

We define the *implicit likelihood* as:

$$P_\nu(Y|\theta) \triangleq \lim_{M\to\infty} P_\nu^{(M)}(Y|\theta) \;=\; \frac{\exp\left(s_\nu(Y,\theta)\right)}{\mathbb{E}_{P(\theta)}\Big[\exp\left(s_\nu(Y,\theta)\right)\Big]}$$

Hence, the asymptotic adjusted loss becomes:

$$\bar{\mathcal{L}}_{\text{InfoNCE}}(\nu) \;=\; -\mathbb{E}_{P(Y,\theta)}\Big[\log P_\nu(Y|\theta)\Big]$$

Asymptotic Loss and KL Decomposition

Now, add and subtract $\log P(Y|\theta)$ inside the expectation:

$$\begin{aligned}\bar{\mathcal{L}}_{\text{InfoNCE}}(\nu) &= -\mathbb{E}_{P(Y,\theta)}\Big[\log P_\nu(Y|\theta)\Big] \\ &= \mathbb{E}_{P(Y,\theta)}\Big[\log P(Y|\theta) - \log P_\nu(Y|\theta)\Big] - \mathbb{E}_{P(Y,\theta)}\Big[\log P(Y|\theta)\Big].\end{aligned}$$

The left term is the KL divergence:

$$\mathbb{E}_{P(Y,\theta)}\Big[\log P(Y|\theta) - \log P_\nu(Y|\theta)\Big] = D_{KL}\Big(P(Y,\theta) \,\|\, P_\nu(Y|\theta)P(\theta)\Big),$$

and the right term is the expected entropy of the likelihood under $P(\theta)$:

$$-\mathbb{E}_{P(Y,\theta)}\Big[\log P(Y|\theta)\Big] = \mathbb{E}_{P(\theta)}\Big[H(P(Y|\theta))\Big].$$

Hence, we have the decomposition:

$$\boxed{\bar{\mathcal{L}}_{\text{InfoNCE}}(\nu) = \mathbb{E}_{P(\theta)}\Big[H(P(Y|\theta))\Big] + D_{KL}\Big(P(Y,\theta) \,\|\, P_\nu(Y|\theta)P(\theta)\Big)}.$$

Theorem 1: KL Divergence Minimization

Statement

$$\min_{\nu} \mathcal{L}_{\text{InfoNCE}}(\nu) \;=\; \min_{\nu} D_{KL}\Big(P(Y,\theta) \,\|\, P_\nu(Y|\theta)P(\theta)\Big).$$

Proof Sketch

1. $\min_\nu \mathcal{L}_{\text{InfoNCE}}(\nu) \;= \min_\nu \bar{\mathcal{L}}_{\text{InfoNCE}}(\nu)$, since $-log(M)$ is a constant
2. The asymptotic loss:

$$\bar{\mathcal{L}}_{\text{InfoNCE}}(\nu) = \mathbb{E}_{P(\theta)}\Big[H(P(Y|\theta))\Big] + D_{KL}\Big(P(Y,\theta) \,\|\, P_\nu(Y|\theta)P(\theta)\Big).$$

3. Since the term $\mathbb{E}_{P(\theta)}\Big[H(P(Y|\theta))\Big]$ does not depend on ν, minimizing $\bar{\mathcal{L}}_{\text{InfoNCE}}(\nu)$ is equivalent to minimizing the KL divergence.

Theorem 2: Consistency

Statement

If $\exists \nu^*$ such that $P_{\nu^*}(Y|\theta) = P(Y|\theta)$ a.e. under $P(\theta)$ or equivalently $P(Y, \theta) = P_{\nu^*}(Y|\theta)P(\theta)$ then:

$$\nu^* = \arg\min_{\nu} \mathcal{L}_{\text{InfoNCE}}(\nu) \implies \hat{P}_{\nu^*}(Y|\theta) = P(Y|\theta)$$

Proof Sketch

1. At ν^*, $D_{KL}(P(Y, \theta) \| \hat{P}_{\nu^*}(Y|\theta)P(\theta)) = 0$.
2. Thus, $\mathcal{L}_{\text{InfoNCE}}(\nu^*) = \mathbb{E}_{P(\theta)}\Big[H(P(Y|\theta))\Big]$ is the global minimum.
3. And, $P(Y|\theta) = \hat{P}_{\nu^*}(Y|\theta)$ (a.e.)

Complete Estimation Procedure

1. **Simulation Phase**:
   - Sample $\theta_i \sim P(\theta)$.
   - Generate $Y_i \sim G(\theta_i)$.
2. **Training Phase**:

$$\hat{\nu} = \arg\min_{\nu} -\frac{1}{M}\sum_{i=1}^{M} \log\left(\frac{\exp(s_\nu(Y_i, \theta_i))}{\sum_{j=1}^{M} \exp(s_\nu(Y_i, \theta_j))}\right)$$

3. **Inference Phase**: For a new observation Y we do MLE!

$$\hat{\theta}(Y) = \arg\max_{\theta} \ \hat{P}_{\hat{\nu}}(Y|\theta)$$

$$\approx s_{\hat{\nu}}(Y, \theta)$$

SME Architecture: Encoders and Embeddings

Encoder: $f_\xi : Y \to \mathbb{S}^{n-1}$

- f_ξ maps observations Y to the unit hypersphere $\mathbb{S}^{n-1} \subset \mathbb{R}^n$
- ξ: encoder network parameters

Emulator: $g_\eta : \theta \to \mathbb{S}^{n-1}$

- g_η maps model parameters θ to the same hypersphere
- η: emulator network parameters

Normalization (last layer):

$$f_\xi(Y) \leftarrow \frac{f_\xi(Y)}{\|f_\xi(Y)\|}, \quad g_\eta(\theta) \leftarrow \frac{g_\eta(\theta)}{\|g_\eta(\theta)\|}$$

So that: $\|f_\xi(Y)\| = \|g_\eta(\theta)\| = 1$

(Jiang, Lu, and Willett 2024)

Cosine Similarity & Scoring Function

Scoring function:

$$s_\nu(Y,\theta) = f_\xi(Y) \cdot g_\eta(\theta)/\tau$$

Since vectors are unit-norm: $s_\nu(Y,\theta) = \cos(\angle(f_\xi(Y), g_\eta(\theta)))/\tau$

- Measures alignment between embedded data and parameters
- Acts as an *implicit likelihood*
- $\nu = [\xi, \eta]$: joint parameter vector
- τ is a tuning parameter.
- The dot product forces the learned likelihood to be in the exponential family. In the future, I need to use a more flexible scoring function. Another network $s_v(Y,\theta) = T_\delta(f_\xi(Y), g_\eta(\theta))$?. One network $s_v(Y,\theta) = T_\delta(Y,\theta)$?

Total Loss Function (InfoNCE Variants)

SME training objective:

$$\mathcal{L}_{\text{loss}} = \mathcal{L}_{\text{sym}} + \mathcal{L}_{\text{aug}} + \mathcal{L}_{\text{intra}}$$

- **Symmetric InfoNCE** aligns $f_\xi(Y_i)$ with $g_\eta(\theta_i)$
- **Augmented InfoNCE** aligns augmented $\tilde{Y}_i \sim G(\theta_i)$
- **Intra InfoNCE** aligns Y_i with its own augmentation $\tilde{Y}_i$
- Symmetric InfoNCE and Intra InfoNCE from Jiang, Lu, and Willett 2024.

Data generation:

$$\theta_i \sim P(\theta), \quad Y_i, \tilde{Y}_i \sim G(\theta_i)$$

Symmetric InfoNCE Loss

$$\mathcal{L}_{\text{sym}} = \mathcal{L}_{\theta \to Y} + \mathcal{L}_{Y \to \theta}$$

First direction (fix θ_i):

$$\mathcal{L}_{\theta Y} = -\frac{1}{M} \sum_{i=1}^{M} \log \frac{\exp\left(f_\xi(Y_i) \cdot g_\eta(\theta_i)/\tau\right)}{\sum_{j=1}^{M} \exp\left(f_\xi(Y_j) \cdot g_\eta(\theta_i)/\tau\right)}$$

Second direction (fix Y_i):

$$\mathcal{L}_{Y\theta} = -\frac{1}{M} \sum_{i=1}^{M} \log \frac{\exp\left(f_\xi(Y_i) \cdot g_\eta(\theta_i)/\tau\right)}{\sum_{j=1}^{M} \exp\left(f_\xi(Y_i) \cdot g_\eta(\theta_j)/\tau\right)}$$

Goal: Ensure $f_\xi(Y_i)$ is aligned with $g_\eta(\theta_i)$

Augmented Symmetric InfoNCE Loss

Loss on augmented data:

$$\mathcal{L}_{\text{aug}} = \mathcal{L}_{\theta\tilde{Y}} + \mathcal{L}_{\tilde{Y}\theta}$$

- $\tilde{Y}_i \sim G(\theta_i)$: augmentation of original data
- Ensures robustness of encoder to noise or transformations
- Pulls $f_\xi(\tilde{Y}_i)$ close to $g_\eta(\theta_i)$

Same formula structure as symmetric loss, using $\tilde{Y}_i$ instead of Y_i

Intra InfoNCE Loss (Encoder Self-Consistency)

$$\mathcal{L}_{\text{intra}} = -\frac{1}{M}\sum_{i=1}^{M} \log \frac{\exp\left(f_\xi(Y_i)\cdot f_\xi(\tilde{Y}_i)/\tau\right)}{\sum_{j=1}^{M}\exp\left(f_\xi(Y_j)\cdot f_\xi(\tilde{Y}_i)/\tau\right)}$$

Purpose:

- Encourages embeddings of Y_i and its augmentation $\tilde{Y}_i$ to be close
- Stabilizes encoder learning

The SME Estimator

$$\hat{\theta}(Y) = \arg\max_{\theta \in \Theta} \; f_{\xi^*}(Y) \cdot g_{\eta^*}(\theta)$$

Equivalent forms:

$$= \arg\min_{\theta} \; 1 - f_{\xi^*}(Y) \cdot g_{\eta^*}(\theta)$$
$$= \arg\min_{\theta} \; \|f_{\xi^*}(Y) - g_{\eta^*}(\theta)\|^2$$

Why it works:

- Maximizing cosine similarity = minimizing Euclidean distance (since vectors are normalized)
- At optimum: $f_{\xi^*}(Y) = g_{\eta^*}(\theta)$
- This mimics a moment condition!

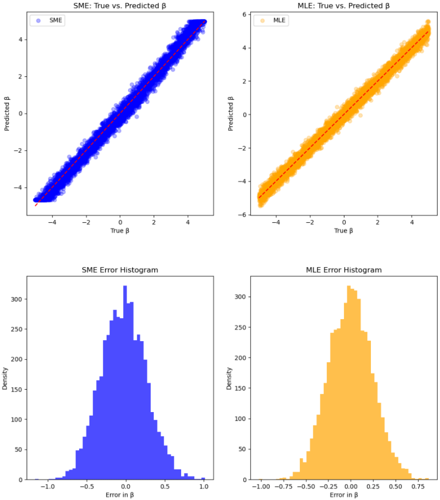
Optimization in Embedding Space

Two options for solving:

- **(1) Nearest Neighbor Search:** Fast if embedding space is low-dimensional
- **(2) Gradient-based Optimization:** Use similarity/distance as differentiable objective

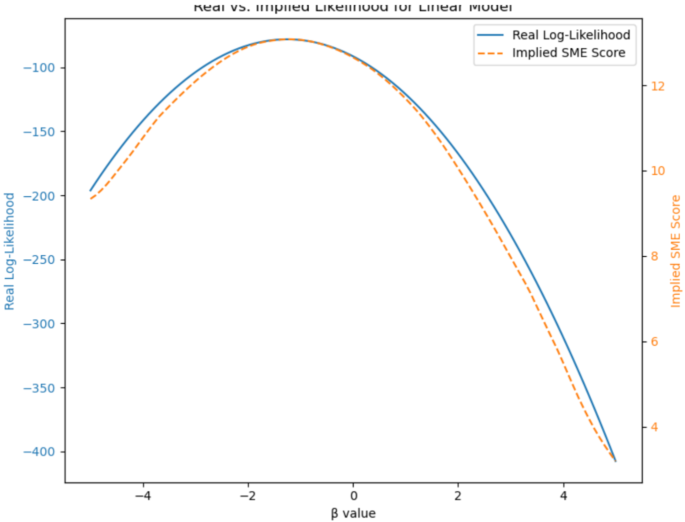
Results: $Y = \beta X + \epsilon, \quad \epsilon \sim N(0, \sigma)$

Figure: SME vs Linear MLE Estimates



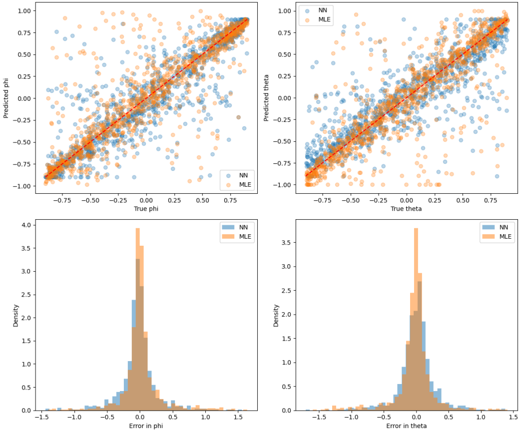
Results: $Y = \beta X + \epsilon, \quad \epsilon \sim N(0, \sigma)$

Figure: SME implied Likelihood vs Linear MLE likehood



Results: ARMA(1,1) $X_t = \phi X_{t-1} + \epsilon_t + \theta\epsilon_{t-1} \quad \epsilon_t \sim N(0, \sigma)$

Figure: SME vs ARMA MLE Estimates



The CKLS Model of the Short-Term Interest Rate

Chan–Karolyi–Longstaff–Sanders (1992) propose a flexible class of continuous-time models for the short-term interest rate r_t, given by:

$$\boxed{dr_t = (\alpha + \beta r_t)\, dt + \sigma r_t^{\gamma}\, dW_t}$$

Terms:

- Drift: $\alpha + \beta r_t$ controls mean reversion
  - $\beta < 0$: rate reverts to long-run mean $-\alpha/\beta$
- Diffusion: σr_t^{γ} governs volatility
  - γ: elasticity of volatility with respect to level of r_t
  - Allows testing whether volatility is constant, linear, square-root, etc.

Special Cases:

- Vasicek (1977): $\gamma = 0$
- CIR (1985): $\gamma = 0.5$
- Dothan (1978), GBM: $\gamma = 1$
- CEV model: general $\gamma > 0$

CKLS Simulation: Euler–Maruyama Scheme

Discretized Equation:

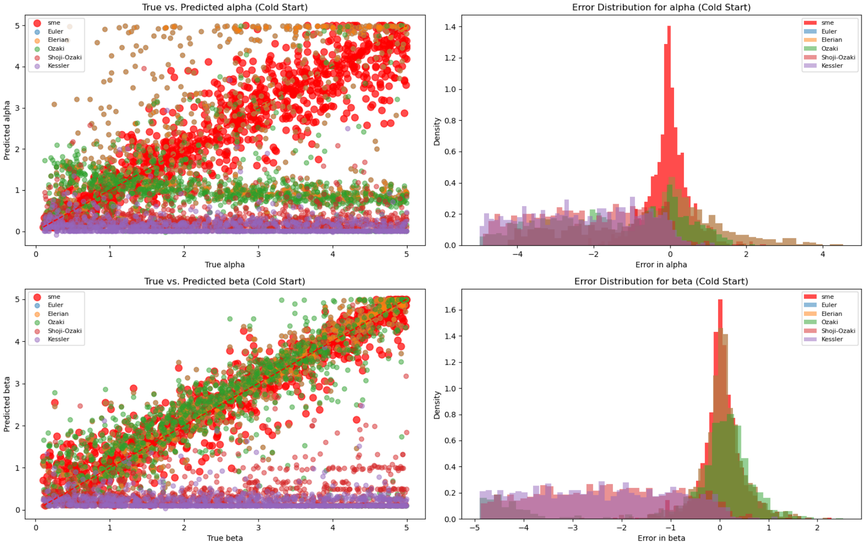
$$\boxed{r_{t+\Delta t} = r_t + (\alpha + \beta X_t)\Delta t + \sigma r_t^{\gamma}\sqrt{\Delta t} \cdot Z_t}$$

- $Z_t \sim \mathcal{N}(0, 1)$: Gaussian noise

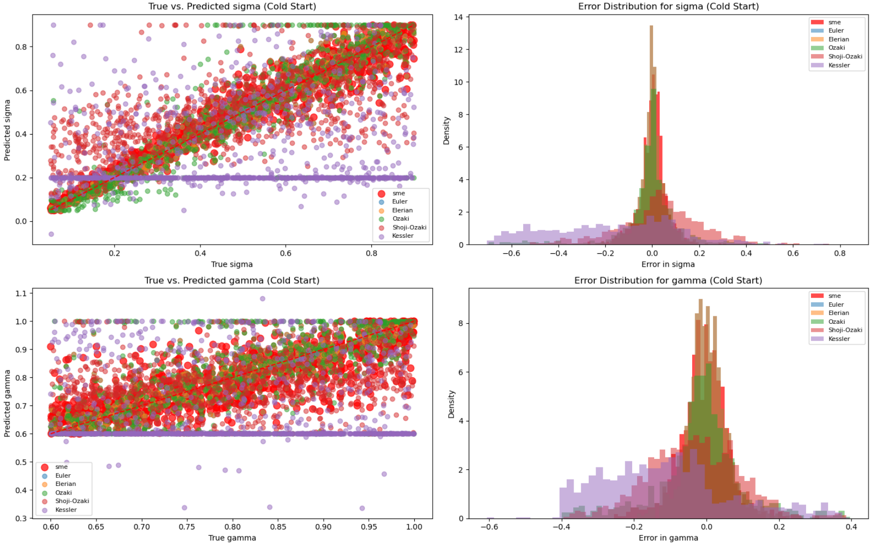
Simulation Setup:

- Time step: $\Delta t = \frac{1}{252}$
- Initial value: $X_0 = 0.05$
- Parameters:
  - $\alpha, \beta \in [0.1, 5.0]$
  - $\sigma \in [0.05, 0.9]$
  - $\gamma \in [0.6, 1.0]$

Cold Start (Fixed Guess) - Graphs (1/2)



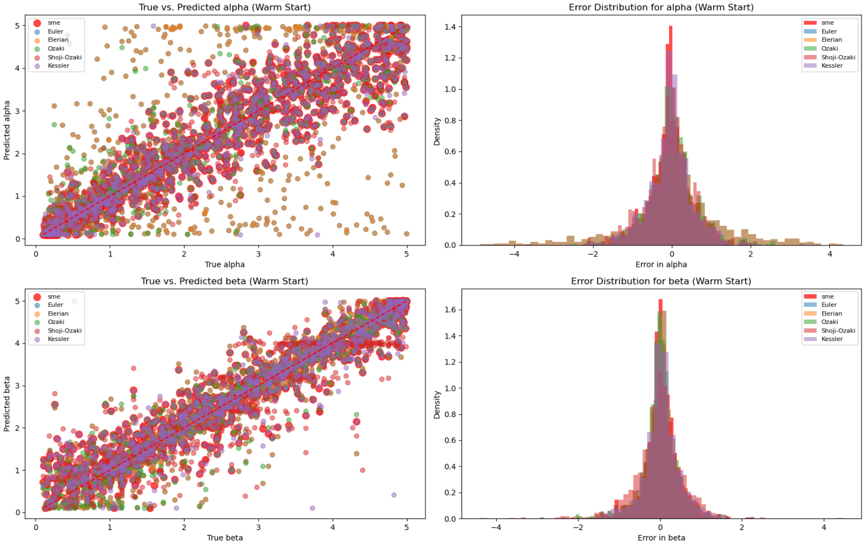
Cold Start (Fixed Guess) - Graphs (2/2)



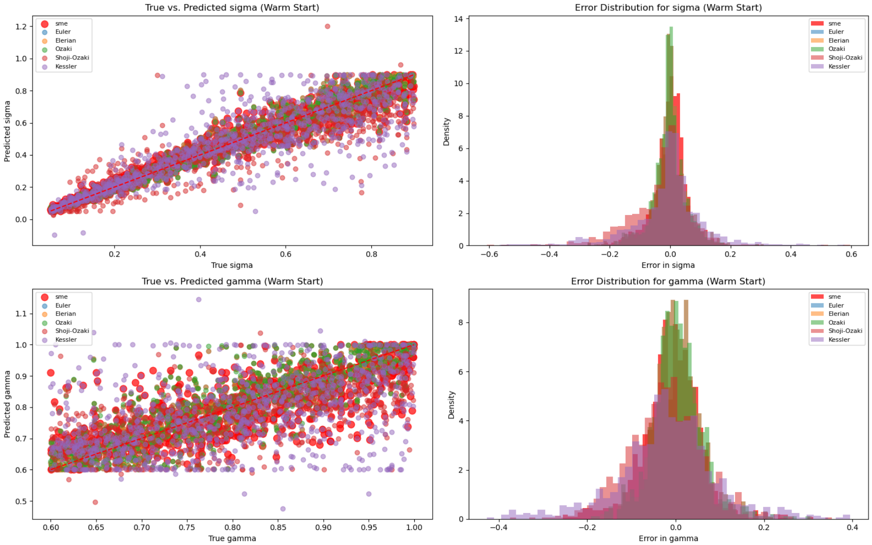
Cold Start (Fixed Guess) – Bias, Variance, and MSE Comparison $\alpha = 1, \beta = 1, \sigma = 0.3, \gamma = 0.7$

Method	Parameter	Bias	Variance	MSE
SME	alpha	0.0412	0.0583	0.0600
SME	beta	0.0993	0.1276	0.1375
SME	sigma	0.0199	0.0003	0.0007
SME	gamma	0.0015	0.0023	0.0023
Euler	alpha	0.2209	0.3941	0.4429
Euler	beta	-0.1011	0.2244	0.2346
Euler	sigma	-0.0009	0.0003	0.0003
Euler	gamma	0.0123	0.0064	0.0066
Elerian	alpha	0.2209	0.3941	0.4429
Elerian	beta	-0.1011	0.2244	0.2346
Elerian	sigma	-0.0009	0.0003	0.0003
Elerian	gamma	0.0123	0.0064	0.0066
Ozaki	alpha	0.2103	0.3652	0.4095
Ozaki	beta	-0.1002	0.2105	0.2205
Ozaki	sigma	-0.0018	0.0004	0.0004
Ozaki	gamma	0.0139	0.0065	0.0067
Shoji-Ozaki	alpha	-0.7379	0.0689	0.6133
Shoji-Ozaki	beta	-0.7062	0.0614	0.5602
Shoji-Ozaki	sigma	0.0261	0.0017	0.0024
Shoji-Ozaki	gamma	-0.0496	0.0048	0.0072
Kessler	alpha	-0.7537	0.1271	0.6952
Kessler	beta	-0.7184	0.0974	0.6135
Kessler	sigma	0.0138	0.0303	0.0305
Kessler	gamma	-0.0100	0.0182	0.0183

Warm Start (SME Initial Guess) - Graphs (1/2)



Warm Start (SME Initial Guess) - Graphs (2/2)



Warm Start (SME Initial Guess)– Bias, Variance, and MSE Comparison $\alpha = 1, \beta = 1, \sigma = 0.3, \gamma = 0.7$

Method	Parameter	Bias	Variance	MSE
SME	α	0.0412	0.0583	0.0600
SME	β	0.0993	0.1276	0.1375
SME	σ	0.0199	0.0003	0.0007
SME	γ	0.0015	0.0023	0.0023
Euler	α	0.1554	0.2351	0.2593
Euler	β	-0.0537	0.1725	0.1753
Euler	σ	-0.0007	0.0003	0.0003
Euler	γ	0.0129	0.0064	0.0065
Elerian	α	0.1554	0.2351	0.2593
Elerian	β	-0.0537	0.1725	0.1753
Elerian	σ	-0.0007	0.0003	0.0003
Elerian	γ	0.0129	0.0064	0.0065
Ozaki	α	0.0769	0.1462	0.1521
Ozaki	β	-0.0081	0.1481	0.1482
Ozaki	σ	-0.0018	0.0003	0.0003
Ozaki	γ	0.0142	0.0063	0.0065
Shoji-Ozaki	α	0.0322	0.1098	0.1109
Shoji-Ozaki	β	0.0854	0.1608	0.1681
Shoji-Ozaki	σ	0.0033	0.0004	0.0005
Shoji-Ozaki	γ	0.0123	0.0034	0.0036
Kessler	α	0.0441	0.0863	0.0882
Kessler	β	0.1036	0.1572	0.1679
Kessler	σ	0.0152	0.0074	0.0077
Kessler	γ	0.0064	0.0081	0.0082

References

- Chan, K. C. et al. (1992). "An Empirical Comparison of Alternative Models of the Short-Term Interest Rate". In: *The Journal of Finance* 47.3, pp. 1209–1227.
- Gourieroux, Christian, Alain Monfort, and Eric Renault (1993). "Indirect Inference". In: *Journal of Applied Econometrics* 8.S1. Special Issue: Econometrics of Simulation-Based Models, S85–S118.
- Jiang, Ruoxi, Peter Y. Lu, and Rebecca Willett (2024). *Embed and Emulate: Contrastive representations for simulation-based inference*. arXiv: 2409.18402 [cs.LG]. URL: https://arxiv.org/abs/2409.18402.
- Kirkby, J.L. et al. (2024). "pymle: A Python Package for Maximum Likelihood Estimation and Simulation of Stochastic Differential Equations". In: *Journal of Statistical Software*. Forthcoming. DOI: 10.18637/jss.v000.i00. URL: https://ssrn.com/abstract=4826948.
- McFadden, Daniel (Aug. 1987). *A Method of Simulated Moments for Estimation of Discrete Response Models Without Numerical*