Signature-Based Generative Models for Time Series

1 Background: Path Signatures

Let $X : [0, T] \to \mathbb{R}^d$ be a continuous path of finite variation. The (*truncated*) signature of X over $[0, T]$ to level m, denoted $S^{(m)}(X)_{0,T} \in \bigoplus_{k=0}^{m} (\mathbb{R}^d)^{\otimes k}$, collects the iterated integrals

$$S^{(k)}(X)_{0,T}^{i_1,\dots,i_k} = \int_{0<t_1<\cdots<t_k<T} dX_{t_k}^{i_k} \cdots dX_{t_1}^{i_1}, \qquad k = 1, 2, \dots, m.$$

In practice, for a one-dimensional series S_t we embed time and value to form a two-dimensional path $Z_t = (t, S_t)$ before computing the signature. Key properties used here: (i) Chen's identity for concatenation, (ii) faithfulness/uniqueness (up to tree-like equivalence), and (iii) *expected signatures* characterize laws, which motivates the linear signature MMD used for evaluation.

Notation. We use sliding windows of length L (lookback) and forward segment length F. For a window ending at index t, write $W_t = Z_{[t-L,\,t]}$ and $s_t = S^{(m)}(W_t) \in \mathbb{R}^{d(m)}$ for the truncated signature feature.

2 Model 1: Path-wise Signature Bootstrap

2.1 Idea

Build an empirical library of *cause/effect* pairs

$$\mathcal{D} = \{(s_t,\, p_t)\}, \quad s_t = S^{(m)}(Z_{[t-L,\,t]}), \quad p_t = S_{t+1:t+F} - S_t,$$

i.e., store the signature of each lookback window and the corresponding future *relative* segment to ensure continuity when stitching. At generation time, compute the current lookback signature s_{gen}, find its k nearest neighbors in $\mathcal{D}$, and *sample* one of the stored future segments to append.

2.2 Pseudo-code

Algorithm 1 Path-wise Signature Bootstrap

Input: historical paths $\mathcal{R}$, lookback L, forward F, level m, k
Library creation:
$\mathcal{D} \leftarrow \emptyset$
for $R \in \mathcal{R}$ **do**
  for $t = L, \ldots, |R| - F$ **do**
    $s_t \leftarrow S^{(m)}\big((\tau, R_\tau)_{\tau=t-L}^{t}\big)$
    $p_t \leftarrow (R_{t+1}, \ldots, R_{t+F}) - R_t$
    $\mathcal{D} \leftarrow \mathcal{D} \cup \{(s_t, p_t)\}$
  end for
end for
Generation: given seed path $R^{\text{seed}}_{0:L}$
$R^{\text{gen}} \leftarrow R^{\text{seed}}$
while length$(R^{\text{gen}}) <$ target **do**
  $s_{\text{gen}} \leftarrow S^{(m)}$ of last $L+1$ points
  $N_k \leftarrow k$-NN of s_{gen} in $\mathcal{D}$
  Sample $(s_j, p_j) \in N_k$ uniformly (or softmax by distance)
  Append p_j shifted by current level: $R^{\text{gen}} \leftarrow R^{\text{gen}} \cup \big(R^{\text{gen}}_{-1} + p_j\big)$
end while

Remarks. Using time-normalized windows (time reparameterized to $[0, 1]$) stabilizes the signature features; log-signatures may further de-correlate components. Soft neighbor sampling (e.g., softmax on distances) reduces jumps.

3 Model 2: Hybrid Drift + Signature Residual

3.1 Idea

Decompose one-step dynamics into a simple parametric drift plus a nonparametric innovation sampled from a signature-conditioned library. With $X_t = \log S_t$ and $\Delta X_t = X_{t+1} - X_t$ (step Δt), learn a mean-reverting drift $\mu_\theta(x) = c\,x$ by least squares on residuals, then build a library of residuals conditioned on window signatures.

3.2 Training

$$\text{Given } \{X_t\}, \quad \text{solve} \quad \min_c \sum_t \Big(\Delta X_t - \mu_c(X_t)\,\Delta t\Big)^2 + \lambda c^2, \qquad \mu_c(x) = c\,x.$$

Then, for each $t \geq L$, define the residual $r_t := \Delta X_t - \mu_c(X_t)\,\Delta t$ and store pairs (s_t, r_t) with $s_t = S^{(m)}(W_t)$ in a residual library $\mathcal{R}$.

3.3 Generation (Euler step with resampled residual)

$$X_{t+1} \;=\; X_t + \mu_c(X_t)\,\Delta t + \widetilde{r}_t, \qquad \widetilde{r}_t \sim \text{Empirical}\big(\{r_j : (s_j, r_j) \in N_k(s_t)\}\big).$$

Pseudocode mirrors Model 1, except the future segment is a scalar residual added per step, not a multi-step block. Soft k-NN sampling is recommended.

Model 2 Pseudocode

Algorithm 2 Hybrid Drift + Signature Residual: Training

Input: detrended log-paths $\{\{X_t^{(n)}\}_t\}_{n=1}^N$, lookback L, step Δt, signature level m, ridge λ
Build one-step dataset
$\mathcal{T} \leftarrow \emptyset$
for $n = 1$ to N **do**
  for $t = L$ to $T_n - 1$ **do**
    $W_t \leftarrow (X_{t-L}^{(n)}, \dots, X_t^{(n)})$
    $s_t \leftarrow S^{(m)}\big((\tau, W_t(\tau))_{\tau \in [0,1]}\big)$ ▷ time normalized to $[0, 1]$
    $y_t \leftarrow X_t^{(n)}, \quad \Delta X_t \leftarrow X_{t+1}^{(n)} - X_t^{(n)}$
    $\mathcal{T} \leftarrow \mathcal{T} \cup \{(s_t, y_t, \Delta X_t)\}$
  end for
end for
Fit linear mean-reversion drift $\mu_c(x) = c\,x$:

$$c^\star \in \arg\min_c \sum_{(s_t, y_t, \Delta X_t) \in \mathcal{T}} \big(\Delta X_t - c\, y_t\, \Delta t\big)^2 + \lambda c^2$$

Build residual library $\mathcal{R}$:

$$r_t \leftarrow \Delta X_t - c^\star\, y_t\, \Delta t, \qquad \mathcal{R} \leftarrow \mathcal{R} \cup \{(s_t, r_t)\}$$

Output: drift coefficient $c^\star$, residual library $\mathcal{R} = \{(s_t, r_t)\}$

Algorithm 3 Hybrid Drift + Signature Residual: Generation

Input: seed log-path $(X_0, \dots, X_L)$, horizon T, step Δt, level m, k-NN, temperature $\tau > 0$, drift $c^\star$, residual library $\mathcal{R}$
for $t = L$ to $T - 1$ **do**
  $W_t \leftarrow (X_{t-L}, \dots, X_t)$
  $s_t \leftarrow S^{(m)}\big((\tau, W_t(\tau))_{\tau \in [0,1]}\big)$
  Find k nearest neighbors $N_k(s_t) \subset \mathcal{R}$ by Euclidean distance in signature space
  Compute weights $w_j \propto \exp\big(-\operatorname{dist}(s_t, s_j)/\tau\big)$ for $(s_j, r_j) \in N_k(s_t)$
  Sample residual $\widetilde{r}_t$ from $N_k(s_t)$ with probabilities $\{w_j\}$
  $X_{t+1} \leftarrow X_t + c^\star X_t \Delta t + \widetilde{r}_t$
end for
Output: generated path $(X_0, \dots, X_T)$ (convert to levels if needed: $S_t = e^{X_t}$)

4 Model 3: Kernel Ridge Regression (KRR) in Signature Space

4.1 Feature construction and targets

Work on $X_t = \log S_t$. For each window W_t (length L) compute $s_t = S^{(m)}(W_t) \in \mathbb{R}^{d(m)}$ and define two targets:

$$y_t^{(\mu)} = \frac{X_{t+1} - X_t}{\Delta t}, \qquad y_t^{(\log \sigma)} = \log\Big(\frac{\operatorname{std}(\{\Delta X \text{ in } W_t\})}{\sqrt{\Delta t}} + \varepsilon\Big).$$

Stacking rows gives the signature design matrix $S \in \mathbb{R}^{N \times d(m)}$.

4.2 Training (linear kernel in signature space)

Form the Gram matrix $K = SS^\top \in \mathbb{R}^{N\times N}$ and solve two ridge systems:

$$\boldsymbol{\alpha}_\mu = (K+\lambda I_N)^{-1}\boldsymbol{y}_\mu, \qquad \boldsymbol{\alpha}_{\log\sigma} = (K+\lambda I_N)^{-1}\boldsymbol{y}_{\log\sigma}.$$

For a new window with signature s_{new}, predict via the kernel trick

$$\widehat{\mu} = (Ss_{\text{new}})^\top\boldsymbol{\alpha}_\mu, \qquad \widehat{\log\sigma} = (Ss_{\text{new}})^\top\boldsymbol{\alpha}_{\log\sigma}, \quad \widehat{\sigma} = e^{\widehat{\log\sigma}}.$$

4.3 Generation (Euler–Maruyama)

$$X_{t+1} \;=\; X_t + \widehat{\mu}\,\Delta t + \widehat{\sigma}\,\Delta W_t, \quad \Delta W_t \sim \mathcal{N}(0,\Delta t).$$

Repeat with a rolling window to refresh s_{new} at each step.

Model 3 Pseudocode

Algorithm 4 KRR in Signature Space: Training (drift and log-vol)

Input: training log-paths $\{\{X_t^{(n)}\}_t\}_{n=1}^N$, lookback L, step Δt, level m, ridge λ, floor $\varepsilon > 0$
Build windowed signature dataset
$S \leftarrow [\,]$ ▷ design matrix rows
$\boldsymbol{y}_\mu \leftarrow [\,]$, $\boldsymbol{y}_{\log\sigma} \leftarrow [\,]$
for $n = 1$ to N **do**
  for $t = L$ to $T_n - 1$ **do**
    $W_t \leftarrow (X_{t-L}^{(n)}, \dots, X_t^{(n)})$
    $s_t \leftarrow S^{(m)}\big((\tau, W_t(\tau))_{\tau\in[0,1]}\big)$
    $\Delta X_t \leftarrow X_{t+1}^{(n)} - X_t^{(n)}$
    $y_t^{(\mu)} \leftarrow \Delta X_t/\Delta t$
    $\sigma_t \leftarrow \max\big(\text{std}(\{\Delta X \text{ inside } W_t\})/\sqrt{\Delta t},\ \varepsilon\big)$
    $y_t^{(\log\sigma)} \leftarrow \log\sigma_t$
    Append row s_t to S; append $y_t^{(\mu)}$ to $\boldsymbol{y}_\mu$; append $y_t^{(\log\sigma)}$ to $\boldsymbol{y}_{\log\sigma}$
  end for
end for
(Optional) feature scaling: center/scale columns of S to get Z
Kernel (linear in signature space): $K \leftarrow ZZ^\top$
Solve

$$\boldsymbol{\alpha}_\mu = (K+\lambda I)^{-1}\boldsymbol{y}_\mu, \qquad \boldsymbol{\alpha}_{\log\sigma} = (K+\lambda I)^{-1}\boldsymbol{y}_{\log\sigma}.$$

Output: $(\boldsymbol{\alpha}_\mu, \boldsymbol{\alpha}_{\log\sigma}, Z)$ and scaling stats for signatures

Algorithm 5 KRR in Signature Space: Generation (Euler–Maruyama)

Input: seed log-path $(X_0, \ldots, X_L)$, horizon T, step Δt, level m, training data $(\boldsymbol{\alpha}_\mu, \boldsymbol{\alpha}_{\log\sigma}, Z)$, signature scaling stats
for $t = L$ to $T-1$ **do**
  $W_t \leftarrow (X_{t-L}, \ldots, X_t)$
  $s_t \leftarrow S^{(m)}\big((\tau, W_t(\tau))_{\tau\in[0,1]}\big)$
  Standardize s_t with saved stats to get z_t; $k_t \leftarrow Z z_t$ ▷ k_t is kernel vector
  $\widehat{\mu}_t \leftarrow k_t^\top \boldsymbol{\alpha}_\mu$
  $\widehat{\log\sigma}_t \leftarrow k_t^\top \boldsymbol{\alpha}_{\log\sigma}$; $\widehat{\sigma}_t \leftarrow \exp(\widehat{\log\sigma}_t)$
  Sample $\Delta W_t \sim \mathcal{N}(0, \Delta t)$
  $X_{t+1} \leftarrow X_t + \widehat{\mu}_t\,\Delta t + \widehat{\sigma}_t\,\Delta W_t$
end for
Output: generated path $(X_0, \ldots, X_T)$ (levels via $S_t = e^{X_t}$ if desired)

5 Model 4: Hybrid KRR–KNN Signature Generator

5.1 Idea

We model the log–return process $\{X_t\}_{t\geq 0}$ as

$$X_t = \mu\big(\mathcal{S}(X)\big)\,dt + \sigma_t\,dW_t,$$

where $\mu(\mathcal{S}(X))$ denotes a *drift term* predicted from the signature of the recent history of returns, and $\sigma_t dW_t$ represents the unpredictable residual. The drift is estimated *parametrically* by kernel ridge regression (KRR) in the signature feature space, while the residual innovation is drawn *nonparametrically* from a signature–conditioned k–nearest–neighbor (KNN) library. This combines the smooth, mean behavior learned from KRR with the distributional realism of the nonparametric residuals.

Data representation. For each asset, let the log–price process be $Y_t = \log S_t$ and the discrete log–return at step Δt be

$$X_t = Y_t - Y_{t-\Delta t}.$$

Define a sliding window of the past L returns as

$$W_t = (X_{t-L+1}, X_{t-L+2}, \ldots, X_t),$$

and embed it together with normalized time into a two–dimensional path

$$Z_t(\tau) = \big(\tau,\, W_t(\tau)\big), \qquad \tau \in [0,1].$$

The truncated signature of this window to level m, $s_t = S^{(m)}(Z_t) \in \mathbb{R}^{d(m)}$, serves as the feature vector representing the local dynamics around time t.

5.2 Drift Estimation via Kernel Ridge Regression

Given training sequences of log–returns $\{X_t^{(n)}\}$, we construct the dataset

$$\mathcal{D}_\mu = \Big\{\big(s_t,\, y_t^{(\mu)}\big) \;:\; y_t^{(\mu)} = \tfrac{X_{t+1}}{\Delta t}\Big\}.$$

Stacking features row–wise yields a design matrix $S \in \mathbb{R}^{N\times d(m)}$ and response vector $\boldsymbol{y}_\mu$. With a linear kernel $K = SS^\top$, the KRR estimator solves

$$(K + \lambda I_N)\boldsymbol{\alpha}_\mu = \boldsymbol{y}_\mu,$$

where $\lambda > 0$ is a ridge regularization parameter. For a new window with signature s_{new}, the predicted drift is

$$\widehat{\mu} = (S s_{\text{new}})^{\top} \boldsymbol{\alpha}_{\mu}.$$

5.3 Residual Library Construction

For each training window (s_t, X_{t+1}), define the empirical residual as

$$r_t = X_{t+1} - \widehat{\mu}(s_t)\,\Delta t,$$

and store the pairs (s_t, r_t) into a residual library $\mathcal{R} = \{(s_t, r_t)\}_{t=1}^{N}$. The residuals capture the high–frequency innovations that are not explained by the parametric drift term.

5.4 Path Generation

Starting from a seed sequence of L past returns $(X_0, \ldots, X_{L-1})$, we iteratively generate new steps.

At each step t:

1. Form the latest window W_t of length L and compute its signature $s_t = S^{(m)}(Z_t)$.
2. Predict the drift $\widehat{\mu}_t$ via the trained KRR: $\widehat{\mu}_t = (S s_t)^{\top} \boldsymbol{\alpha}_{\mu}$.
3. Find k nearest neighbors of s_t in $\mathcal{R}$ under Euclidean distance in signature space, denoted $N_k(s_t)$.
4. Sample a residual $\tilde{r}_t$ from $\{r_j : (s_j, r_j) \in N_k(s_t)\}$ with probability proportional to $\exp(-\text{dist}(s_t, s_j)/\tau)$, where τ is a temperature parameter.
5. Update the next log–return by

$$X_{t+1} = \widehat{\mu}_t\,\Delta t + \tilde{r}_t.$$

Aggregating $\{X_t\}$ yields a generated log–return path, and the synthetic price path is obtained by exponential accumulation:

$$S_t = S_0 \exp\Big(\sum_{u=1}^{t} X_u\Big).$$

Model 4 Pseudocode

Algorithm 6 Hybrid KRR–KNN Signature Generator: Training

Input: training log–returns $\{\{X_t^{(n)}\}_t\}_{n=1}^N$, lookback L, step Δt, signature level m, ridge λ
Initialize empty feature matrix S and target vector $\boldsymbol{y}_\mu$
for $n = 1$ to N **do**
    for $t = L$ to $T_n - 1$ **do**
        $W_t \leftarrow (X_{t-L}^{(n)}, \ldots, X_t^{(n)})$
        $Z_t(\tau) \leftarrow (\tau, W_t(\tau))_{\tau \in [0,1]}$
        $s_t \leftarrow S^{(m)}(Z_t)$
        $y_t^{(\mu)} \leftarrow X_{t+1}^{(n)} / \Delta t$
        Append row s_t to S, append $y_t^{(\mu)}$ to $\boldsymbol{y}_\mu$
    end for
end for
Train KRR drift: $\boldsymbol{\alpha}_\mu = (SS^\top + \lambda I)^{-1} \boldsymbol{y}_\mu$
Compute residual library:

$$r_t \leftarrow X_{t+1}^{(n)} - \big((Ss_t)^\top \boldsymbol{\alpha}_\mu\big)\Delta t, \qquad \mathcal{R} \leftarrow \mathcal{R} \cup \{(s_t, r_t)\}$$

Output: $\boldsymbol{\alpha}_\mu$, feature matrix S, residual library $\mathcal{R}$

Algorithm 7 Hybrid KRR–KNN Signature Generator: Generation

Input: seed log–return path $(X_0, \ldots, X_{L-1})$, horizon T, step Δt, level m, drift model $(\boldsymbol{\alpha}_\mu, S)$, residual library $\mathcal{R}$, KNN size k, temperature τ
for $t = L$ to $T - 1$ **do**
    $W_t \leftarrow (X_{t-L}, \ldots, X_t)$
    $Z_t(\tau) \leftarrow (\tau, W_t(\tau))_{\tau \in [0,1]}$
    $s_t \leftarrow S^{(m)}(Z_t)$
    $\widehat{\mu}_t \leftarrow (Ss_t)^\top \boldsymbol{\alpha}_\mu$
    Find $N_k(s_t) \subset \mathcal{R}$, the k nearest neighbors of s_t
    Compute weights $w_j \propto \exp(-\mathrm{dist}(s_t, s_j)/\tau)$
    Sample residual $\tilde{r}_t$ from $N_k(s_t)$ with probabilities $\{w_j\}$
    $X_{t+1} \leftarrow \widehat{\mu}_t \, \Delta t + \tilde{r}_t$
end for
Output: generated log–return path $\{X_t\}_{t=0}^T$ and reconstructed price path $S_t = S_0 \exp(\sum_{u \le t} X_u)$

6 Evaluation method

We benchmark generative quality along four complementary axes: (1) geometry via signature MMD, (2) marginal distributions (KS/Wasserstein and moments), (3) temporal dependence (ACF and volatility clustering), and (4) downstream ML utility.

Throughout, let $\mathcal{P}_{\text{real}}$ be the set of real paths and $\mathcal{P}_{\text{gen}}$ the set of generated paths. For a path $S = (S_0, \ldots, S_T)$ define log-returns $R_t = \log S_t - \log S_{t-1}$.

6.1 Signature MMD (linear kernel)

Fix a window length L and signature level m. For each path, slide a window of length L and form a two-channel path $Z(\tau) = (\tau, S(\tau))$ with the time channel normalized to $\tau \in [0, 1]$. Let

$s \in \mathbb{R}^{d(m)}$ denote the truncated signature (or log-signature) of the window.

Let $\{s_i^{(\text{real})}\}_{i=1}^{n_r}$ be the collection of window-signatures from $\mathcal{P}_{\text{real}}$ and $\{s_j^{(\text{gen})}\}_{j=1}^{n_g}$ from $\mathcal{P}_{\text{gen}}$. With the linear kernel $k(u,v) = u^\top v$, the MMD reduces to the distance between mean signatures:

$$\text{MMD}^2_{\text{sig}} = \| \bar{s}_{\text{real}} - \bar{s}_{\text{gen}} \|_2^2, \qquad \bar{s}_{\text{real}} = \frac{1}{n_r}\sum_{i=1}^{n_r} s_i^{(\text{real})}, \quad \bar{s}_{\text{gen}} = \frac{1}{n_g}\sum_{j=1}^{n_g} s_j^{(\text{gen})}.$$

6.2 Moment-based evaluation

Let $\mathcal{P}_{\text{real}} = \{S^{(i)}\}_{i=1}^{n_r}$ be the set of training (real) paths and $\mathcal{P}_{\text{gen}} = \{\tilde{S}^{(j)}\}_{j=1}^{n_g}$ the set of generated paths. For a path $S = (S_0, \dots, S_T)$ define log-returns $R_t = \log S_t - \log S_{t-1}$ for $t = 1, \dots, T$. From each path we extract a *moment vector* in $\mathbb{R}^4$:

$$\mu = \frac{1}{T}\sum_{t=1}^{T} R_t, \qquad s^2 = \frac{1}{T-1}\sum_{t=1}^{T}(R_t - \mu)^2,$$

$$\text{skew} = \frac{T}{(T-1)(T-2)} \frac{\sum_{t=1}^{T}(R_t-\mu)^3}{(s^2)^{3/2}},$$

$$\text{kurt}_{\text{ex}} = \frac{T(T+1)}{(T-1)(T-2)(T-3)} \frac{\sum_{t=1}^{T}(R_t-\mu)^4}{(s^2)^2} - \frac{3(T-1)^2}{(T-2)(T-3)}.$$

(We use the Fisher–Pearson bias-corrected skew and excess kurtosis.) Denote $m^{(i)} = (\mu_i, s_i^2, \text{skew}_i, \text{kurt}_{\text{ex},i})^\top$ for the i-th real path and $\tilde{m}^{(j)}$ for the j-th generated path.

We visualize $\{m^{(i)}\}_{i=1}^{n_r}$ and $\{\tilde{m}^{(j)}\}_{j=1}^{n_g}$ via pairwise scatterplots (off-diagonal) and KDEs (diagonal). Overlapping colored clouds indicate similar marginal distributions and similar low-dimensional dependencies between moments.

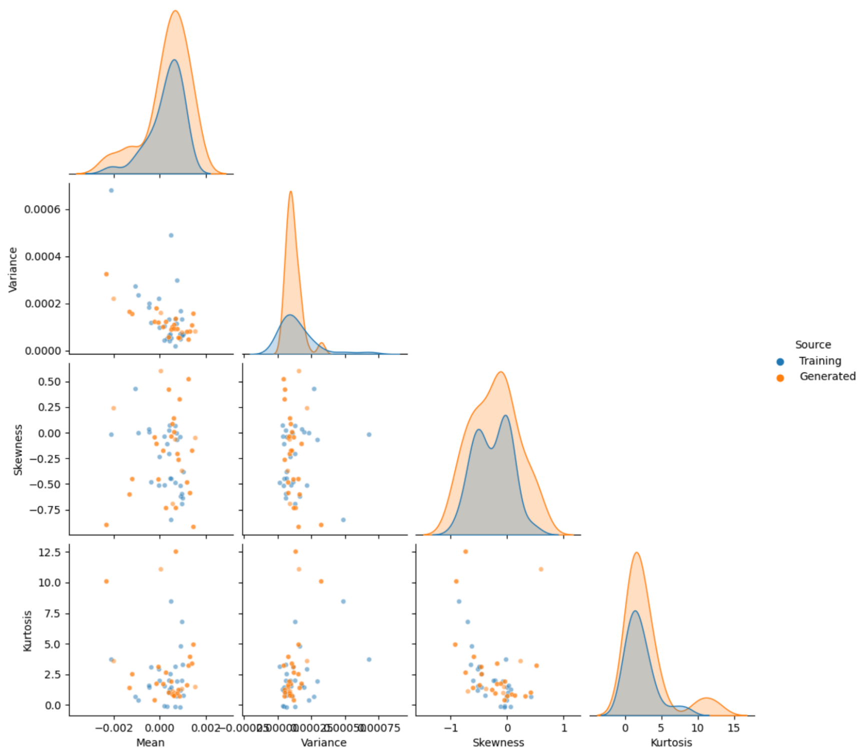


Figure 1: Pair plot of per-path moments (Mean, Variance, Skewness, Kurtosis): blue = training; orange = generated. Off-diagonal panels show joint scatter; diagonal panels show KDE marginals. Large overlap suggests good generative fidelity in these summary statistics.

6.3 Kolmogorov–Smirnov (KS) tests on the four moments

For each coordinate $k \in \{\text{mean}, \text{variance}, \text{skew}, \text{kurt}\}$, let $\{x_i\}_{i=1}^{n_r}$ be the real samples and $\{y_j\}_{j=1}^{n_g}$ the generated samples of that coordinate. The two-sample KS statistic is the supremum of the absolute difference of empirical CDFs:

$$D_{n_r,n_g} \;=\; \sup_{x\in\mathbb{R}} \left|\hat{F}_{n_r}(x) - \hat{G}_{n_g}(x)\right|, \qquad \hat{F}_{n_r}(x) = \frac{1}{n_r}\sum_{i=1}^{n_r} \mathbf{1}\{x_i \le x\},\;\; \hat{G}_{n_g}(x) = \frac{1}{n_g}\sum_{j=1}^{n_g} \mathbf{1}\{y_j \le x\}.$$

Under the null $H_0 : F = G$ (same continuous distribution) and independent samples, the asymptotic p-value is obtained from the Kolmogorov distribution of $D_{n_r,n_g}\sqrt{\frac{n_r n_g}{n_r+n_g}}$.

Assumptions and remarks. (i) Samples are i.i.d. within each group and independent across groups; (ii) the null distribution is continuous (ties require care); (iii) the test is most sensitive near the center and less so in the tails. We report four p-values (one per coordinate); large p-values ($\gtrsim 0.05$) indicate no evidence of a marginal difference for that moment.

6.4 MMD on moment vectors: linear and RBF kernels

Let $X = \{m^{(i)}\}_{i=1}^{n_r}$ and $Y = \{\tilde{m}^{(j)}\}_{j=1}^{n_g}$, with $m^{(i)}, \tilde{m}^{(j)} \in \mathbb{R}^4$. For a positive-definite kernel k, the (unbiased) empirical MMD^2 is

$$\widehat{\mathrm{MMD}}_k^2(X,Y) = \frac{1}{n_r(n_r-1)}\sum_{i\neq i'} k\big(m^{(i)}, m^{(i')}\big) + \frac{1}{n_g(n_g-1)}\sum_{j\neq j'} k\big(\tilde{m}^{(j)}, \tilde{m}^{(j')}\big) - \frac{2}{n_r n_g}\sum_{i=1}^{n_r}\sum_{j=1}^{n_g} k\big(m^{(i)}, \tilde{m}^{(j)}\big).$$

Linear kernel (mean-only check). With $k(u,v) = u^\top v$ one obtains

$$\widehat{\mathrm{MMD}}_{\mathrm{lin}}^2(X,Y) \;=\; \left\|\bar{m}_{\mathrm{real}} - \bar{m}_{\mathrm{gen}}\right\|_2^2, \qquad \bar{m}_{\mathrm{real}} = \frac{1}{n_r}\sum_{i=1}^{n_r} m^{(i)}, \quad \bar{m}_{\mathrm{gen}} = \frac{1}{n_g}\sum_{j=1}^{n_g} \tilde{m}^{(j)}.$$

This checks only differences in the *mean* moment vector.

Gaussian RBF kernel (global distributional check). With

$$k_\sigma(u,v) = \exp\Big(-\tfrac{\|u-v\|_2^2}{2\sigma^2}\Big),$$

$\widehat{\mathrm{MMD}}_{k_\sigma}^2$ captures broader distributional differences (location, scale, and higher-order structure). The bandwidth $\sigma > 0$ controls sensitivity: small σ emphasizes local differences; large σ emphasizes global shifts. In practice we use the median heuristic for σ, i.e. half the median pairwise distance among $\{m^{(i)}\} \cup \{\tilde{m}^{(j)}\}$.

7 Experiment 1: Multi paths and one path comparison on S&P 500 (2010–2024)

7.1 Data and Preprocessing

We source daily close prices via `yfinance` for S&P 500 (`^GSPC`) over `2010-01-01` to `2025-01-01`.

We consider two training datasets:

1. **Yearly paths (*multi*)**: for each year $y \in \{2010, \dots, 2024\}$ we extract the first 250 trading days of `^GSPC` as one path $S^{(y)}$, normalize by $S_0^{(y)}$ to obtain a price index $I_t^{(y)} = S_t^{(y)}/S_0^{(y)}$, then remove a linear trend on the fixed grid $t = 0, \dots, 249$ via least squares: $I_t^{(y)} = \hat{a}^{(y)}t + \hat{b}^{(y)} + R_t^{(y)}$, keep residuals $R^{(y)}$, and store the trend line $\hat{T}_t^{(y)} = \hat{a}^{(y)}t + \hat{b}^{(y)}$ for re-adding after generation.

2. **Whole path (*one*)**: we take the entire `^GSPC` series up to 2024 end as one long path S, form the index $I_t = S_t/S_0$, detrend it linearly on its native grid to get residuals R_t and trend $\hat{T}_t$.

We compute per-window signatures on the 2D path (τ, X_τ) where τ is reparameterized to $[0, 1]$ within each window and X is the detrended index level (residual). Generation is performed in residual space and the saved linear trend is added back to produce final levels for evaluation.

7.2 Model and Training Setup

We use the *path-wise signature bootstrap* library: for each lookback window of length $L = 20$ and forward window $F = 5$, we record (s_t, p_t) where s_t is the truncated signature at level $m = 3$ (and in a second variant, the truncated *log-signature*), and $p_t = (X_{t+1}, \dots, X_{t+F}) - X_t$ is the relative future segment. At generation time, we roll a size-$L+1$ window on the evolving path, compute its signature, find $k = 10$ nearest neighbors in the library (Euclidean distance), sample one neighbor (uniform), and append the stored segment. For the multi dataset we generate 50 paths (seed = 1234); for the one dataset we generate 10 paths. MMD window = 15. All signatures are computed on time-normalized windows ($\tau \in [0, 1]$). Generation is performed in residual space; linear trends are re-added for evaluation.

7.3 Evaluation Metric

We report the *linear signature MMD* (window size 15, level $m = 3$), i.e., the squared ℓ_2 distance between mean (log-)signatures of sliding windows from real vs generated sets:

$$\mathrm{MMD}^2_{\mathrm{sig}} \;=\; \| \, \bar{s}_{\mathrm{real}} - \bar{s}_{\mathrm{gen}} \, \|_2^2 \, .$$

Lower is better.

7.4 Results

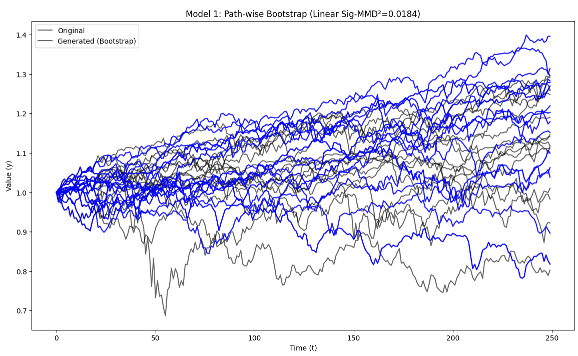
7.4.1 Model 1: Pathwise bootstrap

Configuration constants. Lookback $L = 20$, forward $F = 5$, level $m = 3$, neighbors $k = 10$, window for MMD = 15. Number of generated paths: multi = 50, one = 10.

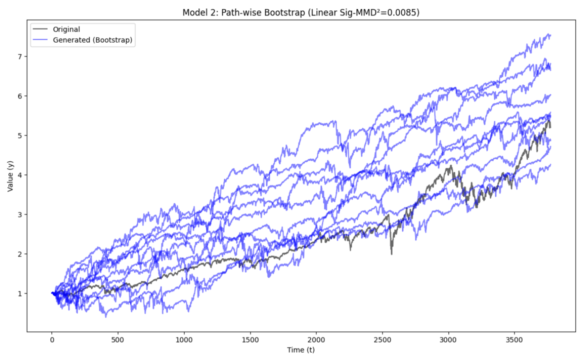
Table 1: Linear Signature MMD^2 across datasets and feature types (lower is better).

Dataset & Feature	Sig-MMD^2	Notes
Yearly paths (*multi*) + Signature	<0.018412>	$N_{\text{gen}} = 50$, seed $= 1234$
Whole path (*one*) + Signature	<0.008518>	$N_{\text{gen}} = 10$, seed unset
Yearly paths (*multi*) + Log-signature	<0.028589>	$N_{\text{gen}} = 50$, seed $= 1234$
Whole path (*one*) + Log-signature	<0.116951>	$N_{\text{gen}} = 10$, seed unset

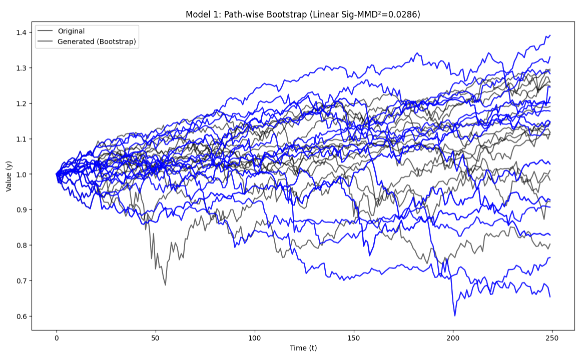
Qualitative samples. Figure 3 shows representative generated samples for each setting (residual paths with trend re-added to produce index levels).



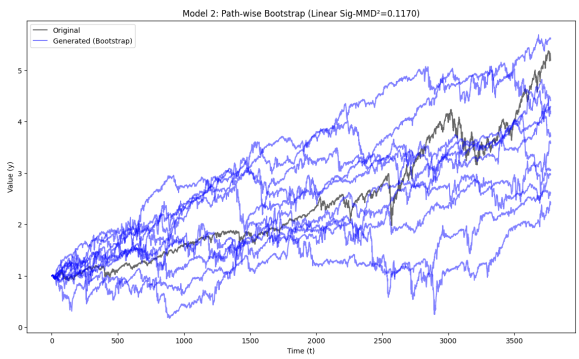
(a) Yearly paths + Signature (multi).



(b) Whole path + Signature (one).



(c) Yearly paths + Log-signature (multi).



(d) Whole path + Log-signature (one).

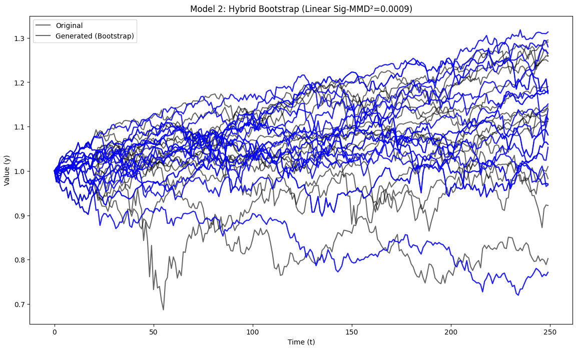
Figure 2: Path-wise bootstrap samples under four settings (trend added back).

7.4.2 Model 2: Hybrid bootstrap

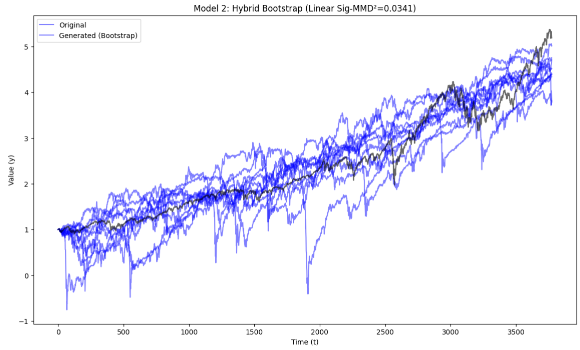
Table 2: Linear Signature MMD^2 across datasets and feature types.

Dataset & Feature	Sig-MMD^2	Notes
Yearly paths (*multi*) + Signature	<0.000867>	$N_{\text{gen}} = 50$, seed $= 1234$
Whole path (*one*) + Signature	<0.034129>	$N_{\text{gen}} = 10$, seed unset
Yearly paths (*multi*) + Log-signature	<0.004026>	$N_{\text{gen}} = 50$, seed $= 1234$
Whole path (*one*) + Log-signature	<0.046949>	$N_{\text{gen}} = 10$, seed unset

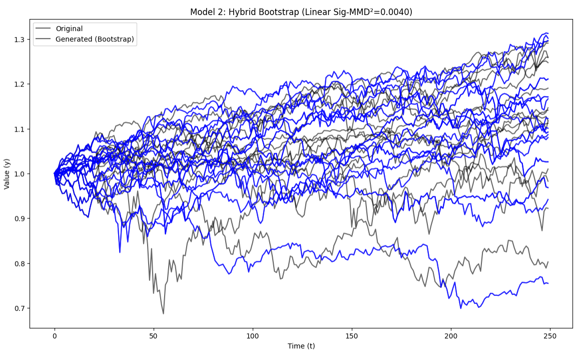
Qualitative samples. Figure 3 shows representative generated samples for each setting (residual paths with trend re-added to produce index levels).



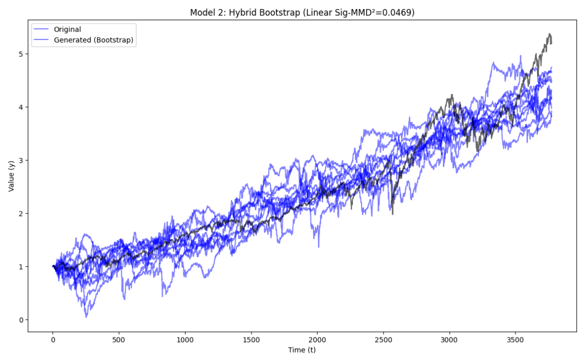
(a) Yearly paths + Signature (multi).



(b) Whole path + Signature (one).



(c) Yearly paths + Log-signature (multi).



(d) Whole path + Log-signature (one).

Figure 3: Hybrid bootstrap samples under four settings (trend added back).

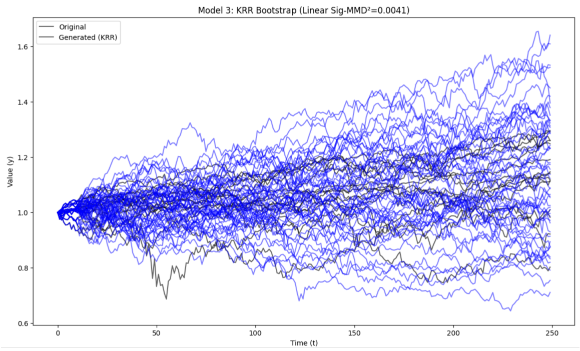
7.4.3 Model 3: Kernel Ridge Regression

Configuration constants. Lookback $L = 10$, signature level $m = 4$, linear kernel in signature space, ridge $\lambda = 2.0$, Euler–Maruyama with $\Delta W_t \sim \mathcal{N}(0, \Delta t)$, MMD window $= 15$. Number of generated paths: *multi* $= 50$, *one* $= 15$. (For the "one" + logsig run, $\widehat{\log \sigma}$ is clipped to the $[1, 99]$th percentiles as in code.)

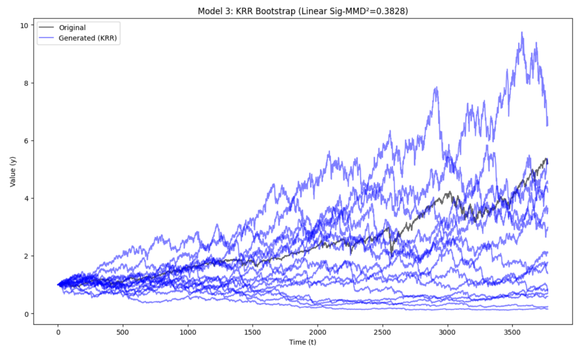
Table 3: Model 3 (KRR) — Linear Signature MMD^2 across datasets and feature types (lower is better).

Dataset & Feature	**Sig-MMD^2**	**Notes**
Yearly paths (*multi*) + Signature	`<0.004088>`	$N_{\text{gen}} = 50$, seed if set
Whole path (*one*) + Signature	`<0.382780>`	$N_{\text{gen}} = 15$, seed unset
Yearly paths (*multi*) + Log-signature	`<0.000223>`	$N_{\text{gen}} = 50$, seed if set
Whole path (*one*) + Log-signature	`<0.531072>`	$N_{\text{gen}} = 15$, $\widehat{\log \sigma}$ clipped

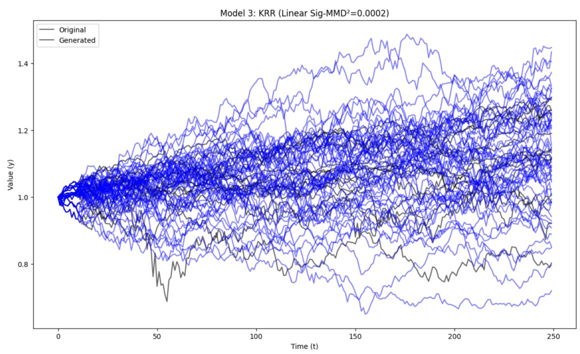
Qualitative samples. Figure 4 shows representative generated log-price paths under the four settings.



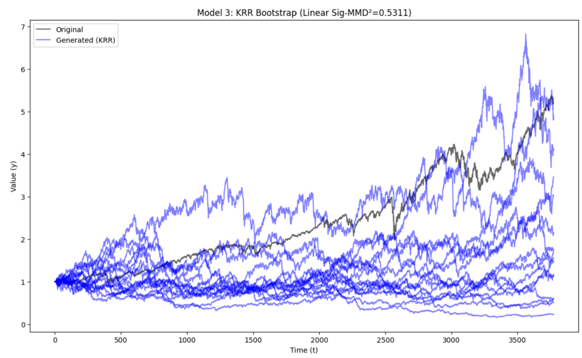
(a) Yearly paths + Signature (multi).



(b) Whole path + Signature (one).



(c) Yearly paths + Log-signature (multi).

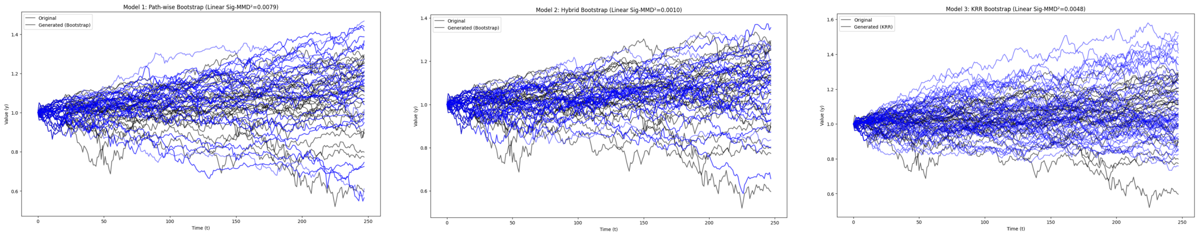


(d) Whole path + Log-signature (one).

Figure 4: Kernel Ridge Regression (signature–SDE) samples under four settings.

8 Experiment 2: Models evaluation on S&P 500 (1997–2024)

We evaluate three generators (`bootstrap`, `hybrid`, `KRR`) on S&P 500 yearly paths (1997–2024), detrended and normalized as described in Section 7.1. Each model produces a set of generated price paths which we compare to the training set using: (i) signature MMD (linear), (ii) KS p-values on per-path moments (mean, variance, skewness, kurtosis), and (iii) MMD on the 4-moment vectors with a Gaussian RBF kernel (bandwidth σ chosen by the median heuristic).



(a) `bootstrap` (b) `hybrid` (c) `KRR`

Figure 5: Pair plots of per-path moment vectors (Mean, Variance, Skewness, Kurtosis): blue = training, orange = generated. Overlap indicates distributional similarity; displacement or spread differences indicate mismatch.

Table 4: Experiment 2: signature and moment-MMD metrics. RBF MMD^2 uses a Gaussian kernel with bandwidth σ chosen by the median heuristic. Smaller is better.

Method	**sig MMD^2**	**linear MMD^2**	**RBF MMD^2**	$\boldsymbol{\sigma}$
`bootstrap`	7.9414×10^{-3}	3.4308×10^{-2}	2.0730×10^{-2}	6.1530×10^{-1}
`hybrid`	1.0370×10^{-3}	2.1179×10^{-3}	-2.1170×10^{-2}	5.4653×10^{-1}
`KRR`	4.7504×10^{-3}	9.4136×10^{-2}	1.9601×10^{-3}	4.8450×10^{-1}

Table 5: Experiment 2: two-sample KS p-values on per-path moments (higher is better; values $\gtrsim 0.05$ indicate no evidence of a marginal difference).

Method	**KS mean**	**KS var**	**KS skew**	**KS kurt**
`bootstrap`	1.3558×10^{-1}	2.6918×10^{-1}	9.5834×10^{-1}	3.2042×10^{-3}
`hybrid`	8.3495×10^{-1}	9.8285×10^{-1}	2.8189×10^{-1}	8.3495×10^{-1}
`KRR`	3.8140×10^{-1}	2.9686×10^{-1}	2.0981×10^{-1}	6.7546×10^{-2}

Notes. (i) Smaller MMD values indicate closer distributions. The linear MMD^2 reflects only the difference in the *mean* of the moment vectors; RBF MMD^2 captures broader distributional differences. (ii) KS entries are p-values; larger values ($\gtrsim 0.05$) suggest no evidence of marginal distributional difference for that moment. (iii) A small negative RBF MMD^2 (as seen for `hybrid`) can occur with the unbiased estimator due to finite-sample noise; it should be interpreted as effectively zero.

Takeaways. Across the three methods, `hybrid` exhibits the smallest linear and RBF MMD^2 (values near zero), with uniformly large KS p-values, indicating strong agreement with the training distribution of per-path moments. `bootstrap` shows a small but positive RBF MMD^2 and a low KS p-value for kurtosis (more tail mismatch), while `KRR` shows a larger linear MMD^2 (mean shift in moments) but very small RBF MMD^2, suggesting modest mean bias with otherwise close higher-order structure.