

# Fast Signature-Based Generative Models for Time Series

(Notes synthesizing code + PDF)

October 3, 2025

## 1 Background: Path Signatures

Let  $X : [0, T] \rightarrow \mathbb{R}^d$  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

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

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  $s_{\text{gen}}$ , 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

---

```

1: Input: historical paths  $\mathcal{R}$ , lookback  $L$ , forward  $F$ , level  $m$ ,  $k$ 
2: Library creation:
3:  $\mathcal{D} \leftarrow \emptyset$ 
4: for  $R \in \mathcal{R}$  do
5:   for  $t = L, \dots, |R| - F$  do
6:      $s_t \leftarrow S^{(m)}((\tau, R_\tau)_{\tau=t-L}^t)$ 
7:      $p_t \leftarrow (R_{t+1}, \dots, R_{t+F}) - R_t$ 
8:      $\mathcal{D} \leftarrow \mathcal{D} \cup \{(s_t, p_t)\}$ 
9:   end for
10: end for

11: Generation: given seed path  $R_{0:L}^{\text{seed}}$ 
12:  $R^{\text{gen}} \leftarrow R^{\text{seed}}$ 
13: while  $\text{length}(R^{\text{gen}}) < \text{target}$  do
14:    $s_{\text{gen}} \leftarrow S^{(m)}$  of last  $L+1$  points
15:    $N_k \leftarrow k\text{-NN}$  of  $s_{\text{gen}}$  in  $\mathcal{D}$ 
16:   Sample  $(s_j, p_j) \in N_k$  uniformly (or softmax by distance)
17:   Append  $p_j$  shifted by current level:  $R^{\text{gen}} \leftarrow R^{\text{gen}} \cup (R_{-1}^{\text{gen}} + p_j)$ 
18: 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  $\Delta t$ ), learn a mean-reverting drift  $\mu_\theta(x) = cx$  by least squares on residuals, then build a library of residuals conditioned on window signatures.

### 3.2 Training

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

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)

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

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

---

- 1: **Input:** detrended log-paths  $\{\{X_t^{(n)}\}_t\}_{n=1}^N$ , lookback  $L$ , step  $\Delta t$ , signature level  $m$ , ridge  $\lambda$
- 2: **Build one-step dataset**
- 3:  $\mathcal{T} \leftarrow \emptyset$
- 4: **for**  $n = 1$  to  $N$  **do**
- 5:   **for**  $t = L$  to  $T_n - 1$  **do**
- 6:      $W_t \leftarrow (X_{t-L}^{(n)}, \dots, X_t^{(n)})$
- 7:      $s_t \leftarrow S^{(m)}((\tau, W_t(\tau))_{\tau \in [0,1]})$   $\triangleright$  time normalized to  $[0, 1]$
- 8:      $y_t \leftarrow X_t^{(n)}, \quad \Delta X_t \leftarrow X_{t+1}^{(n)} - X_t^{(n)}$
- 9:      $\mathcal{T} \leftarrow \mathcal{T} \cup \{(s_t, y_t, \Delta X_t)\}$
- 10:   **end for**
- 11: **end for**
- 12: **Fit linear mean-reversion drift**  $\mu_c(x) = cx$ :

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

- 13: **Build residual library**  $\mathcal{R}$ :

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

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

---

### Algorithm 3 Hybrid Drift + Signature Residual: Generation

---

- 1: **Input:** seed log-path  $(X_0, \dots, X_L)$ , horizon  $T$ , step  $\Delta t$ , level  $m$ ,  $k$ -NN, temperature  $\tau > 0$ , drift  $c^*$ , residual library  $\mathcal{R}$
  - 2: **for**  $t = L$  to  $T - 1$  **do**
  - 3:    $W_t \leftarrow (X_{t-L}, \dots, X_t)$
  - 4:    $s_t \leftarrow S^{(m)}((\tau, W_t(\tau))_{\tau \in [0,1]})$
  - 5:   Find  $k$  nearest neighbors  $N_k(s_t) \subset \mathcal{R}$  by Euclidean distance in signature space
  - 6:   Compute weights  $w_j \propto \exp(-\text{dist}(s_t, s_j)/\tau)$  for  $(s_j, r_j) \in N_k(s_t)$
  - 7:   Sample residual  $\tilde{r}_t$  from  $N_k(s_t)$  with probabilities  $\{w_j\}$
  - 8:    $X_{t+1} \leftarrow X_t + c^* X_t \Delta t + \tilde{r}_t$
  - 9: **end for**
  - 10: **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:

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

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:

$$\alpha_\mu = (K + \lambda I_N)^{-1} \mathbf{y}_\mu, \quad \alpha_{\log \sigma} = (K + \lambda I_N)^{-1} \mathbf{y}_{\log \sigma}.$$

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

$$\hat{\mu} = (S s_{\text{new}})^\top \alpha_\mu, \quad \widehat{\log \sigma} = (S s_{\text{new}})^\top \alpha_{\log \sigma}, \quad \hat{\sigma} = e^{\widehat{\log \sigma}}.$$

## 4.3 Generation (Euler–Maruyama)

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

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

## Model 3 Pseudocode

---

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

---

- 1: **Input:** training log-paths  $\{\{X_t^{(n)}\}_t\}_{n=1}^N$ , lookback  $L$ , step  $\Delta t$ , level  $m$ , ridge  $\lambda$ , floor  $\varepsilon > 0$
  - 2: **Build windowed signature dataset**
  - 3:  $S \leftarrow []$   $\triangleright$  design matrix rows
  - 4:  $\mathbf{y}_\mu \leftarrow [], \mathbf{y}_{\log \sigma} \leftarrow []$
  - 5: **for**  $n = 1$  to  $N$  **do**
  - 6:   **for**  $t = L$  to  $T_n - 1$  **do**
  - 7:      $W_t \leftarrow (X_{t-L}^{(n)}, \dots, X_t^{(n)})$
  - 8:      $s_t \leftarrow S^{(m)}((\tau, W_t(\tau))_{\tau \in [0,1]})$
  - 9:      $\Delta X_t \leftarrow X_{t+1}^{(n)} - X_t^{(n)}$
  - 10:      $y_t^{(\mu)} \leftarrow \Delta X_t / \Delta t$
  - 11:      $\sigma_t \leftarrow \max(\text{std}(\{\Delta X \text{ inside } W_t\}) / \sqrt{\Delta t}, \varepsilon)$
  - 12:      $y_t^{(\log \sigma)} \leftarrow \log \sigma_t$
  - 13:     Append row  $s_t$  to  $S$ ; append  $y_t^{(\mu)}$  to  $\mathbf{y}_\mu$ ; append  $y_t^{(\log \sigma)}$  to  $\mathbf{y}_{\log \sigma}$
  - 14:   **end for**
  - 15: **end for**
  - 16: **(Optional) feature scaling:** center/scale columns of  $S$  to get  $Z$
  - 17: **Kernel (linear in signature space):**  $K \leftarrow ZZ^\top$
  - 18: Solve
 
$$\alpha_\mu = (K + \lambda I)^{-1} \mathbf{y}_\mu, \quad \alpha_{\log \sigma} = (K + \lambda I)^{-1} \mathbf{y}_{\log \sigma}.$$
  - 19: **Output:**  $(\alpha_\mu, \alpha_{\log \sigma}, Z)$  and scaling stats for signatures
-

---

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

---

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

---

## 5 Evaluation Protocol

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  $S = (S_0, \dots, S_T)$  define log-returns  $R_t = \log S_t - \log S_{t-1}$ .

### 5.1 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:

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

**Variants.** (i) *log-signature* features in place of signatures; (ii) *multi-scale* aggregation over window lengths  $\mathcal{L}$  and levels  $\mathcal{M}$ :

$$\text{MMD}_{\text{multi}}^2 = \sum_{L \in \mathcal{L}} \sum_{m \in \mathcal{M}} w_{L,m} \| \bar{s}_{\text{real}}^{(L,m)} - \bar{s}_{\text{gen}}^{(L,m)} \|_2^2, \quad w_{L,m} \geq 0, \quad \sum w_{L,m} = 1.$$

**Notes.** Match the number of windows (or reweight) to mitigate small-sample bias.

### 5.2 Distributional congruence (KS/Wasserstein and moments)

We compare (a) terminal levels  $S_T$ , (b) terminal log-returns  $G = \log(S_T/S_0)$ , and (c) pooled per-step log-returns  $\{R_t\}$ .

**Two-sample KS.** Let  $F_n$  and  $G_m$  be empirical CDFs of samples  $x_{1:n}$  and  $y_{1:m}$ . The KS statistic is

$$D_{n,m} = \sup_x |F_n(x) - G_m(x)|.$$

Lower  $D_{n,m}$  (higher p-value) indicates closer marginals.

**Wasserstein-1.** The one-dimensional  $W_1$  distance admits a quantile representation:

$$W_1(F, G) = \int_0^1 |F^{-1}(u) - G^{-1}(u)| du \approx \frac{1}{N} \sum_{i=1}^N |x_{(i)} - y_{(i)}|,$$

where  $x_{(i)}$  and  $y_{(i)}$  are order statistics (with interpolation if  $n \neq m$ ). Smaller  $W_1$  indicates closer distributions.

**Moment diagnostics.** Report (robust) moments for  $X \in \{S_T, G, R_t\}$ :

$$\text{mean} = \mathbb{E}[X], \quad \text{stdev} = \sqrt{\text{Var}(X)}, \quad \text{skew} = \frac{\mathbb{E}[(X - \mu)^3]}{\sigma^3}, \quad \text{kurt} = \frac{\mathbb{E}[(X - \mu)^4]}{\sigma^4}.$$

Optionally include robust analogues (median, MAD, trimmed moments).

### 5.3 Temporal dependence: ACF, squared-ACF, Ljung–Box

For a series  $x_1, \dots, x_n$  with mean  $\bar{x}$ , the sample autocorrelation at lag  $k$  is

$$\hat{\rho}_k = \frac{\sum_{t=1}^{n-k} (x_t - \bar{x})(x_{t+k} - \bar{x})}{\sum_{t=1}^n (x_t - \bar{x})^2}, \quad k = 1, 2, \dots$$

We compute mean ACF curves across paths for  $x_t = R_t$  (linear dependence) and for  $x_t = R_t^2$  (volatility clustering). Compare curves by an  $\ell_2$  gap:

$$\text{ACFGap}(h) = \left( \frac{1}{h} \sum_{k=1}^h (\hat{\rho}_k^{\text{real}} - \hat{\rho}_k^{\text{gen}})^2 \right)^{1/2},$$

and analogously for squared returns.

**Ljung–Box test (iid check).** For a chosen horizon  $h$ , define the Q-statistic on  $x_t$  as

$$Q(h) = n(n+2) \sum_{k=1}^h \frac{\hat{\rho}_k^2}{n-k},$$

which is asymptotically  $\chi_h^2$  under the iid null. We report  $(Q, p)$  on the concatenated *returns* and on *squared* returns to probe linear and second-order dependence, respectively. Higher p-values imply closer-to-iid behavior.

### 5.4 Downstream ML utility (task-based evaluation)

We quantify whether synthetic data help or harm learning under realistic splits.

**Tasks.** (1) One-step *regression* of log-return  $R_{t+1}$  (report MSE/RMSE, MAE), (2) *classification* of return sign  $\mathbb{I}\{R_{t+1} > 0\}$  (report AUC/accuracy), and (3) *volatility* forecasting (regress  $|R_{t+1}|$  or  $R_{t+1}^2$ ; report MSE).

**Regimes.** Let  $\mathcal{D}_{\text{real}}^{\text{train}}$  and  $\mathcal{D}_{\text{real}}^{\text{test}}$  be disjoint real splits, and  $\mathcal{D}_{\text{gen}}$  be generated samples matched in horizon and sampling. We compare:

$$\begin{aligned} \mathbf{R} : & \text{ train on } \mathcal{D}_{\text{real}}^{\text{train}}, \text{ test on } \mathcal{D}_{\text{real}}^{\text{test}}, \\ \mathbf{G} : & \text{ train on } \mathcal{D}_{\text{gen}}, \text{ test on } \mathcal{D}_{\text{real}}^{\text{test}}, \\ \mathbf{R+G} : & \text{ train on } \mathcal{D}_{\text{real}}^{\text{train}} \cup \mathcal{D}_{\text{gen}}, \text{ test on } \mathcal{D}_{\text{real}}^{\text{test}}. \end{aligned}$$

For a loss functional  $\mathcal{L}$  (e.g., MSE or cross-entropy), define

$$\Delta_{\text{aug}} = \mathcal{L}(\mathbf{R+G}) - \mathcal{L}(\mathbf{R}),$$

with  $\Delta_{\text{aug}} < 0$  indicating that synthetic data *improve* generalization. We also report  $\mathcal{L}(\mathbf{G})$  to gauge domain shift and realism.

**Reporting.** For each metric, provide mean  $\pm$  standard error over  $B$  bootstrap resamples of the test set. When comparing methods, include paired confidence intervals for deltas (e.g.,  $\Delta_{\text{aug}}$ ).

## 5.5 Summary score (optional)

To summarize across views, combine standardized distances:

$$\mathcal{S} = \alpha \widetilde{\text{MMD}}_{\text{multi}} + \beta \widetilde{W}_1(\text{returns}) + \gamma \text{ACFGap}(h) + \eta \max\{0, \mathcal{L}(\mathbf{R+G}) - \mathcal{L}(\mathbf{R})\},$$

where tildes denote z-scored metrics across models to make scales comparable, and  $(\alpha, \beta, \gamma, \eta)$  are user-chosen weights. Lower  $\mathcal{S}$  is better.

**Practical defaults.** Use  $L \in \{10, 15, 20\}$  and  $m \in \{3, 4\}$  for signatures; compare terminal values, terminal log-returns, and pooled per-step log-returns; set  $h \in [10, 20]$  for ACF/Ljung–Box; in ML tasks, hold out the last 20% of each series for testing.

## 6 Practical Notes (matching the code)

- **Path representation.** If you store *levels* or *log-levels*, use relative segments  $p_t = S_{t+1:t+F} - S_t$ ; if you store *returns*, do not re-center segments.
- **Time normalization.** Normalize the time channel within each window to  $[0, 1]$  for stable signatures.
- **k-NN sampling.** Prefer soft sampling (e.g., softmax on distances) to reduce discontinuities.
- **KRR solver.** Use Cholesky with jitter on  $K + \lambda I$ ; center/scale signature features column-wise.

## 7 Experiment: Path-wise Signature Bootstrap on S&P 500 (2010–2024)

### 7.1 Data and Preprocessing

We source daily close prices via `yfinance` for S&P 500 (`^GSPC`), DJIA (`^DJI`), and Nasdaq (`^IXIC`) over 2010-01-01 to 2025-01-01. For this experiment we use only S&P 500 for generation/evaluation; the others are reserved for future multi-asset tests.

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  $\hat{\text{GSPC}}$  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  $\hat{\text{GSPC}}$  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  $(\tau, X_\tau)$  where  $\tau$  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  $\ell_2$  distance between mean (log-)signatures of sliding windows from real vs generated sets:

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

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<sup>2</sup> across datasets and feature types (lower is better).

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

**Qualitative samples.** Figure 2 shows representative generated samples for each setting (residual paths with trend re-added to produce index levels).



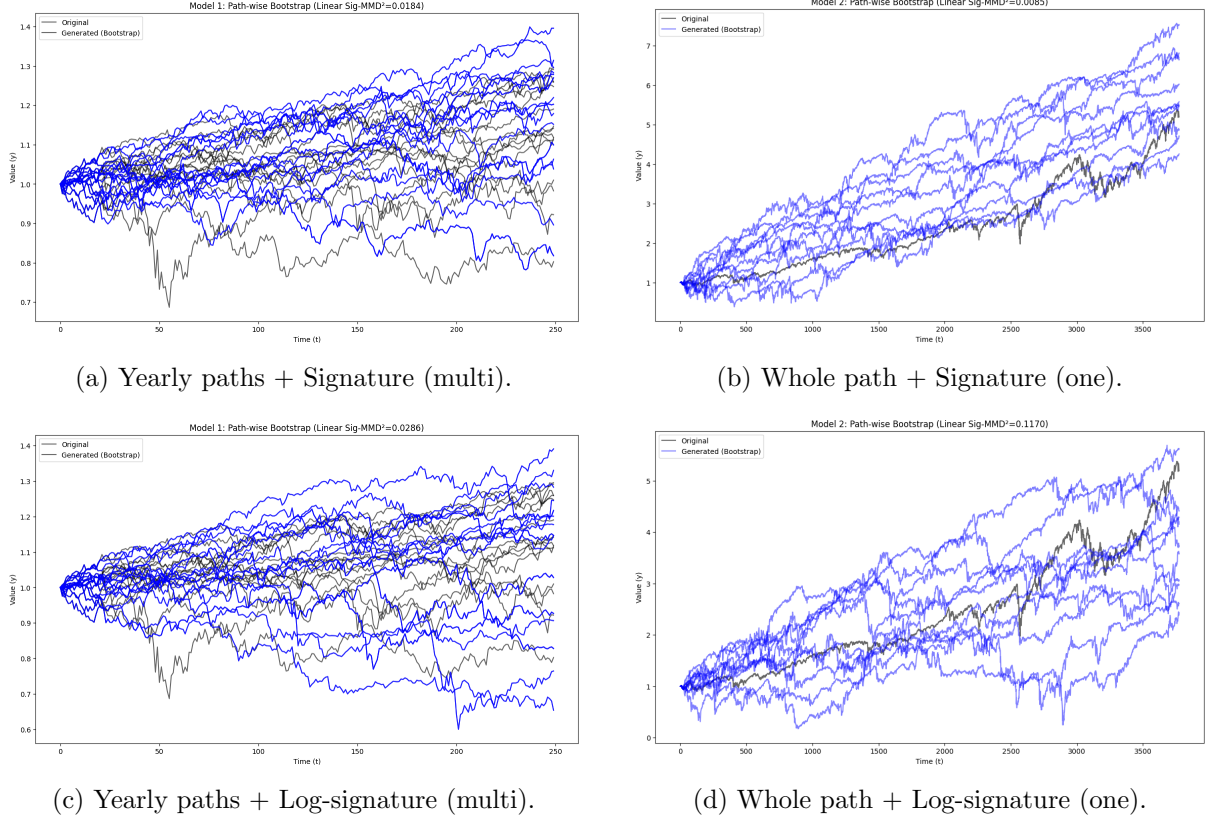


Figure 1: Path-wise bootstrap samples under four settings (trend added back).

#### 7.4.2 Model 2: Hybrid bootstrap

Table 2: Linear Signature MMD<sup>2</sup> across datasets and feature types.

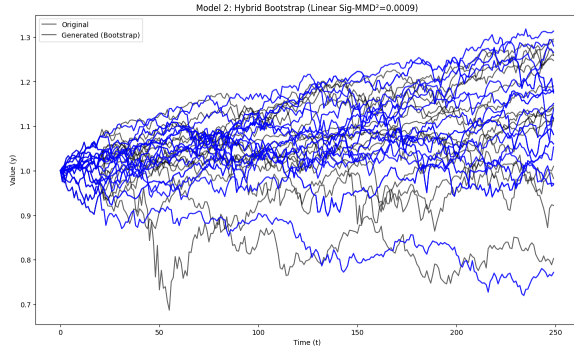
Dataset & Feature	Sig-MMD <sup>2</sup>	Notes
Yearly paths ( <i>multi</i> ) + Signature	<0.000867>	$N_{\text{gen}} = 50$ , seed = 1234
Whole path ( <i>one</i> ) + Signature	<0.034129>	$N_{\text{gen}} = 10$ , seed unset
Yearly paths ( <i>multi</i> ) + Log-signature	<0.004026>	$N_{\text{gen}} = 50$ , seed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

**Qualitative samples.** Figure 2 shows representative generated samples for each setting (residual paths with trend re-added to produce index levels).

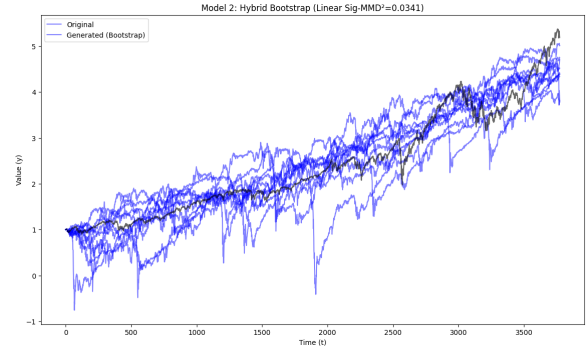
#### 7.4.3 Model 3: Kernel Ridge Regression

**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.)

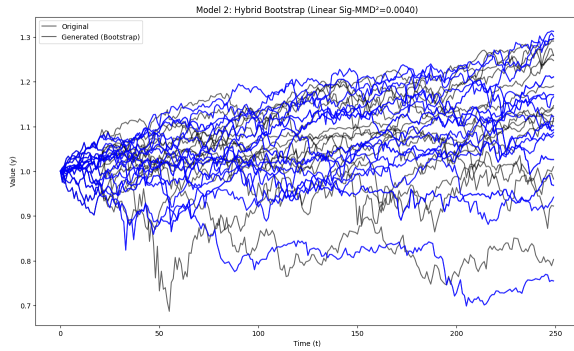
**Qualitative samples.** Figure 3 shows representative generated log-price paths under the four settings.



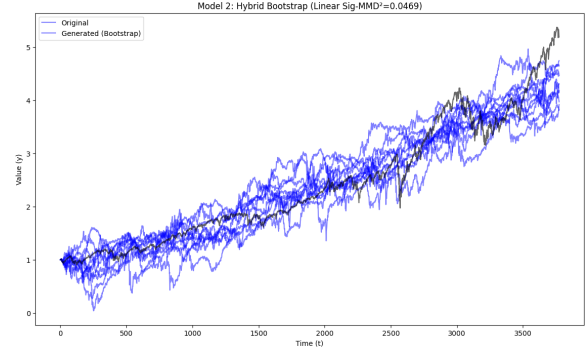
(a) Yearly paths + Signature (multi).



(b) Whole path + Signature (one).



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

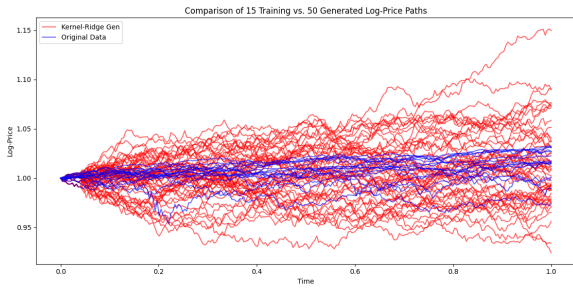


(d) Whole path + Log-signature (one).

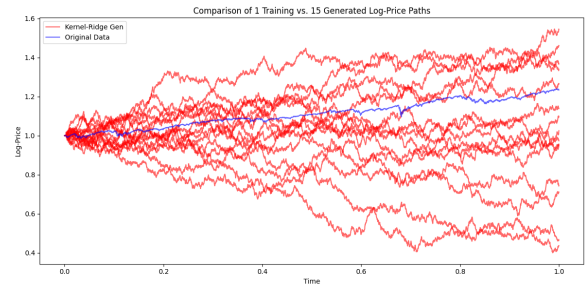
Figure 2: Hybrid bootstrap samples under four settings (trend added back).

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

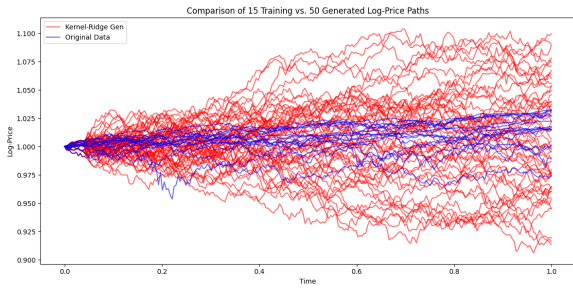
Dataset & Feature	Sig-MMD <sup>2</sup>	Notes
Yearly paths ( <i>multi</i> ) + Signature	$<2.0218544503313323e-09>$	$N_{\text{gen}} = 50$ , seed if set
Whole path ( <i>one</i> ) + Signature	$<6.119611899692986e-07>$	$N_{\text{gen}} = 15$ , seed unset
Yearly paths ( <i>multi</i> ) + Log-signature	$<1.3504824216178214e-07>$	$N_{\text{gen}} = 50$ , seed if set
Whole path ( <i>one</i> ) + Log-signature	$<4.066745963097663e-07>$	$N_{\text{gen}} = 15$ , $\widehat{\log \sigma}$ clipped



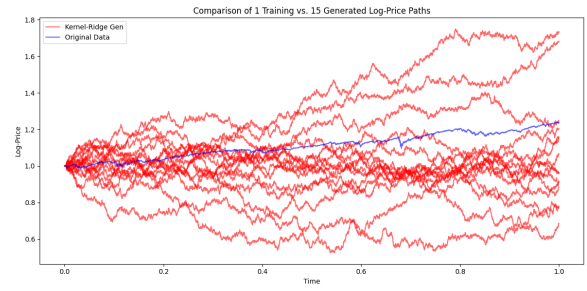
(a) Yearly paths + Signature (multi).



(b) Whole path + Signature (one).



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



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

Figure 3: Kernel Ridge Regression (signature-SDE) samples under four settings.