

A DATA-DRIVEN FOURIER-MIXTURE NEURAL-NETWORK METHOD FOR DENSITY ESTIMATION

DUY-MINH DANG* AND VOLTER ENTOMA †

Abstract. We propose a data-driven Fourier-trained neural-network method for estimating fixed-horizon probability densities from empirical characteristic-function (CF) information. The estimator is a positive Gaussian–Laplace mixture with closed-form CF, so training can be performed directly in Fourier space while preserving nonnegativity and unit mass. We consider two sampling settings. In the direct i.i.d. sampling setting, the method is trained against an empirical CF constructed from i.i.d. samples. In the resampling-based pseudo-sampling setting, it is trained against an empirical pseudo-CF constructed from dependent data by resampling. For the direct i.i.d. case, we derive an expected L_2 error bound that separates Fourier truncation, empirical training error, discretization, and CF sampling error. For the pseudo-sampling case, we obtain a conditional analogue with two additional pseudo-law discrepancy terms. We develop a multidimensional extension of the framework and analyze its computational complexity. Numerical experiments show competitive performance relative to Expectation–Maximization on Gaussian-mixture benchmarks, clear gains on heavy-tailed targets, L_2 error decay consistent with the theory in a well-specified setting, and effective estimation of one-year Australian equity return law from resampled dependent data.

Key words. empirical characteristic function, density estimation, Fourier-trained neural networks, Gaussian–Laplace mixture, dependent data

MSC codes. 65D15, 62G07, 60E10, 68T07, 41A30

1 Introduction Estimating a fixed-horizon probability law from data is a basic task in computational science. Such laws arise whenever one seeks to approximate the distribution of a state increment over a prescribed time interval, for example in stochastic simulation, uncertainty quantification, filtering, and data-driven decision problems. In many applications, the target density is unavailable in closed form and only sample information is available. This is especially true when the underlying observations form a dependent time series and the quantity of interest is a law over a longer horizon than the sampling frequency of the data.

Two broad classes of density estimation methods are commonly used in this setting. The first consists of physical-space estimators, such as kernel density estimation [22] and finite-mixture models fitted by likelihood-based procedures [18]. Among these, Gaussian mixtures estimated by Expectation–Maximization (EM) provide a classical positive parametric benchmark [6, 18].

The second class uses Fourier information, exploiting the fact that a characteristic function (CF) always exists and may be easier to analyze or approximate than the density itself. A prominent example is the COS method [9], which approximates densities by finite cosine-series expansions with expansion coefficients obtained from the CF. Data-driven variants have also been developed; in particular, the data-driven COS method of [15] uses samples to construct a Fourier-cosine density estimator without requiring a closed-form CF. However, truncated Fourier-cosine expansions do not in general enforce nonnegativity of the recovered density, and this loss of positivity can be problematic for short-horizon transition densities and for subsequent stochastic-control computations [7, 10]. Fourier-trained neural-network (NN) methods provide another Fourier-domain route in the exact-transform setting; see, for example,

*School of Mathematics and Physics, The University of Queensland, St Lucia, Brisbane 4072, Australia, (duyminh.dang@uq.edu.au).

†School of Mathematics and Physics, The University of Queensland, St Lucia, Brisbane 4072, Australia, (v.entoma@student.uq.edu.au).

[7, 4], where the target CF is assumed known in closed form.

The present paper focuses on the data-driven Fourier-trained NN setting, where the exact CF is unavailable but empirical transform information can still be constructed from samples. One may have access to direct i.i.d. samples from the target fixed-horizon law, from which an empirical CF is constructed. In the presence of temporal dependence, one instead observes a dependent time series, from which an empirical pseudo-CF is constructed by a suitable resampling procedure. This motivates a Fourier-trained NN model that remains a proper density throughout optimization, is expressive enough to capture non-Gaussian features, and admits a closed-form CF so that the learning problem can be posed directly in Fourier space.

The main contributions of the paper are as follows.

- We introduce a data-driven Fourier-trained NN method based on a positive Gaussian–Laplace mixture parametrized by a bounded single-hidden-layer feedforward NN. The resulting density is nonnegative and integrates to one by construction, while its CF is available in closed form. This allows the learning problem to be posed directly against empirical transform data.
- For the direct i.i.d. sampling setting, we derive an expected L_2 error bound for the estimated density. A central technical ingredient is a Fourier-to-real-space analysis, via Plancherel’s theorem, that transfers control of the Fourier-domain training error to control of the density error in physical space. The resulting decomposition separates Fourier truncation, empirical training error, discretization error, and empirical-CF sampling error.
- For the resampling-based pseudo-sampling setting, we show that the same analysis carries over conditionally on the observed history. The resulting error bound has the same structural form as in the direct i.i.d. case, together with two additional pseudo-law discrepancy terms that quantify the gap between the resampling law and the true fixed-horizon law. This provides a theoretical basis for Fourier-trained NN estimation from dependent time-series data.
- We extend the method to multiple dimensions, analyze its computational complexity, and validate it numerically on a range of benchmark and data-driven examples. These include Gaussian-mixture sanity checks, heavy-tailed and jump benchmarks, a two-dimensional (2D) Cauchy example, experiments on the dependence of the observed L_2 error on the sample size and Fourier grid size, and a resampling-based pseudo-sampling experiment built from Australian equity data.

The paper therefore combines several features that are rarely treated together: data-driven Fourier training of a single-hidden-layer NN, a positivity-preserving Gaussian–Laplace output model, Fourier-to-real-space analysis underpinning rigorous density error bounds, and a treatment of both direct i.i.d. sampling and dependent-data pseudo-sampling. The numerical results support this picture: the method is competitive with EM on Gaussian-mixture benchmarks, improves over the Gaussian subclass on heavy-tailed targets, shows error decay consistent with the theory in a well-specified setting, and remains effective in the resampling-based Australian equity experiment.

Although some numerical illustrations use financial return data, the methodology is not finance-specific. The central problem is more general: estimating a positive fixed-horizon density from empirical transform information, including settings where the available observations are dependent rather than i.i.d. In this sense, the present work can be viewed as a data-driven Fourier counterpart to positive finite-mixture density estimation, with emphasis on Fourier-trained NN models, transform-domain learning, and rigorous error analysis.

The remainder of the paper is organized as follows. Section 2 introduces the neural network Fourier-mixture method in the direct i.i.d. sampling setting. Section 3 presents the corresponding error analysis. Section 4 extends the framework to resampling-based pseudo-sampling from dependent data. Section 5 gives the multi-dimensional extension, and Section 6 discusses computational complexity. Numerical experiments are reported in Section 7. Finally, Section 8 provides concluding remarks and outlines future directions.

2 A neural network Fourier-mixture method

2.1 Data-driven setting We work in a data-driven setting. Given observed data points $X_1, \dots, X_M \in \mathbb{R}$, we seek to estimate an unknown probability density g on \mathbb{R} from Fourier-domain information. Throughout, we use the Fourier pair

$$(2.1) \quad G(\eta) = \int_{\mathbb{R}} e^{i\eta x} g(x) dx, \quad \eta \in \mathbb{R}, \quad g(x) = \frac{1}{2\pi} \int_{\mathbb{R}} e^{-i\eta x} G(\eta) d\eta, \quad x \in \mathbb{R}.$$

In contrast with many Fourier-based approaches, including [7], where the CF $G(\cdot)$ is available in closed form, here we work only with the empirical CF given by

$$(2.2) \quad \widehat{G}_M(\eta) = \frac{1}{M} \sum_{m=1}^M e^{i\eta X_m}, \quad \eta \in \mathbb{R}.$$

In this section, our objective is to recover g from \widehat{G}_M by fitting a bounded class of Gaussian–Laplace mixtures parametrized by a single-hidden-layer FFNN, whose Fourier transforms are available in closed form.

2.2 Description We first state the regularity conditions used in the subsequent error analysis.

ASSUMPTION 2.1 (Modelling assumptions). *Let g denote the target density, and let G denote its Fourier transform, i.e. the CF of the associated probability law. The following conditions hold.*

- (A1) *The observations X_1, \dots, X_M are i.i.d. with common density g .*
- (A2) *The density satisfies $g \in L_1(\mathbb{R}) \cap L_\infty(\mathbb{R})$.*
- (A3) *For any finite interval $[\eta_1, \eta_2] \subset \mathbb{R}$ with $\eta_1 < \eta_2$, the CF G is Lipschitz continuous on $[\eta_1, \eta_2]$, i.e. there exists a finite constant $L_{[\eta_1, \eta_2]} > 0$ such that*

$$|G(\eta') - G(\eta'')| \leq L_{[\eta_1, \eta_2]} |\eta' - \eta''|, \quad \forall \eta', \eta'' \in [\eta_1, \eta_2].$$

Remark 2.1 (Discussion of Assumption 2.1). Assumption 2.1 (A1) is a sampling condition. Under (A1), for each fixed $\eta \in \mathbb{R}$, the empirical CF in (2.2) satisfies

$$(2.3) \quad \mathbb{E}[\widehat{G}_M(\eta)] = G(\eta), \quad \text{Var}(\widehat{G}_M(\eta)) = \frac{1 - |G(\eta)|^2}{M}.$$

Hence \widehat{G}_M is an unbiased estimator of the true CF G . Dependent time-series data are addressed in Section 4 through a resampling-based pseudo-sampling framework. As a first step, we work here in the direct i.i.d. setting to isolate the Fourier-inversion and approximation errors and to establish the baseline analysis for that extension.

Assumption 2.1 (A2) implies $g \in L_2(\mathbb{R})$, since $\|g\|_{L_1} = 1$ and

$$\|g\|_{L_2}^2 = \int_{\mathbb{R}} |g(x)|^2 dx \leq \|g\|_{L_\infty} \|g\|_{L_1} = \|g\|_{L_\infty} < \infty.$$

By Plancherel’s theorem, this yields $G \in L_2(\mathbb{R})$, and therefore for every $\varepsilon > 0$ there exists $\eta' > 0$ such that $\int_{\mathbb{R} \setminus [-\eta', \eta']} |G(\eta)|^2 d\eta \leq \varepsilon$.

Assumption 2.1 (A3) is a local regularity condition. It requires Lipschitz continuity only on the compact Fourier interval used for training and discretization bounds. Assumptions (A2) and (A3) are mild for many smooth parametric families, including Gaussian laws; jump–diffusion models with a nondegenerate Gaussian component, such as the Merton model [19] and the Kou model [12]; and many tempered-tail Lévy models with closed-form CFs and bounded densities, including the Normal–Inverse–Gaussian model [1] and CGMY/tempered-stable families with $Y \in (0, 2)$ under standard parameter regimes [2].

Gaussian–Laplace mixture representation. Let the Gaussian and Laplace activation functions be defined, respectively, by

$$\phi_{\mathcal{G}}(z) = e^{-z^2}, \quad \phi_{\mathcal{L}}(z) = e^{-|z|}.$$

We define the class of mixed single-layer FFNN

$$(2.4) \quad \Sigma(\phi_{\mathcal{G}}, \phi_{\mathcal{L}}) = \left\{ \hat{g} : \mathbb{R} \rightarrow \mathbb{R} \mid \hat{g}(x; \vartheta) = \sum_{k=1}^{K_{\mathcal{G}}} \tilde{\rho}_k^{(\mathcal{G})} \phi_{\mathcal{G}}(\tilde{w}_k^{(\mathcal{G})} x + \tilde{b}_k^{(\mathcal{G})}) + \sum_{\ell=1}^{K_{\mathcal{L}}} \tilde{\rho}_{\ell}^{(\mathcal{L})} \phi_{\mathcal{L}}(\tilde{w}_{\ell}^{(\mathcal{L})} x + \tilde{b}_{\ell}^{(\mathcal{L})}) \right\},$$

where $K_{\mathcal{G}}, K_{\mathcal{L}} \in \mathbb{N}$ and ϑ collects the raw coefficients, weights, and biases.

For density estimation, we work instead with a positive normalized subclass parameterized by mixture logits, locations, and raw scales. To this end, we define the normalized Gaussian and Laplace kernels

$$(2.5) \quad \varphi_{\mathcal{G}}(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right), \quad \varphi_{\mathcal{L}}(x; \nu, b) = \frac{1}{2b} \exp\left(-\frac{|x - \nu|}{b}\right),$$

with $\sigma > 0$ and $b > 0$. The raw hidden-unit form and the normalized kernels are related by

$$(2.6) \quad \begin{aligned} \mu_k &= -\frac{\tilde{b}_k^{(\mathcal{G})}}{\tilde{w}_k^{(\mathcal{G})}}, & \sigma_k &= \frac{1}{\sqrt{2} |\tilde{w}_k^{(\mathcal{G})}|}, & \tilde{\beta}_k^{(\mathcal{G})} &= \frac{\sqrt{\pi} \tilde{\rho}_k^{(\mathcal{G})}}{|\tilde{w}_k^{(\mathcal{G})}|}, \\ \nu_{\ell} &= -\frac{\tilde{b}_{\ell}^{(\mathcal{L})}}{\tilde{w}_{\ell}^{(\mathcal{L})}}, & b_{\ell} &= \frac{1}{|\tilde{w}_{\ell}^{(\mathcal{L})}|}, & \tilde{\beta}_{\ell}^{(\mathcal{L})} &= \frac{2 \tilde{\rho}_{\ell}^{(\mathcal{L})}}{|\tilde{w}_{\ell}^{(\mathcal{L})}|}. \end{aligned}$$

We now introduce the positive normalized parametrization used for training. For $k = 1, \dots, K_{\mathcal{G}}$ and $\ell = 1, \dots, K_{\mathcal{L}}$, the trainable variables are the Gaussian mixture logits $\rho_k^{(\mathcal{G})}$, locations μ_k , and raw scales $w_k^{(\mathcal{G})}$, together with the Laplace mixture logits $\rho_{\ell}^{(\mathcal{L})}$, locations ν_{ℓ} , and raw scales $w_{\ell}^{(\mathcal{L})}$. The mixture weights are obtained from the joint softmax map

$$(2.7) \quad \begin{aligned} \beta_k^{(\mathcal{G})} &= \exp(\rho_k^{(\mathcal{G})}) \left(\sum_{j=1}^{K_{\mathcal{G}}} \exp(\rho_j^{(\mathcal{G})}) + \sum_{m=1}^{K_{\mathcal{L}}} \exp(\rho_m^{(\mathcal{L})}) \right)^{-1}, \\ \beta_{\ell}^{(\mathcal{L})} &= \exp(\rho_{\ell}^{(\mathcal{L})}) \left(\sum_{j=1}^{K_{\mathcal{G}}} \exp(\rho_j^{(\mathcal{G})}) + \sum_{m=1}^{K_{\mathcal{L}}} \exp(\rho_m^{(\mathcal{L})}) \right)^{-1}. \end{aligned}$$

The Gaussian and Laplace scales are generated by the softplus map

$$(2.8) \quad \sigma_k = \text{softplus}(w_k^{(\mathcal{G})}) + \varepsilon, \quad b_{\ell} = \text{softplus}(w_{\ell}^{(\mathcal{L})}) + \varepsilon,$$

for $k = 1, \dots, K_{\mathcal{G}}$ and $\ell = 1, \dots, K_{\mathcal{L}}$, with fixed $\varepsilon > 0$. The resulting Gaussian–Laplace mixture class is

$$(2.9) \quad \Sigma_{\mathcal{GL}} = \left\{ \hat{g} : \mathbb{R} \rightarrow \mathbb{R} \mid \hat{g}(x; \theta) = \sum_{k=1}^{K_{\mathcal{G}}} \beta_k^{(\mathcal{G})} \varphi_{\mathcal{G}}(x; \mu_k, \sigma_k) + \sum_{\ell=1}^{K_{\mathcal{L}}} \beta_{\ell}^{(\mathcal{L})} \varphi_{\mathcal{L}}(x; \nu_{\ell}, b_{\ell}), \theta \in \Theta \right\}.$$

The effective parameter vector is

$$(2.10) \quad \theta = \left(\{\beta_k^{(\mathcal{G})}, \mu_k, \sigma_k\}_{k=1}^{K_{\mathcal{G}}}, \{\beta_\ell^{(\mathcal{L})}, \nu_\ell, b_\ell\}_{\ell=1}^{K_{\mathcal{L}}} \right).$$

We restrict the effective mixture parameters to a bounded set

$$(2.11) \quad \Theta = \left\{ \theta : \beta_k^{(\mathcal{G})} \geq 0, \beta_\ell^{(\mathcal{L})} \geq 0, \sum_{k=1}^{K_{\mathcal{G}}} \beta_k^{(\mathcal{G})} + \sum_{\ell=1}^{K_{\mathcal{L}}} \beta_\ell^{(\mathcal{L})} = 1, \right. \\ \left. |\mu_k|, |\nu_\ell| \leq \bar{\mu}, \quad 0 < \sigma_{\min} \leq \sigma_k \leq \sigma_{\max}, \quad 0 < b_{\min} \leq b_\ell \leq b_{\max} \right\}.$$

Under (2.11), each $\widehat{g}(\cdot; \theta) \in \Sigma_{\mathcal{G}\mathcal{L}}$ is a probability density.

Closed-form characteristic function. Direct calculation gives

$$(2.12) \quad \widehat{G}(\eta; \theta) = \sum_{k=1}^{K_{\mathcal{G}}} \beta_k^{(\mathcal{G})} \exp(i\eta\mu_k - \sigma_k^2\eta^2/2) + \sum_{\ell=1}^{K_{\mathcal{L}}} \beta_\ell^{(\mathcal{L})} \frac{\exp(i\eta\nu_\ell)}{1 + b_\ell^2\eta^2}.$$

Hence

$$(2.13) \quad \operatorname{Re} \widehat{G}(\eta; \theta) = \sum_{k=1}^{K_{\mathcal{G}}} \beta_k^{(\mathcal{G})} \exp(-\sigma_k^2\eta^2/2) \cos(\eta\mu_k) + \sum_{\ell=1}^{K_{\mathcal{L}}} \beta_\ell^{(\mathcal{L})} \frac{\cos(\eta\nu_\ell)}{1 + b_\ell^2\eta^2},$$

$$(2.14) \quad \operatorname{Im} \widehat{G}(\eta; \theta) = \sum_{k=1}^{K_{\mathcal{G}}} \beta_k^{(\mathcal{G})} \exp(-\sigma_k^2\eta^2/2) \sin(\eta\mu_k) + \sum_{\ell=1}^{K_{\mathcal{L}}} \beta_\ell^{(\mathcal{L})} \frac{\sin(\eta\nu_\ell)}{1 + b_\ell^2\eta^2}.$$

In particular, $\widehat{G}(0; \theta) = 1$.

L_2 -approximation error and Fourier-domain invariance. Since $g \in L_1(\mathbb{R}) \cap L_2(\mathbb{R})$ and every $\widehat{g}(\cdot; \theta) \in \Sigma_{\mathcal{G}\mathcal{L}}$ with $\theta \in \Theta$ also belongs to $L_1(\mathbb{R}) \cap L_2(\mathbb{R})$, Plancherel's theorem gives

$$(2.15) \quad \int_{\mathbb{R}} |g(x) - \widehat{g}(x; \theta)|^2 dx = \frac{1}{2\pi} \int_{\mathbb{R}} |G(\eta) - \widehat{G}(\eta; \theta)|^2 d\eta.$$

This identity is the key structural fact used below: it allows both training and error analysis to be carried out through \widehat{G}_M and $\widehat{G}(\cdot; \theta)$ rather than directly in real space.

2.3 Training loss and regularization The exact CF G is not available. Instead, training is based on \widehat{G}_M . We restrict the Fourier domain to $[-\eta', \eta']$ and choose training nodes $\{\eta_p\}_{p=1}^P \subset [-\eta', \eta']$ induced by a deterministic, potentially non-uniform, partition

$$-\eta' = \xi_0 < \xi_1 < \dots < \xi_P = \eta'.$$

For each cell $[\xi_{p-1}, \xi_p]$, let $\eta_p \in [\xi_{p-1}, \xi_p]$ be the training node and set

$$\delta_p = \xi_p - \xi_{p-1}, \quad p = 1, \dots, P.$$

We assume

$$(2.16) \quad \delta_{\min} = \frac{C_0}{P}, \quad \delta_{\max} = \frac{C_1}{P}, \quad \delta_{\min} = \min_{1 \leq p \leq P} \delta_p, \quad \delta_{\max} = \max_{1 \leq p \leq P} \delta_p,$$

for finite constants $C_0, C_1 > 0$ independent of P .

For each training node η_p , define the exact real and imaginary residuals

$$(2.17) \quad \Delta_{p, \operatorname{Re}}(\theta) = \operatorname{Re} \widehat{G}_M(\eta_p) - \operatorname{Re} \widehat{G}(\eta_p; \theta), \quad \Delta_{p, \operatorname{Im}}(\theta) = \operatorname{Im} \widehat{G}_M(\eta_p) - \operatorname{Im} \widehat{G}(\eta_p; \theta).$$

We define the Fourier-domain loss function

$$(2.18) \quad \operatorname{Loss}_{M,P}(\theta) = \operatorname{MSE}_{M,P}(\theta) + \lambda R_{M,P}(\theta), \quad \theta \in \widehat{\Theta} \subseteq \Theta,$$

where

$$(2.19) \quad \begin{aligned} \text{MSE}_{M,P}(\theta) &= \frac{1}{P} \sum_{p=1}^P \left(\Delta_{p,\text{Re}}(\theta)^2 + \Delta_{p,\text{Im}}(\theta)^2 \right), \\ R_{M,P}(\theta) &= \frac{1}{P} \sum_{p=1}^P \left(|\Delta_{p,\text{Re}}(\theta)| + |\Delta_{p,\text{Im}}(\theta)| \right). \end{aligned}$$

The first term in (2.18) is the discrete Fourier-space L_2 misfit between the learned CF $\widehat{G}(\cdot; \theta)$ and the empirical CF \widehat{G}_M . The second term is an MAE contribution that aims to improve robustness in oscillatory or high-curvature frequency regions.

Data-driven minimizer. Let $\widehat{\Theta} \subseteq \Theta$ denote the feasible search set induced by the chosen component counts (K_G, K_L) and the corresponding bounds on the effective parameters. The data-driven minimizer is

$$(2.20) \quad \widehat{\theta} \in \arg \min_{\theta \in \widehat{\Theta}} \text{Loss}_{M,P}(\theta),$$

and the estimated density is $\widehat{g}(\cdot; \widehat{\theta}) \in \Sigma_{\mathcal{GL}}$.

Optional affine preprocessing. When the empirical CF is highly oscillatory or supported over a large frequency range, it is often convenient to apply an affine change of variables $Y = aX + c$ with $a > 0$ before training [7]. The empirical CF of the transformed variable then satisfies $\widehat{G}_M^Y(\eta) = e^{inc} \widehat{G}_M^X(a\eta)$. After training in the Y -variable, the estimated density in the original variable is recovered by $\widehat{g}_X(x; \widehat{\theta}) = a \widehat{g}_Y(ax + c; \widehat{\theta})$.

Training algorithm. We use a two-stage optimization strategy. The first stage performs a coarse exploration of the loss landscape, while the second stage refines the solution using a smaller learning rate. The use of AMSGrad and Adam in these two stages follows standard NN practice; see [23, 11].

Algorithm 2.1 Data-driven Fourier-trained Gaussian–Laplace mixture model

- 1: **Input:** data $\{X_m\}_{m=1}^M$, component counts (K_G, K_L) , Fourier window $[-\eta', \eta']$, training nodes $\{\eta_p\}_{p=1}^P$, and MAE weight λ .
 - 2: If needed, apply affine preprocessing $Y_m = aX_m + c$ and replace the data by $\{Y_m\}_{m=1}^M$.
 - 3: Compute $\widehat{G}_M(\eta_p) = \frac{1}{M} \sum_{m=1}^M e^{i\eta_p Y_m}$ for $p = 1, \dots, P$.
 - 4: Initialize the trainable logits, locations, and raw scales.
 - 5: **Stage I:** minimize $\text{Loss}_{M,P}$ by AMSGrad (or Adam) with a relatively large learning rate.
 - 6: **Stage II:** continue minimization with Adam at a smaller learning rate until convergence, or stop by early stopping / validation loss.
 - 7: Return $\widehat{\theta}$, the estimated density $\widehat{g}(\cdot; \widehat{\theta})$, and the learned CF $\widehat{G}(\cdot; \widehat{\theta})$.
-

3 Error analysis This section analyzes the expected L_2 error under direct i.i.d. sampling. We first introduce the exact Fourier-domain loss on the training nodes, then quantify the empirical-CF sampling error, and finally combine this with discretization and truncation estimates.

For each training node η_p , define the exact real and imaginary residuals

$$(3.1) \quad \Delta_{p,\text{Re}}^*(\theta) = \text{Re } G(\eta_p) - \text{Re } \widehat{G}(\eta_p; \theta), \quad \Delta_{p,\text{Im}}^*(\theta) = \text{Im } G(\eta_p) - \text{Im } \widehat{G}(\eta_p; \theta).$$

Let

$$(3.2) \quad \begin{aligned} \text{MSE}_P^*(\theta) &= \frac{1}{P} \sum_{p=1}^P \left(\Delta_{p,\text{Re}}^*(\theta)^2 + \Delta_{p,\text{Im}}^*(\theta)^2 \right), \\ R_P^*(\theta) &= \frac{1}{P} \sum_{p=1}^P \left(|\Delta_{p,\text{Re}}^*(\theta)| + |\Delta_{p,\text{Im}}^*(\theta)| \right), \end{aligned}$$

and define the exact loss

$$(3.3) \quad \text{Loss}_P^*(\theta) = \text{MSE}_P^*(\theta) + \lambda R_P^*(\theta), \quad \theta \in \widehat{\Theta} \subseteq \Theta.$$

3.1 Empirical-CF sampling error We first quantify how finite-sample error in the empirical-CF propagates into the Fourier-domain loss evaluated on the training nodes.

LEMMA 3.1 (Empirical-CF sampling error). *For the training nodes $\{\eta_p\}_{p=1}^P$, define*

$$(3.4) \quad V_P = \frac{1}{P} \sum_{p=1}^P (1 - |G(\eta_p)|^2), \quad W_P = \frac{1}{P} \sum_{p=1}^P \sqrt{1 - |G(\eta_p)|^2}.$$

Then, for every fixed $\theta \in \widehat{\Theta}$,

$$(3.5) \quad \mathbb{E}[\text{MSE}_{M,P}(\theta)] = \text{MSE}_P^*(\theta) + \frac{V_P}{M}, \quad \mathbb{E}[R_{M,P}(\theta)] \leq R_P^*(\theta) + \sqrt{\frac{2}{M}} W_P.$$

Consequently,

$$(3.6) \quad \mathbb{E}[\text{Loss}_{M,P}(\theta)] \leq \text{Loss}_P^*(\theta) + \frac{V_P}{M} + \lambda \sqrt{\frac{2}{M}} W_P.$$

Proof of Lemma 3.1. For each fixed node η_p , write

$$\widehat{G}_M(\eta_p) - \widehat{G}(\eta_p; \theta) = (G(\eta_p) - \widehat{G}(\eta_p; \theta)) + (\widehat{G}_M(\eta_p) - G(\eta_p)).$$

Taking squared moduli and then expectations yields

$$(3.7) \quad \mathbb{E} \left[|\widehat{G}_M(\eta_p) - \widehat{G}(\eta_p; \theta)|^2 \right] = |G(\eta_p) - \widehat{G}(\eta_p; \theta)|^2 + \mathbb{E} \left[|\widehat{G}_M(\eta_p) - G(\eta_p)|^2 \right],$$

since the cross term vanishes by $\mathbb{E}[\widehat{G}_M(\eta_p) - G(\eta_p)] = 0$ from (2.3). Moreover,

$$(3.8) \quad \mathbb{E} \left[|\widehat{G}_M(\eta_p) - G(\eta_p)|^2 \right] = \text{Var} \left(\widehat{G}_M(\eta_p) \right) = \frac{1 - |G(\eta_p)|^2}{M},$$

again by (2.3). Now sum (3.7) over $p = 1, \dots, P$ and divide by P . Using (3.8), we obtain

$$\frac{1}{P} \sum_{p=1}^P \mathbb{E} \left[|\widehat{G}_M(\eta_p) - \widehat{G}(\eta_p; \theta)|^2 \right] = \frac{1}{P} \sum_{p=1}^P |G(\eta_p) - \widehat{G}(\eta_p; \theta)|^2 + \frac{1}{MP} \sum_{p=1}^P (1 - |G(\eta_p)|^2).$$

Using the definitions of $\text{MSE}_{M,P}(\theta)$, $\text{MSE}_P^*(\theta)$, and V_P yields the first identity in (3.5).

For the MAE term in (3.5), the triangle inequality gives

$$(3.9) \quad \begin{aligned} |\Delta_{p,\text{Re}}(\theta)| + |\Delta_{p,\text{Im}}(\theta)| &\leq |\Delta_{p,\text{Re}}^*(\theta)| + |\Delta_{p,\text{Im}}^*(\theta)| + |\text{Re}(\widehat{G}_M(\eta_p) - G(\eta_p))| \\ &\quad + |\text{Im}(\widehat{G}_M(\eta_p) - G(\eta_p))|. \end{aligned}$$

The last two terms on the right-hand side of (3.9) are bounded by¹

$$(3.10) \quad |\text{Re}(\widehat{G}_M(\eta_p) - G(\eta_p))| + |\text{Im}(\widehat{G}_M(\eta_p) - G(\eta_p))| \leq \sqrt{2} |\widehat{G}_M(\eta_p) - G(\eta_p)|.$$

¹We use the inequality $|a| + |b| \leq \sqrt{2} (a^2 + b^2)^{1/2}$.

Taking expectations both sides of (3.10) and applying Cauchy–Schwarz yields

$$\mathbb{E}[|\operatorname{Re}(\widehat{G}_M(\eta_p) - G(\eta_p))| + |\operatorname{Im}(\widehat{G}_M(\eta_p) - G(\eta_p))|] \leq \sqrt{2} (\mathbb{E}|\widehat{G}_M(\eta_p) - G(\eta_p)|^2)^{1/2}.$$

This and (3.8) give

$$(3.11) \quad \mathbb{E}[|\operatorname{Re}(\widehat{G}_M(\eta_p) - G(\eta_p))| + |\operatorname{Im}(\widehat{G}_M(\eta_p) - G(\eta_p))|] \leq \sqrt{\frac{2}{M}} \sqrt{1 - |G(\eta_p)|^2}.$$

Now sum (3.9) over $p = 1, \dots, P$ and divide by P . Using (3.11) and the definitions of $R_{M,P}(\theta)$, $R_P^*(\theta)$, and W_P yields the second bound in (3.5). Finally, (3.6) follows from (3.5) together with (2.18) and (3.3). \square

3.2 Discretization and truncation in the Fourier domain The next two lemmas quantify the discretization and truncation errors in the Fourier domain.

LEMMA 3.2 (Discretization). *Fix $\eta' > 0$. For $\theta \in \Theta$, define*

$$F_\theta(\eta) = |G(\eta) - \widehat{G}(\eta; \theta)|^2, \quad \eta \in [-\eta', \eta'].$$

Then there exists a finite constant $C' = C'(\eta') > 0$, independent of P and θ , such that

$$(3.12) \quad \left| \sum_{p=1}^P \delta_p F_\theta(\eta_p) - \int_{-\eta'}^{\eta'} F_\theta(\eta) d\eta \right| \leq \frac{C' C_1^2}{P}, \quad \forall \theta \in \Theta.$$

For a proof of Lemma 3.2, see Appendix A.

LEMMA 3.3 (Fourier-domain truncation error). *For every $\varepsilon > 0$, the training interval $[-\eta', \eta']$ can be chosen sufficiently large so that*

$$(3.13) \quad \int_{\mathbb{R} \setminus [-\eta', \eta']} |G(\eta)|^2 d\eta < \varepsilon, \quad \sup_{\theta \in \Theta} \int_{\mathbb{R} \setminus [-\eta', \eta']} |\widehat{G}(\eta; \theta)|^2 d\eta < \varepsilon.$$

Consequently, for every $\theta \in \Theta$,

$$(3.14) \quad \int_{\mathbb{R} \setminus [-\eta', \eta']} |G(\eta) - \widehat{G}(\eta; \theta)|^2 d\eta < 4\varepsilon.$$

A proof of Lemma 3.3 is given in Appendix B.

3.3 Main L_2 error bound We now state the main theorem giving an expected L_2 error bound for the direct i.i.d. sampling setting.

THEOREM 3.4 (Data-driven L_2 error bound). *Fix $\varepsilon > 0$, and choose $[-\eta', \eta']$ sufficiently large so that Lemma 3.3 applies. Let $\widehat{\theta} \in \widehat{\Theta}$ be the parameter obtained by minimizing (2.18), and suppose that*

$$(3.15) \quad \operatorname{Loss}_{M,P}(\widehat{\theta}) \leq \varepsilon_1$$

for some $\varepsilon_1 > 0$. Then

$$(3.16) \quad \mathbb{E} \left[\int_{\mathbb{R}} |g(x) - \widehat{g}(x; \widehat{\theta})|^2 dx \right] < \frac{1}{2\pi} \left(4\varepsilon + C_1 \left(2\varepsilon_1 + \frac{2V_P}{M} + \lambda \sqrt{\frac{2}{M}} W_P \right) + \frac{C' C_1^2}{P} \right).$$

Here, C_1 is defined in (2.16), V_P and W_P in (3.4), and C' in Lemma 3.2.

Proof of Theorem 3.4. By Plancherel's theorem,

$$(3.17) \quad \begin{aligned} 2\pi \int_{\mathbb{R}} |g(x) - \widehat{g}(x; \widehat{\theta})|^2 dx &= \int_{\mathbb{R}} |G(\eta) - \widehat{G}(\eta; \widehat{\theta})|^2 d\eta \\ &= \int_{\mathbb{R} \setminus [-\eta', \eta']} |G(\eta) - \widehat{G}(\eta; \widehat{\theta})|^2 d\eta + \int_{-\eta'}^{\eta'} |G(\eta) - \widehat{G}(\eta; \widehat{\theta})|^2 d\eta. \end{aligned}$$

The first term on the right-hand side is bounded by 4ε by Lemma 3.3. For the second term, with

$$F_{\hat{\theta}}(\eta) = |G(\eta) - \widehat{G}(\eta; \hat{\theta})|^2,$$

Lemma 3.2 gives

$$\int_{-\eta'}^{\eta'} F_{\hat{\theta}}(\eta) d\eta \leq \sum_{p=1}^P \delta_p F_{\hat{\theta}}(\eta_p) + \frac{C' C_1^2}{P} \leq \frac{C_1}{P} \sum_{p=1}^P F_{\hat{\theta}}(\eta_p) + \frac{C' C_1^2}{P},$$

since $\delta_p \leq C_1/P$. Therefore

$$(3.18) \quad \int_{-\eta'}^{\eta'} F_{\hat{\theta}}(\eta) d\eta \leq C_1 \text{MSE}_P^*(\hat{\theta}) + \frac{C' C_1^2}{P} \leq C_1 \text{Loss}_P^*(\hat{\theta}) + \frac{C' C_1^2}{P}.$$

It remains to bound $\mathbb{E}[\text{Loss}_P^*(\hat{\theta})]$. For each p ,

$$|G(\eta_p) - \widehat{G}(\eta_p; \hat{\theta})|^2 \leq 2|\widehat{G}_M(\eta_p) - \widehat{G}(\eta_p; \hat{\theta})|^2 + 2|\widehat{G}_M(\eta_p) - G(\eta_p)|^2.$$

Also, by the triangle inequality,

$$\begin{aligned} |\Delta_{p,\text{Re}}^*(\hat{\theta})| + |\Delta_{p,\text{Im}}^*(\hat{\theta})| &\leq |\Delta_{p,\text{Re}}(\hat{\theta})| + |\Delta_{p,\text{Im}}(\hat{\theta})| \\ &\quad + |\text{Re}(\widehat{G}_M(\eta_p) - G(\eta_p))| + |\text{Im}(\widehat{G}_M(\eta_p) - G(\eta_p))|. \end{aligned}$$

Averaging over $p = 1, \dots, P$ gives

$$\begin{aligned} \text{Loss}_P^*(\hat{\theta}) &\leq 2 \text{MSE}_{M,P}(\hat{\theta}) + \lambda R_{M,P}(\hat{\theta}) + \frac{2}{P} \sum_{p=1}^P |\widehat{G}_M(\eta_p) - G(\eta_p)|^2 \\ &\quad + \lambda \frac{1}{P} \sum_{p=1}^P \left(|\text{Re}(\widehat{G}_M(\eta_p) - G(\eta_p))| + |\text{Im}(\widehat{G}_M(\eta_p) - G(\eta_p))| \right) \\ &\leq 2 \text{Loss}_{M,P}(\hat{\theta}) + \frac{2}{P} \sum_{p=1}^P |\widehat{G}_M(\eta_p) - G(\eta_p)|^2 \\ &\quad + \lambda \frac{1}{P} \sum_{p=1}^P \left(|\text{Re}(\widehat{G}_M(\eta_p) - G(\eta_p))| + |\text{Im}(\widehat{G}_M(\eta_p) - G(\eta_p))| \right), \end{aligned}$$

since $\text{MSE}_{M,P}(\hat{\theta}) \leq \text{Loss}_{M,P}(\hat{\theta})$. Taking expectations and using (3.15) yields

$$\mathbb{E}[\text{Loss}_P^*(\hat{\theta})] \leq 2\varepsilon_1 + \frac{2V_P}{M} + \lambda \sqrt{\frac{2}{M}} W_P,$$

by Lemma 3.1. Combining this with (3.17) and (3.18) yields (3.16). \square

3.4 Discussion of Theorem 3.4 The bound (3.16) decomposes the expected L_2 error of the estimated density in the direct i.i.d. sampling setting into four contributions: Fourier-domain truncation, represented by ε ; empirical training error, represented by ε_1 ; discretization error from the finite frequency partition, represented by $C' C_1^2/P$; and empirical-CF sampling error, represented by V_P/M and W_P/\sqrt{M} .

The terms V_P/M and W_P/\sqrt{M} are the genuinely new contributions in the data-driven setting. The quantity V_P/M measures the average squared sampling fluctuation of the empirical CF across the training nodes, while W_P/\sqrt{M} measures the corresponding average absolute fluctuation entering through the MAE contribution. Both depend on $|G(\eta_p)|$: they are small near the origin and typically more pronounced at larger frequencies. Since $|G(\eta_p)| \leq 1$ for every p ,

$$0 \leq V_P \leq 1, \quad 0 \leq W_P \leq 1,$$

and therefore

$$\frac{V_P}{M} = O(M^{-1}), \quad \frac{W_P}{\sqrt{M}} = O(M^{-1/2}), \quad M \rightarrow \infty,$$

uniformly in P . In particular, unless $\lambda = 0$, the MAE-driven sampling term is asymptotically dominant among the statistical contributions.

4 Pseudo-samples from dependent data Sections 2–3 treat the idealized setting of direct i.i.d. sampling from the target law. In many applications, however, the available observations form a dependent time series, while the object of interest is the law or transition density over a fixed horizon $\tau > 0$. A practical way to recover conditionally i.i.d. training data is to construct pseudo-samples from the observed history by a suitable resampling procedure; see, for example, [14]. Representative techniques include the moving block bootstrap [13], the stationary bootstrap [20], subsampling [21], and, when an explicit parametric model is postulated, the parametric bootstrap [8, 5].

A key advantage of the Fourier-mixture framework is that dependence in the observed data can be handled without changing the basic structure of the error analysis: conditional on the observed history, the argument of Section 3 carries over, and the only new error source is a pseudo-law approximation term measuring the discrepancy between the resampling-based law and the true τ -horizon law.

4.1 Resampling-based pseudo-sampling Fix a horizon $\tau > 0$, interpreted as the length of one decision period or one transition step. Let X_τ denote a generic true increment over horizon τ , and let $\mathcal{L}_\tau = \mathcal{L}(X_\tau)$ be its law. We assume that this target law on \mathbb{R} admits a density g and CF G , exactly as in Assumption 2.1.

Let $\mathcal{H}_N = (Y_1, \dots, Y_N)$ denote an observed dependent sample recorded at the sampling frequency available in the data. This sampling frequency need not coincide with the target horizon τ . A resampling procedure applied to \mathcal{H}_N is intended to approximate the target law \mathcal{L}_τ by a random pseudo-law $\mathcal{L}_{\tau,N}^\dagger$. For example, when the data are observed at a finer frequency than τ , one may resample shorter-horizon observations by a dependence-preserving scheme and then aggregate them over intervals of length τ .

We write

$$\mathbb{E}^\dagger[\cdot] = \mathbb{E}[\cdot | \mathcal{H}_N], \quad \text{Var}^\dagger(\cdot) = \text{Var}(\cdot | \mathcal{H}_N)$$

for conditional expectation and variance given the observed history. The basic conditional assumption is that, given \mathcal{H}_N , the pseudo-samples $X_1^\dagger, \dots, X_M^\dagger$ are i.i.d. from the common pseudo-law $\mathcal{L}_{\tau,N}^\dagger$, namely

$$X_1^\dagger, \dots, X_M^\dagger | \mathcal{H}_N \stackrel{\text{i.i.d.}}{\sim} \mathcal{L}_{\tau,N}^\dagger, \quad \mathcal{L}_{\tau,N}^\dagger \approx \mathcal{L}_\tau.$$

Let X^\dagger denote a generic draw from $\mathcal{L}_{\tau,N}^\dagger$. The corresponding pseudo-CF is

$$G_N^\dagger(\eta) = \mathbb{E}^\dagger \left[e^{i\eta X^\dagger} \right], \quad \eta \in \mathbb{R}.$$

The associated empirical pseudo-CF is

$$(4.1) \quad \widehat{G}_M^\dagger(\eta) = \frac{1}{M} \sum_{m=1}^M e^{i\eta X_m^\dagger}.$$

On the training nodes $\{\eta_p\}_{p=1}^P \subset [-\eta', \eta']$, we define

$$(4.2) \quad \begin{aligned} B_{P,N}^{(1)} &= \frac{1}{P} \sum_{p=1}^P \left(|\operatorname{Re}(G(\eta_p) - G_N^\dagger(\eta_p))| + |\operatorname{Im}(G(\eta_p) - G_N^\dagger(\eta_p))| \right), \\ B_{P,N}^{(2)} &= \frac{1}{P} \sum_{p=1}^P |G(\eta_p) - G_N^\dagger(\eta_p)|^2. \end{aligned}$$

These quantities are random through their dependence on \mathcal{H}_N and measure how accurately the pseudo-law reproduces the target law on the training Fourier grid.

The Fourier-domain pseudo-sample loss is defined exactly as in Section 2.3, but with \widehat{G}_M replaced by \widehat{G}_M^\dagger :

$$(4.3) \quad \operatorname{Loss}_{M,P}^\dagger(\theta) = \operatorname{MSE}_{M,P}^\dagger(\theta) + \lambda R_{M,P}^\dagger(\theta), \quad \theta \in \widehat{\Theta} \subseteq \Theta,$$

where

$$(4.4) \quad \begin{aligned} \operatorname{MSE}_{M,P}^\dagger(\theta) &= \frac{1}{P} \sum_{p=1}^P \left(\Delta_{p,\operatorname{Re}}^\dagger(\theta)^2 + \Delta_{p,\operatorname{Im}}^\dagger(\theta)^2 \right), \\ R_{M,P}^\dagger(\theta) &= \frac{1}{P} \sum_{p=1}^P \left(|\Delta_{p,\operatorname{Re}}^\dagger(\theta)| + |\Delta_{p,\operatorname{Im}}^\dagger(\theta)| \right), \end{aligned}$$

with

$$\Delta_{p,\operatorname{Re}}^\dagger(\theta) = \operatorname{Re} \widehat{G}_M^\dagger(\eta_p) - \operatorname{Re} \widehat{G}(\eta_p; \theta), \quad \Delta_{p,\operatorname{Im}}^\dagger(\theta) = \operatorname{Im} \widehat{G}_M^\dagger(\eta_p) - \operatorname{Im} \widehat{G}(\eta_p; \theta).$$

Similarly, define the exact loss associated with $\mathcal{L}_{\tau,N}^\dagger$ by replacing \widehat{G}_M^\dagger with G_N^\dagger :

$$(4.5) \quad \operatorname{Loss}_{P,N}^{\dagger,*}(\theta) = \operatorname{MSE}_{P,N}^{\dagger,*}(\theta) + \lambda R_{P,N}^{\dagger,*}(\theta),$$

where $\operatorname{MSE}_{P,N}^{\dagger,*}(\theta)$ and $R_{P,N}^{\dagger,*}(\theta)$ are defined analogously to (4.4), with residuals

$$\Delta_{p,\operatorname{Re}}^{\dagger,*}(\theta) = \operatorname{Re} G_N^\dagger(\eta_p) - \operatorname{Re} \widehat{G}(\eta_p; \theta), \quad \Delta_{p,\operatorname{Im}}^{\dagger,*}(\theta) = \operatorname{Im} G_N^\dagger(\eta_p) - \operatorname{Im} \widehat{G}(\eta_p; \theta).$$

4.2 Error analysis under resampling-based pseudo-sampling We now show that the error analysis of Section 3 extends to the pseudo-sample setting. The first step is the pseudo-sample analogue of Lemma 3.1 presented in Lemma 4.1 below.

LEMMA 4.1 (Pseudo-sample unbiasedness and variance). *For each fixed $\eta \in \mathbb{R}$,*

$$(4.6) \quad \mathbb{E}^\dagger \left[\widehat{G}_M^\dagger(\eta) \right] = G_N^\dagger(\eta), \quad \operatorname{Var}^\dagger \left(\widehat{G}_M^\dagger(\eta) \right) = \frac{1 - |G_N^\dagger(\eta)|^2}{M}.$$

Moreover, with

$$(4.7) \quad V_{P,N}^\dagger = \frac{1}{P} \sum_{p=1}^P (1 - |G_N^\dagger(\eta_p)|^2), \quad W_{P,N}^\dagger = \frac{1}{P} \sum_{p=1}^P \sqrt{1 - |G_N^\dagger(\eta_p)|^2},$$

we have, for every fixed $\theta \in \widehat{\Theta}$,

$$(4.8) \quad \mathbb{E}^\dagger [\operatorname{MSE}_{M,P}^\dagger(\theta)] = \operatorname{MSE}_{P,N}^{\dagger,*}(\theta) + \frac{V_{P,N}^\dagger}{M}, \quad \mathbb{E}^\dagger [R_{M,P}^\dagger(\theta)] \leq R_{P,N}^{\dagger,*}(\theta) + \sqrt{\frac{2}{M}} W_{P,N}^\dagger.$$

Consequently,

$$(4.9) \quad \mathbb{E}^\dagger \left[\operatorname{Loss}_{M,P}^\dagger(\theta) \right] \leq \operatorname{Loss}_{P,N}^{\dagger,*}(\theta) + \frac{V_{P,N}^\dagger}{M} + \lambda \sqrt{\frac{2}{M}} W_{P,N}^\dagger.$$

For a proof of Lemma 4.1, see Appendix C.

We now state the pseudo-sample analogue of the expected L_2 error bound for the direct i.i.d. sampling setting in Theorem 3.4. The pseudo-sample bound has the same structure, with only the additional discrepancy terms associated with $\mathcal{L}_{\tau,N}^\dagger$.

THEOREM 4.2 (L_2 error bound under resampling-based pseudo-sampling). *Fix $\varepsilon > 0$, and choose $[-\eta', \eta']$ sufficiently large so that Lemma 3.3 applies. Let $\hat{\theta}^\dagger \in \widehat{\Theta}$ be obtained by minimizing (4.3), and suppose that*

$$(4.10) \quad \text{Loss}_{M,P}^\dagger(\hat{\theta}^\dagger) \leq \varepsilon_1$$

for some $\varepsilon_1 > 0$. Then

$$(4.11) \quad \mathbb{E}^\dagger \left[\int_{\mathbb{R}} |g(x) - \widehat{g}(x; \hat{\theta}^\dagger)|^2 dx \right] < \frac{1}{2\pi} \left(4\varepsilon + C_1 \left(4\varepsilon_1 + 2B_{P,N}^{(2)} + \lambda B_{P,N}^{(1)} \right. \right. \\ \left. \left. + \frac{4V_{P,N}^\dagger}{M} + 2\lambda \sqrt{\frac{2}{M}} W_{P,N}^\dagger \right) + \frac{C' C_1^2}{P} \right).$$

Here, C_1 is defined in (2.16), $B_{P,N}^{(2)}$ and $B_{P,N}^{(1)}$ in (4.2), $V_{P,N}^\dagger$ and $W_{P,N}^\dagger$ in (4.7), and C' in Lemma 3.2.

Proof. By Plancherel's theorem,

$$2\pi \int_{\mathbb{R}} |g(x) - \widehat{g}(x; \hat{\theta}^\dagger)|^2 dx = \int_{\mathbb{R}} |G(\eta) - \widehat{G}(\eta; \hat{\theta}^\dagger)|^2 d\eta.$$

Applying the same truncation and discretization arguments as in the proof of Theorem 3.4 gives

$$(4.12) \quad \int_{\mathbb{R}} |G(\eta) - \widehat{G}(\eta; \hat{\theta}^\dagger)|^2 d\eta < 4\varepsilon + C_1 \text{Loss}_P^*(\hat{\theta}^\dagger) + \frac{C' C_1^2}{P},$$

where Loss_P^* is the exact loss associated with the true law \mathcal{L}_τ .

We next compare the true-law loss with its pseudo-law counterpart. Since

$$|G(\eta_p) - \widehat{G}(\eta_p; \hat{\theta}^\dagger)|^2 \leq 2|G(\eta_p) - G_N^\dagger(\eta_p)|^2 + 2|G_N^\dagger(\eta_p) - \widehat{G}(\eta_p; \hat{\theta}^\dagger)|^2,$$

we obtain

$$\text{MSE}_P^*(\hat{\theta}^\dagger) \leq 2B_{P,N}^{(2)} + 2\text{MSE}_{P,N}^{\dagger,*}(\hat{\theta}^\dagger).$$

Likewise,

$$R_P^*(\hat{\theta}^\dagger) \leq B_{P,N}^{(1)} + R_{P,N}^{\dagger,*}(\hat{\theta}^\dagger),$$

so

$$(4.13) \quad \text{Loss}_P^*(\hat{\theta}^\dagger) \leq 2\text{Loss}_{P,N}^{\dagger,*}(\hat{\theta}^\dagger) + 2B_{P,N}^{(2)} + \lambda B_{P,N}^{(1)}.$$

Repeating the argument used in the proof of Theorem 3.4, but with G replaced by G_N^\dagger , yields

$$\mathbb{E}^\dagger \left[\text{Loss}_{P,N}^{\dagger,*}(\hat{\theta}^\dagger) \right] \leq 2\varepsilon_1 + \frac{2V_{P,N}^\dagger}{M} + \lambda \sqrt{\frac{2}{M}} W_{P,N}^\dagger.$$

Combining this with (4.13) gives

$$\mathbb{E}^\dagger \left[\text{Loss}_P^*(\hat{\theta}^\dagger) \right] \leq 4\varepsilon_1 + 2B_{P,N}^{(2)} + \lambda B_{P,N}^{(1)} + \frac{4V_{P,N}^\dagger}{M} + 2\lambda \sqrt{\frac{2}{M}} W_{P,N}^\dagger.$$

Substituting this into (4.12) proves (4.11). \square

Remark 4.3 (Interpretation of the pseudo-law terms). Compared with Theorem 3.4, the new contributions are $B_{P,N}^{(2)}$ and $B_{P,N}^{(1)}$. These quantify the discrepancy between the true law \mathcal{L}_τ and the resampling law $\mathcal{L}_{\tau,N}^\dagger$ on the chosen Fourier grid. The terms $V_{P,N}^\dagger/M$ and $W_{P,N}^\dagger/\sqrt{M}$ play the same roles as the corresponding empirical-CF sampling terms in Theorem 3.4, but now relative to the pseudo-law rather than the true law. Hence the error decomposition contains five ingredients: truncation, empirical training error, discretization, empirical pseudo-CF sampling error, and pseudo-law approximation error.

Remark 4.4 (Resampling procedures and practical interpretation). Assume that the resampling procedure is consistent for the fixed-horizon law on the chosen Fourier grid, in the sense that

$$B_{P,N}^{(2)} \rightarrow 0, \quad B_{P,N}^{(1)} \rightarrow 0, \quad N \rightarrow \infty,$$

in probability or in mean. Then, the additional pseudo-law discrepancy terms in Theorem 4.2 vanish asymptotically. Moreover, since G_N^\dagger is a CF, $|G_N^\dagger(\eta_p)| \leq 1$ for every p , and thus we have

$$0 \leq V_{P,N}^\dagger \leq 1, \quad 0 \leq W_{P,N}^\dagger \leq 1.$$

Hence Theorem 4.2 implies the simplified bound

$$\mathbb{E}^\dagger \left[\int_{\mathbb{R}} |g(x) - \widehat{g}(x; \widehat{\theta}^\dagger)|^2 dx \right] < \frac{1}{2\pi} \left(4\varepsilon + C_1 \left(4\varepsilon_1 + 2B_{P,N}^{(2)} + \lambda B_{P,N}^{(1)} + \frac{4}{M} + 2\lambda \sqrt{\frac{2}{M}} \right) + \frac{C' C_1^2}{P} \right).$$

Thus, the pseudo-sample estimate has the same dependence on M and P as in the direct i.i.d. setting, up to the additional N -dependent discrepancy terms $B_{P,N}^{(1)}$ and $B_{P,N}^{(2)}$. For large N , these terms become negligible when the resampling procedure is consistent. This provides a theoretical basis for real-data experiments in which a dependent time series is first converted into approximately i.i.d. pseudo-samples by a resampling procedure and the Fourier-mixture model is then trained on the corresponding empirical pseudo-CF. For financial time series, block bootstrap methods are a natural example; see Section 7.

5 Multi-dimensional extension We now extend the construction to dimension $d \geq 1$. We distinguish between the direct i.i.d. sampling setting and its pseudo-sample extension.

5.1 Direct i.i.d. sampling Throughout this section, bold symbols denote vectors or matrices in \mathbb{R}^d or $\mathbb{R}^{d \times d}$. Let $\mathbf{X}_1, \dots, \mathbf{X}_M$ be i.i.d. observations in \mathbb{R}^d from an unknown density g , and let G denote the Fourier transform of g . We seek to estimate g from the empirical CF \widehat{G}_M . Our Fourier convention is

$$(5.1) \quad G(\boldsymbol{\eta}) = \int_{\mathbb{R}^d} e^{i\boldsymbol{\eta}^\top \mathbf{x}} g(\mathbf{x}) d\mathbf{x}, \quad g(\mathbf{x}) = \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} e^{-i\boldsymbol{\eta}^\top \mathbf{x}} G(\boldsymbol{\eta}) d\boldsymbol{\eta},$$

and the empirical CF is

$$(5.2) \quad \widehat{G}_M(\boldsymbol{\eta}) = \frac{1}{M} \sum_{m=1}^M e^{i\boldsymbol{\eta}^\top \mathbf{x}_m}, \quad \boldsymbol{\eta} \in \mathbb{R}^d.$$

For $\boldsymbol{\mu}, \boldsymbol{\nu} \in \mathbb{R}^d$ and symmetric positive-definite matrices $\boldsymbol{\Sigma}, \boldsymbol{\Lambda} \in \mathbb{R}^{d \times d}$, define the multivariate Gaussian and elliptical Laplace kernels

$$(5.3) \quad \varphi_G^{(d)}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})\right),$$

$$(5.4) \quad \varphi_{\mathcal{L}}^{(d)}(\mathbf{x}; \boldsymbol{\nu}, \boldsymbol{\Lambda}) = \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Lambda}|^{1/2}} r(\mathbf{x}; \boldsymbol{\nu}, \boldsymbol{\Lambda})^{1-d/2} K_{d/2-1}(r(\mathbf{x}; \boldsymbol{\nu}, \boldsymbol{\Lambda})),$$

where

$$r(\mathbf{x}; \boldsymbol{\nu}, \boldsymbol{\Lambda}) = \sqrt{(\mathbf{x} - \boldsymbol{\nu})^\top \boldsymbol{\Lambda}^{-1} (\mathbf{x} - \boldsymbol{\nu})},$$

and $K_\alpha(\cdot)$ is the modified Bessel function of the second kind.

The corresponding Gaussian–Laplace mixture is

$$(5.5) \quad \widehat{g}(\mathbf{x}; \theta) = \sum_{k=1}^{K_G} \beta_k^{(G)} \varphi_G^{(d)}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \sum_{\ell=1}^{K_{\mathcal{L}}} \beta_\ell^{(\mathcal{L})} \varphi_{\mathcal{L}}^{(d)}(\mathbf{x}; \boldsymbol{\nu}_\ell, \boldsymbol{\Lambda}_\ell),$$

with nonnegative weights summing to one. As in 1D, the weights are obtained by a joint softmax, while the covariance matrices are parameterized by Cholesky factors with softplus-transformed diagonals.

The learned CF again admits a closed form:

$$(5.6) \quad \widehat{G}(\boldsymbol{\eta}; \theta) = \sum_{k=1}^{K_G} \beta_k^{(G)} \exp\left(i\boldsymbol{\eta}^\top \boldsymbol{\mu}_k - \frac{1}{2}\boldsymbol{\eta}^\top \boldsymbol{\Sigma}_k \boldsymbol{\eta}\right) + \sum_{\ell=1}^{K_{\mathcal{L}}} \beta_\ell^{(\mathcal{L})} \frac{\exp(i\boldsymbol{\eta}^\top \boldsymbol{\nu}_\ell)}{1 + \boldsymbol{\eta}^\top \boldsymbol{\Lambda}_\ell \boldsymbol{\eta}}.$$

Accordingly,

$$(5.7) \quad \operatorname{Re} \widehat{G}(\boldsymbol{\eta}; \theta) = \sum_{k=1}^{K_G} \beta_k^{(G)} \exp\left(-\frac{1}{2}\boldsymbol{\eta}^\top \boldsymbol{\Sigma}_k \boldsymbol{\eta}\right) \cos(\boldsymbol{\eta}^\top \boldsymbol{\mu}_k) + \sum_{\ell=1}^{K_{\mathcal{L}}} \beta_\ell^{(\mathcal{L})} \frac{\cos(\boldsymbol{\eta}^\top \boldsymbol{\nu}_\ell)}{1 + \boldsymbol{\eta}^\top \boldsymbol{\Lambda}_\ell \boldsymbol{\eta}},$$

$$(5.8) \quad \operatorname{Im} \widehat{G}(\boldsymbol{\eta}; \theta) = \sum_{k=1}^{K_G} \beta_k^{(G)} \exp\left(-\frac{1}{2}\boldsymbol{\eta}^\top \boldsymbol{\Sigma}_k \boldsymbol{\eta}\right) \sin(\boldsymbol{\eta}^\top \boldsymbol{\mu}_k) + \sum_{\ell=1}^{K_{\mathcal{L}}} \beta_\ell^{(\mathcal{L})} \frac{\sin(\boldsymbol{\eta}^\top \boldsymbol{\nu}_\ell)}{1 + \boldsymbol{\eta}^\top \boldsymbol{\Lambda}_\ell \boldsymbol{\eta}}.$$

Training proceeds exactly as in Subsection 2.3, with Fourier domain $[-\eta', \eta']^d$. For training nodes $\{\boldsymbol{\eta}_p\}_{p=1}^P \subset [-\eta', \eta']^d$, define the residuals

$$\Delta_{p,\operatorname{Re}}(\theta) = \operatorname{Re} \widehat{G}_M(\boldsymbol{\eta}_p) - \operatorname{Re} \widehat{G}(\boldsymbol{\eta}_p; \theta), \quad \Delta_{p,\operatorname{Im}}(\theta) = \operatorname{Im} \widehat{G}_M(\boldsymbol{\eta}_p) - \operatorname{Im} \widehat{G}(\boldsymbol{\eta}_p; \theta),$$

and use the loss

$$(5.9) \quad \operatorname{Loss}_{M,P}^{(d)}(\theta) = \frac{1}{P} \sum_{p=1}^P \left(\Delta_{p,\operatorname{Re}}(\theta)^2 + \Delta_{p,\operatorname{Im}}(\theta)^2 \right) + \frac{\lambda}{P} \sum_{p=1}^P \left(|\Delta_{p,\operatorname{Re}}(\theta)| + |\Delta_{p,\operatorname{Im}}(\theta)| \right).$$

Assume that P is a perfect d th power and that $[-\eta', \eta']^d$ is partitioned by a deterministic tensor-product grid with $P^{1/d}$ nodes in each coordinate direction. Let $\delta_{\max} = C_1/P^{1/d}$. We define

$$(5.10) \quad V_P^{(d)} = \frac{1}{P} \sum_{p=1}^P (1 - |G(\boldsymbol{\eta}_p)|^2), \quad W_P^{(d)} = \frac{1}{P} \sum_{p=1}^P \sqrt{1 - |G(\boldsymbol{\eta}_p)|^2}.$$

THEOREM 5.1 (Expected L_2 error bound in dimension d). *Assume that g is a probability density in $L_\infty(\mathbb{R}^d)$ and that G is Lipschitz continuous on every compact subset of \mathbb{R}^d . Fix $\varepsilon > 0$, and choose $\eta' > 0$ sufficiently large so that the Fourier tails of G and $\widehat{G}(\cdot; \theta)$ outside $[-\eta', \eta']^d$ are bounded by ε , uniformly in $\theta \in \Theta$. Let $\widehat{\theta}$ be obtained by minimizing (5.9), and suppose that*

$$\operatorname{Loss}_{M,P}^{(d)}(\widehat{\theta}) \leq \varepsilon_1$$

for some $\varepsilon_1 > 0$. Then

$$\mathbb{E} \left[\int_{\mathbb{R}^d} |g(\mathbf{x}) - \widehat{g}(\mathbf{x}; \widehat{\theta})|^2 d\mathbf{x} \right] < \frac{1}{(2\pi)^d} \left(4\varepsilon + C_1^d (2\varepsilon_1 + \frac{2V_P^{(d)}}{M} + \lambda \sqrt{\frac{2}{M}} W_P^{(d)}) + \frac{C'_d C_1^{d+1}}{P^{1/d}} \right),$$

where $C'_d = C'_d(\eta')$ is independent of P . Since $|G(\boldsymbol{\eta}_p)| \leq 1$ for every p , we have

$$0 \leq V_P^{(d)} \leq 1, \quad 0 \leq W_P^{(d)} \leq 1,$$

and hence

$$\mathbb{E} \left[\int_{\mathbb{R}^d} |g(\mathbf{x}) - \widehat{g}(\mathbf{x}; \widehat{\theta})|^2 d\mathbf{x} \right] < \frac{1}{(2\pi)^d} \left(4\varepsilon + C_1^d (2\varepsilon_1 + \frac{2}{M} + \lambda \sqrt{\frac{2}{M}}) + \frac{C'_d C_1^{d+1}}{P^{1/d}} \right).$$

For brevity, a proof is omitted, since the argument follows the same steps as in Section 3: Plancherel's theorem is replaced by its d -dimensional counterpart, and the 1D quadrature estimate is replaced by the tensor-product version on $[-\eta', \eta']^d$. The empirical CF sampling terms remain of order M^{-1} and $M^{-1/2}$, respectively, whereas the deterministic discretization term deteriorates to order $P^{-1/d}$.

5.2 Resampling-based pseudo-sampling The pseudo-sample framework of Section 4 also extends directly to dimension $d \geq 1$. Let \mathcal{H}_N denote the observed dependent history, and suppose that, conditional on \mathcal{H}_N , the pseudo-samples

$$\mathbf{X}_1^\dagger, \dots, \mathbf{X}_M^\dagger \in \mathbb{R}^d \quad \text{are i.i.d. from a common pseudo-law} \quad \mathcal{L}_{\tau, N}^{\dagger, (d)}.$$

Let $G_N^{\dagger, (d)}$ denote the CF of this pseudo-law, and define the empirical pseudo-CF by

$$(5.11) \quad \widehat{G}_M^{\dagger, (d)}(\boldsymbol{\eta}) = \frac{1}{M} \sum_{m=1}^M e^{i\boldsymbol{\eta}^\top \mathbf{x}_m^\dagger}.$$

Define the multivariate pseudo-law discrepancy terms

$$(5.12) \quad B_{P, N}^{(1, d)} = \frac{1}{P} \sum_{p=1}^P \left(\left| \operatorname{Re}(G(\boldsymbol{\eta}_p) - G_N^{\dagger, (d)}(\boldsymbol{\eta}_p)) \right| + \left| \operatorname{Im}(G(\boldsymbol{\eta}_p) - G_N^{\dagger, (d)}(\boldsymbol{\eta}_p)) \right| \right),$$

$$B_{P, N}^{(2, d)} = \frac{1}{P} \sum_{p=1}^P |G(\boldsymbol{\eta}_p) - G_N^{\dagger, (d)}(\boldsymbol{\eta}_p)|^2,$$

as well as

$$(5.13) \quad V_{P, N}^{\dagger, (d)} = \frac{1}{P} \sum_{p=1}^P (1 - |G_N^{\dagger, (d)}(\boldsymbol{\eta}_p)|^2), \quad W_{P, N}^{\dagger, (d)} = \frac{1}{P} \sum_{p=1}^P \sqrt{1 - |G_N^{\dagger, (d)}(\boldsymbol{\eta}_p)|^2}.$$

If $\widehat{\theta}^\dagger$ is obtained by minimizing the multivariate pseudo-sample loss and satisfies

$$\operatorname{Loss}_{M, P}^{\dagger, (d)}(\widehat{\theta}^\dagger) \leq \varepsilon_1,$$

then the same argument as in Section 4 yields the conditional estimate

$$(5.14) \quad \mathbb{E}^\dagger \left[\int_{\mathbb{R}^d} |g(\mathbf{x}) - \widehat{g}(\mathbf{x}; \widehat{\theta}^\dagger)|^2 d\mathbf{x} \right] < \frac{1}{(2\pi)^d} \left(4\varepsilon + C_1^d (4\varepsilon_1 + 2B_{P, N}^{(2, d)} + \lambda B_{P, N}^{(1, d)} + \frac{4V_{P, N}^{\dagger, (d)}}{M} + 2\lambda \sqrt{\frac{2}{M}} W_{P, N}^{\dagger, (d)}) + \frac{C'_d C_1^{d+1}}{P^{1/d}} \right).$$

Thus the pseudo-sample extension in \mathbb{R}^d has the same structure as in one dimension: one retains truncation, empirical training error, discretization, and empirical pseudo-CF sampling error, together with the additional pseudo-law approximation terms $B_{P, N}^{(1, d)}$ and $B_{P, N}^{(2, d)}$.

6 Complexity analysis We now discuss the computational complexity in both the direct i.i.d. sampling setting and the resampling-based pseudo-sampling setting. Recall that M denotes the number of samples used to construct the empirical CF, P the number of training frequencies, and (K_G, K_L) the numbers of Gaussian and Laplace components. We write $K = K_G + K_L$ and let B and E respectively denote the minibatch size and the total number of training epochs in Algorithm 2.1.

6.1 Direct i.i.d. sampling

One-dimensional case. The computational cost of Algorithm 2.1 is determined by: (i) preprocessing of the empirical CF, (ii) repeated forward evaluations of the learned CF on minibatches of training nodes, and (iii) the corresponding backward passes and parameter updates.

The empirical CF values are computed via $\widehat{G}_M(\eta_p) = \frac{1}{M} \sum_{m=1}^M e^{i\eta_p X_m}$, $p = 1, \dots, P$, which costs $\mathcal{O}(MP)$. This is a one-off cost for a fixed dataset and node set. Optional affine preprocessing requires only $\mathcal{O}(M)$ operations.

At one training node η_p , each Gaussian component contributes $\beta_k^{(G)} e^{i\eta_p \mu_k - \sigma_k^2 \eta_p^2 / 2}$, and each Laplace component contributes $\beta_\ell^{(L)} \frac{e^{i\eta_p \nu_\ell}}{1 + b_\ell^2 \eta_p^2}$. Hence, in the forward pass, the cost at one node is $\mathcal{O}(K)$, and over a minibatch of size B it is $\mathcal{O}(BK)$. The softmax and softplus evaluations contribute only $\mathcal{O}(K)$ per forward pass. The loss function evaluation is $\mathcal{O}(B)$, and the backward pass plus parameter update has the same asymptotic order as the forward pass. Therefore one optimization step costs $\mathcal{O}(BK)$, one epoch costs $\mathcal{O}(PK)$, and E epochs cost $\mathcal{O}(EPK)$.

Combining preprocessing and training gives

$$(6.1) \quad \mathcal{O}(MP) + \mathcal{O}(EPK).$$

Thus, once \widehat{G}_M is formed, the iterative part of the algorithm no longer depends on M .

d-dimensional case. If the model is extended to \mathbb{R}^d with full covariance matrices, then constructing the empirical CF at P nodes costs $\mathcal{O}(MPd)$, since each factor $e^{i\eta_p^\top \mathbf{X}_m}$ requires one inner product in \mathbb{R}^d . Each Gaussian or Laplace component at one training node requires evaluating a linear form $\boldsymbol{\eta}^\top \boldsymbol{\mu}$ and a quadratic form $\boldsymbol{\eta}^\top \boldsymbol{\Sigma} \boldsymbol{\eta}$, which cost $\mathcal{O}(d)$ and $\mathcal{O}(d^2)$, respectively. Hence, the forward cost over a minibatch of size B is $\mathcal{O}(BKd^2)$, and the total training cost over E epochs is $\mathcal{O}(EPKd^2)$. Therefore, the total complexity in the full-covariance setting is

$$\mathcal{O}(MPd) + \mathcal{O}(EPKd^2).$$

This assumes that the covariance matrices are handled through their Cholesky factors, so that $\boldsymbol{\eta}^\top \boldsymbol{\Sigma} \boldsymbol{\eta}$ is evaluated directly without explicitly forming $\boldsymbol{\Sigma} = \mathbf{L}\mathbf{L}^\top$.

6.2 Resampling-based pseudo-sampling In the pseudo-sample framework of Section 4, the empirical CF is built from pseudo-samples rather than directly from the observed dependent sample. The total complexity therefore acquires an additional preprocessing term corresponding to pseudo-sample generation.

One-dimensional case. Let $\mathcal{C}_{\text{res}}(N, M, \tau)$ denote the cost of generating M pseudo-samples over horizon τ from the observed history \mathcal{H}_N . This cost depends on the chosen resampling procedure; for block-resampling methods for dependent data, see, for example, [13, 20, 14]. Since the exact cost depends on the resampling scheme and on how the horizon- τ pseudo-samples are represented, we keep $\mathcal{C}_{\text{res}}(N, M, \tau)$ abstract in the subsequent complexity analysis.

Once the pseudo-samples have been generated, the empirical pseudo-CF \widehat{G}_M^\dagger is formed exactly as in the direct-sampling setting, so the Fourier preprocessing cost

remains $\mathcal{O}(MP)$ and the iterative training cost remains $\mathcal{O}(EPK)$. Hence the total one-dimensional complexity under resampling-based pseudo-sampling is

$$(6.2) \quad \mathcal{O}(\mathcal{C}_{\text{res}}(N, M, \tau)) + \mathcal{O}(MP) + \mathcal{O}(EPK).$$

d-dimensional case. Let $\mathcal{C}_{\text{res}}^{(d)}(N, M, \tau)$ denote the cost of generating pseudo-samples $\mathbf{X}_1^\dagger, \dots, \mathbf{X}_M^\dagger \in \mathbb{R}^d$ by the resampling procedure. The empirical pseudo-CF construction then costs $\mathcal{O}(MPd)$, while training the full-covariance Gaussian–Laplace mixture costs $\mathcal{O}(EPKd^2)$. Therefore the total d -dimensional complexity under resampling-based pseudo-sampling is

$$(6.3) \quad \mathcal{O}(\mathcal{C}_{\text{res}}^{(d)}(N, M, \tau)) + \mathcal{O}(MPd) + \mathcal{O}(EPKd^2).$$

6.3 Comparison with Expectation–Maximization Consider a 1D Gaussian-mixture model fitted by the Expectation–Maximization (EM) algorithm; see [17]. If the model has K components, then one expectation-step and one maximization-step over M observations both scale linearly in MK , so the cost per iteration is $\mathcal{O}(MK)$. If T_{EM} iterations are required, the total cost of one fit is $\mathcal{O}(T_{\text{EM}}MK)$. With R random restarts and a search over several candidate values of K , the total cost grows as

$$(6.4) \quad \mathcal{O}(RT_{\text{EM}}M \sum_{K' \in \mathcal{K}} K'),$$

where \mathcal{K} is the set of tested model sizes.

Direct i.i.d. sampling setting. In one-dimension, the essential difference between (6.4) and (6.1) is that EM remains linear in M at every iteration, whereas the Fourier-mixture method depends on M only through the one-off construction of the empirical CF. After preprocessing, the iterative part of the algorithm scales only with P and K .

For full-covariance Gaussian components in d dimensions, one EM iteration scales as $\mathcal{O}(MKd^2)$, with an additional $\mathcal{O}(Kd^3)$ cost for covariance factorizations or inversions. Thus one full fit scales as $\mathcal{O}(T_{\text{EM}}(MKd^2 + Kd^3))$, and with restarts and model-order search the total cost becomes

$$\mathcal{O}(RT_{\text{EM}} \sum_{K' \in \mathcal{K}} (MK'd^2 + K'd^3)).$$

Resampling-based pseudo-sampling setting. A fair comparison should also include the cost of generating pseudo-samples. Thus, in one-dimension, EM trained on pseudo-samples scales as

$$\mathcal{O}(\mathcal{C}_{\text{res}}(N, M, \tau)) + \mathcal{O}(RT_{\text{EM}}M \sum_{K' \in \mathcal{K}} K'),$$

while in d dimensions the corresponding full-covariance complexity becomes

$$\mathcal{O}(\mathcal{C}_{\text{res}}^{(d)}(N, M, \tau)) + \mathcal{O}(RT_{\text{EM}} \sum_{K' \in \mathcal{K}} (MK'd^2 + K'd^3)).$$

By comparison, the Fourier-mixture method scales as (6.2) in one-dimension and (6.3) in d -dimensions. Thus the qualitative distinction persists in both settings: once the empirical (pseudo-)CF has been formed, the iterative stage of the Fourier-mixture method is decoupled from the sample size M , whereas EM continues to scale linearly in M at every iteration.

7 Numerical experiments This section is organized according to the two data settings developed in the theory. Subsection 7.1 reports experiments under direct i.i.d. sampling from known target laws, including Gaussian-mixture sanity checks, non-Gaussian benchmarks, a two-dimensional example, and an error-decay study linked to the error analysis of Section 3. Subsection 7.2 reports a resampling-based pseudo-sampling experiment built from Australian equity data, aligned with Section 4.

Unless otherwise stated, the one-dimensional experiments use $M = 10^6$ observations, a train/validation/test split $M_{\text{tr}} : M_{\text{val}} : M_{\text{te}} = 40\% : 30\% : 30\%$, $P = 10^3$ Fourier training nodes, Fourier truncated domain $[-\eta', \eta'] = [-50, 50]$, and minibatch size 1024. The training set \mathcal{D}_{tr} is used to construct the empirical CF (or empirical pseudo-CF), the validation set \mathcal{D}_{val} is used for model selection and early stopping when needed, and the test set \mathcal{D}_{te} is reserved for out-of-sample evaluation.

When the exact CF G is available in closed form, we report

$$L_2^{\text{Re}}(G, \hat{G}) = \int_{[-\eta', \eta']} |\text{Re } G(\eta) - \text{Re } \hat{G}(\eta)|^2 d\eta,$$

together with

$$\text{MPE}^{\text{Re}}(G, \hat{G}) = \max_{1 \leq q \leq Q} |\text{Re } G(\eta_q) - \text{Re } \hat{G}(\eta_q)|,$$

where $\{\eta_q\}_{q=1}^Q$ is a dense evaluation grid on $[-\eta', \eta']$, independent of the training grid; the imaginary-part quantities are defined analogously. When the exact density g is available, we also report

$$L_2(g, \hat{g}) = \int_{[-A, A]} |g(x) - \hat{g}(x)|^2 dx,$$

for $A > 0$ sufficiently large. When a test dataset is available, we report the per-sample negative log-likelihood

$$\text{NLL}(\hat{g}; \mathcal{D}_{\text{te}}) = -\frac{1}{M_{\text{te}}} \sum_{x \in \mathcal{D}_{\text{te}}} \log \hat{g}(x),$$

so that smaller values indicate a better out-of-sample fit.

7.1 Direct i.i.d. sampling

7.1.1 Gaussian-mixture sanity checks For the synthetic Gaussian-mixture benchmarks, the data are generated directly from the target densities by standard mixture sampling: a component label is drawn from the categorical distribution determined by the mixture weights, and then a sample is drawn from the corresponding Gaussian component. This yields i.i.d. observations from the target density and introduces no additional approximation error at the data-generation stage.

We consider two three-component Gaussian-mixture targets of the form

$$g(x) = \sum_{j=1}^3 \pi_j \varphi_{\mathcal{G}}(x; \mu_j, \sigma_j),$$

with common weights $(\pi_1, \pi_2, \pi_3) = (0.5, 0.3, 0.2)$. In the well-separated case we take $(\mu_1, \mu_2, \mu_3) = (-4, 0, 4)$ and $(\sigma_1, \sigma_2, \sigma_3) = (1, 1, 1)$, whereas in the overlapping case we take $(\mu_1, \mu_2, \mu_3) = (-2, 0, 2)$ and $(\sigma_1, \sigma_2, \sigma_3) = (1, 1, 2)$.

For these tests, we restrict the learned model to the Gaussian subclass by setting $K_{\mathcal{L}} = 0$ and varying $K_{\mathcal{G}}$. Figure 7.1 shows the effect of $K_{\mathcal{G}}$ on the density L_2 error, per-sample NLL, and training time. In both targets, the best mean density accuracy

is attained at $K_G = 3$, matching the number of components in the ground truth. Increasing the model size beyond this value does not improve the fit: the average L_2 error increases gradually, while the NLL remains nearly flat. This is the expected behaviour for a positive finite mixture trained on a well-specified target.

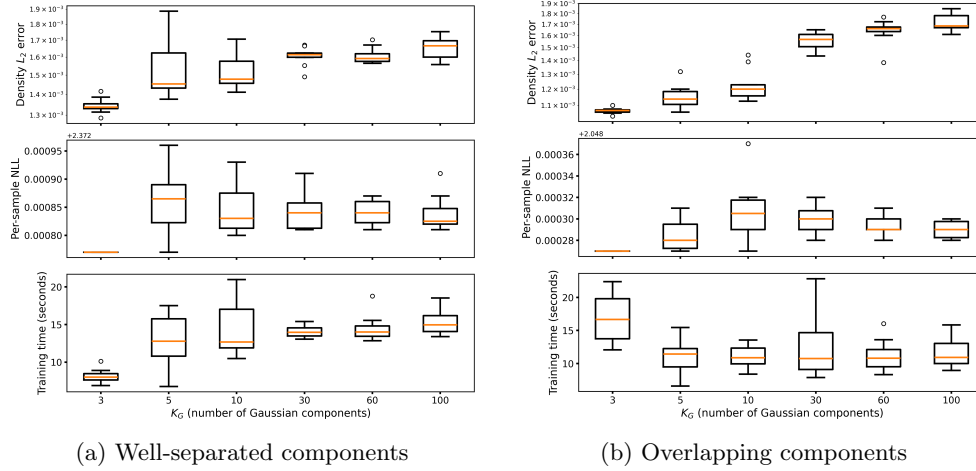


Fig. 7.1: Effect of the number K_G of learned Gaussian components on the density L_2 error, per-sample NLL, and training time. In both examples, the best mean density accuracy is attained at $K_G = 3$.

Table 7.1 compares the data-driven Fourier-mixture method with a capacity-matched Gaussian-mixture model fitted by EM, with $K_G = K_{EM} = 3$.

Table 7.1: Capacity-matched comparison between the data-driven Fourier-mixture method (Gaussian subclass) and EM on the three-component Gaussian-mixture targets.

Method	Target	Components	$L_2(g, \hat{g})$	NLL
Fourier-mixture	Well-separated	$K_G = 3$	1.32×10^{-3}	2.373
EM	Well-separated	$K_{EM} = 3$	1.07×10^{-3}	2.370
Fourier-mixture	Overlapping	$K_G = 3$	1.05×10^{-3}	2.048
EM	Overlapping	$K_{EM} = 3$	2.29×10^{-3}	2.048

In the well-separated case, the two methods are essentially comparable in both NLL and density L_2 error, with EM slightly better in L_2 . In the overlapping case, however, the proposed method attains a substantially smaller density L_2 error while maintaining essentially the same NLL. Thus the method behaves correctly on standard Gaussian-mixture benchmarks and remains competitive with EM in the classical Gaussian setting.

7.1.2 Heavy-tail and jump examples We next consider two non-Gaussian examples. The first is the Cauchy distribution, a canonical heavy-tailed benchmark for which both the density and the CF are available in closed form. The second is the Kou jump-diffusion model [12], a non-Gaussian jump model whose CF is also available in closed form.

For the Cauchy test, the training data are generated directly from the Cauchy distribution with location $x_0 = 0$ and scale $\gamma = 0.5$, whose density and CF are

$$g_{\text{Cau}}(x) = \frac{1}{\pi\gamma} \frac{1}{1 + ((x - x_0)/\gamma)^2}, \quad G_{\text{Cau}}(\eta) = \exp(ix_0\eta - \gamma|\eta|).$$

The Gaussian-mixture model corresponds to $(K_{\mathcal{G}}, K_{\mathcal{L}}) = (10, 0)$, while the Gaussian-Laplace mixture uses $(K_{\mathcal{G}}, K_{\mathcal{L}}) = (5, 5)$ and is trained against the empirical CF computed from the same sample. The exact CF is used only as an analytic reference for evaluation.

For the Kou example, we use the increment law of the Kou jump-diffusion process [12] over a fixed horizon $T = 10^{-3}$, with parameters

$$\mu = 0.05, \quad \sigma = 0.15, \quad \lambda = 0.1, \quad p = 0.3445, \quad \zeta_1 = 3.0465, \quad \zeta_2 = 3.0775,$$

following [7, Section 6.5]. Training samples are generated by the standard Brownian-plus-compound-Poisson representation of the increment law, and the exact Kou CF is used only as an analytic reference for evaluation. In this example, we use only a Gaussian mixture with $K_{\mathcal{G}} = 45$ and $K_{\mathcal{L}} = 0$, since the Gaussian subclass already provides a sufficiently accurate fit.

Table 7.2: Fourier-domain errors between G and \widehat{G} for the Cauchy and Kou examples.

Model (mixture)	$L_2^{\text{Re}}(\cdot)$	$L_2^{\text{Im}}(\cdot)$	$\text{MPE}^{\text{Re}}(\cdot)$	$\text{MPE}^{\text{Im}}(\cdot)$
Cauchy (Gaussian)	1.3×10^{-6}	4.2×10^{-7}	2.0×10^{-2}	1.6×10^{-3}
Cauchy (Gaussian-Laplace)	1.2×10^{-8}	3.0×10^{-9}	5.0×10^{-3}	7.2×10^{-4}
Kou (Gaussian)	1.8×10^{-9}	2.5×10^{-11}	1.3×10^{-4}	1.6×10^{-5}

Table 7.2 shows that adding Laplace components reduces the Fourier-domain error by one to two orders of magnitude for the Cauchy target in both the real and imaginary parts. By contrast, for the Kou model a Gaussian mixture with $K_{\mathcal{G}} = 45$ already achieves extremely small CF errors. Thus the Gaussian-Laplace extension is most beneficial when the target exhibits heavier tails or sharper local structure, rather than for every non-Gaussian model indiscriminately.

7.1.3 Two-dimensional Cauchy example To demonstrate that the multivariate extension is more than a formal generalization, we consider a genuinely heavy-tailed 2D benchmark. In the present test, the target is the product of two independent centered Cauchy marginals with common scale $\gamma = 0.5$, so that

$$g_{\text{Cau},2}(\mathbf{x}) = \prod_{j=1}^2 \frac{1}{\pi\gamma} \frac{1}{1 + (x_j/\gamma)^2}, \quad G_{\text{Cau},2}(\boldsymbol{\eta}) = \exp(-\gamma(|\eta_1| + |\eta_2|)),$$

for $\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$ and $\boldsymbol{\eta} = (\eta_1, \eta_2) \in \mathbb{R}^2$. The training data are generated directly from this product-Cauchy law and then used to form the empirical CF. We compare three learned models: two Gaussian-mixture models with $(K_{\mathcal{G}}, K_{\mathcal{L}}) = (30, 0)$ and $(40, 0)$, and one Gaussian-Laplace mixture with $(K_{\mathcal{G}}, K_{\mathcal{L}}) = (30, 20)$.

Table 7.3 shows that the Gaussian-Laplace model improves all reported CF error metrics relative to the Gaussian-only baselines. Enlarging the Gaussian-only model from $(30, 0)$ to $(40, 0)$ improves some imaginary-part errors but does not improve the real-part errors, whereas adding Laplace components improves both. Thus the Fourier-domain training strategy remains effective in two dimensions, and the Gaussian-Laplace model continues to offer a clear advantage on a genuinely heavy-tailed target.

Table 7.3: Fourier-domain errors between G and \widehat{G} for the 2D Cauchy example.

Model	(K_G, K_L)	$L_2^{\text{Re}}(\cdot)$	$L_2^{\text{Im}}(\cdot)$	$\text{MPE}^{\text{Re}}(\cdot)$	$\text{MPE}^{\text{Im}}(\cdot)$
Gauss.	(30,0)	3.30×10^{-2}	1.59×10^{-3}	2.42×10^{-2}	1.76×10^{-3}
Gauss.	(40,0)	4.65×10^{-2}	1.01×10^{-3}	2.83×10^{-2}	1.29×10^{-3}
Gauss.-Laplace	(30,20)	1.51×10^{-2}	5.36×10^{-4}	1.20×10^{-2}	2.93×10^{-4}

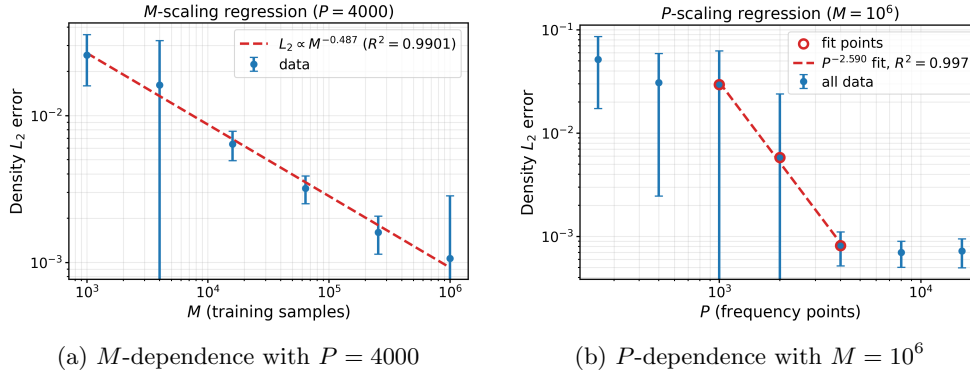
7.1.4 L_2 error decay study We conclude the direct i.i.d. experiments with an L_2 error-decay study motivated by Theorem 3.4. We use the Gaussian subclass $(K_G, K_L) = (3, 0)$ and target density $g(x) = \sum_{j=1}^3 \pi_j \varphi_G(x; \mu_j, \sigma_j)$, with $(\pi_1, \pi_2, \pi_3) = (0.5, 0.3, 0.2)$, $(\mu_1, \mu_2, \mu_3) = (-4, 0, 4)$, and $(\sigma_1, \sigma_2, \sigma_3) = (1, 1, 1)$. The model is therefore well specified. We fix $\eta' = 50$, use 30 independent repetitions per setting, take $P = 4000$ for the M -dependence study, and vary

$$M \in \{10^3, 4 \times 10^3, 1.6 \times 10^4, 6.4 \times 10^4, 2.56 \times 10^5, 10^6\}.$$

For the P -dependence study, we fix $M = 10^6$ and vary

$$P \in \{250, 500, 1000, 2000, 4000, 8000, 16,000\}.$$

The theory-linked quantities MSE_P^* , R_P^* , Loss_P^* , V_P , and W_P are also recorded. The

Fig. 7.2: L_2 error decay for the well-specified 3-GMM target.

fitted log-log slope for the M -dependence study is -0.4868 with $R^2 = 0.9901$, which is close to the theoretical $-1/2$ rate. This provides direct numerical support for the expected L_2 error bound in the well-specified case. The P -dependence curve shows a steep initial decrease followed by saturation, indicating that once the Fourier grid is sufficiently fine, the residual error is governed primarily by finite-sample noise in the empirical CF rather than by the grid itself.

As a secondary check, we also repeated the L_2 error-decay experiment for the Cauchy target from Subsection 7.1.2, with $x_0 = 0$ and $\gamma = 0.5$ (results not shown). This case is intentionally misspecified: a fixed finite Gaussian–Laplace mixture cannot represent the Cauchy density exactly, so the total error is expected to contain an approximation floor in addition to the empirical-CF sampling and discretization errors. This is precisely what is observed. For example, at $M = 10^6$ and $P = 4000$, increasing the model size from $(K_G, K_L) = (3, 3)$ to $(100, 30)$ and $(150, 45)$ reduces the realized L_2 error floor from about 1.82×10^{-2} to 9.04×10^{-3} and 8.80×10^{-3} , respectively.

The fitted transient M -dependence also moves closer to the well-specified $M^{-1/2}$ behavior, with the log-log slope changing from about -0.642 for $(K_G, K_L) = (3, 3)$ to about -0.604 for $(K_G, K_L) = (150, 45)$. Thus the Cauchy study illustrates the expected bias–variance picture: increasing model capacity reduces the approximation floor, while the well-specified 3-GMM study gives the cleanest direct validation of the theoretical $M^{-1/2}$ rate.

7.2 Resampling-based pseudo-sampling

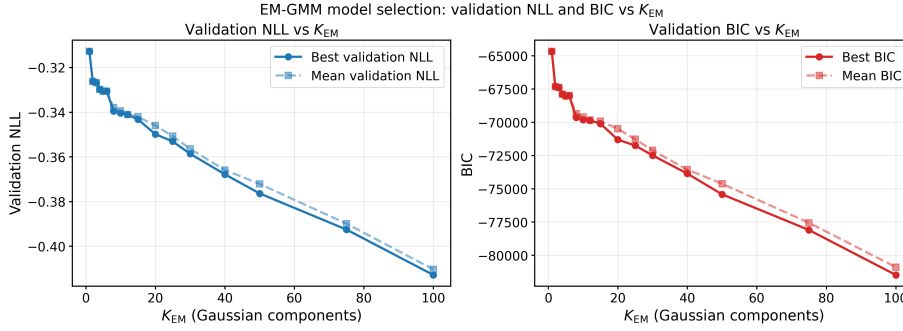
7.2.1 Data and processing The resampling-based pseudo-sampling experiment is built from a long monthly real return series for Australian equity, expressed in AUD. As a proxy for the Australian equity market, we use a capitalization-weighted total return series for the All Ordinaries index. Monthly price levels are taken from the Bloomberg All Ordinaries Price Index (AS30). Prior to the start of the Bloomberg All Ordinaries Accumulation Index in 1999, reinvested dividends are imputed using trailing dividend-yield data from [16] and the Reserve Bank of Australia. The nominal index is then deflated by the Australian CPI, yielding monthly real Australian equity returns over 1935:1–2022:12.

The target horizon in this experiment is one year. Starting from the monthly real return series, we generate many independent 30-year bootstrap paths using the stationary bootstrap of [20]. From each path, we extract one fixed annual observation and use the resulting one-year total returns $X_1^\dagger, \dots, X_M^\dagger$, with $M = 2.56 \times 10^5$, to construct the empirical pseudo-CF $\widehat{G}_M^\dagger(\eta) = \frac{1}{M} \sum_{m=1}^M e^{i\eta X_m^\dagger}$. Thus the experiment is based on a resampling law for one-year Australian equity returns rather than on directly observed i.i.d. annual data.

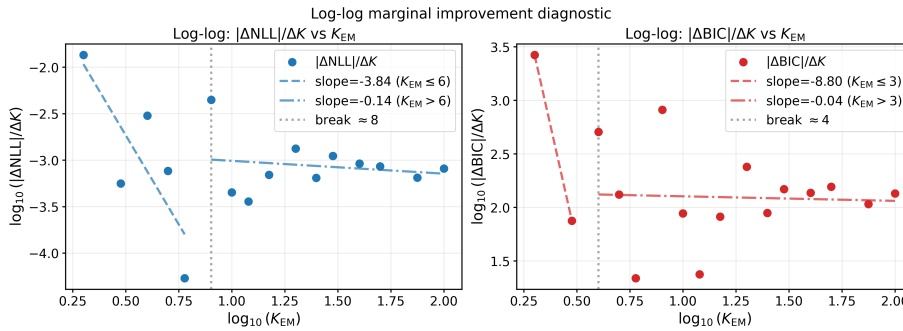
7.2.2 Australian equity pseudo-sample experiment We use a train/validation/test split $M_{\text{tr}} : M_{\text{val}} : M_{\text{te}} = 102,400 : 76,800 : 76,800$, $P = 1000$ uniformly spaced Fourier nodes on $[-50, 50]$, and the same training schedule as in the preceding experiments. Model sizes are chosen by validation NLL. For EM-GMM, however, naive minimum-NLL selection is unreliable in this pseudo-sample setting: because the resampled dataset contains repeated values, the likelihood can continue to improve as Gaussian components shrink around discrete atoms. We therefore test EM-GMM component counts across a range of K_{EM} and inspect the validation-NLL and marginal-improvement profiles in Figure 7.3. The validation NLL continues to improve for larger models, but the marginal gain drops sharply after $K_{\text{EM}} = 8$; beyond this point, additional components mainly add local flexibility around repeated pseudo-sample values. We therefore use the parsimonious EM-GMM benchmark with $K_{\text{EM}} = 8$ as the main comparison.

Table 7.4 shows that Fourier-GLMM achieves the best validation and test NLL under the parsimonious EM comparison, while also giving the smallest real-part and imaginary-part CF errors. The advantage over EM is clearest in Fourier space, which is consistent with the fact that the Fourier models are trained against CF data rather than directly by maximum likelihood. Figure 7.4 confirms this visually: the Fourier-based fitted CFs track the empirical test pseudo-CF more closely than the EM benchmark over the diagnostic frequency range.

Since Fourier-domain accuracy and NLL do not by themselves guarantee accurate tail behavior, and since one motivation for the Gaussian–Laplace model is improved flexibility for non-Gaussian tails, we also examine tail calibration. Tail calibration is adequate for all three methods, with no uniform dominance across tail levels. At the deepest lower tail ($\alpha = 0.5\%$), EM-GMM attains the smallest lower-tail probability



(a) Validation NLL and BIC.



(b) Marginal improvement.

Fig. 7.3: EM-GMM model selection for the Australian equity pseudo-sample. The diagnostics identify $K_{EM} = 8$ as a parsimonious benchmark size.

Table 7.4: Selected models and test performance for the Australian equity pseudo-sample. The upper panel reports model size and NLL; the lower panel reports Fourier-domain errors against the empirical test pseudo-CF.

Method	$(K_{\mathcal{G}}, K_{\mathcal{L}})$	Val NLL	Test NLL
EM-GMM	(8, 0)	-0.3308	-0.3250
Fourier-GMM	(15, 0)	-0.3329	-0.3269
Fourier-GLMM	(15, 6)	-0.3344	-0.3286

Method	$L_2^{\text{Re}}(\cdot)$	$L_2^{\text{Im}}(\cdot)$	$\text{MPE}^{\text{Re}}(\cdot)$	$\text{MPE}^{\text{Im}}(\cdot)$
EM-GMM	4.194×10^{-3}	4.723×10^{-3}	2.393×10^{-2}	2.632×10^{-2}
Fourier-GMM	1.446×10^{-3}	1.409×10^{-3}	1.137×10^{-2}	9.414×10^{-3}
Fourier-GLMM	1.359×10^{-3}	1.375×10^{-3}	8.886×10^{-3}	8.832×10^{-3}

error (1.6×10^{-4}), whereas Fourier-GLMM is most accurate in the upper tail (1.2×10^{-4}). At more moderate levels, the methods remain close. This is consistent with the modest excess kurtosis of the pseudo-sample: the train, validation, and test splits have excess kurtosis 0.743, 0.735, and 0.723, respectively. Since these values are close to the Gaussian benchmark of 0 and well below the Laplace value 3, the one-year

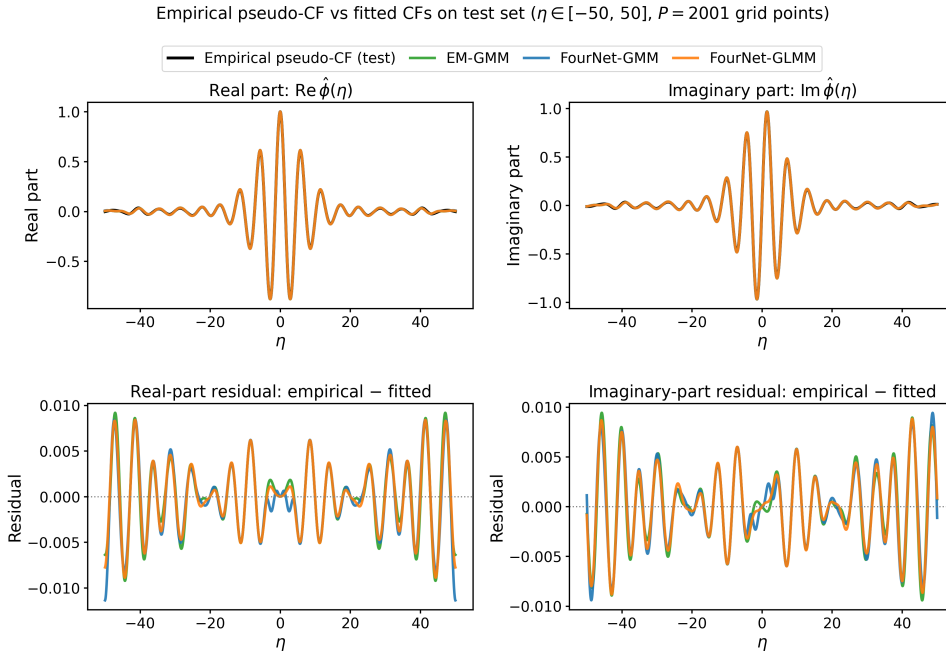


Fig. 7.4: Australian equity pseudo-sample experiment: empirical test pseudo-CF, fitted CFs, and residual curves under the parsimonious EM-GMM benchmark.

total-return distribution is only moderately heavy-tailed, so the Laplace part of the GLMM improves the Fourier-domain fit without producing a uniformly dominant tail-calibration result.

8 Conclusion and future work We introduced a data-driven Fourier-trained neural-network method for estimating fixed-horizon probability densities from empirical transform information. The method combines Fourier-domain training with a positive Gaussian–Laplace output structure, so the learned model remains a proper density throughout optimization while retaining a closed-form CF. Thus, the estimation problem can be formulated directly in Fourier space rather than through physical-space likelihood fitting alone.

The analysis was developed for two sampling settings. In the direct i.i.d. case, we derived an expected L_2 bound for the density estimation error, separating Fourier truncation, empirical training error, discretization, and empirical-CF sampling error. In the resampling-based pseudo-sampling case, the same Fourier-to-real-space argument carries over conditionally on the observed history, with two additional pseudo-law discrepancy terms measuring the gap between the resampling law and the true fixed-horizon law. We also extended the framework to multiple dimensions and discussed its computational complexity.

The numerical experiments support the main conclusions. The method is competitive with EM on well-specified Gaussian-mixture benchmarks, improves over the Gaussian subclass on heavy-tailed targets such as the Cauchy law, exhibits the predicted $M^{-1/2}$ error decay in a well-specified setting, and remains viable in two dimensions. The Australian equity experiment further illustrates that the resampling-based framework can be used to estimate fixed-horizon laws from dependent time-series

data.

Several directions merit further study. On the theoretical side, it would be useful to derive explicit convergence rates for the pseudo-law discrepancy terms under specific resampling schemes, and to extend the analysis beyond conditionally i.i.d. pseudo-samples to more general dependent settings. On the application side, the resampling-based pseudo-sampling framework provides a route toward data-driven fixed-horizon laws in stochastic control and portfolio optimization, complementing existing Fourier-trained transition-kernel approaches based on closed-form CFs [4, 3].

Appendix A. Proof of Lemma 3.2. By Assumption 2.1 (A3), G is Lipschitz on $[-\eta', \eta']$. Under (2.11), the real and imaginary parts of $\widehat{G}(\cdot; \theta)$ given by (2.13)–(2.14) are uniformly Lipschitz on $[-\eta', \eta']$, and hence so is $F_\theta(\cdot)$. Thus

$$|F_\theta(\eta) - F_\theta(\xi)| \leq C'|\eta - \xi|, \quad \forall \eta, \xi \in [-\eta', \eta'], \quad \forall \theta \in \Theta,$$

for some $C' = C'(\eta') > 0$. A standard composite left-rule estimate on the partition (2.16) then yields (3.12).

Appendix B. Proof of Lemma 3.3. By Assumption 2.1 (A2) and Plancherel's theorem, $G \in L_2(\mathbb{R})$, which gives the first bound in (3.13) for $\eta' > 0$ sufficiently large. For the second, (2.12) and (2.11) imply

$$|\widehat{G}(\eta; \theta)| \leq \sum_{k=1}^{K_G} \beta_k^{(G)} \exp(-\sigma_k^2 \eta^2 / 2) + \sum_{\ell=1}^{K_C} \beta_\ell^{(C)} \frac{1}{1 + b_\ell^2 \eta^2} \leq \exp(-\sigma_{\min}^2 \eta^2 / 2) + \frac{1}{1 + b_{\min}^2 \eta^2},$$

whose square is integrable on \mathbb{R} , uniformly in θ . The bound (3.14) then follows from

$$|G(\eta) - \widehat{G}(\eta; \theta)|^2 \leq 2|G(\eta)|^2 + 2|\widehat{G}(\eta; \theta)|^2.$$

Appendix C. Proof of Lemma 4.1. Given \mathcal{H}_N , the pseudo-samples $X_1^\dagger, \dots, X_M^\dagger$ are i.i.d. with common law $\mathcal{L}_{\tau, N}^\dagger$ and CF G_N^\dagger . Therefore,

$$\mathbb{E}^\dagger \left[e^{i\eta X_m^\dagger} \right] = G_N^\dagger(\eta), \quad \mathbb{E}^\dagger \left[|e^{i\eta X_m^\dagger}|^2 \right] = 1.$$

The first equation of (4.6) follows from (4.1). Using the same variance calculation as in (2.3) gives the second equation of (4.6). The identities (4.8)–(4.9) then follow by repeating the proof of Lemma 3.1 with G replaced by G_N^\dagger and \widehat{G}_M replaced by \widehat{G}_M^\dagger .

REFERENCES

- [1] O. E. BARNDORFF-NIELSEN, *Normal inverse gaussian distributions and stochastic volatility modelling*, Scandinavian Journal of statistics, 24 (1997), pp. 1–13.
- [2] P. CARR, H. GEMAN, D. B. MADAN, AND M. YOR, *The fine structure of asset returns: An empirical investigation*, The Journal of Business, 75 (2002), pp. 305–332.
- [3] D.-M. DANG AND C. CHEN, *Multi-period mean-buffered probability of exceedance in Defined Contribution portfolio optimization*, SIAM Journal on Financial Mathematics, (2026). to appear.
- [4] D.-M. DANG AND H. ZHOU, *Monotone 2D integration scheme for Mean-CVaR optimization via Fourier-trained transition kernels*, arXiv preprint arXiv:2603.26291, (2026).
- [5] A. C. DAVISON AND D. V. HINKLEY, *Bootstrap Methods and Their Application*, Cambridge University Press, Cambridge, 1997.
- [6] A. P. DEMPSTER, N. M. LAIRD, AND D. B. RUBIN, *Maximum likelihood from incomplete data via the EM algorithm*, Journal of the Royal Statistical Society. Series B, 39 (1977), pp. 1–38.
- [7] R. DU AND D.-M. DANG, *Fourier neural network approximation of transition densities in finance*, SIAM Journal on Scientific Computing, 47 (2025), pp. C529–C557.
- [8] B. EFRON AND R. J. TIBSHIRANI, *An Introduction to the Bootstrap*, Chapman and Hall, New York, 1993.

- [9] F. FANG AND C. W. OOSTERLEE, *A novel pricing method for european options based on fourier-cosine series expansions*, SIAM Journal on Scientific Computing, 31 (2008), pp. 826–848.
- [10] P. A. FORSYTH AND G. LABAHN, *ϵ -monotone Fourier methods for optimal stochastic control in finance*, Journal of Computational Finance, 22(4) (2019), pp. 25–71.
- [11] D. P. KINGMA AND J. BA, *Adam: A method for stochastic optimization*, arXiv preprint arXiv:1412.6980, (2014).
- [12] S. KOU, *A jump diffusion model for option pricing*, Management Science, 48 (2002), pp. 1086–1101.
- [13] H. R. KÜNSCH, *The jackknife and the bootstrap for general stationary observations*, The Annals of Statistics, 17 (1989), pp. 1217–1241.
- [14] S. N. LAHIRI, *Resampling Methods for Dependent Data*, Springer, New York, 2003.
- [15] Á. LEITAO, C. W. OOSTERLEE, L. ORTIZ-GRACIA, AND S. M. BOHTE, *On the data-driven cosine method*, Applied Mathematics and Computation, 317 (2018), pp. 68–84.
- [16] T. MATHEWS, *A history of australian equities*, Research Discussion Paper RDP 2019-04, Reserve Bank of Australia, 2019, <https://www.rba.gov.au/publications/rdp/2019/2019-04.html>.
- [17] G. J. MCLACHLAN, S. X. LEE, AND S. I. RATHNAYAKE, *Finite mixture models*, Annual review of statistics and its application, 6 (2019), pp. 355–378.
- [18] G. J. MCLACHLAN AND D. PEEL, *Finite Mixture Models*, Wiley, New York, 2000.
- [19] R. C. MERTON, *Option pricing when underlying stock returns are discontinuous*, Journal of financial economics, 3 (1976), pp. 125–144.
- [20] D. N. POLITIS AND J. P. ROMANO, *The stationary bootstrap*, Journal of the American Statistical Association, 89 (1994), pp. 1303–1313.
- [21] D. N. POLITIS, J. P. ROMANO, AND M. WOLF, *Subsampling*, Springer, New York, 1999.
- [22] B. W. SILVERMAN, *Density Estimation for Statistics and Data Analysis*, Chapman and Hall, London, 1986.
- [23] P. T. TRAN ET AL., *On the convergence proof of AMSGrad and a new version*, IEEE Access, 7 (2019), pp. 61706–61716.