DENSITYFM: FOUNDATION-STYLE PRETRAINING FOR DENSITY ESTIMATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Estimating probability densities is at the heart of many data science and machine learning pipelines. However, existing estimators face a fundamental accuracy–efficiency trade-off: classic non-parametric methods are fast but struggle in high dimensions, while more expressive flow-based methods require costly training and tuning for each new dataset. In this work, we present DensityFM, a pretrained, foundation model-style density estimator achieving the best of both worlds. Pretrained once on large-scale and diverse distributions, DensityFM performs zero-shot density estimation on unseen datasets with frozen parameters at test time. Experiments across diverse synthetic distributions and multiple 64-dimensional real-world datasets demonstrate that DensityFM processes each dataset in 0.5 seconds, delivering a speedup up to $156\times$ over the fastest neural baseline, while achieving comparable or even superior accuracy, establishing a new foundation for scalable data science and knowledge discovery.

1 INTRODUCTION

A probability density characterizes how observations are distributed, capturing their concentration, geometry, multimodality, and uncertainty. Estimating this density is a fundamental task in statistics and machine learning, underpinning applications such as anomaly detection, uncertainty quantification, data generation, and probabilistic inference. For instance, in single-cell studies, density estimation can reveal rare resistant populations and developmental shifts induced by genetic or drug perturbations (Dixit et al., 2016; Srivatsan et al., 2020), while modeling accelerometer and gyroscope signals can identify atypical movement patterns relevant to fall detection and mobility monitoring (Anguita et al., 2013). A density estimator that is both scalable and broadly applicable would therefore provide a versatile tool for understanding complex, high-dimensional data across real-world domains.

A wide spectrum of density estimation methods has been developed, each offering different trade-offs. Classical non-parametric methods such as kernel density estimators (KDEs) construct a density by smoothing around individual observations (Silverman, 1986), while Gaussian mixture models approximate the underlying distribution using a weighted combination of Gaussian components (McLachlan & Peel, 2000). Although these estimators are inexpensive to use, they often scale poorly with dimensionality and may lack the flexibility required to capture complex distributions in reality (Silverman, 1986; Nagler & Czado, 2016; McLachlan & Peel, 2000). In contrast, modern neural density estimators such as normalizing flows (Dinh et al., 2017; Durkan et al., 2019; Papamakarios et al., 2017) model complex distributions with expressive deep neural networks. While flexible, they typically require costly optimization for every new dataset (Papamakarios et al., 2021), making their computational overhead prohibitive in real-time or streaming applications. Thus, existing density estimators face a fundamental dilemma between accuracy and computational efficiency.

Inspired by recent advances in foundation models for data science (Müller et al., 2022; Hollmann et al., 2023; 2025; Das et al., 2024; Hu et al., 2026), we introduce DensityFM, a foundation-model-style neural estimator for fast and accurate density estimation. Given a dataset $\mathcal{D}$ as context, DensityFM evaluates a normalized density at any query point in a single forward pass, without fitting a separate model for each incoming dataset. Unlike conventional approaches that train a density estimator directly on $\mathcal{D}$, we feed $\mathcal{D}$ as context to a pretrained model, which directly predicts the corresponding context-conditioned density. Through large-scale pretraining on a broad collection

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

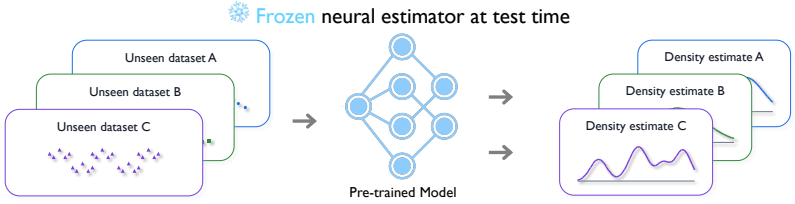

Figure 1: **Conceptual framework of DensityFM.** One pretrained network turns an unlabeled dataset into a normalized density in a single forward pass. No per-dataset fitting is required. The dataset itself is used the *context*, and density estimation is done by a single forward pass based on the context.

of synthetic distributions, DensityFM learns to automatically adapt to different context datasets and generalizes effectively to unseen distributions. This eliminates the need for costly per-dataset optimization while achieving accuracy comparable to or even exceeding that of popular neural density estimators (Dinh et al., 2017; Durkan et al., 2019; Papamakarios et al., 2017). This enables flexible density estimation, providing new foundation for data science and machine learning applications.

Contributions. Our main contributions are threefold.

- We develop DensityFM, the first in-context architecture for zero-shot density estimation of data up to 64 dimensions. Given a variable-sized dataset, DensityFM directly produces a normalized density that can be evaluated at arbitrary query points, without any test-time optimization.
- We introduce a large-scale, diversity-aware pretraining recipe that exposes DensityFM to a broad family of synthetic distributions with diverse statistical patterns. The recipe requires no ground-truth density values, simplifying pretraining while enabling effective generalization to unseen distributions.
- We conduct extensive evaluations on synthetic and real-world benchmarks, demonstrating strong and generalizable zero-shot density estimation. DensityFM achieves state-of-the-art performance across all three real-world datasets and processes each 64-dimensional dataset in only 0.447 seconds, delivering up to a $156\times$ speedup over the fastest neural baseline.

2 Problem Formulation

The task of density estimation. Density estimation seeks to approximate an unknown probability density from a finite sample. Let $\mathcal{D} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\} \subset \mathbb{R}^d$ contain i.i.d. observations from an unknown density $p^\star$. Given $\mathcal{D}$, the task is to construct an estimate $\hat{p}_\mathcal{D}$ of $p^\star$ satisfying

$$\hat{p}_\mathcal{D} : \mathbb{R}^d \to \mathbb{R}^+, \qquad \int_{\mathbb{R}^d} \hat{p}_\mathcal{D}(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x} = 1.$$

The estimated density can then be evaluated at any query point $\boldsymbol{x}^{\mathrm{qry}} \in \mathbb{R}^d$.

A promising venue to perform accurate density estimation is to leverage neural density estimators. These methods use neural networks with parameters θ to represent flexible densities p_θ. Normalizing flows, for example, learn an invertible transformation $\boldsymbol{z} = f_\theta(\boldsymbol{x})$ from the data to a simple latent distribution with known density p_0 (Papamakarios et al., 2021). They assign density by

$$p_\theta(\boldsymbol{x}) = p_0(f_\theta(\boldsymbol{x}))\, |\det \nabla_{\boldsymbol{x}} f_\theta(\boldsymbol{x})|\,. \tag{1}$$

Representative architectures include Real NVP, MAF, and neural spline flows (Dinh et al., 2017; Papamakarios et al., 2017; Durkan et al., 2019). More recently, flow matching trains continuous normalizing flows by regressing time-dependent velocity fields along prescribed probability paths (Lipman et al., 2023; Albergo & Vanden-Eijnden, 2023; Tong et al., 2024). Their densities can be recovered by integrating the instantaneous change-of-variables equation along the learned ODE. For likelihood-based estimators, the network is trained by minimizing the negative log-likelihood $\mathcal{L}_\mathcal{D}(\theta)$, and the optimized parameters $\theta^\star_\mathcal{D}$ define the fitted density $p_{\theta^\star_\mathcal{D}}$:

$$\mathcal{L}_\mathcal{D}(\theta) = -\mathbb{E}_{\boldsymbol{x}}[\log p_\theta(\boldsymbol{x})]\,, \qquad \theta^\star_\mathcal{D} = \arg\min_\theta \mathcal{L}_\mathcal{D}(\theta), \tag{2}$$

Table 1: Conceptual comparison of density estimators. "Normalized" denotes an architectural normalization guarantee. "Training" denotes required target-specific neural-network optimization, excluding pretraining. Flexibility and efficiency rate representational breadth and target-time computation; higher is better. Max dim. is the largest reported pretrained dimension; "–" denotes methods fitted separately to each target dataset.

Method	Normalized	Training	Flexibility	Efficiency	Max dim.
Gaussian	✓	×	Low	High	–
KDE	✓	×	Medium	High	–
GMM	✓	×	Medium	Medium	–
Flows (e.g., MAF, RealNVP, NSF)	✓	✓	High	Low	–
DiScoFormer	×	×	High	High	10
DensityFM	✓	×	High	High	64

Challenge of real-time neural density estimation. Regardless of the training objective, conventional neural density estimators optimize their parameters separately for each target dataset. For the likelihood objective in equation 2, gradient-based training follows

$$\theta^{t+1} \leftarrow \theta^t - \eta \nabla_\theta \mathcal{L}_{\mathcal{D}}(\theta^t), \qquad t = 1, \ldots, T, \tag{3}$$

where η is the learning rate. Each new dataset thus requires T sequential parameter updates before its fitted density is available. Suppressing the cost of each update, the per-dataset optimization complexity is $\mathcal{O}(T)$, which prevents immediate density evaluation when datasets arrive continually or must be processed at scale.

DensityFM learns a reusable density estimator from sample sets to densities through pretraining:

$$\mathcal{D} \longmapsto \hat{p}_\theta(\cdot \mid \mathcal{D}).$$

At test time, the pretrained model takes $\mathcal{D}$ as context while θ remains fixed. For every fixed $\mathcal{D}$, the function $\boldsymbol{x}^{\mathrm{qry}} \mapsto \hat{p}_\theta(\boldsymbol{x}^{\mathrm{qry}} \mid \mathcal{D})$ satisfies the same nonnegativity and unit-integral conditions above. Thus, a new dataset changes the estimated density without fitting new model parameters.

2.1 Copula–Marginal Disentanglement

Design principle. A joint density combines the marginal distribution of each variable with the dependence among variables. We assign these two parts to separate components using Sklar's theorem (Sklar, 1959):

$$\hat{p}_\theta(\boldsymbol{x}^{\mathrm{qry}} \mid \mathcal{D}) = \hat{c}_\theta\Big(\big(\hat{F}_j(x_j^{\mathrm{qry}})\big)_{j=1}^d \mid \mathcal{D}\Big) \times \prod_{j=1}^d \hat{f}_j(x_j^{\mathrm{qry}}). \tag{4}$$

where $\hat{f}_j$ and $\hat{F}_j$ are the estimated marginal density and CDF, while $\hat{c}_\theta$ captures the cross-coordinate copula. This decomposition is the central design principle of DensityFM: a marginal estimator handles the one-dimensional marginals, and a pretrained neural model handles the multivariate copula.

- **Marginal estimation.** Given $\mathcal{D}$, we estimate each coordinate's marginal density $\hat{f}_j$ and CDF $\hat{F}_j$ using Gaussian KDE (Silverman, 1986) or a pretrained one-dimensional DensityFM. At a query $\boldsymbol{x}^{\mathrm{qry}}$, the product $\prod_{j=1}^d \hat{f}_j(x_j^{\mathrm{qry}})$ gives the marginal contribution in equation 4.
- **Copula estimation.** We apply the estimated CDFs coordinate-wise to construct the CDF-scale context $\mathcal{D}_u$ and query $\boldsymbol{u}^{\mathrm{qry}}$. A pretrained context encoder summarizes $\mathcal{D}_u$, after which a pretrained autoregressive decoder evaluates $\hat{c}_\theta(\boldsymbol{u}^{\mathrm{qry}} \mid \mathcal{D}_u)$, providing the dependence contribution in equation 4.

This divide-and-conquer design avoids requiring a single pretrained model to capture both heterogeneous marginal distributions and multivariate dependence. By estimating the marginals with KDE or a one-dimensional DensityFM, the neural model can focus on multivariate dependence, simplifying pretraining and modeling.

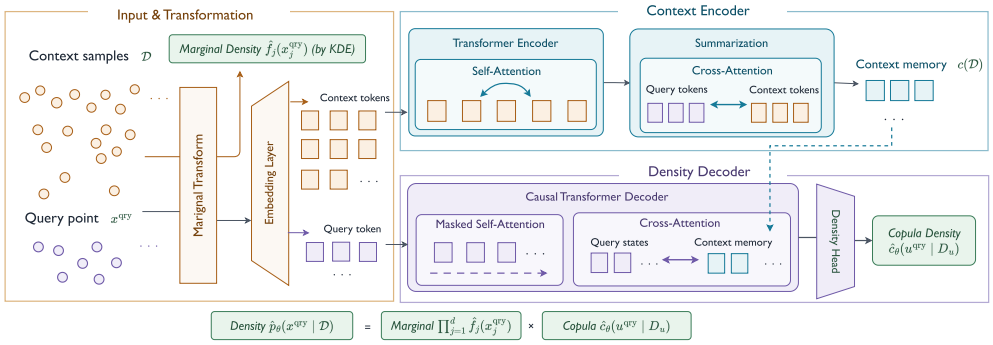

Figure 2: **Three-step architecture of DensityFM.** KDE-based marginal transforms map the context set $\mathcal{D}$ and query $\boldsymbol{y}$ to copula coordinates. The set encoder summarizes the transformed context, and the autoregressive decoder evaluates the copula density at the transformed query. Multiplying the copula and marginal terms yields $\hat{p}_\theta(\boldsymbol{y} \mid \mathcal{D})$.

2.2 Neural Copula Estimation

Conventional neural estimators encode a target distribution in parameters fitted specifically to its dataset, while DensityFM instead keeps θ fixed and provides the dataset as an input *context*. Specifically, $\mathcal{D} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ is the context that identifies the target distribution from its observations, while $\boldsymbol{x}^{\text{qry}}$ is a *query* specifying the point at which its density should be evaluated. Each pair $(\mathcal{D}, \boldsymbol{x}^{\text{qry}})$ therefore requests one value $\hat{p}_\theta(\boldsymbol{x}^{\text{qry}} \mid \mathcal{D})$; holding $\mathcal{D}$ fixed and varying $\boldsymbol{x}^{\text{qry}}$ defines the entire density for that dataset.

Architecture overview. Under the decomposition in equation 4, the marginal estimator operates on the original context $\mathcal{D}$ and query $\boldsymbol{x}^{\text{qry}}$, whereas the neural copula estimator uses their CDF-scale representations

$$\boldsymbol{u}_i = \big(\hat{F}_1(x_{i,1}), \ldots, \hat{F}_d(x_{i,d})\big), \quad \mathcal{D}_u = \{\boldsymbol{u}_i\}_{i=1}^n. \tag{5}$$

The context encoder processes $\mathcal{D}_u$, and the density decoder combines the resulting context representation with $\boldsymbol{u}^{\text{qry}}$ to evaluate $\hat{c}_\theta(\boldsymbol{u}^{\text{qry}} \mid \mathcal{D}_u)$. Equation equation 4 then combines this copula value with the KDE marginals to produce $\hat{p}_\theta(\boldsymbol{x}^{\text{qry}} \mid \mathcal{D})$.

Context encoder. The context encoder converts the variable-size context $\mathcal{D}_u$ into a fixed-length summary, giving the density decoder a consistent representation of the target dependence structure regardless of n. Each copula-scale sample $\boldsymbol{u}_i$ is first mapped to a context embedding $\boldsymbol{e}_i$, after which self-attention contextualizes these embeddings:

$$\boldsymbol{e}_i = \text{Embed}_\theta^{\text{ctx}}(\boldsymbol{u}_i), \qquad \mathbf{H}_{\mathcal{D}_u} = \text{Encoder}_\theta\big([\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n]^\top\big) = [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_n]^\top.$$

Each $\boldsymbol{h}_i$ therefore represents $\boldsymbol{u}_i$ in relation to all context samples. Because $\mathbf{H}_{\mathcal{D}_u}$ still contains n tokens, m learned summary queries Q_θ cross-attend to it as keys and values:

$$\boldsymbol{c}(\mathcal{D}_u) = \text{CrossAttn}_\theta(Q_\theta, \mathbf{H}_{\mathcal{D}_u}, \mathbf{H}_{\mathcal{D}_u}).$$

The resulting m tokens form the fixed-length, order-invariant context memory $\boldsymbol{c}(\mathcal{D}_u)$.

Density decoder. The density decoder combines the copula-scale query $\boldsymbol{u}^{\text{qry}}$ with the context memory to approximate the copula density through an autoregressive model. It applies the probability chain rule

$$\hat{c}_\theta(\boldsymbol{u}^{\text{qry}} \mid \mathcal{D}_u) = \prod_{j=1}^d q_{\theta,j}\big(u_j^{\text{qry}} \mid u_{<j}^{\text{qry}}, \boldsymbol{c}(\mathcal{D}_u)\big). \tag{6}$$

Analogously, each copula-scale query coordinate u_j^{qry} is first mapped to a query embedding $\boldsymbol{r}_j$. When parameterizing the jth conditional, the decoder uses causal self-attention to integrate the

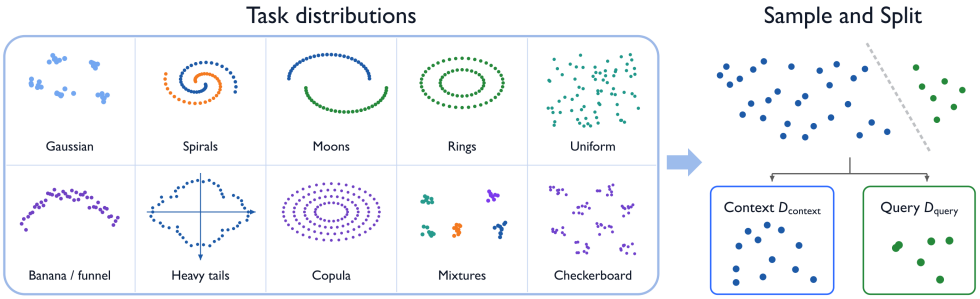

Figure 3: **On-the-fly pretraining tasks.** Each task samples and freezes one compositional distribution, then draws observations and splits them into independent context and query sets.

preceding query embeddings $\boldsymbol{r}_{<j}$ and cross-attention to retrieve the relevant dependence information from the context memory. The resulting decoder state is

$$\boldsymbol{r}_j = \mathrm{Embed}_\theta^{\mathrm{qry}}(u_j^{\mathrm{qry}}), \qquad \boldsymbol{s}_j = \mathrm{Decoder}_\theta(\boldsymbol{r}_{<j}, \boldsymbol{c}(\mathcal{D}_u)) .$$

The head maps $\boldsymbol{s}_j$ to a context-dependent Student-t base and a stacked transform of eight-bin monotone rational-quadratic splines (Durkan et al., 2019):

$$(\nu_{\theta,j}, S_{\theta,j}) = \mathrm{Head}_{\theta,j}(\boldsymbol{s}_j) . \tag{7}$$

Together, these outputs define the conditional through

$$z_j \sim t_{\nu_{\theta,j}}, \qquad u_j^{\mathrm{qry}} = S_{\theta,j}(z_j). \tag{8}$$

The learned degrees of freedom adapt the tail decay, while the invertible spline stack captures asymmetric and multimodal conditionals.

Substituting the neural copula in equation 6 and the KDE marginals into equation 4 yields the final normalized density $\hat{p}_\theta(\cdot \mid \mathcal{D})$. The following result characterizes the approximation capacity of the complete estimator.

Theorem 1 (Architectural universality of DensityFM)**.** *For any target density p on $\mathbb{R}^d$ satisfying standard regularity conditions and any $\epsilon, \delta > 0$, there exist a context size n and parameters θ of a sufficiently expressive DensityFM model such that*

$$\mathbb{P}_{\mathcal{D}\sim p^n}(\|\hat{p}_\theta(\cdot \mid \mathcal{D}) - p\|_1 < \epsilon) \geq 1 - \delta.$$

For a new dataset, the marginal estimators and context memory are computed once, after which query densities are evaluated with frozen θ.

Proof. See Appendix A. □

3 Large-Scale Synthetic Pretraining

The architecture determines which densities DensityFM can represent, while the pretraining tasks determine which unseen distributions it can recognize from samples. We pretrain DensityFM across a broad collection of synthetic density-estimation tasks. The diversity of these tasks supports zero-shot generalization to unseen distributions. We next define the shared learning objective and the distribution of pretraining tasks.

3.1 Pretraining Paradigm

DensityFM learns one density-estimation rule across tasks drawn from a meta-distribution $P_{\mathcal{D}}$. For each task, we draw a dataset $\mathcal{D} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}_{i=1}^n$ via $\mathcal{D} \sim P_{\mathcal{D}}$. We split the dataset into $\mathcal{D}_{\text{context}}$ and $\mathcal{D}_{\text{query}}$, so that $\mathcal{D} = \mathcal{D}_{\text{context}} \cup \mathcal{D}_{\text{query}}$.

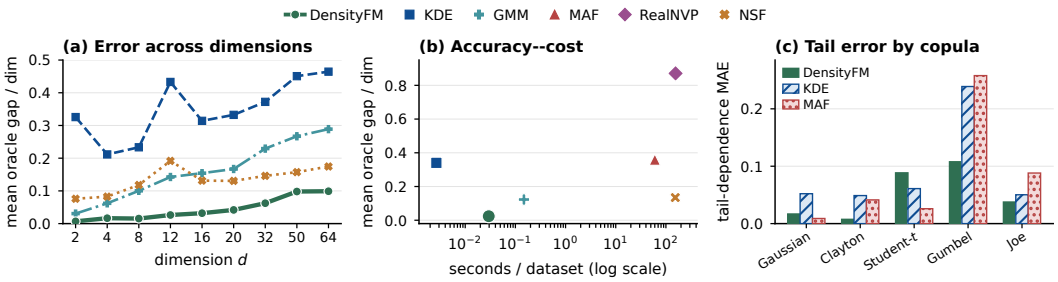

Figure 4: Synthetic density estimation on 22 common density families spanning Gaussian, GMM, and other popular distributions. (a) Mean oracle gap per dimension across all density families at each evaluated dimension. (b) Accuracy–cost trade-off. Each marker represents one estimator, positioned by its mean wall-clock time per dataset and mean oracle gap per dimension over the same 594 workloads (22 families $\times$ nine dimensions $\times$ three seeds). Runtime includes target-dataset fitting when required. (c) Mean absolute error of finite-threshold tail-dependence estimates for five copulas. Marginal probability mass $q \in \{0.001, 0.01, 0.025\}$ corresponds to the most extreme 0.1%, 1%, and 2.5% of each marginal, respectively. Bars average the error across these three thresholds, with each estimate computed from ten paired repetitions.

We train the shared parameters θ by minimizing the expected negative log-likelihood under this sampling process:

$$\mathcal{L}(\theta) = \mathbb{E}_{\mathcal{D}\sim P_{\mathcal{D}}} \mathbb{E}_{\boldsymbol{x}\sim \mathcal{D}_{\text{query}}} \Big[-\log \hat{p}_\theta(\boldsymbol{x} \mid \mathcal{D}_{\text{context}}) \Big]. \tag{9}$$

This objective requires only context and query samples, not the ground-truth density value $p(\boldsymbol{x})$. Any simulator that generates continuous samples can therefore contribute tasks to $P_{\mathcal{D}}$. The next subsection constructs the meta-distribution $P_{\mathcal{D}}$ through compositional distribution generation.

3.2 Pretraining Task Generation

Compositional distribution construction. For each dataset $\mathcal{D}$ sampled from $P_{\mathcal{D}}$, the generator constructs it in two stages. It first draws an intermediate variable $\boldsymbol{z}$ from a mixture of base distributions, which provides the initial dependence pattern and geometry. It then maps $\boldsymbol{z}$ to the observed variable $\boldsymbol{x}$ through a transformation T

$$\boldsymbol{z} \sim \sum_{m=1}^{M} \pi_m q_m, \qquad \boldsymbol{x} = T(\boldsymbol{z}). \tag{10}$$

Here, q_m denotes a component of the sampled structure and π_m is its mixture weight. Combining the same dependence structure with different transformations produces distributions with different marginals but related dependencies. This construction follows the copula–marginal separation in Section 2.1.

Diversity-aware task generation. The structural base distribution library includes Gaussian and Student-t copulas, Archimedean copulas, mixtures, and geometric or hierarchical distributions such as rings, spirals, bananas, and funnels. The transformation T further varies the marginal families, tail behavior, coordinate warps, and rotations. Resampling these choices produces different observed densities from related dependence structures.

4 Experiments

We evaluate frozen DensityFM checkpoints without target-specific optimization on analytic densities and real-world datasets, followed by two conditional-inference tasks.

Protocol. Within each experiment, methods share the same cohorts or outer splits, and hyperparameter selection does not use test rows. Unless stated otherwise, estimators are fitted or conditioned

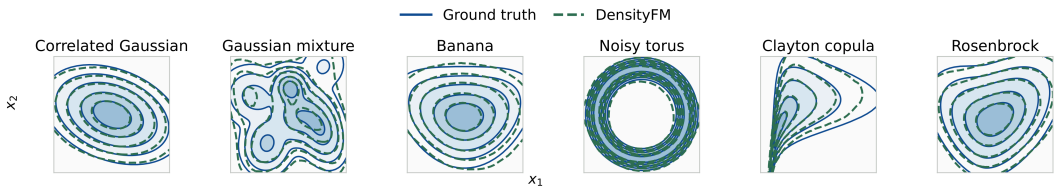

Figure 5: Two-dimensional density recovery. Ground-truth contours (solid blue) and DensityFM estimates (dashed green) for six representative Common Densities Family 22 instances at $d = 2$, drawn at matched ground-truth log-density levels.

Table 2: Accuracy and cost on six of the 22 distribution families at $d = 50$. Gap per dimension: mean $\pm$ standard deviation over five instances; time is averaged across the six families. Dirichlet uses ALR coordinates. Lower is better.

Method	Gaussian	Gauss. copula	t copula	Dirichlet	GMM	Noisy torus	Time (s)
DensityFM	**0.002 ± .001**	0.361 ± .066	**0.329 ± .055**	**0.039 ± .003**	**0.029 ± .013**	**0.061 ± .014**	0.140
KDE	0.184 ± .003	0.852 ± .080	0.915 ± .075	0.223 ± .018	0.236 ± .037	0.270 ± .020	0.015
GMM	0.014 ± .001	0.399 ± .067	0.386 ± .052	0.051 ± .002	0.071 ± .047	0.081 ± .020	6.42
MAF	0.043 ± .003	0.374 ± .059	0.578 ± .205	0.067 ± .004	0.091 ± .042	0.134 ± .024	57.4
NSF	0.048 ± .001	**0.356 ± .052**	0.391 ± .041	0.060 ± .015	0.104 ± .043	0.132 ± .024	194

on at most 1024 context rows. We report oracle gap per dimension on synthetic data (lower is better) and held-out log-likelihood per dimension on real data (higher is better). Bold and underlined entries denote the best and second-best displayed means.

4.1 Synthetic density estimation

Density estimation quality. We consider 22 distribution families with analytic densities, including Gaussian, t-copula, noisy torus, and Gaussian mixtures. For each distribution, we draw 1024 context samples and 2000 query points. We measure accuracy by the per-dimension Kullback–Leibler gap between the target density p and estimate $\hat{p}$:

$$\text{gap}/d = \text{KL}(p \,\|\, \hat{p})/d,$$

approximated by Monte Carlo estimation using the sampled data. DensityFM has the lowest mean gap at all nine dimensions in figure 4a. Across the 396 hardware-matched workloads in panel (b), its mean gap is 0.024 at 0.029 seconds per dataset, versus 0.123 at 0.148 seconds for GMM and 0.134 at 153.6 seconds for NSF. In Table 2, it is best or tied on all six displayed families at $d = 8$, and ranks first on five and second on the Gaussian copula at $d = 50$. Complete family-by-method results appear in section D.4.

Visualization of contours. The estimated contours in figure 5 follow the ground-truth contours across anisotropic, multimodal, curved, annular, and asymmetric examples.

Tail dependence modeling effect. Average likelihood can conceal errors in rare joint events. For copula-uniform coordinates (U_1, U_2), we estimate $\lambda_L(q) = \Pr(U_2 \le q \mid U_1 \le q)$ and $\lambda_U(q) = \Pr(U_2 > 1-q \mid U_1 > 1-q)$ at $q \in \{0.001, 0.01, 0.025\}$. This experiment uses 4096 context points and ten paired repetitions. Figure 4c compares three methods on five dependence classes. Averaged over these five copulas, the absolute error is 0.052 for DensityFM, 0.084 for MAF, and 0.090 for KDE; MAF is lower on the Gaussian and Student-t cases.

4.2 Real-world density estimation

Setup. We evaluate five standard tabular density benchmarks in their prepared native coordinates and four high-dimensional datasets after PCA and standardization, both fitted strictly within each

Table 3: Real-world density estimation in native coordinates (top) and after PCA fitted on training data to $d = 64$ (bottom). Held-out log-likelihood per dimension: mean $\pm$ sample standard deviation over three native-data splits and five PCA splits; higher is better. Average time includes target fitting or context processing and query scoring, with equal weight across the nine datasets; loading and external preprocessing are excluded.

Dataset	d	DensityFM	Gaussian	KDE	GMM	MAF	RealNVP	NSF
Log likelihood with native data (↑)								
POWER	5	**−0.79±.01**	−0.95±.01	−0.82±.01	−0.81±.00	−0.82±.01	−0.87±.07	−0.80±.01
GAS	8	**+0.57±.02**	−0.40±.03	−0.13±.18	+0.37±.05	−0.11±.26	+0.16±.13	+0.50±.04
HEPMASS	22	**−1.17±.00**	−1.29±.01	−1.36±.00	−1.20±.01	−1.22±.01	−1.27±.03	−1.17±.00
MiniBooNE	50	**−0.52±.08**	−0.88±.05	−1.00±.05	−0.78±.13	−0.91±.41	−0.81±.04	−0.57±.04
BSDS300	63	**−0.16±.04**	−0.94±.03	−1.09±.04	−0.68±.04	−0.69±.07	−0.66±.16	−0.53±.04
Log likelihood with PCA compressed data (↑)								
Olivetti	64	−1.33±.02	−1.33±.02	**−1.22±.04**	−1.33±.02	−1.28±.02	−1.38±.08	−1.28±.02
MNIST	64	**−1.37±.01**	−1.44±.00	−1.39±.00	−1.37±.05	−1.37±.00	−1.40±.01	−1.38±.00
HAR	64	−1.39±.00	−1.44±.00	−1.50±.00	**−1.34±.07**	−1.42±.01	−1.48±.00	−1.41±.01
ISOLET	64	−1.39±.00	−1.42±.00	−1.45±.00	**−1.38±.05**	−1.39±.00	−1.42±.00	−1.40±.00
Average time (s)		0.312	0.553	0.250	5.24	35.7	50.1	81.4

outer-training split. Both protocols use at most 1024 context or fitting rows. For the 63-dimensional BSDS300 dataset, DensityFM uses the $d = 64$ checkpoint with zero padding and retains the first 63 autoregressive factors.

Results. In prepared native coordinates, DensityFM has the highest mean on all five datasets; its largest margins over the next-best method are 0.363 on BSDS300 and 0.064 on GAS. At $d = 64$, it ranks first on MNIST, second on HAR and ISOLET, and fourth on Olivetti Faces. On BSDS300, context processing and scoring 2048 queries take 0.452 seconds on one H200. Further protocol details and dimension sweeps on Breast Cancer, Digits, and MiniBooNE are given in section D.1.

4.3 Downstream Applications

We test whether the frozen joint density supports conditional inference without retraining DensityFM, using latent-coordinate reconstruction and simulation-based inference.

Conditional reconstruction. For Default Credit, Spambase, MNIST, BSDS300, and MiniBooNE, clipping, standardization, and PCA are fitted on 2048 context rows and held fixed for 512 test rows, reducing each dataset to $d = 12$. The task is to infer x_{12} from $x_{1:11}$. DensityFM conditions its frozen joint model, while the baselines use fitted joint or conditional estimators. We evaluate conditional NLL and RMSE, with conditional means for DensityFM and the flows estimated from 128 draws. DensityFM achieves the lowest conditional NLL on MiniBooNE, MNIST, and BSDS300, and the second lowest on Default Credit and Spambase. Its RMSE is lowest on MiniBooNE and second lowest on BSDS300. Across four datasets with complete timing records, DensityFM averages 2.50 seconds, substantially below that for GMM, MAF, and NSF. Full results are reported in Table 8.

Simulation-based inference. Each estimator receives the same 8192 prior–simulator pairs. DensityFM conditions its frozen joint model, whereas KDE and the flows fit conditional posteriors to the same bank. We evaluate five benchmark simulators using eight observations and 2000 inferred draws per observation; the first four tasks have 12-dimensional joint spaces and SLCP has dimension 16. C2ST accuracy measures posterior quality, with values closer to 0.5 indicating better agreement with the reference posterior. DensityFM performs best on the linear-Gaussian, Student-t, and four-mode tasks, while NSF performs best on Two Moons and SLCP. Mean runtime is 3.92 seconds for DensityFM, compared with 4.10, 26.8, and 53.2 seconds for KDE, MAF, and NSF. Full results are reported in Table 9.

5 Related Work

Table 4: Conditional reconstruction ($d = 12$). Mean RMSE over five splits; lower is better. Time averages four datasets with complete records.

Dataset	DensityFM	GMM	MAF	NSF
MiniBooNE	**0.664**	0.744	0.669	0.670
Credit	0.435	0.483	0.419	**0.419**
Spambase	0.678	0.724	**0.575**	0.588
MNIST	1.530	1.719	**1.381**	1.429
BSDS300	**0.971**	0.973	1.048	0.995
Time (s)	2.50	9.7	11.3	17.4

Table 5: Simulation-based inference ($d = 12$; SLCP: 16). Mean C2ST over five cohorts; 0.5 is ideal. Time includes fitting and sampling, averaged over five tasks.

Task	DensityFM	KDE	MAF	NSF
Linear Gauss.	**0.515**	0.812	0.540	0.570
Student-t	**0.534**	0.617	0.543	0.545
Four-mode	**0.590**	0.779	0.591	0.592
Two Moons	0.753	0.896	0.699	**0.606**
SLCP	0.878	0.893	0.842	**0.810**
Time (s)	3.92	4.10	26.8	53.2

Neural density estimation. A wide spectrum of density estimation methods has been developed, each offering different trade-offs. Classical methods such as kernel density estimators (KDEs) (Silverman, 1986) and Gaussian mixture models (McLachlan & Peel, 2000) require zero or little fitting efforts, yet may struggle to accurately capture complex high-dimensional distributions (Silverman, 1986; Nagler & Czado, 2016; McLachlan & Peel, 2000). In contrast, modern neural density estimators such as flow models (Dinh et al., 2017; Durkan et al., 2019; Papamakarios et al., 2017; Lipman et al., 2023; Zhai et al., 2025) can model complex distributions using expressive deep neural networks, but typically require computationally expensive iterative optimization for every new dataset (Papamakarios et al., 2021). In this work, we aim to combine accuracy and efficiency through large-scale pretraining, enabling zero-shot density estimation without per-dataset optimization.

Meta-learning and pretraining for density estimation. Similar to our work, DiscoFormer (Ilin et al., 2026) also pretrains a model that maps a context set of samples to density and score estimates. However, it only supports data up to 10 dimensions, while scaling pretrained density estimation to substantially higher dimensions (e.g., 64D) requires non-trivial architectural and pretraining innovations, namely our copula–marginal separation and diversity-aware large-scale pretraining. NPE-PFN (Vetter et al., 2025) recursively applies a pretrained conditional prediction model (TabPFN (Hollmann et al., 2023)) for zero-shot density estimation, but focuses primarily on conditional estimation, unlike ours which focus on unconditional density estimation. Estimating a d-dimensional density with NPE-PFN also requires d sequential TabPFN evaluations, compared with a single forward pass in our method. More broadly, our work relates to meta-learning and prior-fitted approaches (Müller et al., 2022; Reed et al., 2018; Garnelo et al., 2018; Nagler, 2023), with differences in architecture, pretraining paradigm, and the goal of general zero-shot density estimation.

6 Discussion and Conclusion

We introduced DensityFM, the first pretrained density estimator that enables zero-shot neural density estimation for data up to 64 dimensions in a single forward pass. This capability is enabled by a novel copula–marginal separation design, which simplifies pretraining and allows the pretrained model to focus on cross-variable dependence, together with a large-scale pretraining recipe that exposes the model to a broad family of diverse distributions. Across a rich collection of commonly encountered distribution families, nine real-world datasets up to 64 dimensions, and two real-world downstream applications, DensityFM delivers accurate density estimates across diverse regimes while being orders of magnitude faster than alternative neural baselines. By shifting density estimation from per-dataset optimization to amortized inference with a reusable pretrained model, DensityFM provides a versatile foundation for density-driven data analysis and learning workflows.

The main limitation of DensityFM is that the current checkpoints support only $d \leq 64$. Future work will focus on scaling to higher-dimensional data, either through substantially larger-scale pretraining covering even more diverse spretrum of data, or by developing mechanisms that approximate high-dimensional densities through novel compositions of low-dimensional pretrained density estimators.

USE OF AI ASSISTANCE

AI coding assistants were used for software engineering (training/evaluation scripts, plotting, and LaTeX) and for editorial polishing. The authors are responsible for the experimental design, numerical verification, and manuscript content.

REPRODUCIBILITY STATEMENT

The model architecture and learning objective are described in Sections 2.1 and 3. Appendices B and C provide further details on the synthetic training prior, training procedure, and pretraining compute. Evaluation protocols, including data preprocessing, context sizes, metrics, and repeated runs, are described in Section 4 and Appendix D. We provide an anonymous code supplement containing the core model, inference routines, and baseline implementations.

REFERENCES

Michael S. Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *International Conference on Learning Representations (ICLR)*, 2023.

Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, and Jorge L. Reyes-Ortiz. A public domain dataset for human activity recognition using smartphones. In *Proceedings of the 21st European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pp. 437–442, 2013.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In *International Conference on Machine Learning (ICML)*, 2024.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using Real NVP. In *International Conference on Learning Representations (ICLR)*, 2017.

Atray Dixit, Oren Parnas, Biyu Li, Jenny Chen, Charles P. Fulco, Livnat Jerby-Arnon, Nemanja D. Marjanovic, Danielle Dionne, Tyler Burks, Raktima Raychowdhury, Britt Adamson, Thomas M. Norman, Eric S. Lander, Jonathan S. Weissman, Nir Friedman, and Aviv Regev. Perturb-Seq: Dissecting molecular circuits with scalable single-cell RNA profiling of pooled genetic screens. *Cell*, 167(7):1853–1866.e17, 2016. doi: 10.1016/j.cell.2016.11.038.

Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural spline flows. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Marta Garnelo, Dan Rosenbaum, Christopher Maddison, Tiago Ramalho, David Saxton, Murray Shanahan, Yee Whye Teh, Danilo Rezende, and S. M. Ali Eslami. Conditional neural processes. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1704–1713. PMLR, 2018. URL https://proceedings.mlr.press/v80/garnelo18a.html.

Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. TabPFN: A transformer that solves small tabular classification problems in a second. In *International Conference on Learning Representations (ICLR)*, 2023.

Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637:319–326, 2025.

Zhengyang Hu, Yanzhi Chen, Hanxiang Ren, Qunsong Zeng, Youyi Zheng, Adrian Weller, Kaibin Huang, and Yanchao Yang. Infoatlas: A foundation model for zero-shot statistical dependence estimate. *arXiv preprint arXiv:2606.00241*, 2026.

Vasily Ilin, Peter Sushko, and Ranjay Krishna. DiScoFormer: Plug-in density and score estimation with transformers. *arXiv preprint arXiv:2511.05924*, 2026. URL https://arxiv.org/abs/2511.05924v4.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023.

Geoffrey McLachlan and David Peel. *Finite Mixture Models*. Wiley Series in Probability and Statistics. Wiley, 2000. ISBN 9780471006268. doi: 10.1002/0471721182.

Samuel Müller, Noah Hollmann, Sebastian Pineda Arango, Josif Grabocka, and Frank Hutter. Transformers can do bayesian inference. In *International Conference on Learning Representations (ICLR)*, 2022.

Thomas Nagler. Statistical foundations of prior-data fitted networks. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 25660–25676. PMLR, 2023. URL https://proceedings.mlr.press/v202/nagler23a.html.

Thomas Nagler and Claudia Czado. Evading the curse of dimensionality in nonparametric density estimation with simplified vine copulas. *Journal of Multivariate Analysis*, 151:69–89, 2016. doi: 10.1016/j.jmva.2016.07.003.

George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density estimation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64, 2021.

Scott Reed, Yutian Chen, Thomas Paine, Aäron van den Oord, S. M. Ali Eslami, Danilo Rezende, Oriol Vinyals, and Nando de Freitas. Few-shot autoregressive density estimation: Towards learning to learn distributions. In *International Conference on Learning Representations (ICLR)*, 2018. URL https://openreview.net/forum?id=r1wEFyWCW.

Bernard W Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, 1986.

Abe Sklar. Fonctions de répartition à n dimensions et leurs marges. *Publications de l'Institut de Statistique de l'Université de Paris*, 8:229–231, 1959.

Sanjay R. Srivatsan, José L. McFaline-Figueroa, Vijay Ramani, Lauren Saunders, Junyue Cao, Jonathan Packer, Hannah A. Pliner, Dana L. Jackson, Riza M. Daza, Lena Christiansen, Fan Zhang, Frank Steemers, Jay Shendure, and Cole Trapnell. Massively multiplex chemical transcriptomics at single-cell resolution. *Science*, 367(6473):45–51, 2020. doi: 10.1126/science.aax6234.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *Transactions on Machine Learning Research*, 2024.

Julius Vetter, Manuel Gloeckler, Daniel Gedon, and Jakob H Macke. Effortless, simulation-efficient bayesian inference using tabular foundation models. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 82685–82724. Curran Associates, Inc., 2025. doi: 10.52202/085713-2770. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/770cabd044c4eacb6dc5924d9a686dce-Paper-Conference.pdf.

Shuangfei Zhai, Ruixiang Zhang, Preetum Nakkiran, David Berthelot, Jiatao Gu, Huangjie Zheng, Tianrong Chen, Miguel Angel Bautista, Navdeep Jaitly, and Joshua M Susskind. Normalizing flows are capable generative models. In *Forty-second International Conference on Machine Learning*, 2025.

A Proof of Theorem 1

Fix dimension d and an absolutely continuous target density p with marginal densities f_j, CDFs F_j, and copula density c. Let $F = (F_1, \dots, F_d)$ and $\hat{F} = (\hat{F}_1, \dots, \hat{F}_d)$. Sklar's factorization gives

$$p(\boldsymbol{x}) = c(F(\boldsymbol{x})) \prod_{j=1}^{d} f_j(x_j).$$

Replacing c with q_θ and (F_j, f_j) with their KDE estimates $(\hat{F}_j, \hat{f}_j)$ gives

$$\hat{p}_\theta(\boldsymbol{x} \mid \mathcal{D}) = q_\theta\Big(\hat{F}(\boldsymbol{x})\Big) \prod_{j=1}^{d} \hat{f}_j(x_j).$$

Thus it suffices to control the marginal-estimation error and the copula approximation error.

Marginal estimation error. Let $h_{n,j} > 0$ be the bandwidth of the Gaussian KDE for marginal j. Assume $h_{n,j} \to 0$ and $nh_{n,j} \to \infty$. Shrinking the bandwidth removes smoothing bias, while the growing effective local sample size controls variance. Since $f_j \in L_1(\mathbb{R})$, these conditions imply

$$\|\hat{f}_j - f_j\|_1 \xrightarrow{p} 0, \qquad \|\hat{F}_j - F_j\|_\infty \xrightarrow{p} 0. \tag{11}$$

Let $a > 0$ be a common tolerance for the aggregate marginal-density and CDF errors. Equation (11) and fixed d imply that, for any $\delta > 0$ and all sufficiently large n, the event

$$\mathcal{E}_n(a) = \left\{ \sum_{j=1}^{d} \|\hat{f}_j - f_j\|_1 < a, \quad \sum_{j=1}^{d} \|\hat{F}_j - F_j\|_\infty < a \right\}$$

has probability at least $1 - \delta$ under $\mathcal{D} \sim p^n$.

Copula approximation error. Recall that c is the target copula density. We approximate c by a smooth, strictly positive auxiliary density g on $[0,1]^d$ satisfying $\|g - c\|_1 < \epsilon/6$. The smoothness of g allows its conditional transports to be approximated by splines. Its autoregressive factorization is

$$g(\boldsymbol{u}) = \prod_{j=1}^{d} g_j(u_j \mid u_{<j}).$$

Consider a coordinate j and a prefix $u_{<j}$. Let $G_j(\cdot \mid u_{<j})$ be the CDF of the conditional density $g_j(\cdot \mid u_{<j})$. For $Z \sim t_\nu$, the probability integral transform gives

$$V = T_\nu(Z) \sim \mathrm{Unif}(0,1).$$

Inverse-CDF sampling then gives

$$U_j = G_j^{-1}(V \mid u_{<j}) \sim g_j(\cdot \mid u_{<j}).$$

Thus the exact map from the Student-t base to this conditional distribution is

$$S_j^\star(z; u_{<j}) = G_j^{-1}(T_\nu(z) \mid u_{<j}).$$

Smoothness and positivity of g make $S_j^\star$ a smooth monotone map that varies continuously with $u_{<j}$. A sufficiently expressive stacked-spline decoder can approximate the map and its derivative uniformly over all prefixes on any compact z-interval. The change-of-variables formula then controls the conditional-density error on that interval. Choosing the interval to contain at least $1 - \eta$ of the Student-t probability bounds the remaining contribution to the L_1 error by 2η.

Consequently, the conditional errors can be made smaller than $\epsilon/(6d)$. A telescoping expansion of the autoregressive products then gives

$$\|q_\theta - g\|_1 \le \sum_{j=1}^{d} \sup_{u_{<j}} \|q_{\theta,j}(\cdot \mid u_{<j}) - g_j(\cdot \mid u_{<j})\|_1 < \epsilon/6,$$

and hence

$$\|q_\theta - c\|_1 < \epsilon/3. \tag{12}$$

Theorem 1 concerns approximation capacity, so θ may be chosen for the target density p. The finite spline approximation above can be chosen so that q_θ is bounded and Lipschitz. Let $M = \|q_\theta\|_\infty$, and let L be its Lipschitz constant under the ℓ_1 norm.

Combining the errors. The total error comes from estimating the marginal CDFs, estimating the marginal densities, and approximating the target copula. Insert the two intermediate expressions $q_\theta(F(\boldsymbol{x}))\prod_j \hat{f}_j(x_j)$ and $q_\theta(F(\boldsymbol{x}))\prod_j f_j(x_j)$. The triangle inequality separates the three errors. Lipschitz continuity controls the CDF error, boundedness controls the marginal-density error, and the change of variables $u_j = F_j(x_j)$ identifies the copula error. Together these steps give

$$\begin{aligned}\|\hat{p}_\theta(\cdot \mid \mathcal{D}) - p\|_1 \le L\sum_{j=1}^{d}\|\hat{F}_j - F_j\|_\infty \\ + M\sum_{j=1}^{d}\|\hat{f}_j - f_j\|_1 \\ + \|q_\theta - c\|_1. \end{aligned} \tag{13}$$

Equation (12) bounds the copula term by $\epsilon/3$. Set $a = 2\epsilon/[3(L+M)]$. On $\mathcal{E}_n(a)$, the two marginal terms sum to less than $2\epsilon/3$. The total error is therefore less than ϵ. Finally, KDE consistency allows n to be chosen so that $\mathbb{P}(\mathcal{E}_n(a)) \ge 1-\delta$, which proves the theorem. □

B Broad synthetic prior

Each training task samples a structure and an observation channel, then *freezes* all parameters so that context and query points share one law (resampling parameters per draw is a silent way to make the context and query distributions disagree, which inflates the loss and is easily mistaken for a modeling floor). Structures: Gaussian, t, Clayton, Gumbel, and Frank copulas with random parameters; geometric generators (rings, moons, spirals, pinwheels) embedded with a noise floor; low-rank latent push-forwards through random nonlinear maps; and the curved *banana* (chained quadratic shears) and hierarchical *funnel* (Neal-style log-scale) families. Observation channel: 22+ marginal families, monotone asinh warps, a global rotation, and heavy t/Cauchy/Pareto tails. We deliberately omit discretization, many-to-one maps, and hard outliers, all of which create atoms that make continuous-density maximum likelihood ill-posed. Low-dimensional Gaussian-mixture components carry a diagonal floor on the Cholesky factor (≥ 0.35) to avoid near-singular tasks whose differential entropy diverges.

C Architecture and training

DensityFM follows the architecture in section 2.1. Per-coordinate Gaussian KDEs map the context and query to the copula scale. A Transformer set encoder then compresses the variable-size context into m summary tokens, and a causal decoder parameterizes the copula through Student-t bases and stacks of eight-bin rational-quadratic splines. We train one checkpoint per dimension by maximum likelihood (Adam with a cosine schedule), selecting the exponential-moving-average checkpoint by synthetic validation loss. Pretraining the 64-dimensional model takes approximately five days on four NVIDIA H200 GPUs. The released metadata records the width, depth, and spline configuration used for each result. At evaluation time, the checkpoint remains frozen and the target context is only an input to the estimator. Training uses samples alone; no oracle density or score enters the objective.

Pretrained dimensions. Our pretrained model collection contains separate checkpoints for $d \in \{1, 2, 3, 4, 5, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 32, 50, 64\}$. The main density-estimation benchmarks use dimensions up to 64; the one-dimensional model also provides a learned marginal estimator. When the data dimension matches a checkpoint, we use that checkpoint directly. For example, the native-coordinate POWER, GAS, HEPMASS, and MiniBooNE evaluations use 5-, 8-, 22-, and 50-dimensional checkpoints, respectively, while the PCA64 evaluations use a 64-dimensional checkpoint. The neural parameters remain frozen when conditioning on a new dataset.

Reusing a checkpoint at a smaller dimension. A checkpoint with input dimension D can also define a density for $d < D$ through its autoregressive prefix. Let $\boldsymbol{z} = T_{\mathcal{D}}(\boldsymbol{x})$ denote an invertible transformation acting only on the d real coordinates. We zero-pad the transformed context and query

Table 6: Marginal–copula inference-time decomposition on one H200 GPU. For each of three Gaussian-correlation instances, we average 10 timed runs and then average across instances. Model and data loading and host-to-device transfer are excluded.

d	n_c/n_q	Latency (ms)		
		Marginal	Copula	Total
2	256/512	0.568	6.71	7.28
2	1024/512	0.679	7.03	7.71
2	1024/2000	1.69	15.1	16.8
16	256/512	0.627	12.6	13.2
16	1024/512	0.975	12.6	13.6
16	1024/2000	2.53	34.2	36.7
64	256/512	0.736	118	119
64	1024/512	2.93	122	125
64	1024/2000	8.21	448	457

to dimension D, encode the context once as $\boldsymbol{c}$, and retain the first d density factors:

$$\log \hat{p}_d(\boldsymbol{x} \mid \mathcal{D}) = \sum_{j=1}^{d} \log q_{\theta,j}(z_j \mid z_{<j}, \boldsymbol{c}) + \log \left|\det J_{T_{\mathcal{D}}}(\boldsymbol{x})\right| . \tag{14}$$

The real coordinates precede the padding in the autoregressive order, and the transformation does not mix the two. Each omitted conditional integrates to one, so the retained prefix defines a normalized d-dimensional density. For example, a 64-dimensional checkpoint can score 63-dimensional BSDS300 inputs using its first 63 factors, with one decoder evaluation per query and no auxiliary-variable sampling or weight updates. We use this construction for BSDS300 in Table 3.

Exploratory design studies. Exploratory per-axis Gaussianization runs did not improve on linear whitening. Because those runs also varied context size, they do not isolate transform depth as the cause. Ensembling identical architectures likewise did not improve performance, whereas averaging different architectures did, suggesting that ensemble gains depend on model diversity.

D Additional experimental results

We first report the inference-time decomposition and the smaller supporting evaluations for tail dependence, real-data density estimation, conditional reconstruction, and simulation-based inference. The complete results for the 22 common density families are collected at the end of this section because they span several consecutive pages. The complete $d = 64$ real-data results already appear in the main text and are not repeated. Unless stated otherwise, these experiments use the same splits, preprocessing, context limits, and metrics as the main-text evaluations. The expanded tables retain the complete method rosters and variation across repeated runs, making the aggregate comparisons traceable to individual datasets and distribution families.

Inference-time decomposition. We separately profile the context-fitted marginal front and the frozen neural copula. Marginal time includes the per-coordinate Gaussian KDE and CDF transforms for the context and queries, together with the final marginal Jacobian; copula time includes context encoding and query scoring. As shown in Table 6, neither stage requires target-specific gradient updates. The $d = 64$, $n_c = 1024$ rows match the context size used in the paper and serve as a stress setting beyond that checkpoint's training maximum of 768. These fixed-workload latencies are not directly comparable to the dataset-level runtimes in the main tables, which use different datasets and aggregation protocols.

D.1 Complete real-data dimension sweep

Native-coordinate protocol. The main-text native-coordinate table uses no PCA or pair-copula composition. A flow result is marked unsuccessful if any split is nonfinite or has held-out log-likelihood below -10 nats per dimension. BSDS300 has 63 prepared coordinates; we evaluate it

with the $d = 64$ model using zero padding and the first 63 autoregressive factors, as described in Appendix C. Runtime excludes model loading and data preparation. After one warmup, we take the median of three timed runs per split and average across splits.

Table 7 reports the complete 11-method dimension sweep referenced in the main text. The PCA, clipping, and standardization maps are fitted on the outer-training split and then held fixed for validation and test rows. Input continuization precedes PCA for the $d = 64$ image experiments, and the context or fitting set is capped at 1024 rows as stated in the main-text protocol.

D.2 Complete latent-coordinate imputation results

Table 8 extends the corresponding main-text comparison to every evaluated method. The native dimensions are 23 (Default Credit), 50 (MiniBooNE), 57 (Spambase), 63 (BSDS300), and 784 (MNIST). For every outer split, clipping, standardization, and PCA are fitted on the 2048 context rows and then held fixed for the 512 test rows. KNN and MICE are point predictors and therefore enter RMSE but not conditional NLL.

D.3 Simulation-based inference protocol

Table 9 extends the corresponding main-text comparison to the complete five-method roster. For each task and cohort, all methods receive the same bank of 8192 prior–simulator pairs; 10% of the bank is reserved for selecting KDE and flow hyperparameters. DensityFM remains frozen. Evaluation uses eight held-out observations, 2000 inferred posterior draws per observation, and 4000 reference draws. The linear-Gaussian, multivariate Student-t, and four-component mixture tasks have analytic conditional posteriors. Two Moons uses exact-likelihood rejection sampling, and SLCP uses mode-stratified, exact-likelihood multichain MCMC. Their joint dimensions are respectively $6+6$, $6+6$, $6+6$, $10+2$, and $11+5$, where the first term is the observation dimension.

C2ST trains a three-fold neural classifier on equal-sized sets of 512 inferred and reference draws for each observation. We symmetrize accuracy about chance, so 0.5 is ideal and larger values indicate a more distinguishable approximate posterior. Reported variation is across five independently simulated cohorts.

D.4 Complete results on 22 common density families

Table 11 defines the 22 analytic constructions, and Table 12 reports the complete estimator roster at $d \in \{2, 4, 8, 12, 16, 20, 32, 50, 64\}$. Figure 4a aggregates these families, while panel (b) uses the hardware-matched workloads for which timings are available.

Table 7: Complete real-data dimension sweep. Held-out log-likelihood per dimension (higher is better), mean ± sample standard deviation over three outer splits. Bold and underlined values mark the best and second-best mean over the complete 11-method roster. PCA, clipping, and standardization are fit on outer-training data only.

Classical and copula estimators							
Dataset	d	DensityFM	Gaussian	KDE	GMM	FA	Gauss. copula
Breast Cancer	12	**−1.357 ± 0.025**	−1.523 ± 0.045	−1.489 ± 0.049	−1.441 ± 0.033	−1.528 ± 0.044	−1.496 ± 0.048
	14	**−1.264 ± 0.024**	−1.426 ± 0.030	−1.397 ± 0.045	−1.315 ± 0.015	−1.431 ± 0.029	−1.420 ± 0.051
	16	**−1.093 ± 0.047**	−1.290 ± 0.041	−1.257 ± 0.053	−1.186 ± 0.053	−1.295 ± 0.041	−1.289 ± 0.071
	18	**−0.898 ± 0.039**	−1.153 ± 0.046	−1.083 ± 0.049	−1.058 ± 0.076	−1.157 ± 0.045	−1.136 ± 0.057
	20	**−0.756 ± 0.046**	−1.019 ± 0.046	−0.951 ± 0.043	−0.923 ± 0.074	−1.024 ± 0.046	−0.999 ± 0.044
Digits	12	**−1.309 ± 0.032**	−1.878 ± 0.024	−1.479 ± 0.024	−1.451 ± 0.088	−1.875 ± 0.028	−1.784 ± 0.015
	14	**−1.261 ± 0.004**	−1.825 ± 0.020	−1.423 ± 0.021	−1.376 ± 0.015	−1.822 ± 0.023	−1.738 ± 0.016
	16	**−1.121 ± 0.012**	−1.769 ± 0.024	−1.345 ± 0.016	−1.258 ± 0.008	−1.767 ± 0.027	−1.669 ± 0.026
	18	**−1.063 ± 0.018**	−1.722 ± 0.021	−1.308 ± 0.022	−1.236 ± 0.050	−1.721 ± 0.021	−1.614 ± 0.017
	20	**−1.012 ± 0.025**	−1.674 ± 0.023	−1.234 ± 0.022	−1.156 ± 0.076	−1.675 ± 0.019	−1.560 ± 0.022
MiniBooNE	12	**−1.549 ± 0.001**	−1.849 ± 0.005	−1.704 ± 0.007	−1.621 ± 0.018	−1.851 ± 0.005	−1.811 ± 0.003
	14	**−1.465 ± 0.002**	−1.776 ± 0.005	−1.656 ± 0.003	−1.557 ± 0.022	−1.778 ± 0.004	−1.743 ± 0.003
	16	**−1.403 ± 0.001**	−1.710 ± 0.004	−1.602 ± 0.003	−1.506 ± 0.019	−1.712 ± 0.004	−1.681 ± 0.003
	18	**−1.344 ± 0.004**	−1.648 ± 0.004	−1.545 ± 0.003	−1.457 ± 0.014	−1.651 ± 0.004	−1.623 ± 0.003
	20	**−1.298 ± 0.004**	−1.593 ± 0.005	−1.489 ± 0.003	−1.401 ± 0.007	−1.597 ± 0.005	−1.569 ± 0.004

Vines and flows							
Dataset	d	DensityFM	TLL	Param. vine	MAF	RealNVP	NSF
Breast Cancer	12	**−1.357 ± 0.025**	−1.598 ± 0.098	−1.459 ± 0.044	−1.454 ± 0.068	−1.530 ± 0.053	−1.433 ± 0.057
	14	**−1.264 ± 0.024**	−1.528 ± 0.052	−1.375 ± 0.045	−1.319 ± 0.068	−1.511 ± 0.125	−1.328 ± 0.035
	16	**−1.093 ± 0.047**	−1.559 ± 0.203	−1.245 ± 0.065	−1.174 ± 0.088	−1.375 ± 0.158	−1.199 ± 0.049
	18	**−0.898 ± 0.039**	−1.676 ± 0.205	−1.103 ± 0.040	−1.089 ± 0.174	−2.444 ± 2.182	−1.046 ± 0.041
	20	**−0.756 ± 0.046**	−1.669 ± 0.289	−0.962 ± 0.035	−0.984 ± 0.216	−4.905 ± 6.628	−0.915 ± 0.060
Digits	12	**−1.309 ± 0.032**	−1.602 ± 0.021	−1.694 ± 0.004	−1.451 ± 0.057	−1.486 ± 0.029	−1.372 ± 0.017
	14	**−1.261 ± 0.004**	−1.577 ± 0.027	−1.651 ± 0.012	−1.359 ± 0.036	−1.464 ± 0.004	−1.334 ± 0.029
	16	**−1.121 ± 0.012**	−1.503 ± 0.041	−1.552 ± 0.012	−1.291 ± 0.042	−1.413 ± 0.029	−1.252 ± 0.019
	18	**−1.063 ± 0.018**	−1.465 ± 0.031	−1.490 ± 0.022	−1.304 ± 0.121	−1.324 ± 0.019	−1.196 ± 0.006
	20	**−1.012 ± 0.025**	−1.445 ± 0.073	−1.409 ± 0.023	−1.262 ± 0.112	−1.296 ± 0.040	−1.127 ± 0.010
MiniBooNE	12	**−1.549 ± 0.001**	−1.730 ± 0.017	−1.763 ± 0.006	−1.589 ± 0.011	−1.644 ± 0.008	−1.622 ± 0.012
	14	**−1.465 ± 0.002**	−1.690 ± 0.012	−1.687 ± 0.005	−1.526 ± 0.007	−1.609 ± 0.069	−1.556 ± 0.006
	16	**−1.403 ± 0.001**	−1.676 ± 0.026	−1.626 ± 0.004	−1.520 ± 0.045	−1.530 ± 0.006	−1.507 ± 0.006
	18	**−1.344 ± 0.004**	−1.677 ± 0.042	−1.569 ± 0.005	−1.442 ± 0.040	−1.493 ± 0.021	−1.450 ± 0.004
	20	**−1.298 ± 0.004**	−1.661 ± 0.051	−1.514 ± 0.005	−1.376 ± 0.025	−1.425 ± 0.007	−1.391 ± 0.017

Table 8: Complete conditional reconstruction results. Each dataset is reduced by outer-context-fitted PCA to $d = 12$, and x_{12} is inferred from $x_{1:11}$. Entries are mean ± sample standard deviation over five outer splits. Bold and underlined values mark the best and second-best mean in each row. KNN and MICE provide point imputations only and therefore do not appear in the conditional-NLL panel. The context-mean predictor is omitted because it is numerically identical to the Gaussian conditional mean after PCA.

Conditional NLL ↓

Dataset	DensityFM	Gaussian	KDE	GMM	MAF	RealNVP	NSF
MiniBooNE	**0.910 ± 0.071**	1.365 ± 0.046	1.160 ± 0.058	1.034 ± 0.081	0.951 ± 0.078	0.973 ± 0.097	1.008 ± 0.061
Default Credit	0.237 ± 0.102	1.240 ± 0.017	0.848 ± 0.041	**0.089 ± 0.205**	0.484 ± 0.119	0.467 ± 0.110	0.393 ± 0.125
Spambase	0.607 ± 0.135	1.420 ± 0.069	0.817 ± 0.089	0.779 ± 0.204	0.686 ± 0.125	0.698 ± 0.059	**0.521 ± 0.083**
MNIST	**1.507 ± 0.046**	2.442 ± 0.083	1.904 ± 0.045	1.719 ± 0.049	1.517 ± 0.038	1.621 ± 0.153	1.589 ± 0.070
BSDS300	**0.589 ± 0.040**	1.410 ± 0.033	1.249 ± 0.031	0.620 ± 0.035	0.834 ± 0.066	0.999 ± 0.131	0.807 ± 0.116

Imputation RMSE: statistical and nonparametric methods

Dataset	DensityFM	Gaussian	KDE	GMM	KNN
MiniBooNE	**0.664 ± 0.043**	0.947 ± 0.043	0.774 ± 0.046	0.744 ± 0.056	0.761 ± 0.042
Default Credit	0.435 ± 0.056	0.835 ± 0.013	0.534 ± 0.031	0.483 ± 0.029	0.547 ± 0.033
Spambase	0.678 ± 0.071	0.991 ± 0.080	0.638 ± 0.080	0.724 ± 0.070	0.581 ± 0.041
MNIST	1.531 ± 0.051	2.762 ± 0.260	1.640 ± 0.069	1.719 ± 0.092	1.609 ± 0.093
BSDS300	0.971 ± 0.040	0.982 ± 0.036	0.998 ± 0.040	0.973 ± 0.034	**0.966 ± 0.025**

Imputation RMSE: imputation and flow methods

Dataset	DensityFM	MICE	MAF	RealNVP	NSF
MiniBooNE	**0.664 ± 0.043**	0.947 ± 0.043	0.669 ± 0.052	0.687 ± 0.053	0.670 ± 0.050
Default Credit	0.435 ± 0.056	0.835 ± 0.013	**0.419 ± 0.026**	0.457 ± 0.075	**0.419 ± 0.035**
Spambase	0.678 ± 0.071	0.991 ± 0.080	**0.575 ± 0.073**	0.589 ± 0.053	0.588 ± 0.047
MNIST	1.531 ± 0.051	2.762 ± 0.260	**1.381 ± 0.061**	1.407 ± 0.070	1.429 ± 0.096
BSDS300	0.971 ± 0.040	0.982 ± 0.036	1.048 ± 0.102	1.067 ± 0.058	0.995 ± 0.042

Table 9: Complete simulation-based inference results. C2ST accuracy, mean ± sample standard deviation over five independent cohorts; values closer to 0.5 are better. Runtime is H200 wall time for fitting, where required, and drawing posteriors for eight observations. Bold and underlined values mark the best and second-best displayed mean.

Method	Linear	Student-t	Four-mode	Two Moons	SLCP	Time (s)
DensityFM	**0.515 ± 0.004**	**0.534 ± 0.011**	**0.590 ± 0.021**	0.753 ± 0.029	0.878 ± 0.015	3.92
KDE	0.812 ± 0.007	0.617 ± 0.009	0.779 ± 0.025	0.896 ± 0.008	0.893 ± 0.009	4.10
MAF	0.540 ± 0.011	0.543 ± 0.009	0.591 ± 0.011	0.699 ± 0.030	0.842 ± 0.037	26.8
RealNVP	0.552 ± 0.021	0.538 ± 0.010	0.591 ± 0.022	0.715 ± 0.072	0.842 ± 0.019	27.9
NSF	0.570 ± 0.006	0.545 ± 0.001	0.592 ± 0.014	**0.606 ± 0.023**	**0.810 ± 0.033**	53.2

Table 10: Complete finite-threshold tail-dependence results underlying figure 4c. Absolute error of the ten-repetition mean estimate of $\lambda_L(q)$ or $\lambda_U(q)$ (lower is better). Bold and underlined values mark the best and second-best mean in each family–threshold cell; values below .0005 are printed as $< .001$. Neural-flow estimates pass mass and grid-convergence checks.

Family	Tail	q	DensityFM	VDC	TLL	Gaussian	KDE	GMM	MAF	RealNVP	NSF
Gaussian ($\rho = .7$)	L	0.001	0.019	0.143	0.066	0.011	0.094	0.087	0.021	0.098	**¡0.001**
Gaussian ($\rho = .7$)	L	0.010	0.016	0.096	0.020	0.010	0.034	0.064	**0.005**	0.033	0.008
Gaussian ($\rho = .7$)	L	0.025	0.015	0.028	0.011	0.009	0.029	0.033	0.001	0.015	**¡0.001**
Clayton ($\theta = 2$)	L	0.001	**0.008**	0.658	0.376	0.551	0.095	0.515	0.075	0.020	0.237
Clayton ($\theta = 2$)	L	0.010	**0.007**	0.213	0.107	0.444	0.051	0.128	0.020	0.016	0.037
Clayton ($\theta = 2$)	L	0.025	0.007	0.031	0.073	0.380	**0.001**	0.004	0.029	0.007	0.020
Clayton ($\theta = 5$)	L	0.001	0.599	0.807	0.799	**0.595**	0.871	0.863	0.871	0.871	0.871
Clayton ($\theta = 5$)	L	0.010	0.347	**0.239**	0.359	0.474	0.859	0.731	0.861	0.863	0.859
Clayton ($\theta = 5$)	L	0.025	0.073	**0.064**	0.148	0.408	0.243	0.308	0.205	0.226	0.146
Gumbel ($\theta = 2.5$)	U	0.001	**0.176**	0.632	0.483	0.418	0.678	0.525	0.659	0.673	0.676
Gumbel ($\theta = 2.5$)	U	0.010	0.087	0.200	0.097	0.308	0.036	0.247	0.099	0.088	**0.016**
Gumbel ($\theta = 2.5$)	U	0.025	0.061	0.027	0.053	0.250	0.003	0.099	0.015	**0.001**	0.035
Frank ($\theta = 5$)	L	0.001	0.002	**¡0.001**	**¡0.001**	0.103	**¡0.001**	0.003	0.004	**¡0.001**	0.001
Frank ($\theta = 5$)	L	0.010	0.011	**0.001**	0.002	0.156	0.003	0.010	0.009	0.002	0.005
Frank ($\theta = 5$)	L	0.025	0.022	**0.002**	0.008	0.154	0.022	**0.002**	0.010	0.017	0.004
Student-t ($\rho = .7, \nu = 3$)	L	0.001	0.103	0.423	0.246	0.306	0.141	0.111	0.034	0.079	**0.001**
Student-t ($\rho = .7, \nu = 3$)	L	0.010	0.087	0.179	0.066	0.214	**0.018**	0.059	0.021	0.031	0.025
Student-t ($\rho = .7, \nu = 3$)	L	0.025	0.075	0.034	0.050	0.165	0.024	0.033	0.023	**0.010**	0.020
Joe ($\theta = 3$)	U	0.001	0.066	0.686	0.319	0.574	**0.059**	0.524	0.240	0.105	0.473
Joe ($\theta = 3$)	U	0.010	0.024	0.202	0.095	0.466	0.066	0.180	0.015	0.009	**0.002**
Joe ($\theta = 3$)	U	0.025	0.023	0.029	0.074	0.401	0.026	0.042	**0.009**	0.018	0.013
BB1 ($\theta = .5, \delta = 2$)	U	0.001	0.191	0.552	0.270	0.295	**0.026**	0.414	0.169	0.089	0.414
BB1 ($\theta = .5, \delta = 2$)	U	0.010	0.126	0.246	0.048	0.185	0.030	0.151	0.007	**0.006**	0.029
BB1 ($\theta = .5, \delta = 2$)	U	0.025	0.096	0.077	0.033	0.130	**0.002**	0.033	0.019	0.005	0.024

Table 11: The 22 common density families. Each instance fixes its parameters before drawing the paired context and query cohorts.

Family	Construction
Banana	Gaussian transformed by chained quadratic shears
Clayton copula	non-Gaussian marginals joined by a Clayton copula
Gaussian copula	non-Gaussian marginals joined by a Gaussian copula
t copula	non-Gaussian marginals joined by a Student-t copula
Dirichlet (ALR)	Dirichlet density in additive-log-ratio coordinates
Funnel	hierarchical funnel with input-dependent scales
Gaussian, corr.	Gaussian with dense correlation
Gaussian, ill-cond.	Gaussian with an ill-conditioned covariance
Gaussian, isotropic	isotropic multivariate Gaussian
Generalized Gaussian	independent generalized-Gaussian marginals
GMM, separated	three-component, well-separated Gaussian mixture
GMM, overlapping	eight-component, overlapping Gaussian mixture
Heavy-tailed t	multivariate Student-t with randomized heavy tails
Laplace	independent Laplace marginals
Noisy torus	noisy toroidal manifold embedded in d dimensions
Rosenbrock	Rosenbrock-type curved density
Sinh–arcsinh	Gaussian push-forward through a sinh–arcsinh warp
Skew normal	multivariate skew-normal density
Student-t, $\nu = 2$	elliptical Student-t with two degrees of freedom
Student-t, $\nu = 5$	elliptical Student-t with five degrees of freedom
Uniform box	uniform density on an axis-aligned box
Warped GMM	Gaussian mixture followed by a smooth nonlinear warp

Table 12: Complete results on the 22 common density families. Oracle gap per dimension, reported as mean ± sample standard deviation over repeated paired evaluations; lower is better. Bold and underlined values mark the best and second-best mean in each row. Finite means of at least 10 are displayed as > 10 for readability; – denotes a result withheld by the fixed numerical-stability gate.

	$d = 2$						
Family	DensityFM	Gaussian	KDE	GMM	MAF	RealNVP	NSF
Banana	**0.003** ± **0.003**	0.007 ± 0.007	0.016 ± 0.011	0.005 ± 0.004	0.015 ± 0.008	–	0.032 ± 0.013
Clayton copula	**0.009** ± **0.003**	0.330 ± 0.259	0.243 ± 0.171	0.050 ± 0.034	0.017 ± 0.010	0.023 ± 0.017	0.071 ± 0.043
Gaussian copula	**0.006** ± **0.004**	0.220 ± 0.207	0.170 ± 0.166	0.029 ± 0.024	0.077 ± 0.132	0.021 ± 0.010	0.085 ± 0.060
t copula	**0.008** ± **0.003**	0.213 ± 0.185	0.119 ± 0.082	0.034 ± 0.028	0.023 ± 0.006	0.040 ± 0.023	0.075 ± 0.045
Dirichlet (ALR)	**0.003** ± **0.002**	0.016 ± 0.012	0.019 ± 0.016	0.008 ± 0.001	0.010 ± 0.003	0.016 ± 0.007	0.032 ± 0.019
Funnel	**0.013** ± **0.003**	0.797 ± 0.385	3.937 ± 5.575	0.189 ± 0.202	0.016 ± 0.008	0.018 ± 0.010	0.314 ± 0.268
Gaussian, corr.	0.003 ± 0.002	**0.002** ± **0.002**	0.013 ± 0.004	0.006 ± 0.007	0.010 ± 0.004	0.014 ± 0.005	0.028 ± 0.007
Gaussian, ill-cond.	0.003 ± 0.002	**0.002** ± **0.002**	0.223 ± 0.147	0.005 ± 0.005	0.015 ± 0.003	0.022 ± 0.007	0.040 ± 0.011
Gaussian, isotropic	0.003 ± 0.002	**0.002** ± **0.002**	0.012 ± 0.004	0.008 ± 0.007	0.010 ± 0.005	0.014 ± 0.005	0.030 ± 0.006
Generalized Gaussian	**0.006** ± **0.001**	0.040 ± 0.024	0.025 ± 0.006	0.016 ± 0.004	0.027 ± 0.010	0.033 ± 0.027	0.044 ± 0.015
GMM, separated	0.011 ± 0.010	0.384 ± 0.352	0.038 ± 0.027	**0.010** ± **0.008**	0.049 ± 0.049	0.506 ± 1.021	0.085 ± 0.074
GMM, overlapping	**0.010** ± **0.007**	0.137 ± 0.081	0.022 ± 0.008	0.013 ± 0.010	0.028 ± 0.013	0.038 ± 0.019	0.035 ± 0.006
Heavy-tailed t	**0.007** ± **0.002**	0.177 ± 0.027	0.121 ± 0.083	0.030 ± 0.017	0.063 ± 0.057	0.038 ± 0.020	0.118 ± 0.045
Laplace	**0.005** ± **0.002**	0.071 ± 0.003	0.036 ± 0.011	0.019 ± 0.007	0.050 ± 0.053	1.452 ± 3.178	0.045 ± 0.015
Noisy torus	**0.007** ± **0.002**	0.833 ± 0.204	0.204 ± 0.118	0.073 ± 0.042	2.255 ± 4.621	–	0.053 ± 0.015
Rosenbrock	**0.004** ± **0.002**	0.022 ± 0.014	0.016 ± 0.008	0.011 ± 0.006	0.019 ± 0.010	0.024 ± 0.011	0.030 ± 0.009
Sinh–arcsinh	**0.008** ± **0.005**	0.080 ± 0.034	0.103 ± 0.161	0.020 ± 0.007	0.019 ± 0.008	0.022 ± 0.013	0.039 ± 0.008
Skew normal	**0.002** ± **0.002**	0.010 ± 0.007	0.025 ± 0.016	0.007 ± 0.003	0.013 ± 0.004	0.016 ± 0.007	0.035 ± 0.017
Student-t, $\nu = 2$	**0.010** ± **0.002**	0.628 ± 0.293	1.678 ± 3.054	0.037 ± 0.014	0.028 ± 0.012	0.024 ± 0.011	0.313 ± 0.319
Student-t, $\nu = 5$	**0.006** ± **0.002**	0.062 ± 0.010	0.055 ± 0.022	0.014 ± 0.010	0.029 ± 0.009	0.026 ± 0.010	0.058 ± 0.009
Uniform box	**0.015** ± **0.004**	0.179 ± 0.008	0.056 ± 0.004	0.072 ± 0.009	0.109 ± 0.092	0.171 ± 0.244	0.066 ± 0.026
Warped GMM	**0.012** ± **0.005**	0.392 ± 0.140	0.037 ± 0.009	0.015 ± 0.005	–	–	0.041 ± 0.011

Table 12 (continued).

$d = 4$							
Family	DensityFM	Gaussian	KDE	GMM	MAF	RealNVP	NSF
Banana	**0.004** ± **0.001**	0.008 ± 0.004	0.022 ± 0.003	0.008 ± 0.003	0.010 ± 0.003	0.022 ± 0.004	0.022 ± 0.002
Clayton copula	**0.015** ± **0.007**	0.292 ± 0.174	0.211 ± 0.198	0.066 ± 0.030	0.042 ± 0.013	0.074 ± 0.042	0.061 ± 0.039
Gaussian copula	**0.009** ± **0.006**	0.155 ± 0.121	0.097 ± 0.052	0.033 ± 0.020	0.026 ± 0.012	0.036 ± 0.023	0.054 ± 0.029
t copula	**0.013** ± **0.003**	0.170 ± 0.116	0.148 ± 0.075	0.044 ± 0.024	0.042 ± 0.040	0.041 ± 0.014	0.058 ± 0.021
Dirichlet (ALR)	**0.005** ± **0.001**	0.022 ± 0.008	0.036 ± 0.003	0.017 ± 0.006	0.021 ± 0.011	0.030 ± 0.011	0.023 ± 0.002
Funnel	**0.022** ± **0.005**	1.157 ± 0.454	1.476 ± 0.877	0.172 ± 0.055	0.055 ± 0.041	–	0.425 ± 0.279
Gaussian, corr.	**0.002** ± **0.001**	**0.002** ± **0.001**	0.022 ± 0.002	**0.002** ± **0.001**	0.007 ± 0.001	0.020 ± 0.005	0.022 ± 0.004
Gaussian, ill-cond.	**0.002** ± **0.001**	**0.002** ± **0.001**	0.208 ± 0.081	0.003 ± 0.001	0.021 ± 0.009	0.029 ± 0.010	0.050 ± 0.015
Gaussian, isotropic	**0.002** ± **0.001**	**0.002** ± **0.001**	0.024 ± 0.006	**0.002** ± **0.001**	0.008 ± 0.002	0.017 ± 0.005	0.021 ± 0.004
Generalized Gaussian	**0.009** ± **0.002**	0.043 ± 0.015	0.042 ± 0.009	0.022 ± 0.009	0.053 ± 0.011	0.051 ± 0.016	0.031 ± 0.007
GMM, separated	0.013 ± 0.011	0.303 ± 0.285	0.055 ± 0.037	**0.010** ± **0.007**	0.057 ± 0.045	0.055 ± 0.031	0.089 ± 0.081
GMM, overlapping	**0.013** ± **0.008**	0.223 ± 0.145	0.048 ± 0.016	0.020 ± 0.016	0.066 ± 0.029	0.086 ± 0.040	0.051 ± 0.021
Heavy-tailed t	**0.010** ± **0.002**	0.182 ± 0.007	0.141 ± 0.033	0.046 ± 0.012	0.083 ± 0.041	0.062 ± 0.013	0.120 ± 0.013
Laplace	**0.010** ± **0.001**	0.076 ± 0.007	0.059 ± 0.009	0.029 ± 0.005	0.792 ± 1.640	0.054 ± 0.012	0.042 ± 0.004
Noisy torus	**0.106** ± **0.079**	0.841 ± 0.245	0.332 ± 0.185	0.536 ± 0.221	0.437 ± 0.167	0.458 ± 0.170	0.180 ± 0.059
Rosenbrock	**0.006** ± **0.002**	0.022 ± 0.003	0.028 ± 0.009	0.013 ± 0.006	0.009 ± 0.002	0.016 ± 0.003	0.023 ± 0.003
Sinh–arcsinh	**0.010** ± **0.003**	0.072 ± 0.038	0.261 ± 0.288	0.028 ± 0.012	0.063 ± 0.091	0.051 ± 0.052	0.056 ± 0.037
Skew normal	**0.003** ± **0.002**	0.011 ± 0.007	0.032 ± 0.009	0.007 ± 0.004	0.016 ± 0.013	0.030 ± 0.015	0.021 ± 0.001
Student-t, $\nu = 2$	**0.062** ± **0.105**	0.890 ± 0.454	1.163 ± 0.836	0.143 ± 0.155	0.073 ± 0.048	–	0.291 ± 0.257
Student-t, $\nu = 5$	**0.006** ± **0.002**	0.064 ± 0.008	0.076 ± 0.018	0.010 ± 0.001	0.040 ± 0.006	0.041 ± 0.012	0.049 ± 0.011
Uniform box	**0.024** ± **0.002**	0.177 ± 0.004	0.108 ± 0.003	0.134 ± 0.005	0.127 ± 0.006	0.113 ± 0.013	0.068 ± 0.005
Warped GMM	**0.016** ± **0.005**	0.274 ± 0.124	0.065 ± 0.014	0.022 ± 0.008	0.056 ± 0.019	0.070 ± 0.030	0.057 ± 0.020

Table 12 (continued).

$d = 8$							
Family	DensityFM	Gaussian	KDE	GMM	MAF	RealNVP	NSF
Banana	**0.006** ± **0.000**	0.009 ± 0.002	0.047 ± 0.002	0.010 ± 0.004	0.012 ± 0.002	0.036 ± 0.004	0.029 ± 0.004
Clayton copula	**0.022** ± **0.004**	0.319 ± 0.143	0.222 ± 0.083	0.133 ± 0.047	–	–	0.083 ± 0.029
Gaussian copula	**0.012** ± **0.004**	0.155 ± 0.088	0.186 ± 0.026	0.071 ± 0.041	0.050 ± 0.022	–	0.072 ± 0.026
t copula	**0.021** ± **0.002**	0.186 ± 0.083	0.233 ± 0.036	0.091 ± 0.040	0.057 ± 0.013	0.101 ± 0.060	0.091 ± 0.023
Dirichlet (ALR)	**0.010** ± **0.002**	0.031 ± 0.004	0.069 ± 0.004	0.026 ± 0.002	0.033 ± 0.008	0.046 ± 0.007	0.030 ± 0.003
Funnel	**0.025** ± **0.005**	1.300 ± 0.410	1.346 ± 0.561	0.618 ± 0.910	–	–	0.438 ± 0.146
Gaussian, corr.	**0.003** ± **0.001**	**0.003** ± **0.001**	0.048 ± 0.002	**0.003** ± **0.001**	0.015 ± 0.002	0.044 ± 0.011	0.025 ± 0.002
Gaussian, ill-cond.	**0.003** ± **0.001**	**0.003** ± **0.001**	0.284 ± 0.077	**0.003** ± **0.001**	0.031 ± 0.005	0.069 ± 0.028	0.057 ± 0.007
Gaussian, isotropic	**0.003** ± **0.001**	**0.003** ± **0.001**	0.045 ± 0.002	**0.003** ± **0.001**	0.010 ± 0.002	0.035 ± 0.004	0.025 ± 0.002
Generalized Gaussian	**0.008** ± **0.001**	0.050 ± 0.008	0.069 ± 0.007	0.039 ± 0.007	0.072 ± 0.034	0.094 ± 0.013	0.040 ± 0.009
GMM, separated	**0.010** ± **0.007**	0.231 ± 0.209	0.120 ± 0.087	0.066 ± 0.116	0.044 ± 0.029	0.072 ± 0.028	0.108 ± 0.112
GMM, overlapping	**0.013** ± **0.009**	0.244 ± 0.171	0.097 ± 0.024	0.028 ± 0.023	0.085 ± 0.051	0.107 ± 0.045	0.091 ± 0.052
Heavy-tailed t	**0.009** ± **0.003**	0.184 ± 0.021	0.145 ± 0.032	0.081 ± 0.019	–	–	0.125 ± 0.019
Laplace	**0.007** ± **0.001**	0.076 ± 0.002	0.082 ± 0.004	0.059 ± 0.004	0.087 ± 0.004	0.128 ± 0.025	0.046 ± 0.002
Noisy torus	**0.089** ± **0.054**	0.422 ± 0.122	0.336 ± 0.098	0.261 ± 0.107	0.355 ± 0.092	0.368 ± 0.122	0.308 ± 0.059
Rosenbrock	**0.011** ± **0.002**	0.036 ± 0.008	0.055 ± 0.004	0.029 ± 0.005	0.025 ± 0.003	0.036 ± 0.004	0.043 ± 0.007
Sinh–arcsinh	**0.015** ± **0.002**	0.069 ± 0.016	0.272 ± 0.083	0.045 ± 0.012	0.050 ± 0.006	0.080 ± 0.019	0.082 ± 0.011
Skew normal	**0.004** ± **0.001**	0.012 ± 0.001	0.057 ± 0.002	0.009 ± 0.003	0.024 ± 0.007	0.047 ± 0.007	0.030 ± 0.004
Student-t, $\nu = 2$	**0.019** ± **0.003**	1.120 ± 0.925	1.005 ± 0.449	0.387 ± 0.391	1.290 ± 2.664	–	0.641 ± 0.717
Student-t, $\nu = 5$	**0.007** ± **0.002**	0.089 ± 0.009	0.129 ± 0.012	0.021 ± 0.003	0.048 ± 0.007	0.081 ± 0.023	0.068 ± 0.008
Uniform box	**0.024** ± **0.002**	0.179 ± 0.004	0.184 ± 0.003	0.171 ± 0.004	0.182 ± 0.003	0.196 ± 0.023	0.071 ± 0.005
Warped GMM	**0.019** ± **0.007**	0.271 ± 0.153	0.102 ± 0.023	0.051 ± 0.020	0.097 ± 0.037	0.142 ± 0.054	0.094 ± 0.037

Table 12 (continued).

$d = 12$							
Family	DensityFM	Gaussian	KDE	GMM	MAF	RealNVP	NSF
Banana	**0.009** ± **0.002**	0.012 ± 0.007	0.069 ± 0.003	0.011 ± 0.006	0.016 ± 0.008	0.048 ± 0.005	0.030 ± 0.007
Clayton copula	**0.045** ± **0.005**	0.384 ± 0.102	0.309 ± 0.070	0.188 ± 0.037	–	–	0.116 ± 0.028
Gaussian copula	**0.028** ± **0.006**	0.254 ± 0.057	0.320 ± 0.024	0.120 ± 0.030	0.170 ± 0.133	–	0.119 ± 0.015
t copula	**0.041** ± **0.003**	0.276 ± 0.043	0.389 ± 0.051	0.159 ± 0.035	0.099 ± 0.018	–	0.129 ± 0.014
Dirichlet (ALR)	**0.013** ± **0.001**	0.037 ± 0.003	0.106 ± 0.009	0.037 ± 0.003	0.089 ± 0.054	0.097 ± 0.008	0.049 ± 0.010
Funnel	**0.047** ± **0.009**	1.569 ± 0.663	1.581 ± 0.585	0.990 ± 1.256	–	–	0.593 ± 0.263
Gaussian, corr.	**0.005** ± **0.002**	**0.005** ± **0.001**	0.072 ± 0.004	**0.005** ± **0.001**	0.023 ± 0.005	0.060 ± 0.006	0.029 ± 0.002
Gaussian, ill-cond.	**0.005** ± **0.002**	**0.005** ± **0.002**	0.362 ± 0.085	**0.005** ± **0.002**	0.053 ± 0.011	0.373 ± 0.599	0.078 ± 0.008
Gaussian, isotropic	**0.005** ± **0.002**	**0.005** ± **0.001**	0.065 ± 0.002	**0.005** ± **0.001**	0.010 ± 0.001	0.049 ± 0.004	0.025 ± 0.001
Generalized Gaussian	**0.013** ± **0.002**	0.055 ± 0.012	0.090 ± 0.007	0.048 ± 0.009	0.066 ± 0.014	0.123 ± 0.021	0.042 ± 0.008
GMM, separated	**0.014** ± **0.008**	0.180 ± 0.160	0.160 ± 0.119	0.073 ± 0.118	0.049 ± 0.030	0.099 ± 0.050	0.095 ± 0.070
GMM, overlapping	**0.017** ± **0.009**	0.204 ± 0.140	0.115 ± 0.027	0.037 ± 0.030	0.083 ± 0.046	0.135 ± 0.057	0.101 ± 0.048
Heavy-tailed t	**0.014** ± **0.002**	0.195 ± 0.018	0.181 ± 0.031	0.107 ± 0.008	1.091 ± 1.953	–	0.142 ± 0.024
Laplace	**0.016** ± **0.002**	0.077 ± 0.002	0.102 ± 0.003	0.070 ± 0.002	0.092 ± 0.002	0.176 ± 0.010	0.046 ± 0.003
Noisy torus	**0.148** ± **0.065**	0.283 ± 0.082	0.303 ± 0.070	0.210 ± 0.089	0.281 ± 0.077	0.306 ± 0.076	0.285 ± 0.081
Rosenbrock	**0.021** ± **0.002**	0.037 ± 0.006	0.078 ± 0.003	0.036 ± 0.005	0.043 ± 0.014	0.050 ± 0.005	0.052 ± 0.007
Sinh–arcsinh	**0.021** ± **0.002**	0.066 ± 0.013	0.391 ± 0.076	0.053 ± 0.011	0.112 ± 0.041	0.182 ± 0.043	0.136 ± 0.031
Skew normal	**0.006** ± **0.001**	0.011 ± 0.002	0.076 ± 0.002	0.011 ± 0.001	0.024 ± 0.004	0.062 ± 0.006	0.033 ± 0.004
Student-t, $\nu = 2$	**0.031** ± **0.004**	2.692 ± 2.079	4.230 ± 3.493	0.716 ± 0.643	–	0.234 ± 0.165	1.856 ± 1.633
Student-t, $\nu = 5$	**0.011** ± **0.002**	0.100 ± 0.004	0.145 ± 0.008	0.025 ± 0.004	0.049 ± 0.004	0.093 ± 0.011	0.073 ± 0.006
Uniform box	**0.046** ± **0.001**	0.181 ± 0.002	0.224 ± 0.001	0.180 ± 0.003	0.183 ± 0.002	0.203 ± 0.005	0.074 ± 0.001
Warped GMM	**0.023** ± **0.005**	0.275 ± 0.086	0.159 ± 0.022	0.049 ± 0.023	0.114 ± 0.030	0.162 ± 0.026	0.115 ± 0.018

Table 12 (continued).

$d = 16$							
Family	DensityFM	Gaussian	KDE	GMM	MAF	RealNVP	NSF
Banana	**0.011** ± **0.003**	0.013 ± 0.007	0.087 ± 0.004	0.014 ± 0.007	0.016 ± 0.006	0.052 ± 0.004	0.031 ± 0.006
Clayton copula	**0.064** ± **0.009**	0.418 ± 0.046	0.363 ± 0.035	0.235 ± 0.028	–	–	0.125 ± 0.008
Gaussian copula	**0.046** ± **0.013**	0.300 ± 0.055	0.403 ± 0.074	0.184 ± 0.051	0.727 ± 1.353	–	0.144 ± 0.027
t copula	**0.058** ± **0.009**	0.324 ± 0.045	0.467 ± 0.066	0.198 ± 0.036	0.142 ± 0.021	0.209 ± 0.018	0.162 ± 0.021
Dirichlet (ALR)	**0.016** ± **0.001**	0.039 ± 0.004	0.132 ± 0.018	0.040 ± 0.004	0.054 ± 0.006	–	0.049 ± 0.008
Funnel	**0.064** ± **0.014**	1.585 ± 0.587	1.590 ± 0.650	0.956 ± 0.989	6.591 ± 14.424	–	0.554 ± 0.231
Gaussian, corr.	0.006 ± 0.001	**0.005** ± **0.001**	0.091 ± 0.003	**0.005** ± **0.001**	0.027 ± 0.004	0.068 ± 0.006	0.031 ± 0.003
Gaussian, ill-cond.	0.006 ± 0.001	**0.005** ± **0.001**	0.411 ± 0.097	**0.005** ± **0.001**	0.073 ± 0.015	0.163 ± 0.060	0.101 ± 0.013
Gaussian, isotropic	0.006 ± 0.001	**0.005** ± **0.001**	0.081 ± 0.003	**0.005** ± **0.001**	0.009 ± 0.001	0.054 ± 0.003	0.023 ± 0.003
Generalized Gaussian	**0.013** ± **0.002**	0.059 ± 0.010	0.108 ± 0.006	0.060 ± 0.013	0.066 ± 0.012	0.142 ± 0.018	0.040 ± 0.006
GMM, separated	**0.012** ± **0.006**	0.151 ± 0.134	0.202 ± 0.128	0.079 ± 0.136	0.085 ± 0.087	0.116 ± 0.056	0.090 ± 0.063
GMM, overlapping	**0.019** ± **0.009**	0.191 ± 0.141	0.137 ± 0.027	0.031 ± 0.023	0.094 ± 0.048	0.143 ± 0.055	0.100 ± 0.047
Heavy-tailed t	**0.016** ± **0.001**	0.197 ± 0.010	0.177 ± 0.017	0.118 ± 0.013	0.229 ± 0.030	–	0.141 ± 0.018
Laplace	**0.014** ± **0.001**	0.077 ± 0.002	0.118 ± 0.003	0.075 ± 0.001	0.086 ± 0.004	0.182 ± 0.012	0.045 ± 0.003
Noisy torus	**0.137** ± **0.061**	0.214 ± 0.061	0.280 ± 0.056	0.180 ± 0.066	0.233 ± 0.064	0.264 ± 0.059	0.236 ± 0.066
Rosenbrock	**0.030** ± **0.005**	0.042 ± 0.010	0.101 ± 0.005	0.042 ± 0.010	0.046 ± 0.011	0.055 ± 0.007	0.059 ± 0.010
Sinh–arcsinh	**0.027** ± **0.004**	0.063 ± 0.014	0.409 ± 0.049	0.056 ± 0.013	0.109 ± 0.034	0.212 ± 0.083	0.144 ± 0.024
Skew normal	**0.007** ± **0.001**	0.010 ± 0.001	0.093 ± 0.003	0.010 ± 0.001	0.023 ± 0.002	0.068 ± 0.003	0.031 ± 0.001
Student-t, $\nu = 2$	**0.045** ± **0.007**	0.993 ± 0.297	1.064 ± 0.441	0.818 ± 0.630	0.363 ± 0.631	0.116 ± 0.034	0.508 ± 0.239
Student-t, $\nu = 5$	**0.013** ± **0.001**	0.111 ± 0.011	0.159 ± 0.013	0.048 ± 0.038	0.055 ± 0.006	0.111 ± 0.013	0.085 ± 0.009
Uniform box	**0.065** ± **0.001**	0.181 ± 0.001	0.253 ± 0.002	0.181 ± 0.001	0.183 ± 0.001	0.209 ± 0.001	0.078 ± 0.002
Warped GMM	**0.027** ± **0.007**	0.253 ± 0.091	0.185 ± 0.025	0.052 ± 0.026	0.106 ± 0.025	0.200 ± 0.082	0.113 ± 0.027

Table 12 (continued).

$d = 20$							
Family	DensityFM	Gaussian	KDE	GMM	MAF	RealNVP	NSF
Banana	**0.012 ± 0.005**	0.013 ± 0.006	0.101 ± 0.004	0.013 ± 0.006	0.016 ± 0.006	0.058 ± 0.003	0.030 ± 0.006
Clayton copula	**0.080 ± 0.010**	0.402 ± 0.064	0.382 ± 0.047	0.237 ± 0.032	–	–	0.126 ± 0.022
Gaussian copula	**0.069 ± 0.023**	0.305 ± 0.080	0.465 ± 0.065	0.210 ± 0.057	0.201 ± 0.054	0.436 ± 0.317	0.162 ± 0.022
t copula	**0.093 ± 0.015**	0.358 ± 0.064	0.561 ± 0.070	0.225 ± 0.040	–	0.329 ± 0.018	0.190 ± 0.014
Dirichlet (ALR)	**0.020 ± 0.002**	0.043 ± 0.005	0.140 ± 0.009	0.043 ± 0.005	0.069 ± 0.010	0.122 ± 0.014	0.047 ± 0.005
Funnel	**0.084 ± 0.020**	1.632 ± 0.642	1.574 ± 0.709	1.322 ± 0.791	–	–	0.554 ± 0.213
Gaussian, corr.	**0.006 ± 0.001**	**0.006 ± 0.001**	0.108 ± 0.005	**0.006 ± 0.001**	0.032 ± 0.007	0.072 ± 0.006	0.035 ± 0.003
Gaussian, ill-cond.	**0.006 ± 0.001**	**0.006 ± 0.001**	0.441 ± 0.089	**0.006 ± 0.001**	0.094 ± 0.014	0.221 ± 0.047	0.117 ± 0.025
Gaussian, isotropic	**0.006 ± 0.001**	**0.006 ± 0.001**	0.097 ± 0.002	**0.006 ± 0.001**	0.008 ± 0.001	0.056 ± 0.004	0.022 ± 0.002
Generalized Gaussian	**0.014 ± 0.002**	0.060 ± 0.011	0.121 ± 0.007	0.060 ± 0.012	0.065 ± 0.011	0.149 ± 0.019	0.040 ± 0.004
GMM, separated	**0.015 ± 0.008**	0.135 ± 0.118	0.195 ± 0.098	0.148 ± 0.241	0.051 ± 0.026	0.113 ± 0.041	0.109 ± 0.078
GMM, overlapping	**0.023 ± 0.013**	0.186 ± 0.124	0.163 ± 0.037	0.057 ± 0.050	0.094 ± 0.052	0.148 ± 0.055	0.103 ± 0.049
Heavy-tailed t	**0.019 ± 0.002**	0.199 ± 0.015	0.205 ± 0.047	0.156 ± 0.017	0.224 ± 0.030	–	0.143 ± 0.013
Laplace	**0.015 ± 0.001**	0.077 ± 0.003	0.131 ± 0.001	0.077 ± 0.003	0.085 ± 0.004	0.188 ± 0.011	0.046 ± 0.004
Noisy torus	**0.149 ± 0.041**	0.174 ± 0.049	0.272 ± 0.050	0.154 ± 0.053	0.207 ± 0.052	0.243 ± 0.048	0.207 ± 0.054
Rosenbrock	**0.037 ± 0.005**	0.044 ± 0.010	0.118 ± 0.007	0.045 ± 0.010	0.050 ± 0.012	0.059 ± 0.003	0.063 ± 0.012
Sinh–arcsinh	**0.035 ± 0.005**	0.068 ± 0.011	0.504 ± 0.098	0.061 ± 0.008	0.216 ± 0.073	0.304 ± 0.070	0.216 ± 0.054
Skew normal	**0.009 ± 0.001**	0.012 ± 0.001	0.108 ± 0.003	0.014 ± 0.003	0.027 ± 0.001	0.071 ± 0.006	0.034 ± 0.002
Student-t, $\nu = 2$	**0.076 ± 0.009**	0.876 ± 0.236	0.973 ± 0.367	0.565 ± 0.486	0.389 ± 0.709	0.124 ± 0.012	0.357 ± 0.147
Student-t, $\nu = 5$	**0.017 ± 0.002**	0.115 ± 0.006	0.156 ± 0.009	0.036 ± 0.004	0.057 ± 0.011	0.111 ± 0.009	0.082 ± 0.008
Uniform box	0.103 ± 0.009	0.183 ± 0.001	0.277 ± 0.003	0.183 ± 0.001	0.183 ± 0.001	0.214 ± 0.003	**0.078 ± 0.002**
Warped GMM	**0.034 ± 0.009**	0.236 ± 0.063	0.222 ± 0.022	0.050 ± 0.017	0.114 ± 0.023	0.173 ± 0.034	0.113 ± 0.016

Table 12 (continued).

$d = 32$							
Family	DensityFM	Gaussian	KDE	GMM	MAF	RealNVP	NSF
Banana	**0.011** ± **0.005**	0.016 ± 0.005	0.135 ± 0.005	0.016 ± 0.005	0.015 ± 0.005	0.064 ± 0.002	0.029 ± 0.005
Clayton copula	0.189 ± 0.050	0.423 ± 0.075	0.447 ± 0.060	0.295 ± 0.032	–	–	**0.126** ± **0.019**
Gaussian copula	**0.082** ± **0.030**	0.352 ± 0.084	0.604 ± 0.055	0.288 ± 0.063	1.961 ± 3.716	–	0.234 ± 0.028
t copula	**0.116** ± **0.020**	0.404 ± 0.062	0.689 ± 0.052	0.281 ± 0.055	0.311 ± 0.053	–	0.265 ± 0.022
Dirichlet (ALR)	**0.044** ± **0.009**	0.045 ± 0.003	0.179 ± 0.012	0.045 ± 0.003	0.071 ± 0.005	0.141 ± 0.010	0.056 ± 0.008
Funnel	2.576 ± 2.619	1.782 ± 0.723	1.587 ± 0.551	1.936 ± 0.796	–	–	**0.587** ± **0.245**
Gaussian, corr.	0.012 ± 0.001	**0.010** ± **0.001**	0.145 ± 0.006	**0.010** ± **0.001**	0.039 ± 0.008	0.083 ± 0.005	0.041 ± 0.004
Gaussian, ill-cond.	0.022 ± 0.004	**0.010** ± **0.001**	0.542 ± 0.087	**0.010** ± **0.001**	0.166 ± 0.017	0.294 ± 0.055	0.175 ± 0.028
Gaussian, isotropic	**0.005** ± **0.001**	0.010 ± 0.001	0.128 ± 0.001	0.010 ± 0.001	0.008 ± 0.001	0.062 ± 0.003	0.023 ± 0.002
Generalized Gaussian	**0.020** ± **0.005**	0.061 ± 0.008	0.151 ± 0.006	0.061 ± 0.008	0.062 ± 0.009	0.153 ± 0.013	0.039 ± 0.003
GMM, separated	**0.041** ± **0.028**	0.100 ± 0.083	0.252 ± 0.125	0.255 ± 0.308	0.057 ± 0.027	0.120 ± 0.040	0.106 ± 0.070
GMM, overlapping	0.060 ± 0.032	0.142 ± 0.094	0.194 ± 0.036	**0.047** ± **0.031**	0.090 ± 0.043	0.155 ± 0.051	0.102 ± 0.044
Heavy-tailed t	0.412 ± 0.063	0.204 ± 0.014	0.220 ± 0.012	0.186 ± 0.008	0.225 ± 0.020	–	**0.144** ± **0.008**
Laplace	**0.024** ± **0.008**	0.081 ± 0.002	0.163 ± 0.002	0.081 ± 0.002	0.084 ± 0.003	0.187 ± 0.009	0.044 ± 0.003
Noisy torus	**0.085** ± **0.020**	0.114 ± 0.031	0.263 ± 0.031	0.116 ± 0.030	0.161 ± 0.035	0.203 ± 0.035	0.158 ± 0.036
Rosenbrock	**0.037** ± **0.010**	0.042 ± 0.010	0.149 ± 0.008	0.042 ± 0.010	0.042 ± 0.011	0.064 ± 0.003	0.056 ± 0.011
Sinh–arcsinh	0.101 ± 0.012	**0.067** ± **0.007**	0.519 ± 0.047	**0.067** ± **0.007**	0.228 ± 0.018	0.333 ± 0.027	0.232 ± 0.020
Skew normal	**0.012** ± **0.001**	0.013 ± 0.001	0.142 ± 0.003	0.013 ± 0.001	0.027 ± 0.002	0.076 ± 0.003	0.036 ± 0.001
Student-t, $\nu = 2$	1.824 ± 2.454	1.071 ± 0.393	0.896 ± 0.238	0.962 ± 0.238	**0.086** ± **0.023**	0.132 ± 0.025	0.454 ± 0.225
Student-t, $\nu = 5$	0.160 ± 0.178	0.156 ± 0.027	0.199 ± 0.031	0.065 ± 0.016	**0.058** ± **0.007**	0.147 ± 0.036	0.113 ± 0.026
Uniform box	**0.064** ± **0.001**	0.185 ± 0.002	0.320 ± 0.003	0.185 ± 0.002	0.182 ± 0.001	0.216 ± 0.002	0.078 ± 0.003
Warped GMM	**0.070** ± **0.010**	0.203 ± 0.047	0.263 ± 0.016	0.074 ± 0.031	0.116 ± 0.024	0.187 ± 0.032	0.112 ± 0.017

Table 12 (continued).

$d = 64$							
Family	DensityFM	Gaussian	KDE	GMM	MAF	RealNVP	NSF
Banana	**0.011** ± **0.004**	0.025 ± 0.003	0.186 ± 0.004	0.025 ± 0.003	0.014 ± 0.004	0.072 ± 0.002	0.033 ± 0.004
Clayton copula	0.186 ± 0.024	0.418 ± 0.035	0.507 ± 0.030	0.347 ± 0.027	–	–	**0.124** ± **0.013**
Gaussian copula	**0.130** ± **0.036**	0.490 ± 0.121	0.965 ± 0.137	0.477 ± 0.112	0.953 ± 0.532	–	0.465 ± 0.093
t copula	**0.259** ± **0.106**	0.586 ± 0.128	1.034 ± 0.142	0.506 ± 0.094	–	–	0.502 ± 0.098
Dirichlet (ALR)	**0.047** ± **0.015**	0.054 ± 0.002	0.240 ± 0.028	0.054 ± 0.002	0.068 ± 0.003	0.154 ± 0.009	0.055 ± 0.010
Funnel	2.849 ± 3.090	2.384 ± 1.352	1.929 ± 1.217	2.491 ± 1.432	–	–	**0.595** ± **0.287**
Gaussian, corr.	**0.018** ± **0.001**	0.019 ± 0.001	0.205 ± 0.004	0.019 ± 0.001	0.047 ± 0.005	0.099 ± 0.003	0.051 ± 0.003
Gaussian, ill-cond.	0.029 ± 0.003	**0.019** ± **0.001**	0.664 ± 0.112	**0.019** ± **0.001**	0.260 ± 0.045	0.434 ± 0.081	0.281 ± 0.044
Gaussian, isotropic	**0.004** ± **0.000**	0.019 ± 0.001	0.182 ± 0.002	0.019 ± 0.001	0.007 ± 0.000	0.070 ± 0.002	0.026 ± 0.001
Generalized Gaussian	**0.019** ± **0.003**	0.068 ± 0.003	0.201 ± 0.005	0.068 ± 0.003	0.059 ± 0.002	0.163 ± 0.007	0.043 ± 0.002
GMM, separated	**0.052** ± **0.033**	0.072 ± 0.050	0.326 ± 0.145	0.054 ± 0.036	0.056 ± 0.023	0.116 ± 0.030	0.111 ± 0.077
GMM, overlapping	0.084 ± 0.042	0.109 ± 0.064	0.250 ± 0.034	**0.075** ± **0.045**	0.091 ± 0.039	0.155 ± 0.049	0.108 ± 0.041
Heavy-tailed t	0.404 ± 0.062	0.216 ± 0.009	0.274 ± 0.005	0.216 ± 0.007	0.220 ± 0.012	–	**0.145** ± **0.008**
Laplace	**0.023** ± **0.004**	0.091 ± 0.001	0.219 ± 0.004	0.091 ± 0.001	0.082 ± 0.002	0.203 ± 0.008	0.047 ± 0.003
Noisy torus	**0.061** ± **0.012**	0.071 ± 0.016	0.280 ± 0.018	0.071 ± 0.016	0.131 ± 0.019	0.171 ± 0.020	0.124 ± 0.018
Rosenbrock	**0.049** ± **0.010**	0.062 ± 0.009	0.214 ± 0.006	0.062 ± 0.009	0.052 ± 0.011	0.081 ± 0.005	0.071 ± 0.010
Sinh–arcsinh	0.134 ± 0.024	0.086 ± 0.016	0.588 ± 0.031	**0.085** ± **0.016**	0.326 ± 0.024	0.427 ± 0.024	0.332 ± 0.021
Skew normal	**0.017** ± **0.000**	0.020 ± 0.001	0.194 ± 0.003	0.020 ± 0.001	0.027 ± 0.001	0.083 ± 0.000	0.040 ± 0.001
Student-t, $\nu = 2$	1.103 ± 0.669	1.092 ± 0.271	0.853 ± 0.169	1.179 ± 0.388	**0.069** ± **0.005**	–	0.380 ± 0.103
Student-t, $\nu = 5$	0.082 ± 0.035	0.161 ± 0.016	0.197 ± 0.015	0.159 ± 0.016	**0.058** ± **0.003**	0.177 ± 0.026	0.109 ± 0.014
Uniform box	**0.064** ± **0.001**	0.194 ± 0.001	0.378 ± 0.001	0.194 ± 0.001	0.181 ± 0.001	0.221 ± 0.002	0.085 ± 0.002
Warped GMM	**0.091** ± **0.019**	0.176 ± 0.044	0.335 ± 0.021	0.124 ± 0.039	0.106 ± 0.019	0.177 ± 0.026	0.113 ± 0.020

Table 12 (continued).

$d = 50$							
Family	DensityFM	Gaussian	KDE	GMM	MAF	RealNVP	NSF
Banana	0.160 ± 0.046	0.021 ± 0.004	0.167 ± 0.004	0.021 ± 0.004	**0.011 ± 0.004**	0.027 ± 0.002	0.032 ± 0.004
Clayton copula	0.227 ± 0.018	0.420 ± 0.040	0.488 ± 0.033	0.334 ± 0.028	> 10	> 10	**0.130 ± 0.012**
Gaussian copula	0.361 ± 0.066	0.431 ± 0.076	0.852 ± 0.080	0.399 ± 0.067	> 10	0.725 ± 0.090	**0.356 ± 0.052**
t copula	**0.329 ± 0.055**	0.506 ± 0.058	0.915 ± 0.075	0.386 ± 0.052	> 10	> 10	0.391 ± 0.041
Dirichlet (ALR)	**0.039 ± 0.003**	0.051 ± 0.002	0.223 ± 0.018	0.051 ± 0.002	0.067 ± 0.004	0.140 ± 0.025	0.060 ± 0.015
Funnel	**0.198 ± 0.071**	2.030 ± 1.011	1.823 ± 0.910	2.139 ± 0.962	> 10	> 10	0.551 ± 0.192
Gaussian, corr.	**0.011 ± 0.001**	0.014 ± 0.001	0.184 ± 0.003	0.014 ± 0.001	0.043 ± 0.003	0.056 ± 0.003	0.048 ± 0.001
Gaussian, ill-cond.	0.025 ± 0.007	**0.014 ± 0.001**	0.628 ± 0.104	**0.014 ± 0.001**	0.217 ± 0.038	0.383 ± 0.079	0.241 ± 0.052
Gaussian, isotropic	**0.002 ± 0.001**	0.014 ± 0.001	0.163 ± 0.003	0.014 ± 0.001	0.004 ± 0.001	0.025 ± 0.001	0.026 ± 0.002
Generalized Gaussian	0.051 ± 0.006	0.068 ± 0.006	0.185 ± 0.004	0.068 ± 0.006	0.061 ± 0.006	0.089 ± 0.007	**0.043 ± 0.003**
GMM, separated	**0.023 ± 0.011**	0.080 ± 0.059	0.311 ± 0.135	0.154 ± 0.236	0.060 ± 0.026	0.102 ± 0.049	0.113 ± 0.074
GMM, overlapping	**0.029 ± 0.013**	0.117 ± 0.073	0.236 ± 0.037	0.071 ± 0.047	0.091 ± 0.042	0.147 ± 0.068	0.104 ± 0.043
Heavy-tailed t	**0.120 ± 0.002**	0.213 ± 0.009	0.275 ± 0.040	0.208 ± 0.012	0.206 ± 0.009	0.249 ± 0.016	0.145 ± 0.011
Laplace	0.075 ± 0.001	0.086 ± 0.002	0.200 ± 0.003	0.086 ± 0.002	0.082 ± 0.002	0.114 ± 0.004	**0.047 ± 0.003**
Noisy torus	**0.061 ± 0.014**	0.081 ± 0.020	0.270 ± 0.020	0.081 ± 0.020	0.134 ± 0.024	0.155 ± 0.030	0.132 ± 0.024
Rosenbrock	0.051 ± 0.008	0.057 ± 0.008	0.193 ± 0.009	0.057 ± 0.008	0.049 ± 0.008	**0.044 ± 0.004**	0.071 ± 0.008
Sinh–arcsinh	**0.072 ± 0.007**	0.079 ± 0.015	0.558 ± 0.032	0.079 ± 0.014	0.278 ± 0.030	0.381 ± 0.035	0.286 ± 0.028
Skew normal	**0.011 ± 0.001**	0.017 ± 0.001	0.176 ± 0.002	0.017 ± 0.001	0.026 ± 0.002	0.042 ± 0.001	0.039 ± 0.001
Student-t, $\nu = 2$	**0.027 ± 0.002**	1.186 ± 0.513	1.187 ± 0.800	1.286 ± 0.698	0.481 ± 0.547	> 10	0.343 ± 0.156
Student-t, $\nu = 5$	**0.013 ± 0.001**	0.152 ± 0.013	0.212 ± 0.051	0.097 ± 0.031	0.061 ± 0.001	0.124 ± 0.008	0.105 ± 0.007
Uniform box	0.189 ± 0.002	0.190 ± 0.002	0.359 ± 0.002	0.190 ± 0.002	0.180 ± 0.002	0.196 ± 0.002	**0.083 ± 0.003**
Warped GMM	**0.082 ± 0.015**	0.180 ± 0.043	0.311 ± 0.021	0.108 ± 0.039	0.113 ± 0.022	0.199 ± 0.040	0.116 ± 0.021