Denoising Lévy Probabilisitc Models - DLPM

Heavy-Tailed Diffusion Models

Umut Simsekli

joint work with Dario Shariatian, Alain Durmus

March 19, 2026

Generative AI for Extreme Events Workshop

Introduction on Diffusion Models

DDPM – Overview

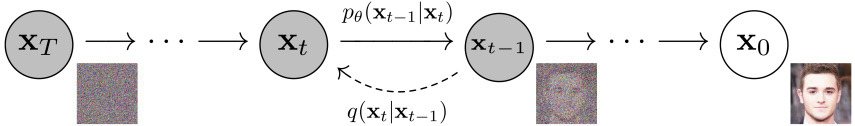


Figure 1: Forward/backward structure, discrete time [HJA20]

Setup, discrete time:

- **Forward process** $\{X_t\}_{t=0}^T$ is a Markov chain with Gaussian transition kernels $p_{t+1|t}(\cdot|\cdot)$, such that

$$X_0 \sim p_0 \text{ (the data)}, \quad X_T \sim p_T \approx \mathcal{N}(0, \mathrm{I}_d) \text{ (the noise)} \tag{1}$$

- **Generative process** $\{\overleftarrow{X}_t^\theta\}_{t=0}^T$ will be a Markov chain running in reverse time ; we want p_t and p_t^θ to match
- **Training loss** Fit the joint distributions with an ELBO loss, like in VAEs.

DDPM – Forward Process

- **Forward process (Markov chain):**

$$X_{t+1} = \sqrt{\alpha_t} X_t + \sqrt{1-\alpha_t}\epsilon_t \,, \tag{2}$$

where $\{\alpha_t\}_{t=0}^{T-1}$ is a noise schedule, $0 < \alpha_t < 1$.

- **Closed form for $X_t \mid X_0$**, by stability of the Gaussian distribution:

$$X_t \mid X_0 \overset{d}{=} \sqrt{\overline{\alpha}_t}\, X_0 + \sqrt{(1-\overline{\alpha}_t)\mathrm{I}_d}\overline{\epsilon}_t \,, \tag{3}$$

with $\overline{\alpha}_t = \prod_{s=1}^{t} \alpha_s$, chosen such that X_T is approximately distributed as $\mathcal{N}(0, \mathrm{I}_d)$.

DDPM – Backward Process

- **Reformulating the forward process** Let us examine its joint distribution

$$
\begin{aligned}
p(x_0, \cdots, x_T) &= p_0(x_0) \cdot \prod_{t=1}^{T} p_{t|t-1}(x_t|x_{t-1}) \\
&= p_0(x_0) \cdot p_{1|0}(x_1|x_0) \cdot \prod_{t=2}^{T} p_{t|t-1}(x_t|x_{t-1}, x_0) \\
&= p_0(x_0) \cdot p_{1|0}(x_1|x_0) \cdot \prod_{t=2}^{T} \frac{p_{t-1|t,0}(x_{t-1}|x_t, x_0) p_{t|0}(x_t|x_0)}{p_{t-1|0}(x_{t-1}|x_0)} \quad \text{, by Bayes rule} \\
&= \underbrace{p_0(x_0)}_{\text{data}} \cdot \underbrace{p_{T|0}(x_T|x_0)}_{\text{noise}} \cdot \prod_{t=2}^{T} \underbrace{p_{t-1|t,0}(x_{t-1}|x_t, x_0)}_{\text{Gaussian transitions}}
\end{aligned}
$$

- **Gaussian transitions** $p_{t-1|t,0}(\cdot|x_t, x_0)$ is the density of $\mathcal{N}(\tilde{\mathbb{m}}_t(x_t, x_0), \tilde{\Sigma}_t)$.

DDPM – Generative Process

- **Backward process**

$$p(x_0, \cdots, x_t) = \underbrace{p_0(x_0)}_{\text{data}} \cdot \underbrace{p_{T|0}(x_T|x_0)}_{\text{noise}} \cdot \prod_{t=2}^{T} \underbrace{p_{t-1|t,0}(x_{t-1}|x_t, x_0)}_{\text{Gaussian transitions}}, \tag{4}$$

where $p_{t-1|t,0}(\cdot|x_t, x_0) = \mathcal{N}(\cdot\ ;\ \tilde{\mathrm{m}}_t(x_t, x_0), \tilde{\Sigma}_t)$.

- **Generative process** This suggests using the following structure for the generative model

$$p^\theta(x_0, \cdots, x_t) = \underbrace{p_T^\theta(x_T)}_{\text{noise}} \cdot \prod_{t=1}^{T} \underbrace{p_{t-1|t}^\theta(x_{t-1}|x_t)}_{\text{Gaussian transitions}}, \tag{5}$$

with $p_{t-1|t}^\theta(\cdot|x_t) = \mathcal{N}(\cdot\ ;\ \hat{\mathrm{m}}_t^\theta(x_t), \tilde{\Sigma}_t)$.

DDPM – Training Objective

- **Variational bound (ELBO)** We want to fit p^θ to p:

$$\begin{aligned}
\log p_\theta(x_0) &= \log\left(\int p^\theta(X_{0:T})dX_{1:T}\right) \\
&\geqslant \log\left(\mathbb{E}_{p(X_{1:T}|x_0)}\frac{p^\theta(X_{0:T})}{p(X_{1:T}|X_0)}\right) \\
&\geqslant \mathbb{E}_{p(X_{1:T})}\log\left(\frac{p^\theta(X_{0:T})}{p(X_{1:T}|X_0)}\right) \quad \text{By Jensen's ineq.} \\
&= -\mathcal{L}_{\mathrm{ELBO}}(\theta)
\end{aligned}$$

Rearranging terms, we obtain

$$\mathcal{L}_{\mathrm{ELBO}}(\theta) = \mathbb{E}\left[\underbrace{\mathrm{KL}(p_{T|0}(\cdot|x_0) \parallel p_T^\theta(\cdot))}_{L_T} + \sum_{t=2}^{T}\underbrace{\mathrm{KL}(p_{t-1|t,0}(\cdot|x_t,x_0) \parallel p_{t-1|t}^\theta(\cdot|x_t))}_{L_{t-1}} \underbrace{-\log p_{0|1}^\theta(x_0|x_1)}_{L_0}\right].$$

The terms L_T, L_0 are typically neglected.

DDPM – Training Objective

- **ELBO loss**

$$\mathcal{L}(\theta) = \mathbb{E}\left[\frac{\|\tilde{\mathfrak{m}}_t(X_t, X_0) - \hat{\mathfrak{m}}_t^\theta(X_t)\|^2}{2\tilde{\Sigma}_t}\right], \tag{6}$$

- **Denoiser reparameterization** Instead of predicting $\tilde{\mathfrak{m}}_t(x_t, x_0)$ from x_t, we can predict x_0 or the noise $\bar{\epsilon}$. Typically, instead of the true ELBO:

$$\mathcal{L}_{\text{simple}}(\theta) = \mathbb{E}_{X_0, t, \bar{\epsilon}_t, X_t = \bar{\alpha}_t X_0 + \sqrt{1-\bar{\alpha}_t}\bar{\epsilon}_t}\left[\|\bar{\epsilon}_t - \hat{\epsilon}_t^\theta(X_t)\|^2\right]. \tag{7}$$

  Thus, the model learns $\mathbb{E}[\bar{\epsilon}_t | X_t]$, or $\mathbb{E}[X_0 | X_t]$...

Advantages

- High quality samples
- Stable/easy training (e.g., contrary to GANs)
- Equivalence between multiple approaches (continuous time with SDEs, flow matching etc.)

Disadvantages

- Lots of diffusion steps $T \gg 1$
- Mode collapse with high class imbalance
- What if initial data distribution is heavy tailed (no variance)?

Figure 2: ASML stock, 1Y

Proposal – Change Noise Distribution

- Previous work:
  - Generalized Gaussian distributions ([DSL21])
  - Gamma distributions ([NRW21])
  - Lévy α-stable distribution ([Yoo+23])
- Limitations:
  - No true time reversal, heuristics for sampling
  - Crude upper bound or unstable training
  - Hyper-parameters to tune

Proposal – α-stable Heavy-tailed Distribution

- We advocate for the α-**stable Lévy distributions**, which generalize Gaussian with heavy tails.
- Better coverage of heavy-tailed data distribution
- Improvements on mode collapse, especially in the context of class imbalance
- Less function evaluation, as large jumps seem to benefit the exploration of the data space

Lévy-Itô Models (LIM): a Continuous-Time Version

Lévy-Itô Models (LIM) vs DLPM

- **LIM** Levy-Ito Models are the continuous time version of α-stable generative models. They extend the SDE formulation to Levy processes. They must resort to complicated tools in fractional stochastic calculus.
- **LIM vs DLPM**
  - DLPM admits an accessible theory.
  - DLPM admits more flexibility; choose any noise schedule (γ_t, σ_t), possibility to learn variance...
  - Both approaches yield different training losses and sampling procedures

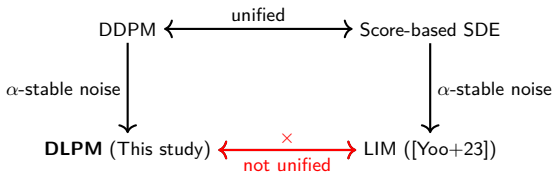
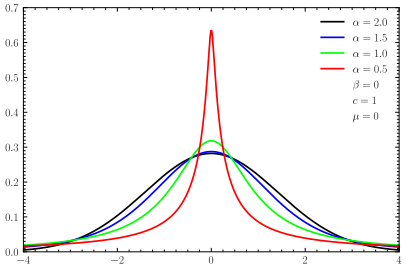


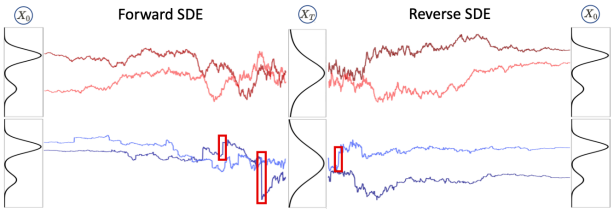
Figure 3: Illustration of available methods.

α-stable Lévy distributions

α-stable Lévy distributions



(a) Symmetric α-Stable distribution, varying α [Wik24]

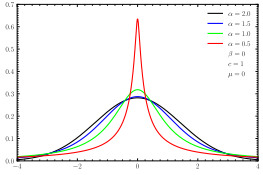


(b) Lévy Process vs Brownian Motion ($\alpha = 2$) [Yoo+23]

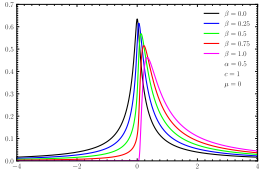
Definition and properties

The α-stable distributions $\mathcal{S}_{\alpha,\beta}(\mu,\sigma)$ are characterized by four parameters $(\alpha,\beta,\mu,\sigma)$:

- $\alpha \in (0,2)$, the tail heaviness parameter
- $\beta \in (-1,1)$, the skewness parameter
- μ, the location parameter
- σ, the scale parameter



(a) $\beta = 0, \mu = 0, \sigma = 1$, varying α [Wik24]



(b) $\alpha = 0.5, \mu = 0, \sigma = 1$, varying β [Wik24]

Notable special cases

- ($\alpha = 2$) In this case, $\mathcal{S}_\alpha(0, \sigma)$ is the Gaussian distribution $\mathcal{N}(0, 2\sigma^2)$

- ($\alpha = 1$) In this case, $\mathcal{S}_\alpha(0, 1)$ is the Cauchy distribution Cauchy$(0, 1)$

Stability

- This family of distributions is **stable** by addition, i.e.,

$$X_{\mathcal{S}_{\alpha,\beta_0}(\mu_0,\sigma_0)} + X_{\mathcal{S}_{\alpha,\beta_1}(\mu_1,\sigma_1)} \sim X_{\mathcal{S}_{\alpha,\beta}(\mu,\sigma)}$$

  where

$$\sigma^\alpha = \sigma_0^\alpha + \sigma_1^\alpha \ , \quad \beta = \frac{\beta_0\sigma_0^\alpha + \beta_1\sigma_1^\alpha}{\sigma^\alpha} \ , \quad \mu = \mu_0 + \mu_1$$

- **Gaussian case ($\alpha = 2, \beta = 0$)**:

$$\sigma^2 = \sigma_0^2 + \sigma_1^2 \ , \quad \mu = \mu_0 + \mu_1$$

Gaussian Trick

Gaussian Trick

Let $A \sim \mathcal{S}_{\alpha/2,1}(0, c_A)$, and $G \sim \mathcal{N}(0,1)$, where $c_A := \cos^{2/\alpha}(\pi\alpha/4)$. Then

$$A^{1/2} G \sim \mathcal{S}_\alpha(0,1)\ . \tag{8}$$

- **Isotropic noise.** $A^{1/2} \cdot G$.
- **Non-isotropic (independent) noise.** With $\mathrm{A} = \{A_i\}_{i=1}^d$ i.i.d.: $\mathrm{A}^{1/2} \odot G$.

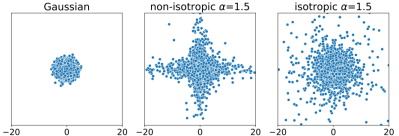


Figure 6: Different multidimensional heavy-tailed noise distributions, Gaussian vs $\alpha = 1.5$ [Yoo+23]

DLPM: Heavy-Tailed Denoising Diffusion

Forward Process - first approach

- **Forward process (Markov chain)** Consider $\{X_t\}_{t=0}^T$ defined by:

$$X_0 \sim p_0 \ , \qquad X_t = \gamma_t X_{t-1} + \sigma_t \epsilon_t^{(\alpha)}, \tag{9}$$

where $\epsilon_t^{(\alpha)} \sim \mathcal{S}_\alpha^{\mathrm{i}}(0, \mathrm{I}_d)$ i.i.d.., and $\{(\gamma_t, \sigma_t)\}_{t=1}^T$ is the noising schedule.

- **Closed form for $X_t|X_0$**

$$X_t \overset{d}{=} \gamma_{1\to t} X_0 + \sigma_{1\to t} {\epsilon^{(\alpha)}}_t \ , \tag{10}$$

where ${\epsilon^{(\alpha)}}_t \sim \mathcal{S}_\alpha^{\mathrm{i}}(0, \mathrm{I}_d)$.

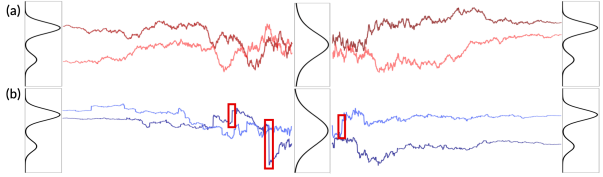


Figure 7: (a) Gaussian transitions (b) Heavy-tailed transitions

Backward Process - first approach

- We want a similar structure for the generative process:

$$p^{\theta}_{0:T}(x_{0:T}) = p^{\theta}_{T}(x_T) \prod_{t=T}^{1} p^{\theta}_{t-1|t}(x_{t-1}|x_t) \ , \tag{11}$$

- **Problem** No known techniques to characterize

$$p_{t-1|t}(x_{t-1}|x_t) \ , \quad p_{t-1|t,0}(x_{t-1}|x_t, x_0) \ . \tag{12}$$

  Moreover, the KL between α-stable distributions is unavailable when $\alpha \neq 2, 1$.
- **How to design the backward process, the generative process, and the training procedure?**
- **Our approach** Data augmentation and the Gaussian trick

Forward Process - Data Augmentation approach

- **Data augmentation approach.** Define $\{Y_t\}_{t=0}^T$ by:

$$Y_0 \sim p_0, \qquad Y_t = \gamma_t Y_{t-1} + \sigma_t A_t^{1/2} G_t \,, \tag{13}$$

where $\{A_t\}_{t=1}^T \sim \mathcal{S}_{\alpha/2,1}(0, c_A)^{\otimes T}$ and $\{G_t\}_{t=1}^T \sim \mathcal{N}(0, \mathrm{I}_d)^{\otimes T}$. This process satisfies

$$Y_t \overset{d}{=} X_t \,. \tag{14}$$

- **Closed form for** $Y_t \mid Y_0, A_{1:t}$

$$Y_t \mid Y_0, A_{1:t} \overset{d}{=} \gamma_{1\to t} Y_0 + \Sigma_{1\to t}(A_{1:t})^{1/2} \bar{G}_t, \tag{15}$$

where $\bar{G}_t \sim \mathcal{N}(0, \mathrm{I}_d)$.

Backward Process – Data Augmentation Approach

- **Conditioning on** $\{A_t\}_{t=1}^T$ The joint distribution admits the decomposition

$$p(x_0, \cdots, x_T, a_{1:T}) = p_0(x_0) \cdot \prod_{t=1}^{T} p_{t|t-1}(x_t|x_{t-1}, a_{1:T}) \psi_{(\alpha)}^{\otimes T}(a_{1:T})$$

$$= \underbrace{p_0(x_0)}_{\text{data}} \cdot \underbrace{p_{T|0}(x_T|x_0, a_{1:T})}_{\text{noise}} \cdot \prod_{t=2}^{T} \underbrace{p_{t-1|t,0}(x_{t-1}|x_t, x_0, a_{1:T})}_{\text{Gaussian transitions}} \psi_{(\alpha)}^{\otimes T}(a_{1:T}) \,,$$

  where $\psi_{(\alpha)}$ is the density of $\mathcal{S}_{\alpha/2,1}(0, c_A)$.
- **Gaussian transitions** $p_{t-1|t,0,a_{1:T}}(\cdot|x_t, x_0, a_{1:T})$ is the density of $\mathcal{N}(\tilde{m}_t(x_t, x_0, a_{1:t}), \tilde{\Sigma}_t(a_{1:t}))$.

Backward process - model

Generative process

$$p^{\theta}(x_0, \cdots, x_t, a_{1:T}) = \underbrace{p^{\theta}_T(x_T)}_{\text{noise}} \cdot \prod_{t=1}^{T} \underbrace{p^{\theta}_{t-1|t,a}(x_{t-1}|x_t, a_{1:t})}_{\text{Gaussian transitions}} \psi^{\otimes T}_{(\alpha)}(a_{1:T}) \ , \tag{16}$$

where $\psi_{(\alpha)}$ is the density of the $\mathcal{S}_{\alpha/2,1}(0, c_A)$ distribution, and

$$p^{\theta}_{t-1|t,a}(\cdot|x_t, a_{1:t}) = \mathcal{N}(\cdot \ ; \ \hat{\mathfrak{m}}^{\theta}_t(x_t, a_{1:t}), \tilde{\Sigma}_t(a_{1:t})) \ . \tag{17}$$

Loss function - alpha-stable case

Reminder: ELBO loss, Gaussian case

$$\mathcal{L}(\theta) = \mathbb{E}\left[\frac{\|\tilde{\mathfrak{m}}_t(X_t, X_0) - \hat{\mathfrak{m}}_t^\theta(X_t)\|^2}{2\tilde{\Sigma}_t}\right], \tag{18}$$

- **A naive solution:** by Jensen's inequality:

$$\mathrm{KL}(p_0 \| p_0^\theta) \leqslant \mathbb{E}\left(\mathrm{KL}\left[p_0(\cdot)\|p_{0|a}^\theta(\cdot|A_{1:T})\right]\right). \tag{19}$$

- As we see in (18), this expression would involve taking expectation of A_t

- However, A_t is distributed as $\mathcal{S}_{\alpha/2,1}(0, c_A)$, and does not admit a first order moment.

Loss function - alpha-stable case

- **Loss function** We directly minimize the following KL divergence:

$$\mathcal{L}(\theta) := \mathbb{E}_{A_{1:T}}\left[\mathrm{KL}(p_{0|a}(\cdot|A_{1:T})\|p^{\theta}_{0|a}(\cdot|A_{1:T}))^{1/2}\right] . \tag{20}$$

To obtain our loss, we employ the usual derivations:

$$\begin{aligned}\mathcal{L}(\theta) &\leqslant \mathbb{E}_{A_{1:T}}\left[L_T(\theta, A_{1:T}) + \sum_{t\geqslant 2} L_{t-1}(\theta, A_{1:t}) + L_0(\theta, A_1)\right]^{1/2} \qquad \text{(ELBO)} ,\\ &\leqslant \mathbb{E}_{A_{1:T}}\left[L_T(\theta, A_{1:T})^{1/2} + \sum_{t\geqslant 2} L_{t-1}(\theta, A_{1:t})^{1/2} + L_0(\theta, A_1)^{1/2}\right] \qquad (\sqrt{a+b} < \sqrt{a} + \sqrt{b}) .\end{aligned} \tag{21}$$

Again, we neglect L_T, L_0, and:

$$\mathscr{L}^{\mathrm{L}}(\theta) = \mathbb{E}_{A_{1:T}}\left[\mathbb{E}_{Y_{0:T}}\left[\frac{1}{2\hat{\Sigma}^{\theta}_{t-1}(A_{1:T})}\|\tilde{\mathrm{m}}_{t-1}(Y_t, Y_0, A_{1:t}) - \hat{\mathrm{m}}^{\theta}_{t-1}(Y_t, A_{1:t})\|^2 \;\middle|\; A_1\right]^{1/2}\right] . \tag{22}$$

Further Design Choices

Further Design Choices

- We can also do x_0, ϵ prediction.

- Each loss term L_t involves $A_{1:t}$ terms; we derive a smart reparameterization of L_t involving a single A_t.

Experiments

Experiments – Setup

- The loss function

$$\mathscr{L}^{\text{SimpleLess}}(\theta) = \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{E}\left(\|\hat{\epsilon}_t^{\theta}(Y_t) - \epsilon_t(Y_t, Y_0)\|^2 \mid \bar{A}_t\right)^{1/2}\right] \tag{23}$$

  involves an expectation with respect to A_t. We propose the *median-of-means* estimator ([LM19]), denoted by DLPM$_5$ ($M = 5$).

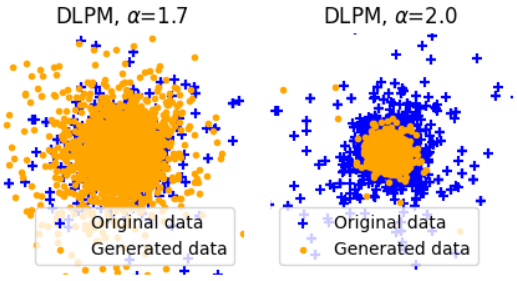
- We consider the range $1.5 \leqslant \alpha \leqslant 2.0$

- (Low-dim) We work on a synthetic mixture of Gaussian/α-Stable distributions

- (High-dim) We work on CIFAR10_LT (long tail), unbalanced modification of the CIFAR10 ([Yoo+23]).
  - Class count: $[5000, 2997, 1796, 1077, 645, 387, 232, 139, 83, 50]$.

2D data - covering the dataset and capturing heavy-tails

- Challenge: cover the dataset and correctly capture the tails.
- **Dataset** 20000 samples of $\mathcal{S}_\alpha^{\mathrm{i}}\left(0, 0.05 \cdot \mathrm{I}_2\right)$, with $\alpha = 1.7$.



(a) DLPM with $\alpha = 1.7$

(b) DLPM with $\alpha = 2.0$ (DDPM)

- The lighter tailed process fails to capture the distribution's tail.

2D data - covering the dataset and capturing heavy-tails

- Drawing inspiration from [AGG22], we define the MSLE:

$$\mathrm{MSLE}(\xi) = \int_{\xi}^{1} \left(\log \hat{F}^{-1}(p) - \log \hat{F^{\theta}}^{-1}(p) \right)^2 dp \,, \tag{24}$$

where $\hat{F}, \hat{F}^{\theta}$ denote respectively the cdf of the true data and the generated data.

Method	$\alpha = 1.5$	$\alpha = 1.6$	$\alpha = 1.7$	$\alpha = 1.8$	$\alpha = 1.9$	$\alpha = 2.0$
DLPM	**0.160** $\pm$ 0.128	**0.081** $\pm$ 0.078	**0.071** $\pm$ 0.028	**0.099** $\pm$ 0.044	**0.132** $\pm$ 0.101	0.798 $\pm$ 0.601
DDPM	-	-	-	-	-	0.528 $\pm$ 0.400
						1.0e-1
LIM	0.743 $\pm$ 0.290	0.497 $\pm$ 0.311	0.267 $\pm$ 0.077	0.653 $\pm$ 0.413	2.444 $\pm$ 1.067	1.239 $\pm$ 0.240
	1.0e-08	*8.6e-06*	*1.3e-10*	*8.8e-06*	*7.9e-09*	*5.0e-3*

Table 1: $\mathrm{MSLE}_{\xi=0.95} \downarrow$ averaged over 20 runs. Figures below scores corresponds to p-values from Welch's t-test (assuming unequal variances), comparing the mean of DLPM with the given method.

2D data - managing class imbalance

- Challenge: correctly approximate the mixture weights. F_1^{pr} score of precision and recall metrics
- **Dataset** Mixture of nine Gaussian distributions arranged in a grid

$$\sum_{i=1}^{9} w_i \mathcal{N}(\mu_i, 0.05^2 \cdot \mathrm{I}_2) \ . \tag{25}$$

Mixture weights range from .01 to .3: $\{.01, .02, .02, .05, .05, .1, .1, .15, .2, .3\}$.

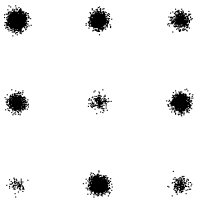


Figure 9: Gaussian grid

Method	$\alpha = 1.5$	$\alpha = 1.6$	$\alpha = 1.7$	$\alpha = 1.8$	$\alpha = 1.9$	$\alpha = 2.0$
DLPM	0.933 ± 0.018	0.923 ± 0.005	0.933 ± 0.028	0.923 ± 0.024	0.907 ± 0.034	0.862 ± 0.028
DLPM_5	**0.944 ± 0.013**	**0.943 ± 0.021**	**0.943 ± 0.010**	**0.941 ± 0.014**	**0.928 ± 0.016**	-
	9.0e-3	*1.6e-05*	*7.4e-2*	*9.0e-4*	*3.9e-3*	
LIM	0.842 ± 0.039	0.850 ± 0.046	0.868 ± 0.034	0.874 ± 0.030	0.884 ± 0.017	0.874 ± 0.027
	1.7e-14	*1.3e-09*	*5.7e-11*	*3.9e-09*	*1.9e-3*	*9.6e-2*
DDPM	-	-	-	-	-	0.867 ± 0.029
						5.0e-1

Table 2: $F_1^{\text{pr}} \uparrow$ score, averaged over 30 runs. Figures below scores corresponds to p-values from Welch's t-test (assuming unequal variances), comparing the mean of DLPM with the given method.

2D data - faster convergence

- Challenge: get to the data distribution with the smallest T possible
- DLIM vs LIM-ODE with varying total diffusion steps T, on the Gaussian grid.

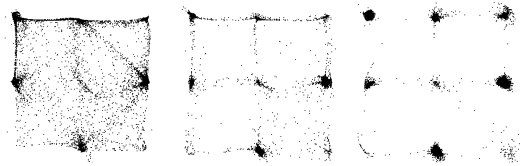


Figure 10: DLIM with $T = 5, 10, 25$ diffusion steps on the Gaussian grid

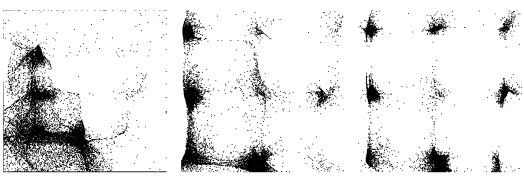


Figure 11: LIM-ODE with $T = 5, 10, 25$ diffusion steps on the Gaussian grid

Image data - LIM vs DLPM

- **Dataset** MNIST and CIFAR10_LT.
- Convergence speed for the different methods, varying total number of diffusion steps T.

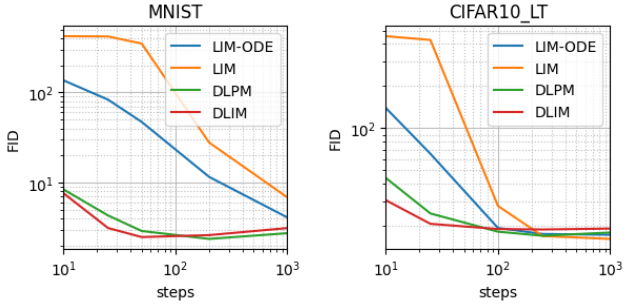


Figure 12: FID↓ with varying step size, $\alpha = 1.7$

Image data - LIM vs DLPM vs DDPM

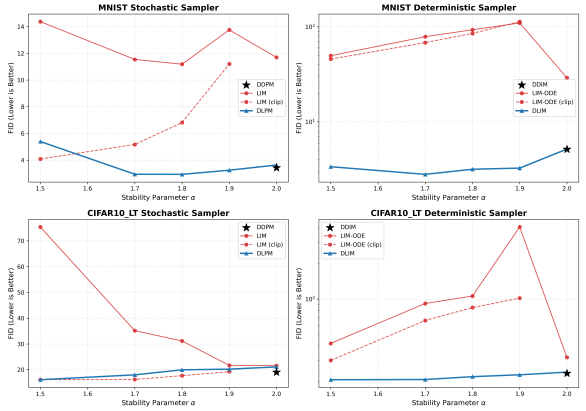


Figure 13: FID↓ on MNIST, CIFAR10_LT for different methods with the stochastic sampler (1000 steps) and the deterministic sampler (25 steps)

DLPM beats LIM, and smaller α induce better performance

Reference I

[LM19] Gábor Lugosi and Shahar Mendelson. "Mean estimation and regression under heavy-tailed distributions: A survey". In: *Foundations of Computational Mathematics* 19.5 (2019), pp. 1145–1190.

[HJA20] Jonathan Ho, Ajay Jain, and Pieter Abbeel. *Denoising Diffusion Probabilistic Models*. 2020. arXiv: 2006.11239 [cs.LG].

[SME20] Jiaming Song, Chenlin Meng, and Stefano Ermon. "Denoising Diffusion Implicit Models". In: *CoRR* abs/2010.02502 (2020). arXiv: 2010.02502. URL: https://arxiv.org/abs/2010.02502.

[DSL21] Jacob Deasy, Nikola Simidjievski, and Pietro Lio'. "Heavy-tailed denoising score matching". In: *ArXiv* abs/2112.09788 (2021). URL: https://api.semanticscholar.org/CorpusID:245334465.

[NRW21] Eliya Nachmani, Robin San Roman, and Lior Wolf. *Denoising Diffusion Gamma Models*. 2021. arXiv: 2110.05948 [eess.SP].

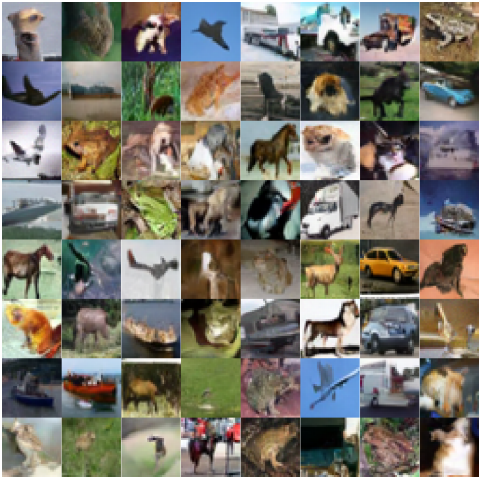
[AGG22] Michaël Allouche, Stéphane Girard, and Emmanuel Gobet. "EV-GAN: Simulation of extreme events with ReLU neural networks". In: *Journal of Machine Learning Research* 23.150 (2022), pp. 1–39. URL: https://hal.science/hal-03250663.

Reference II

[Yoo+23] Eun Bi Yoon et al. "Score-based Generative Models with Lévy Processes". In: *Advances in Neural Information Processing Systems*. Ed. by A. Oh et al. Vol. 36. Curran Associates, Inc., 2023, pp. 40694–40707. URL: https://proceedings.neurips.cc/paper_files/paper/2023/file/8011b23e1dc3f57e1b6211ccad498919-Paper-Conference.pdf.

[Wik24] Wikipedia contributors. *Stable distribution — Wikipedia, The Free Encyclopedia*. [Online; accessed 2-July-2024]. 2024. URL: https://en.wikipedia.org/w/index.php?title=Stable_distribution&oldid=1227672574.

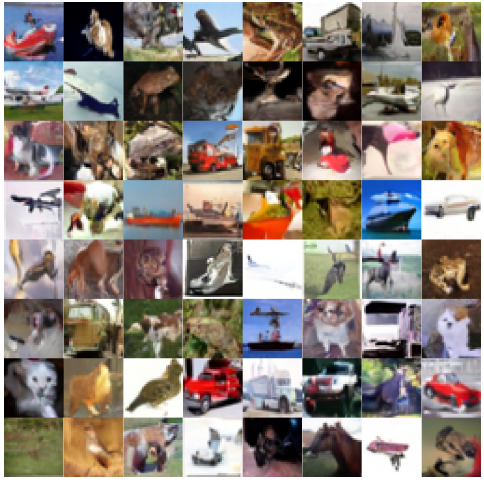
Some images - DLPM



(a) CIFAR10, $T = 4000$

(b) MNIST, $T = 1000$

Some images - DLIM



(a) CIFAR10, $T = 200$

(b) MNIST, $T = 50$

Loss function - design choices **D1**

- **D1 (Fixed variance)** We set $\hat{\Sigma}_t^\theta = \tilde{\Sigma}_t$.

Loss function - design choice **D2**

- **D2 (Denoiser Reparameterization)** We predict the *injected noise* $\epsilon_t(y_t, y_0)$ rather than $\tilde{\mathrm{m}}_{t-1}(y_t, y_0, a_{1:t})$. Since

$$\tilde{\mathrm{m}}_{t-1}(Y_t, Y_0, A_{1:t}) = \frac{1}{\gamma_t}\left(Y_t - \sigma_{1\to t}\Gamma_t(A_{1:t})\epsilon_t(Y_t, Y_0)\right) , \tag{26}$$

  we re-parameterize $\hat{\mathrm{m}}^\theta_{t-1}$ as

$$\hat{\mathrm{m}}^\theta_{t-1}(Y_t, A_{1:t}) = \frac{1}{\gamma_t}\left(Y_t - \sigma_{1\to t}\Gamma_t(A_{1:t})\hat{\epsilon}^\theta_t(Y_t)\right) . \tag{27}$$

  with $\hat{\epsilon}^\theta_t$ the output of the model.
- The model $\hat{\epsilon}^\theta_t$ does not take any heavy-tailed $A_{1:t}$ as input.
- Assuming **D1**, the loss $\mathscr{L}^{\mathrm{L}}$ becomes

$$\mathscr{L}^{\mathrm{L}}(\theta) = \mathbb{E}\left[\lambda^2_{t,A_{1:t}}\|\hat{\epsilon}^\theta_t(Y_t) - \epsilon_t(Y_t, Y_0)\|^2\right] , \tag{28}$$

$$\lambda_{t,a_{1:t}} = \frac{\Gamma_t(a_{1:t})\sigma_{1\to t}}{2\gamma_t\tilde{\Sigma}_{t-1}} , \quad \epsilon_t(Y_t, Y_0) = \frac{(Y_t - \gamma_{1\to t}Y_0)}{\sigma_{1\to t}} . \tag{29}$$

Loss function - design choice **D3**

- **D3 (Simple loss)** With design choices **D1**, **D2**, the loss $\mathscr{L}^{\mathrm{L}}$ is

$$\mathscr{L}^{\mathrm{L}}(\theta) = \mathbb{E}\left[\lambda^2_{t,A_{1:t}} \|\hat{\epsilon}^{\theta}_t(Y_t) - \epsilon_t(Y_t, Y_0)\|^2\right] . \tag{30}$$

We choose to set $\lambda_{t,a_{1:t}} = 1$, which improves performance, and draws similarities to the continuous α-stable score-based perspective.

We obtain a simplified denoising objective function

$$\boxed{\mathscr{L}^{\mathrm{Simple}}(\theta) = \mathbb{E}\left[\mathbb{E}\left(\|\hat{\epsilon}^{\theta}_t(Y_t) - \epsilon_t(Y_t, Y_0)\|^2 \mid A_{1:t}\right)^{1/2}\right] .} \tag{31}$$

Bonus - faster sampling

Assume design choices **D1, D2, D3** are satisfied. Then one can obtain the following simplified denoising objective function:

$$\mathscr{L}_{t-1}^{\text{SimpleLess}}(\theta) = \mathbb{E}\left[\mathbb{E}\left(\|\hat{\epsilon}_t^\theta(Y_t^{\text{Less}}) - \epsilon_t(Y_t^{\text{Less}}, Y_0^{\text{Less}})\|^2 \mid \bar{A}_t\right)\right]^{1/2}, \qquad t \in \{2, \cdots, T\} \tag{32}$$

where

$$Y_t^{\text{Less}} = \gamma_{1\to t} Y_0^{\text{Less}} + \sigma_{1\to t} \bar{A}_t^{1/2} G_t\,, \qquad \epsilon_t(Y_t^{\text{Less}}, Y_0^{\text{Less}}) = \frac{Y_t^{\text{Less}} - \gamma_{1\to t} Y_0^{\text{Less}}}{\sigma_{1\to t}}\,. \tag{33}$$

with $G_t \sim \mathcal{N}(0, \mathrm{I}_d)$, $\bar{A}_t \sim \mathcal{S}_{\alpha/2,1}(0, c_A)$.

- Idea: marginalization/sufficient statistic, as

$$Y_t \overset{d}{=} \gamma_{1\to t} Y_0 + \Sigma_{1\to t}(A_{1:t})^{1/2} \bar{G}_t \overset{d}{=} \gamma_{1\to t} Y_0 + \sigma_{1\to t} \epsilon_t^{(\alpha)} \overset{d}{=} \gamma_{1\to t} Y_0 + \sigma_{1\to t} \bar{A}_t^{1/2} G_t \tag{34}$$

- Cheaper than sampling a list $A_{1:t}$ for each datapoint.
- The final denoising loss is similar to LIM (continuous α-stable case), but guaranteed to be finite.

LIM - forward

- **Forward process** The forward process $\{X_t\}_{0 \leqslant t \leqslant T}$, with $X_0 \sim p_0$, is obtained with

$$\mathrm{d}X_t = \gamma(t, X_{t-})\mathrm{d}t + \sigma(t)\mathrm{d}L_t^\alpha, \tag{35}$$

where X_{t-} denotes the left limit of X at time t. LIM only defines scale-preserving schedule:

$$\gamma(t, x) = -\frac{\beta_t}{\alpha}x, \quad \sigma(t) = \beta_t^{1/\alpha}. \tag{36}$$

- **Closed-form expession of $X_t|X_0$**

$$X_t \overset{d}{=} \gamma_{1\to t}X_0 + \sigma_{1\to t}\bar{\epsilon}, \tag{37}$$

where $\bar{\epsilon}_t \sim \mathcal{S}_\alpha^{\mathrm{i}}(0, \mathrm{I}_d)$. The values of $\gamma_{1\to t}$ and $\sigma_{1\to t}$ match with the DLPM definition on integer timesteps.

LIM - backward

- **Backward process** The following backward process $\bar{X}_t$ is obtained:

$$\mathrm{d}\bar{X}_t = \left(-\gamma(t, \bar{X}_{t+}) + \alpha\sigma^{\alpha}(t, \bar{X}_{t+})s_t(\bar{X}_{t+})\right) dt + \sigma(t)\mathrm{d}\bar{L^{\alpha}}_t + d\bar{Z}_t \tag{38}$$

where

- $\bar{Z}_t$ is the backward version of a Levy-type stochastic integral Z_t s.t $\mathbb{E}[Z_t] = 0$ with finite variation
- s_t is the fractional score function:

$$s_t(x) = \frac{\Delta^{\frac{\alpha-2}{2}} \nabla p_t(x)}{p_t(x)} \ , \tag{39}$$

where $\Delta^{\eta/2}$ is the fractional Laplacian of order $\eta/2$, defined with Fourier transform $\mathcal{F}$:

$$\Delta^{\eta/2} f(x) = \mathcal{F}^{-1}\{\|u\|^{\eta} \mathcal{F}\{f\}(u)\} \ . \tag{40}$$

LIM - training

- The true score $s_t(x_t|x_0)$ can be expressed as

$$s_t(x_t|x_0) = -\frac{1}{\alpha\sigma_{1\to t}^{\alpha-1}(t)}\epsilon_t(x_t, x_0), \tag{41}$$

where $\epsilon_t(x_t, x_0) = \frac{x_t - \gamma_{1\to t}x_0}{\sigma_{1\to t}}$, thus we re-parametrize

$$s_\theta(x_t, t) = -\frac{1}{\alpha\sigma_{1\to t}^{\alpha-1}(t)}\hat{\epsilon}_t^\theta(x_t, x_0), \tag{42}$$

so that we rather work with $\hat{\epsilon}_t^\theta$.

- Training loss obtained using denoising score matching technique:

$$L : \theta \mapsto \mathbb{E}\|s_\theta(X_t, t) - s_t(X_t)\|^2, \qquad L' : \theta \mapsto \mathbb{E}\|s_\theta(X_t, t) - s_t(X_t|X_0)\|^2, \tag{43}$$

are equivalent objective functions, with s_θ the score approximation given by the model.

LIM vs DLPM - forward/backward

With $\{G'_t\}_{t=T}^1$ i.i.d. $\mathcal{N}(0, \mathrm{I}_d)$, $\{\epsilon'_t\}_{t=T}^1$ i.i.d. $\mathcal{S}_\alpha^{\mathrm{i}}(0, \mathrm{I}_d)$, and $\hat{\epsilon}_t^\theta$ the model at time t:

	Stochastic	Deterministic
Continuous (LIM)	$\frac{\bar{X}_t^\theta}{\gamma_t} - \frac{\alpha(1/\gamma_t - 1)}{\sigma_{1\to t}^{\alpha-1}}\hat{\epsilon}_t^\theta + (\frac{1}{\gamma_t^\alpha} - 1)^{1/\alpha}\epsilon'_t$	$\frac{\bar{X}_t^\theta}{\gamma_t} - \left(\frac{\sigma_{1\to t}^{1-\alpha}}{\gamma_t} - \sigma_{1\to t}^{1-\alpha}\right)\hat{\epsilon}_t^\theta$
Denoising (DLPM)	$\frac{\bar{Y}_t^\theta}{\gamma_t} - \Gamma_t\sigma_{1\to t}\hat{\epsilon}_t^\theta + \Gamma_t\Sigma_{1\to t-1}G'_t$	$\frac{\bar{Y}_t^\theta}{\gamma_t} - \left(\frac{\sigma_{1\to t}}{\gamma_t} - \sigma_{1\to t-1}\right)\hat{\epsilon}_t^\theta$

- **Stochastic sampling** Different sampling procedures. Moreover:
  - When $\alpha = 2$, $0 \leqslant \Gamma_t \leqslant 1$ becomes deterministic, and one recovers DDPM formulas
  - Γ_t brings additional stochasticity
  - Γ_t scales (i) the noise added at time $t-1$ (ii) the output of the noise model.
- **Deterministic sampling** Different sampling procedures.

LIM vs DLPM - training

- Alike the Gaussian case ($\alpha = 2$), the score $s_t(x_t|x_0)$ is a linear expression of the noise term:

$$s_t(x_t|x_0) = -\frac{1}{\alpha \sigma_{1\to t}^{\alpha-1}(t)} \epsilon_t(x_t, x_0) \ , \tag{44}$$

  leading to a similar denoising loss:

$$\mathcal{L}_{t-1} : \theta \mapsto \mathbb{E}\left(\|\hat{\epsilon}_t^\theta(X_t) - \epsilon_t(X_t, X_0)\|_p^\eta \right) . \tag{45}$$

- DLPM: use $p = 2$ and $\eta = 1$.
- LIM (theory): use $p = 2$ and $\eta = 2$, for denoising score matching loss equivalence. But $\epsilon_t(X_t, X_0)$ is heavy-tailed: no variance!
- LIM (experiments): use $p = 1$ and $\eta = 1$. Indicates potential shortcoming of the theoretical approach.

Denoising Lévy Implicit Models: Deterministic Generation

Deterministic Generation

- **Directly define the bridges** We did not really need the forward process to be Markovian, but we leveraged the following decomposition:

$$p(x_0, \cdots, x_T, a_{1:T}) = \underbrace{p_0(x_0)}_{\text{data}} \cdot \underbrace{p_{T|0}(x_T|x_0, a_{1:T})}_{\text{noise}} \cdot \prod_{t=2}^{T} \underbrace{p_{t-1|t,0}(x_{t-1}|x_t, x_0, a_{1:T})}_{\text{Gaussian transitions}} \psi_{(\alpha)}^{\otimes T}(a_{1:T}) \,. \tag{46}$$

- **Non-necessarily Markovian process** Sample endpoints first

$$Z_0 \sim p_0 \,, \quad Z_T|Z_0 \sim \mathcal{S}_\alpha\left(\gamma_{1\to T} Z_0, \sigma_{1\to T} \mathrm{I}_d\right) \,, \tag{47}$$

and then the bridges

$$Z_{t-1} = \gamma_{1\to t-1} Z_0 + (\sigma_{1\to t-1}^\alpha - \rho_t^\alpha)^{1/\alpha} \cdot \underbrace{\frac{Z_t - \gamma_{1\to t} Z_0}{\sigma_{1\to t}}}_{\equiv \text{injected noise term } \epsilon_t(Z_t, Z_0)} + \underbrace{\rho_t A_t^{1/2} G_t}_{\text{stochasticity}} \,, \tag{48}$$

with $G_t \sim \mathcal{N}(0, \mathrm{I}_d)$, $A_t \sim \mathcal{S}_{\alpha/2,1}(0, c_A)$ i.i.d..

Deterministic Generation – Gaussian case (DDIM)

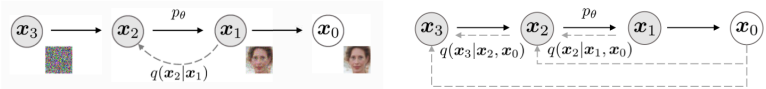


Figure 16: Non-Markovian forward process [SME20]

Distribution of $Z_t|Z_0$ Same as DLPM. Informal proof:

$$Z_{t-1} = \gamma_{1\to t-1} Z_0 + (\sigma^\alpha_{1\to t-1} - \rho_t^\alpha)^{1/\alpha} \cdot \underbrace{\frac{Z_t - \gamma_{1\to t} Z_0}{\sigma_{1\to t}}}_{\equiv \text{injected noise term } \epsilon_t(Z_t, Z_0)} + \underbrace{\rho_t A_t^{1/2} G_t}_{\text{stochasticity}}$$

$$\overset{d}{=} \gamma_{1\to t-1} Z_0 + (\sigma^\alpha_{1\to t-1} - \rho_t^\alpha)^{1/\alpha} \cdot \underbrace{\overset{0}{\epsilon_t}}_{\equiv \text{injected noise term } \epsilon_t(Z_t, Z_0)} + \underbrace{\rho_t \overset{1}{\epsilon_t}}_{\text{stochasticity}}$$

$$\overset{d}{=} \gamma_{1\to t-1} Z_0 + \sigma_{1\to t-1} \epsilon_t \quad \text{(Stability of } \alpha\text{-stable)}$$

DLIM - Denoising Lévy Implicit Models

- **Recovers DLPM loss** Models $\hat{\epsilon}_t^\theta(Z_t)$ trained for DLPM can be reused

- **Possibly better loss** Since $p_{t-1|t,0}$ is α-stable, we can bypass data-augmentation if closed-form KL exists between $\mathcal{S}(\mu_1, \sigma_1)$ and $\mathcal{S}(\mu_2, \sigma_2)$. It is the case for Cauchy ($\alpha = 1$):

$$\mathrm{KL}\left[\text{Cauchy}(\mu_1, \sigma) \;\middle|\; \text{Cauchy}(\mu_2, \sigma)\right] = \log\left(1 + \frac{(\mu_1 - \mu_2)^2}{4}\right) . \tag{49}$$

- **Deterministic generation** Deterministic sampling process with $\rho_t = 0$.