On some modern developments in generative modelling

ISA Oberseminar

Lukas Trottner

18 November 2025

University of Stuttgart

University of Stuttgart
Germany

Motivation:

> *"Creating noise from data is easy; creating data from noise is generative modeling."*

Source: Song et al. (2021). Score based generative modeling through stochastic differential equations. *ICLR*.

Generative modelling

Setup: identically distributed samples $X_1, \dots, X_n$ with unknown distribution P are given

Goal: develop sampling algorithms that do not rely on structural assumptions on P

$\rightsquigarrow$ involves (implicitly) learning the underlying distribution of a dataset to generate new samples that

a) follow approximately the same distribution as the training data;
b) should not be drawn from the training data set

$\rightsquigarrow$ essential in applications like image synthesis, text generation, data augmentation ...

Noise transformation

Inverse transform sampling: for an $\mathbb{R}$-valued random variable X with cdf F and $U \sim \mathcal{U}((0,1))$, we have $F^{-1}(U) \stackrel{d}{=} X$ for the left-inverse F^{-1} of F

- we don't know F, but are only given samples $X_1, \ldots, X_n \stackrel{d}{=} X$
- naïve approach: replace F by empirical cdf $\widehat{F}(x) = \frac{\#\{X_i : X_i \leq x\}}{n}$ and set $\widehat{X} = \widehat{F}^{-1}(U)$ for an independent $U \sim \mathcal{U}((0,1))$
- if $X_{(1)}, \ldots, X_{(n)}$ is an increasing ordering of the data set and $U \in [k/n, (k+1)/n)$, then $\widehat{X} = X_{(k)}$

$\rightsquigarrow$ algorithm learns the empirical distribution $\mathbb{P}_n = \frac{1}{n} \sum_{i=1}^{n} \delta_{X_i}$ $\rightsquigarrow$ overfitting/"no creativity"

Noise transformation

Inverse transform sampling: for an $\mathbb{R}$-valued random variable X with cdf F and $U \sim \mathcal{U}((0,1))$, we have $F^{-1}(U) \overset{d}{=} X$ for the left-inverse F^{-1} of F

- we don't know F, but are only given samples $X_1, \dots, X_n \overset{d}{=} X$
- naïve approach: replace F by empirical cdf $\widehat{F}(x) = \frac{\#\{X_i : X_i \leq x\}}{n}$ and set $\widehat{X} = \widehat{F}^{-1}(U)$ for an independent $U \sim \mathcal{U}((0,1))$
- if $X_{(1)}, \dots, X_{(n)}$ is an increasing ordering of the data set and $U \in [k/n, (k+1)/n)$, then $\widehat{X} = X_{(k)}$

$\rightsquigarrow$ algorithm learns the empirical distribution $\mathbb{P}_n = \frac{1}{n} \sum_{i=1}^n \delta_{X_i}$ $\rightsquigarrow$ overfitting/“no creativity”

To evaluate the performance of an algorithm that outputs $\widehat{X} = \widehat{T}(U)$, for some $\widehat{T} \in \sigma(X_1, \dots, X_n)$ and independent noise U we can

- analyse the rate of convergence of

$$\mathbb{E}\Big[d(\widehat{T}_\sharp \mathbb{P}_U, \mathbb{P}_X)\Big], \qquad d \text{ some probability distance or divergence}$$

- study distance of generated distribution to empirical distribution $\mathbb{P}_n$
- inspect samples visually

Generative Adversial Networks (GANs)

Key idea: GANs use a game-theoretic setup:

- Generator G: maps random, easy to-sample-from noise U to data space to generate samples
- Discriminator D: distinguishes between real samples X and generated samples $G(U)$

Objective: A minimax game is played:

$$\inf_{G \in \mathcal{G}} \sup_{D \in \mathcal{D}} |\mathbb{E}[D(X)] - \mathbb{E}[D(G(U))]|$$

Generative Adversial Networks (GANs)

Key idea: GANs use a game-theoretic setup:

- Generator G: maps random, easy to-sample-from noise U to data space to generate samples
- Discriminator D: distinguishes between real samples X and generated samples $G(U)$

Objective: A minimax game is played:

$$\inf_{G\in\mathcal{G}} \sup_{D\in\mathcal{D}} |\mathbb{E}[D(X)] - \mathbb{E}[D(G(U))]|$$

- for $\mathcal{D} = \{D : D \text{ is 1-Lipschitz}\}$ this is equivalent to

$$\inf_{G\in\mathcal{G}} \mathrm{W}_1\big(\mathbb{P}_X, G_\sharp\mathbb{P}_U\big) = \inf_{G\in\mathcal{G}} \inf_{\gamma\in\Gamma(\mathbb{P}_X, G_\sharp\mathbb{P}_U)} \int \|x-y\|\,\gamma(\mathrm{d}x,\mathrm{d}y), \qquad \text{(Wasserstein-GAN)}$$

where $\gamma \in \Gamma(\mathbb{P}_X, G_\sharp\mathbb{P}_U)$ iff $\gamma(\mathrm{d}x \times \mathbb{R}^d) = \mathbb{P}_X(\mathrm{d}x)$ and $\gamma(\mathbb{R}^d \times \mathrm{d}y) = \mathbb{P}(G(U) \in \mathrm{d}y)$

Generative Adversial Networks (GANs)

Key idea: GANs use a game-theoretic setup:

- Generator G: maps random, easy to-sample-from noise U to data space to generate samples
- Discriminator D: distinguishes between real samples X and generated samples $G(U)$

Objective: A minimax game is played:

$$\inf_{G\in\mathcal{G}} \sup_{D\in\mathcal{D}} |\mathbb{E}[D(X)] - \mathbb{E}[D(G(U))]|$$

- for $\mathcal{D} = \{D : D \text{ is 1-Lipschitz}\}$ this is equivalent to

$$\inf_{G\in\mathcal{G}} \mathrm{W}_1\big(\mathbb{P}_X, G_\sharp\mathbb{P}_U\big) = \inf_{G\in\mathcal{G}} \inf_{\gamma\in\Gamma(\mathbb{P}_X, G_\sharp\mathbb{P}_U)} \int \|x-y\|\,\gamma(\mathrm{d}x,\mathrm{d}y), \qquad \text{(Wasserstein-GAN)}$$

  where $\gamma\in\Gamma(\mathbb{P}_X, G_\sharp\mathbb{P}_U)$ iff $\gamma(\mathrm{d}x\times\mathbb{R}^d) = \mathbb{P}_X(\mathrm{d}x)$ and $\gamma(\mathbb{R}^d\times\mathrm{d}y) = \mathbb{P}(G(U)\in\mathrm{d}y)$
- in practice, $\mathcal{D}$ and $\mathcal{G}$ are chosen as parameterised neural network classes and expectations are replaced by empirical means:

$$\text{ERM:}\quad \widehat{G} \in \arg\min_{G\in\mathcal{G}} \underbrace{\sup_{D\in\mathcal{D}} |\mathbb{E}_n D - \mathbb{E}[D(G(U))]|}_{\approx \mathrm{W}_1(\mathbb{P}_n, G_\sharp\mathbb{P}_U)}$$

Langevin diffusion models

Langevin MCMC algorithm: given target density p_0 simulate diffusion

$$\mathrm{d}Z_t = \nabla \log p_0(Z_t)\,\mathrm{d}t + \sqrt{2}\,\mathrm{d}W_t$$

and output Z_T for T "sufficiently large": if p_0 is sufficiently nice, then $Z_t \xrightarrow{d} p_0$

Langevin diffusion models

Langevin MCMC algorithm: given target density p_0 simulate diffusion

$$\mathrm{d}Z_t = \nabla \log p_0(Z_t)\,\mathrm{d}t + \sqrt{2}\,\mathrm{d}W_t$$

and output Z_T for T "sufficiently large": if p_0 is sufficiently nice, then $Z_t \xrightarrow{d} p_0$

- bad idea: estimate p_0 by some $\hat{p}$ and replace $\nabla \log p_0$ by $\nabla \log \hat{p}$ (explicit approach)

Langevin diffusion models

Langevin MCMC algorithm: given target density p_0 simulate diffusion

$$\mathrm{d}Z_t = \nabla \log p_0(Z_t)\,\mathrm{d}t + \sqrt{2}\,\mathrm{d}W_t$$

and output Z_T for T “sufficiently large”: if p_0 is sufficiently nice, then $Z_t \xrightarrow{d} p_0$

- bad idea: estimate p_0 by some $\hat{p}$ and replace $\nabla \log p_0$ by $\nabla \log \hat{p}$ (explicit approach)
- implicit approach: consider $p_{0,\sigma} = p_0 * \phi_{0,\sigma^2}$ for a Gaussian density ϕ_{0,σ^2} (σ^2 small) instead and target score $\nabla \log p_{0,\sigma^2}$ directly

Langevin diffusion models

Langevin MCMC algorithm: given target density p_0 simulate diffusion

$$\mathrm{d}Z_t = \nabla \log p_0(Z_t)\,\mathrm{d}t + \sqrt{2}\,\mathrm{d}W_t$$

and output Z_T for T "sufficiently large": if p_0 is sufficiently nice, then $Z_t \xrightarrow{d} p_0$

- bad idea: estimate p_0 by some $\hat{p}$ and replace $\nabla \log p_0$ by $\nabla \log \hat{p}$ (explicit approach)
- implicit approach: consider $p_{0,\sigma} = p_0 * \phi_{0,\sigma^2}$ for a Gaussian density ϕ_{0,σ^2} (σ^2 small) instead and target score $\nabla \log p_{0,\sigma^2}$ directly
- denoising score matching: let $X \sim p_0$, $X_\sigma^2 = X + \sigma\varepsilon \sim p_0 * \phi_{0,\sigma^2}$, for indep. noise $\varepsilon \sim \mathcal{N}(0, \mathbb{I}_d)$

$$\nabla \log p_{0,\sigma^2}(x) = \frac{\int \nabla_x \phi_{0,\sigma^2}(x-y)\,p_0(\mathrm{d}y)}{p_{0,\sigma^2}(x)} = \int \nabla_x \log \varphi_{0,\sigma^2}(x-y)\,\underbrace{\frac{\phi_{0,\sigma^2}(x-y)\,p_0(\mathrm{d}y)}{p_{0,\sigma^2}(x)}}_{=\mathbb{P}(X\in\mathrm{d}y|X_{\sigma^2}=x)}$$

$$= \mathbb{E}\big[\nabla \log \phi_{X,\sigma^2}(X_{\sigma^2}) \mid X_{\sigma^2} = x\big]$$

Langevin diffusion models

Langevin MCMC algorithm: given target density p_0 simulate diffusion

$$\mathrm{d}Z_t = \nabla \log p_0(Z_t)\,\mathrm{d}t + \sqrt{2}\,\mathrm{d}W_t$$

and output Z_T for T "sufficiently large": if p_0 is sufficiently nice, then $Z_t \xrightarrow{d} p_0$

- bad idea: estimate p_0 by some $\hat{p}$ and replace $\nabla \log p_0$ by $\nabla \log \hat{p}$ (explicit approach)
- implicit approach: consider $p_{0,\sigma} = p_0 * \phi_{0,\sigma^2}$ for a Gaussian density ϕ_{0,σ^2} (σ^2 small) instead and target score $\nabla \log p_{0,\sigma^2}$ directly
- denoising score matching: let $X \sim p_0$, $X_\sigma^2 = X + \sigma\varepsilon \sim p_0 * \phi_{0,\sigma^2}$, for indep. noise $\varepsilon \sim \mathcal{N}(0, \mathbb{I}_d)$

$$\nabla \log p_{0,\sigma^2}(x) = \frac{\int \nabla_x \phi_{0,\sigma^2}(x-y)\,p_0(\mathrm{d}y)}{p_{0,\sigma^2}(x)} = \int \nabla_x \log \varphi_{0,\sigma^2}(x-y) \underbrace{\frac{\phi_{0,\sigma^2}(x-y)\,p_0(\mathrm{d}y)}{p_{0,\sigma^2}(x)}}_{=\mathbb{P}(X \in \mathrm{d}y \mid X_{\sigma^2}=x)}$$

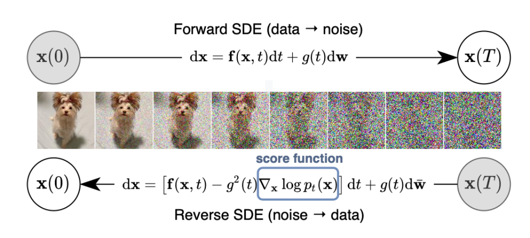
$$= \mathbb{E}\big[\nabla \log \phi_{X,\sigma^2}(X_{\sigma^2}) \mid X_{\sigma^2} = x\big]$$

ERM: $$\hat{s} \in \arg\min_{s \in \mathcal{S}} \frac{1}{M} \sum_{j=1}^{M} \|s(X_{i_j,\sigma^2}) - \nabla \log \phi_{X_{i_j},\sigma^2}(X_{i_j,\sigma^2})\|^2 = \arg\min_{s \in \mathcal{S}} \frac{1}{M} \sum_{j=1}^{M} \|s(X_{i_j,\sigma^2}) + \frac{1}{\sigma}\varepsilon_{i_j}\|^2,$$

where the X_{i_j} are uniformly sampled from $X_1, \dots, X_n$ and $X_{i_j,\sigma^2} = X_{i_j} + \sigma\varepsilon_{i_j}$

Denoising diffusion models

- provide an iterative generative algorithm to create new samples that approximately match the target distribution p_0, given a finite number of samples corresponding to an unknown p_0
- general idea: find a stochastic process that perturbs p_0 to a new distribution p_T such that
  1) p_T or a good approximation thereof is easy to sample from, and
  2) the perturbation is reversible in the sense that we know how to simulate the time-reversed process



Source: Song et al. (2021). Score based generative modeling through stochastic differential equations. *ICLR*.

Denoising Diffusion Models

- for some fixed time $T > 0$ consider the forward model

$$\mathrm{d}X_t = b(t, X_t)\,\mathrm{d}t + \sigma(t, X_t)\,\mathrm{d}W_t, \quad t \in [0, T], X_0 \sim p_0$$

Denoising Diffusion Models

- for some fixed time $T > 0$ consider the forward model

$$\mathrm{d}X_t = b(t, X_t)\,\mathrm{d}t + \sigma(t, X_t)\,\mathrm{d}W_t, \quad t \in [0, T], X_0 \sim p_0$$

- letting $p_t(x) = \int p_{0,t}(y, x)\, p_0(\mathrm{d}y)$ be the marginal densities of (X_t), the time reversal $\overleftarrow{X}_t = X_{T-t}$ solves

$$\mathrm{d}\overleftarrow{X}_t = -\overline{b}(T - t, \overleftarrow{X}_t)\,\mathrm{d}t + \sigma(T - t, \overleftarrow{X}_t)\,\mathrm{d}\overline{W}_t, \quad t \in [0, T], \overleftarrow{X}_0 \sim p_T,$$

where

$$\begin{aligned}\overline{b}_i(t, x) &= b_i(t, x) - \frac{1}{p_t(x)} \sum_{j,k=1}^{d} \frac{\partial}{\partial x_j}\big(p_t(x)\sigma_{ik}(t, x)\sigma_{jk}(t, x)\big) \\ &= b_i(t, x) - (\nabla \cdot \Sigma(t, x))_i - (\nabla \log p_t(x))_i, \quad i = 1, \dots, d, \Sigma = \sigma\sigma^\top\end{aligned}$$

Denoising Diffusion Models

- for some fixed time $T > 0$ consider the forward model

$$\mathrm{d}X_t = b(t, X_t)\,\mathrm{d}t + \sigma(t, X_t)\,\mathrm{d}W_t, \qquad t \in [0, T], X_0 \sim p_0$$

- letting $p_t(x) = \int p_{0,t}(y, x)\, p_0(\mathrm{d}y)$ be the marginal densities of (X_t), the time reversal $\overleftarrow{X}_t = X_{T-t}$ solves

$$\mathrm{d}\overleftarrow{X}_t = -\overline{b}(T-t, \overleftarrow{X}_t)\,\mathrm{d}t + \sigma(T-t, \overleftarrow{X}_t)\,\mathrm{d}\overline{W}_t, \qquad t \in [0, T], \overleftarrow{X}_0 \sim p_T,$$

  where

$$\begin{aligned}\overline{b}_i(t, x) &= b_i(t, x) - \frac{1}{p_t(x)} \sum_{j,k=1}^{d} \frac{\partial}{\partial x_j}\big(p_t(x)\sigma_{ik}(t, x)\sigma_{jk}(t, x)\big) \\ &= b_i(t, x) - (\nabla \cdot \Sigma(t, x))_i - (\nabla \log p_t(x))_i, \qquad i = 1, \ldots, d, \Sigma = \sigma\sigma^\top\end{aligned}$$

$\rightsquigarrow$ time-reversed process solves a time-inhomogeneous SDE, now with drift $-\overline{b}(T - \cdot, \cdot)$ involving the score $\nabla \log p_t$, which depends on the unknown data distribution p_0

$\rightsquigarrow$ score needs to be estimated from the data

Denoising score matching

- denoising score matching:

$$\nabla \log p_t(x) = \frac{\int \nabla_x p_{0,t}(y,x)\, p_0(\mathrm{d}y)}{p_t(x)} = \int \nabla_x \log p_{0,t}(y,x)\, \underbrace{\frac{p_{0,t}(y,x)\, p_0(\mathrm{d}y)}{p_t(x)}}_{=\mathbb{P}(X_0 \in \mathrm{d}y | X_t = x)}$$

$$= \mathbb{E}[\nabla_2 \log p_{0,t}(X_0, X_t) \mid X_t = x]$$

and thus

$$\mathfrak{s} := \nabla \log p_t \in \operatorname*{arg\,min}_{s \text{ meas.}} \mathbb{E}\big[\|s(X_t) - \nabla_2 \log p_{0,t}(X_0, X_t)\|^2\big]$$

$\rightsquigarrow$ given data $(X_0^i)_{i \in [n]} \overset{\text{iid}}{\sim} p_0$ define the denoising score estimator

$$\hat{\mathfrak{s}} \in \operatorname*{arg\,min}_{s \in \mathcal{S}} \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}_{X_0^i}\Big[\int_{\underline{T}}^{T} \|s(t, X_t) - \nabla_2 \log p_{0,t}(X_0, X_t)\|^2 \,\mathrm{d}t\Big],$$

where $0 < \underline{T} \ll T$ and $\mathcal{S}$ is an approximating function class, e.g. space-time neural networks

Generative process

On $[0, T - \underline{T}]$, simulate

$$\mathrm{d}Y_t = \big(-b(T-t, Y_t)+\nabla\cdot\Sigma(T-t, Y_t)+\Sigma(T-t, Y_t)\hat{\mathfrak{s}}(T-t, Y_t)\big)\,\mathrm{d}t+\sigma(T-t, Y_t)\,\mathrm{d}W_t, \quad \mathbb{P}^{Y_0}(\mathrm{d}y) \approx p_T(y)\,\mathrm{d}y$$

Output:

$$Y_{T-\underline{T}} \overset{d}{\approx} \overleftarrow{X}_{T-\underline{T}} = X_{\underline{T}} \overset{d}{\approx} X_0$$

Generative process

On $[0, T - \underline{T}]$, simulate

$$\mathrm{d}Y_t = \big(-b(T-t, Y_t) + \nabla\cdot\Sigma(T-t, Y_t) + \Sigma(T-t, Y_t)\hat{\mathfrak{s}}(T-t, Y_t)\big)\,\mathrm{d}t + \sigma(T-t, Y_t)\,\mathrm{d}W_t, \quad \mathbb{P}^{Y_0}(\mathrm{d}y) \approx p_T(y)\,\mathrm{d}y$$

Output:

$$Y_{T-\underline{T}} \overset{d}{\approx} \overleftarrow{X}_{T-\underline{T}} = X_{\underline{T}} \overset{d}{\approx} X_0$$

Minimax optimality of diffusion models

Assumptions on data distribution p_0 with support $\mathscr{M}$:

- $\mathrm{Leb}(\mathscr{M}) > 0$, $\mathscr{M}$ bounded, $p_0|_{\mathscr{M}} \geq m > 0$ and β-smooth: Oko, Akiyama, Suzuki (*ICML* '23), Dou, Kotekal, Xu and Zhou ('24+) [$d = 1$, no log-factors], Holk, Strauch, LT ('25+) [reflected models]
- $d = 1$, $\mathscr{M} = \mathbb{R}$, p_0 not lower bounded: Zhang et al. ('25, *ICML*)
- $\mathscr{M}$ bounded and $\subset$ linear subspace: Oko, Akiyama, Suzuki (*ICML* '23), Chen et al. (*ICML* '23)
- $\mathscr{M}$ is a d^*-dimensional submanifold: Tang and Yang (*AISTATS* '24), Azangulov, Delegiannidis and Rousseau ('24+) [rates adapt to intrinsic dimension d^*]
- $\log p_0(x) = \sum_{J\subset[d],|J|\leq d^*} f_J(x_J)$, for f_J β-Hölder: Kwon et. al ('25+), Fan, Gu and Li ('25+)
- ... [?]

Generative process

On $[0, T - \underline{T}]$, simulate

$$\mathrm{d}Y_t = \big(-b(T-t, Y_t) + \nabla\cdot\Sigma(T-t, Y_t) + \Sigma(T-t, Y_t)\hat{\mathfrak{s}}(T-t, Y_t)\big)\,\mathrm{d}t + \sigma(T-t, Y_t)\,\mathrm{d}W_t, \quad \mathbb{P}^{Y_0}(\mathrm{d}y) \approx p_T(y)\,\mathrm{d}y$$

Output:

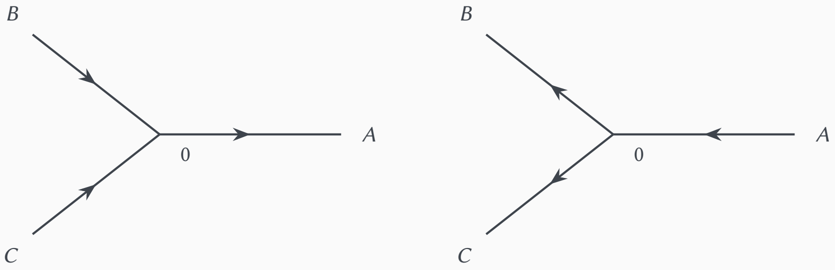
$$Y_{T-\underline{T}} \overset{d}{\approx} \overleftarrow{X}_{T-\underline{T}} = X_{\underline{T}} \overset{d}{\approx} X_0$$

Some fundamental observations

- time reversal at deterministic time T forces the backward process to be time-inhomogeneous
- if p_0 has low-dimensional support $\mathscr{M}$, for small t and x close to $\mathscr{M}$, $\nabla \log p_t(x)$ is approximately orthogonal to $\mathscr{M}$ (Stanczuk et al., *ICML* '24)
- initialising the generative process in a distribution that is not close to $\mathbb{P}^{X_T}$ and simulating for $T - \underline{T}$ time units will not give useful results $\rightsquigarrow$ algorithm is not adaptive to the noise level in the data

Homogeneous time reversal

- Markov property: "the past and future of a Markov process are conditionally independent given the present" $\rightsquigarrow$ time-reversed Markov processes are Markov
- to ensure that a homogeneous Markov process remains homogeneous under time reversal, we need to reverse at a suitable random (life)time ζ. This can be
  - a randomised stopping time such as an independent exponential time;
  - a last exit time;
  - a first hitting time;
  - any terminal time, that is, any stopping time T such that $T = t + T \circ \theta_t$ on $\{T > t\}$
- retaining the strong Markov property under time reversal is a bit more tricky:



h-transforms and time reversal

h-transform

For a possibly killed, homogeneous strong Markov process X with state space S, let h be an excessive function, that is

$$\mathbb{E}_x[h(X_t)] \leq h(x) \quad \text{and} \quad \lim_{t\to 0} \mathbb{E}_x[h(X_t)] = h(x).$$

Then,

$$P_t^h f(x) = \mathbb{E}_x\left[\frac{h(X_t)}{h(x)} f(X_t) \mathbf{1}_{\{X_t \in S\}}\right] \mathbf{1}_{(0,\infty)}(h(x)), \quad f \in \mathcal{B}_b(\mathbb{R}^d),$$

defines a sub-Markov semigroup. The corresponding Markov process X^h is strong Markov and is called h-transform of X.

- suppose that X is a continuous and self-dual Feller process (i.e., its generator satisfies $A = A^*$)
- if X^h has a finite killing time ζ, then the time-reversed process $\overleftarrow{X}_t^h = X_{\zeta-t}^h$ is homogeneous, strong Markov and is a $\overleftarrow{h}$-transform of X.

h-transforming a killed diffusion

- consider a symmetric diffusion process

$$\mathrm{d}X_t = b(X_t)\,\mathrm{d}t + \sigma(X_t)\,\mathrm{d}W_t$$

  with invariant measure m and let Z be its version killed at an independent exponential time with parameter $r > 0$
- as an excessive function for Z use

$$h(x) = \int G_r(x, y)\,\kappa(\mathrm{d}y)$$

  for the Green kernel $G_r(x, y) = \int_0^\infty e^{-rt} p_t(x, y)\,\mathrm{d}y$ and a representing measure κ
  - $\kappa(\mathrm{d}y) = r\,\mathrm{d}y \rightsquigarrow h = 1$ and $Z^h = Z$
  - $\kappa(\mathrm{d}y) = \frac{1}{G_r(x_0, y)}\beta(\mathrm{d}y) \rightsquigarrow Z$ conditioned to have distribution β before killing if started in x_0
  - Z is a killed Brownian motion and $\kappa(\mathrm{d}y) = \sigma_R(\mathrm{d}y)$ for the surface measure σ_R of an R-sphere $\mathbb{S}^{d-1}(R) \rightsquigarrow Z^h$ is killed at last exit from $\mathbb{S}^{d-1}(R)$

A time-homogeneous generative process

Proposition (Christensen, Strauch and LT (2025+))

1. Z^h is an Itô-diffusion with dynamics

$$\mathrm{d}Z_t^h = \big(b(Z_t^h) + \Sigma(X_t)\nabla \log h(X_t)\big)\,\mathrm{d}t + \sigma(Z_t^h)\,\mathrm{d}W_t$$

outside $\operatorname{supp}\kappa$ and its distribution at the lifetime is given by

$$\mathbb{P}_x(Z_{\zeta-}^h \in \mathrm{d}y) = \frac{G_r(x,y)}{h(x)}\kappa(\mathrm{d}y)$$

2. Let $\alpha = \mathbb{P}^{Z_0^h}$. Then $\overleftarrow{Z}_t^h$ is an $\overleftarrow{h}$-transform of Z with initial distribution $\mathbb{P}_\alpha(Z_{\zeta-}^h \in \mathrm{d}y)$ and

$$\overleftarrow{h}(x) := \int \frac{G_r(x,y)}{h(y)}\,\alpha(\mathrm{d}y).$$

In particular, $\overleftarrow{Z}^h$ has dynamics

$$\mathrm{d}\overleftarrow{Z}_t^h = \big(b(\overleftarrow{Z}_t^h) + \Sigma(\overleftarrow{Z}_t^h)\nabla \log \overleftarrow{h}(\overleftarrow{Z}_t^h)\big)\,\mathrm{d}t + \sigma(\overleftarrow{Z}_t^h)\,\mathrm{d}\overline{W}_t,$$

outside $\operatorname{supp}\alpha =: \mathscr{M}$ and $\mathbb{P}_\alpha(\overleftarrow{Z}_{\zeta-}^h \in \mathrm{d}y \mid \overleftarrow{Z}_0^h = x) = \frac{G_r(x,y)}{\overleftarrow{h}(x)h(y)}\alpha(\mathrm{d}y)$ for $\mathbb{P}_\alpha(Z_{\zeta-}^h \in \cdot)$-a.e. x.

A time-homogeneous generative process

Idealised algorithm:

1. Initialise $Z_0^{\overleftarrow{h}} \sim \tilde{\beta} \approx \mathbb{P}_\alpha(Z_{\zeta-}^h)$
   - for ergodic forward process with stationary distribution μ and small exponential killing rate $r > 0$, choose $\tilde{\beta} = \mu$ [$\leftrightsquigarrow$ ergodic diffusion model]
   - for exponentially killed BM with small killing rate $r > 0$, choose $\tilde{\beta} = \mathrm{Laplace}(0, (2r)^{-1/2}\mathbb{I}_d)$ [$\leftrightsquigarrow$ variance exploding diffusion model]
   - for $\kappa(\mathrm{d}y) = \frac{1}{G_r(x_0,y)}\delta_z$, choose $\tilde{\beta} = \delta_z$
2. Simulate diffusion $Z^{\overleftarrow{h}}$ until killing time and output $Z_{\zeta-}^{\overleftarrow{h}}$

A time-homogeneous generative process

Idealised algorithm:

1. Initialise $Z_0^{\overleftarrow{h}} \sim \tilde{\beta} \approx \mathbb{P}_\alpha(Z^h_{\zeta-})$
   - for ergodic forward process with stationary distribution μ and small exponential killing rate $r > 0$, choose $\tilde{\beta} = \mu$ [$\leftrightsquigarrow$ ergodic diffusion model]
   - for exponentially killed BM with small killing rate $r > 0$, choose $\tilde{\beta} = \mathrm{Laplace}(0, (2r)^{-1/2}\mathbb{I}_d)$ [$\leftrightsquigarrow$ variance exploding diffusion model]
   - for $\kappa(\mathrm{d}y) = \frac{1}{G_r(x_0, y)}\delta_z$, choose $\tilde{\beta} = \delta_z$
2. Simulate diffusion $Z^{\overleftarrow{h}}$ until killing time and output $Z^{\overleftarrow{h}}_{\zeta-}$

Requirements for implementation

1. learn $\nabla \log \overleftarrow{h}$ (only a function in space – no time component);

A time-homogeneous generative process

Idealised algorithm:

1. Initialise $Z_0^{\overleftarrow{h}} \sim \tilde{\beta} \approx \mathbb{P}_\alpha(Z_{\zeta-}^h)$
    - for ergodic forward process with stationary distribution μ and small exponential killing rate $r > 0$, choose $\tilde{\beta} = \mu$ [↭ ergodic diffusion model]
    - for exponentially killed BM with small killing rate $r > 0$, choose $\tilde{\beta} = \text{Laplace}(0, (2r)^{-1/2}\mathbb{I}_d)$ [↭ variance exploding diffusion model]
    - for $\kappa(\mathrm{d}y) = \frac{1}{G_r(x_0, y)}\delta_z$, choose $\tilde{\beta} = \delta_z$
2. Simulate diffusion $Z^{\overleftarrow{h}}$ until killing time and output $Z_{\zeta-}^{\overleftarrow{h}}$

Requirements for implementation

1. learn $\nabla \log \overleftarrow{h}$ (only a function in space – no time component);
2. learn killing time ζ of $Z^{\overleftarrow{h}}$

Learning to kill

Polarity hypothesis

Assume that $\mathscr{M} = \operatorname{supp}\alpha$ is polar for X, i.e., for any $x \in \mathbb{R}^d$, $\mathbb{P}_x(\inf\{t > 0 : X_t \in \mathscr{M}\} < \infty) = 0$.

Learning to kill

Polarity hypothesis

Assume that $\mathscr{M} = \operatorname{supp} \alpha$ is polar for X, i.e., for any $x \in \mathbb{R}^d$, $\mathbb{P}_x(\inf\{t > 0 : X_t \in \mathscr{M}\} < \infty) = 0$.

Theorem (Christensen, Strauch and LT (2025+))

Under the polarity hypothesis, the backward process $\overleftarrow{Z}^h$ is killed at first entrance into $\mathscr{M}$.

Learning to kill

Polarity hypothesis

Assume that $\mathscr{M} = \operatorname{supp} \alpha$ is polar for X, i.e., for any $x \in \mathbb{R}^d$, $\mathbb{P}_x(\inf\{t > 0 : X_t \in \mathscr{M}\} < \infty) = 0$.

Theorem (Christensen, Strauch and LT (2025+))

Under the polarity hypothesis, the backward process $\overleftarrow{Z}^h$ is killed at first entrance into $\mathscr{M}$.

Possible strategies to estimate a δ-fattening $\mathscr{M}_\delta = \{x : \operatorname{dist}(x, \mathscr{M}) \leq \delta\}$ given data $X^1, \ldots, X^n \overset{\text{iid}}{\sim} \alpha$ and an estimator $\hat{\mathfrak{s}}$ of $\mathfrak{s} := \nabla \log \overleftarrow{h}$:

- explicit plug-in approach: estimate $\mathscr{M}_\delta$ directly or indirectly by setting $\widehat{\mathscr{M}_\delta} = (\widehat{\mathscr{M}})_\delta$; then set $\hat{\zeta} := \inf\{t \geq 0 : Z_t^{\hat{\mathfrak{s}}} \in \widehat{\mathscr{M}_\delta}\}$
- implicit approach: use explosive behaviour of $\mathfrak{s}$ as $x \to \mathscr{M}$

Theorem (Christensen, Kallsen, Strauch and LT (2025+))

Suppose that $\mathscr{M}$ is polar for X and Y solving $\mathrm{d}Y_t = \sigma(Y_t)\,\mathrm{d}B_t$. Then, it a.s. holds that

$$\zeta = \inf\Big\{t \geq 0 : \sup_{s \leq t} |\mathfrak{s}(\overleftarrow{Z}^h_s)| = \infty\Big\} = \inf\Big\{t \geq 0 : \|\mathfrak{s}(\overleftarrow{Z}^h)\|_{L^2([0,t])} = \infty\Big\}.$$

Denoising score matching

- for $\mathbb{P}_\alpha(Z^h_{\zeta_-} \in \cdot)$-a.e. x

$$\mathfrak{s}(x) = \nabla \log \overleftarrow{h}(x) = \frac{1}{\overleftarrow{h}(x)} \int \nabla_x G_r(x,y) \frac{1}{h(y)}\,\alpha(\mathrm{d}y) = \int \nabla_x \log G_r(x,y) \frac{G_r(x,y)}{\overleftarrow{h}(x)h(y)}\,\alpha(\mathrm{d}y)$$
$$= \mathbb{E}\big[\nabla_x \log G_r(x, Z^{\overleftarrow{h}}_{\zeta-}) \mid Z^{\overleftarrow{h}}_0 = x\big]$$
$$= \mathbb{E}_\alpha\big[\nabla_x \log G_r(x, Z^h_0) \mid Z^h_{\zeta_-} = x\big]$$

- this implies that on $\mathbb{R}^d \setminus \mathscr{M}_\delta$, $\mathfrak{s}$ agrees $\mathbb{P}_\alpha(Z^h_{\zeta-} \in \cdot)$-a.e. with the minimiser of

$$\mathcal{B}(\mathbb{R}^d;\mathbb{R}^d) \ni s \mapsto \mathbb{E}_\alpha\Big[\big\|s(Z^h_{\zeta-}) - \nabla \log G_r(Z^h_0, Z^h_{\zeta-})\big\|^2 \mathbf{1}_{\{\|Z^h_{\zeta-} - Z^h_0\| > \delta\}}\Big]$$

- note that if $Z^h = Z$, then $\zeta \sim \mathrm{Exp}(r)$ independent of X, $Z_{\zeta-} = X_\zeta$ has full support and we have

$$\mathbb{E}_\alpha\Big[\big\|s(Z^h_{\zeta-}) - \nabla \log G_r(Z^h_0, Z^h_{\zeta-})\big\|^2 \mathbf{1}_{\{\|Z^h_{\zeta-} - Z^h_0\| > \delta\}}\Big] = r\mathbb{E}_\alpha\Big[\int_0^\zeta \big\|s(Z^h_t) - \nabla \log G_r(Z^h_0, Z^h_t)\big\|^2 \mathbf{1}_{\{\|Z^h_t - Z^h_0\| > \delta\}}\,\mathrm{d}t\Big]$$

Projection learning

- we don't have to start the backward process approximately in $\mathbb{P}_\alpha(Z^h_{\zeta_-} \in \mathrm{d}y)$: it will always be killed on the data support $\mathscr{M}$ and different initialisations will yield different output distributions supported on $\mathscr{M}$ $\rightsquigarrow$ natural conditioning
- a natural question is therefore what happens if we don't start the generative process from pure noise but something more informative, say a masked or moderately noised picture

- it turns out that the natural conditioning aspect entails a blessing of dimensionality

Projection learning

Let Z be an exponentially killed Brownian motion. Then,

$$\overleftarrow{h}(x) = \int G_r(x,y)\,\alpha(\mathrm{d}y), \quad G_r(x,y) = 2(2\pi)^{-d/2} r\left(\frac{\sqrt{2r}}{|x-y|}\right)^{\frac{d-2}{2}} K_{\frac{d-2}{2}}\left(\frac{\sqrt{2r}}{|x-y|}\right).$$

For large d,

$$\nabla \log \overleftarrow{h}(x) \approx d\,\frac{\int \frac{x-y}{|x-y|^d}\,\alpha(\mathrm{d}y)}{\int |x-y|^{2-d}\,\alpha(\mathrm{d}y)}$$

and thus, if there is a unique projection $x^* \in \arg\min_{y\in\mathscr{M}} |x-y|$ of x onto $\mathscr{M}$, then

$$\nabla \log \overleftarrow{h}(x) \approx d\,\frac{x^*-x}{|x^*-x|^2} = d\,\frac{\operatorname{sign}(x^*-x)}{|x^*-x|}$$

Theorem (Christensen, Kallsen, Strauch and LT (2025+))

Let $\delta, \varepsilon > 0$ and fix an observation $x \in \mathbb{R}^d$. If $\alpha(B(x,r)) > \varepsilon$ for some ball $B(x,r)$ with radius $r > 0$ around y, then

$$\mathbb{P}\left(Z^{\overleftarrow{h}}_{\zeta-} \in \mathscr{M} \cap B(x,(1+\delta)r) \mid Z^{\overleftarrow{h}}_0 = x\right) \geq 1 - \frac{1}{\varepsilon}(1+\delta)^{2-d}.$$

Projection learning

Consider now estimators $\hat{\mathfrak{s}}_n$, an independent Brownian motion W and let $\widehat{Z}^{\hat{\mathfrak{s}}_n}$ be the process solving

$$\mathrm{d}\widehat{Z}_t^{\hat{\mathfrak{s}}_n} = \hat{\mathfrak{s}}_n(\widehat{Z}_t^{\hat{\mathfrak{s}}_n})\mathbf{1}_{\{t\le\hat{\zeta}\}}\,\mathrm{d}t + \mathbf{1}_{\{t\le\hat{\zeta}\}}\,\mathrm{d}W_t, \quad \hat{\zeta} := \inf\Big\{t \ge 0 \,:\, \|\widehat{Z}^{\hat{\mathfrak{s}}_n}\|_{L^2[0,t]} > M\Big\}.$$

Theorem (Christensen, Kallsen, Strauch and LT (2025+))

Fix an observation $x \in \mathbb{R}^d$. Suppose that

- for any $\tilde{\delta}, \delta, \varepsilon > 0$ it holds for sufficiently large n that
$$\mathbb{P}\Big(\Big\|\big(\hat{\mathfrak{s}}_n(Z^{\overleftarrow{h}}) - \mathfrak{s}(Z^{\overleftarrow{h}})\big)\mathbf{1}_{\{Z^{\overleftarrow{h}}\notin\mathscr{M}_{\tilde{\delta}}\}}\Big\|_{L^2(\zeta)} > \delta \,\Big|\, Z_0^{\overleftarrow{h}} = x\Big) < \varepsilon$$
- for any $n \in \mathbb{N}$ and $\tilde{\delta} > 0$, the function $\hat{\mathfrak{s}}_n$ is $L_{\tilde{\delta}}$-Lipschitz on $\mathscr{M}_{\tilde{\delta}}^c$

Let $\delta, \varepsilon, \tilde{\delta}, \tilde{\varepsilon} > 0$. If $\alpha(B(x,r)) > \varepsilon$, then, for sufficiently large $M > 0$ and $n \in \mathbb{N}$,

$$\mathbb{P}\big(\widehat{Z}_{\hat{\zeta}}^{\hat{\mathfrak{s}}_n} \in \mathscr{M}_{\tilde{\delta}} \cap B(x,(1+\delta)r) \,\big|\, \widehat{Z}_0^{\hat{\mathfrak{s}}_n} = x\big) \;>\; 1 - \frac{1}{\varepsilon}(1+\delta)^{2-d} - \tilde{\varepsilon}.$$

Projection learning

Consider now estimators $\hat{\mathfrak{s}}_n$, an independent Brownian motion W and let $\widehat{Z}^{\hat{\mathfrak{s}}_n}$ be the process solving

$$\mathrm{d}\widehat{Z}_t^{\hat{\mathfrak{s}}_n} = \hat{\mathfrak{s}}_n(\widehat{Z}_t^{\hat{\mathfrak{s}}_n})\mathbf{1}_{\{t\le\hat{\zeta}\}}\,\mathrm{d}t + \mathbf{1}_{\{t\le\hat{\zeta}\}}\,\mathrm{d}W_t, \quad \hat{\zeta} := \inf\Big\{t \ge 0 \,:\, \|\widehat{Z}^{\hat{\mathfrak{s}}_n}\|_{L^2[0,t]} > M\Big\}.$$

Theorem (Christensen, Kallsen, Strauch and LT (2025+))

Fix an observation $x \in \mathbb{R}^d$. Suppose that

- for any $\tilde{\delta}, \delta, \varepsilon > 0$ it holds for sufficiently large n that
$$\mathbb{P}\Big(\Big\|(\hat{\mathfrak{s}}_n(Z^{\overleftarrow{h}}) - \mathfrak{s}(Z^{\overleftarrow{h}}))\mathbf{1}_{\{Z^{\overleftarrow{h}}\notin\mathcal{M}_{\tilde{\delta}}\}}\Big\|_{L^2(\zeta)} > \delta \,\Big|\, Z_0^{\overleftarrow{h}} = x\Big) < \varepsilon$$
- for any $n \in \mathbb{N}$ and $\tilde{\delta} > 0$, the function $\hat{\mathfrak{s}}_n$ is $L_{\tilde{\delta}}$-Lipschitz on $\mathcal{M}_{\tilde{\delta}}^{\mathsf{c}}$

Let $\delta, \varepsilon, \tilde{\delta}, \tilde{\varepsilon} > 0$. If $\alpha(B(x,r)) > \varepsilon$, then, for sufficiently large $M > 0$ and $n \in \mathbb{N}$,

$$\mathbb{P}\big(\widehat{Z}_{\hat{\zeta}}^{\hat{\mathfrak{s}}_n} \in \mathcal{M}_{\tilde{\delta}} \cap B(x,(1+\delta)r) \,\big|\, \widehat{Z}_0^{\hat{\mathfrak{s}}_n} = x\big) \; > \; 1 - \frac{1}{\varepsilon}(1+\delta)^{2-d} - \tilde{\varepsilon}.$$

Thank you for your attention!