Training Diffusion Models
(with applications to statistics)

Connie Trojan

November 2024

Diffusion Model Recap

- Perturb the data with a **stochastic differential equation**

$$dX_t = \underbrace{f(t)X_t dt}_{\text{drift}} + \underbrace{g(t)dW_t}_{\text{noise}}$$

- For SDEs of this form, the distribution of $X_t \mid X_0$ will be $N(m(t)X_0, s(t)^2)$ where m and s can be found by integration

- This is a generalisation of the discrete time noising processes used by and Ho et al. [2020] and Song and Ermon [2019]

Time Reversal

- The time reversal of the noising SDE is

$$dX_t = \left[f(t)X_t - g(t)^2 \nabla_x \log p_t(X_t)\right] dt + g(t)dW_t$$

- Here, the **score function** $\nabla_x \log p_t(x)$ is the gradient of the log pdf of X_t *with respect to* x.
- If we can approximate this, the above SDE can be used to generate new samples

Score Matching

- Method for approximating an unnormalised probability density by learning its score function

- If we had access to the true score, the ideal objective would be **explicit score matching**:

$$J(\theta) = \mathbb{E}_{p(x)} \left[\frac{1}{2} \| \psi(x;\theta) - \nabla_x \log p(x) \|_2^2 \right]$$

- This minimises the MSE between the approximation $\psi(x;\theta)$ and the true score $\nabla p(x)$

Denoising Score Matching

- **Denoising score matching** [Vincent, 2011], is an approximation that matches the score function of a kernel density estimate of the target density

- Using kernel q, this can be seen as the noised data distribution $\tilde{x} = x + e$, where $e \sim q(e)$.

- Vincent [2011] showed that the following objective is equivalent to explicit score matching on the score of $\tilde{x}$:

$$L_{DSM}(\theta) = \mathbb{E}_{q(x,\tilde{x})}\left[\frac{1}{2}\|\psi(\tilde{x};\theta) - \nabla_{\tilde{x}} \log q(\tilde{x} - x)\|_2^2\right]$$

Denoising Score Matching

$$L_{DSM}(\theta) = \mathbb{E}_{p_0(x)q(\tilde{x}|x)}\left[\frac{1}{2}\|\psi(\tilde{x};\theta) - \nabla_{\tilde{x}} \log q(\tilde{x}|x)\|_2^2\right]$$

- This does not require the score of the data density, only the score of the noising kernel q.
- For example, for a Gaussian kernel $e \sim N(0, \sigma^2)$ we have $\nabla_{\tilde{x}} \log q(\tilde{x}|x) = \frac{1}{\sigma^2}(x - \tilde{x})$, the direction that removes the noise from $\tilde{x}$.
- If we approximate p_0 with a **finite** sample, this matches the score of a kernel density estimate rather than the true p_t

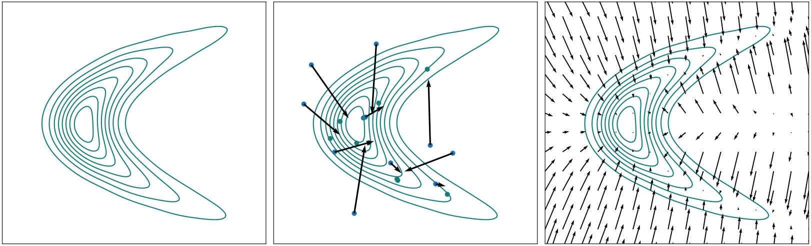


Figure: Score matching on a warped Gaussian: target distribution, noised samples at $t = 0.15$ with direction of $\nabla \log p_{t|0}$ indicated by arrows, learned score function at $t \approx 0$.

Why use a diffusion?

- In principle, we could simply do this for some small noise level and use e.g. unadjusted Langevin dynamics to generate new samples
- There are obvious problems with this - KDE works poorly for high dimensional distributions
- Song and Ermon [2019] showed empirically that doing this fails to recover mode weights in mixture distributions even when the true score is used

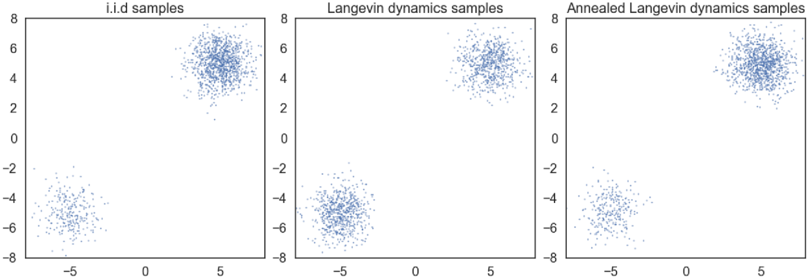


Figure: From Song and Ermon [2019]: target distribution, samples using ULA with the true score function, samples using ULA with a sequence of noised densities

Time Conditioning

- Although we are training with simple KDEs, the ‘magic’ of diffusion models comes from using a sequence of noise levels, and from training a single model across time
- Score estimates are implicitly smoothed through time in a way that can give better estimates in low density regions than could be obtained with separate models
- This benefit comes from the neural network approximation - choosing a good architecture is key

- So, we need an objective function that trains a single model for all noise levels

- We choose the kernel q in DSM to be the transition density $p_{t|0}(x_t \mid x_0)$ of the forward diffusion, to learn the desired $\nabla \log p_t$

- Song and Ermon [2019] incorporate time conditioning in the objective by taking a weighted expectation over t:

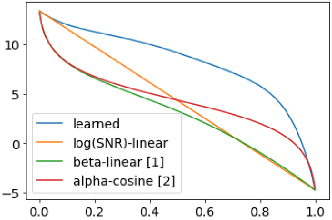
$$\mathbb{E}_t \left\{ \lambda(t)\, \mathbb{E}_{p_0(x)p_t(x_t|x)} \left[\frac{1}{2} \|\psi(x_t, t; \theta) - \nabla_{x_t} \log p_{t|0}(x_t \mid x_0)\|_2^2 \right] \right\}$$

$$\mathbb{E}_t \left\{ \lambda(t)\, \mathbb{E}_{p_0(x) p_t(x_t|x)} \left[\frac{1}{2} \|\psi(x_t, t; \theta) - \nabla_{x_t} \log p_{t|0}(x_t \mid x_0)\|_2^2 \right] \right\}$$

- Here, t is distributed uniformly on $[0, 1]$, and $\lambda : \mathbb{R} \to \mathbb{R}_{>0}$ is a weighting function.
- Song et al. [2021] showed that choosing the weighting $\lambda(t) = g(t)^2$ makes this an upper bound for the model KL divergence

Learning the noising process

- Using the SDE formulation allows us to vary the time discretisation used in sampling
- In continuous time, different noising processes can be equivalent to each other. This means the SDE itself does not have to be fixed in advance either
- Kingma et al. [2023] reparameterise the DSM objective to show that it is invariant to changes to the VP SDE that preserve the signal-to-noise ratio at $t = 0$ and $t = 1$
- This can be used to optimise the noise schedule to reduce the variance of the objective function, speeding up training



(a) log SNR vs time t

SNR(t) schedule	Var(BPD)
Learned (ours)	**0.53**
log SNR-linear	6.35
β-Linear [1]	31.6
α-Cosine [2]	31.1

(b) Variance of VLB estimate

Figure: From Kingma et al. [2023] - comparison of objective function variance for different noise schedules

What if we know p_0?

- There has been much recent interest in using diffusion models for sampling in Bayesian statistics, even when the target density is known:
  - They are useful as surrogate models when the target density is expensive to evaluate
  - They can successfully sample from complex, multi-modal distributions, so are attractive as an alternative to MCMC

Target Score Matching

- A key difficulty is that the noised density p_t is typically intractable even if we know the target density p_0, so we still need to train a score approximation
- Denoising score matching (DSM) often struggles to approximate the score function at low noise levels, since the variance of its score estimates explodes as $t \to 0$
- De Bortoli et al. [2024] proposed an alternative objective, which uses a rescaling of the unnoised score function rather than the score of the noising distribution

Target Score Matching

- The following objective can be used to estimate $\nabla \log p_t$:

$$L_{TSM}(\theta, t) = \mathbb{E}_{x_0, x_t}\left[\|\psi(x_t, t; \theta) - m(t)^{-1}\nabla \log p_0(x_0)\|_2^2\right]$$

- L_{TSM} is very well behaved near $t = 0$ where the regression target is a low variance estimator of the true score, less so for large t since typically $m(t) \to 0$

- We can get an objective that is well behaved across time by taking a weighted combination of TSM and DSM

Diffusion-Based Samplers

- TSM incorporates evaluations of the true density, but it still requires an initial sample from the target distribution to compute the objective function

- Diffusions have desirable properties for sampling from complex distributions (e.g. good mixing for multimodal distributions) so there has been recent interest in using them for sampling

- For example, Phillips et al. [2024] start with an initial approximate sample, which is refined over repeated rounds of training

Motivation: Diffusion vs Tempering

- Diffusion models interpolate between the target distribution and a tractable distribution, much like tempering [Neal, 2001]:

$$p_t(x) = p(x)^{1-t}\phi(x)^t$$

- Unlike diffusions, the intermediate densities p_t in tempering are known and do not have to be estimated
- However, diffusions can outperform tempering on multimodal distributions with differing mode weights

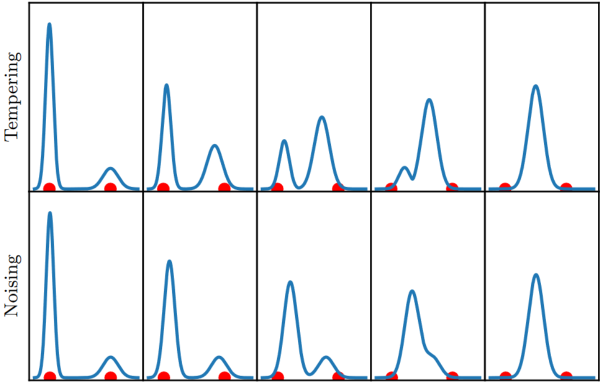


Figure: From Phillips et al. [2024], comparing the intermediate densities in tempering and noising. In tempering, the mode weights can 'switch'.

Score Network

- Can in theory use any architecture that maps $\mathbb{R}^d \to \mathbb{R}^d$ with time as an input
- In practice, the network can have a huge impact on results. Try and use domain knowledge to choose an appropriate architecture, add time as an input to each layer
- For 'low dimensional' distributions ($d \approx 100$), a small feedforward MLP or ResNet will do

The U-net

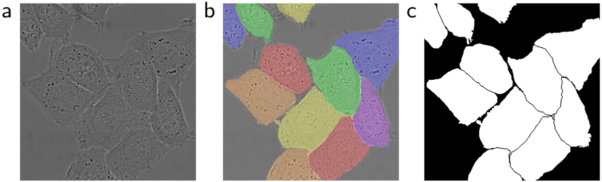


Figure: From Ronneberger et al. [2015]: image segmentation

The U-net is an architecture proposed by Ronneberger et al. [2015] for image-to-image tasks, which has become ubiquitous as an architecture for score nets in image generation

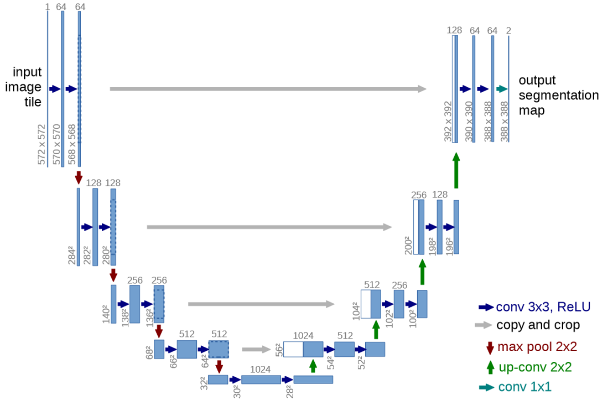
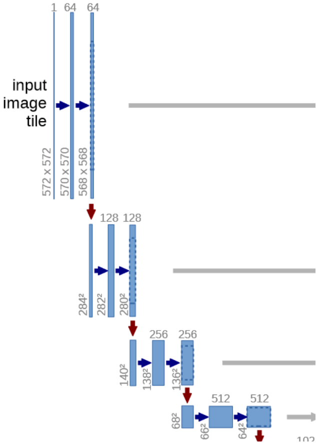


Figure: From Ronneberger et al. [2015]: U-net architecture

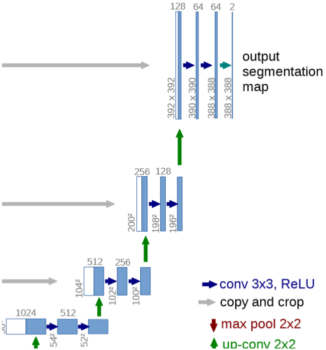
(Exact details like number/type of convolutions and inclusion of dense layers can vary)



- The LHS is a typical CNN architecture, with a sequence of convolutions that decrease the resolution and increase the number of channels
- The idea is for each channel to extract a different key feature of the image

- The RHS mirrors the LHS, with transposed convolutions returning the image to its original dimensions

- This uses the extracted features to construct an output image



- The low-res final output of the LHS does not contain precise information about where features are located, but we need each output pixel to relate to the corresponding input pixel
- So, the full U-net includes skip connections concatenating outputs from the LHS onto the inputs of the RHS to help with localisation

Energy based parameterisation

- Since the score function of a distribution determines its density up to normalising constant, we can use score matching to estimate the target density directly
- This idea was proposed by Salimans and Ho [2021] as a way of ensuring that the score approximation is in fact a valid score function
- This is known as an **energy-based model** (EBM) because we model an energy function $E(x, t; \theta)$ and approximate p_t by $\exp(-E(x, t; \theta))$

- A common parameterisation is:

$$E(x, t; \theta) = \frac{1}{2s(t)} ||x - \psi(x, t; \theta)||_2^2 \,,$$

where $\psi(x, t; \rho) : \mathbb{R}^d \to \mathbb{R}^d$ is a neural network and $s(t)^2$ is the variance of the noising kernel $p_{t|0}$

- The gradient $-\nabla_x E(x, t; \rho)$ is substituted into the usual score matching objective in training

- Salimans and Ho [2021] found that this performed similarly but no better than the usual parameterisation

Composing diffusion models

Du et al. [2023] found that using an EBM to perform MCMC sampling enables sampling from compositions of diffusion models

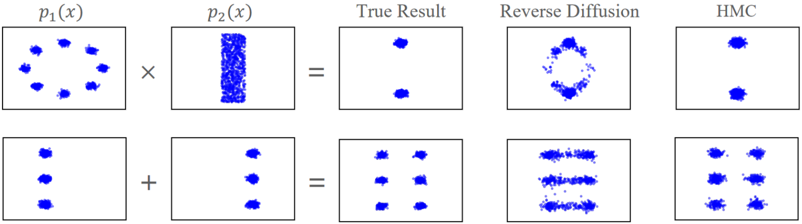


Figure: From Du et al. [2023]: sampling from product and mixture distributions, where a reverse SDE is not available

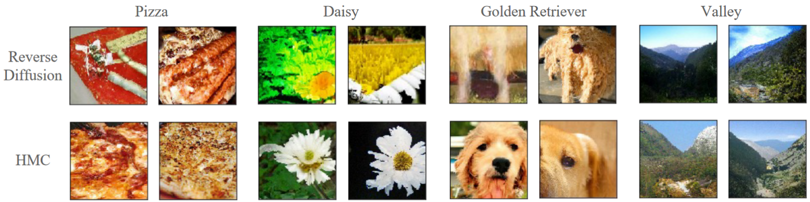


Figure: Du et al. [2023] - classifier guidance as sampling from a product distribution

Figure: Du et al. [2023] - using product distributions to combine prompts

Applications in Statistics

- Sampling from products of posterior distributions can be used in Bayesian statistics to sample from the posterior conditioned on their pooled datasets

- **Simulation-based inference** [Geffner et al., 2023] - approximating single-observation posteriors requires fewer simulator calls than conditioning jointly on larger datasets

- **Divide-and-conquer MCMC** [Trojan et al., 2024] - if the full dataset is very large, it can be computationally intractable to sample directly from the full posterior distribution

References I

V. De Bortoli, M. Hutchinson, P. Wirnsberger, and A. Doucet. Target score matching, 2024. arXiv preprint arXiv:2402.08667.

Y. Du, C. Durkan, R. Strudel, J. B. Tenenbaum, S. Dieleman, R. Fergus, J. Sohl-Dickstein, A. Doucet, and W. S. Grathwohl. Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and MCMC. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 8489–8510. PMLR, 2023.

T. Geffner, G. Papamakarios, and A. Mnih. Compositional score modeling for simulation-based inference, 2023. arXiv preprint arXiv:2209.14249v3.

J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc., 2020.

References II

D. P. Kingma, T. Salimans, B. Poole, and J. Ho. Variational diffusion models. *arXiv preprint 2107.00630*, 2023.

R. M. Neal. Annealed importance sampling. *Statistics and Computing*, 11:125–139, 2001.

A. Phillips, H.-D. Dau, M. J. Hutchinson, V. De Bortoli, G. Deligiannidis, and A. Doucet. Particle denoising diffusion sampler, 2024. arXiv preprint arXiv:2402.06320.

O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241. Springer International Publishing, 2015.

T. Salimans and J. Ho. Should EBMs model the energy or the score? In *Energy Based Models Workshop - ICLR 2021*, 2021.

References III

Y. Song and S. Ermon. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems*, volume 32, pages 11895–11907. Curran Associates, Inc., 2019.

Y. Song, C. Durkan, I. Murray, and S. Ermon. Maximum likelihood training of score-based diffusion models. In *Advances in Neural Information Processing Systems*, volume 34, pages 1415–1428. Curran Associates, Inc., 2021.

C. Trojan, P. Fearnhead, and C. Nemeth. Diffusion generative modelling for divide-and-conquer MCMC. *arXiv preprint 2406.11664*, 2024.

P. Vincent. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 2011.