Learning normalized image densities via dual score matching

Florentin Guth
Center for Data Science, New York University
Flatiron Institute, Simons Foundation
florentin.guth@nyu.edu

Zahra Kadkhodaie
Flatiron Institute, Simons Foundation
zk388@nyu.edu

Eero P. Simoncelli
New York University
Flatiron Institute, Simons Foundation
eero.simoncelli@nyu.edu

Abstract

Learning probability models from data is at the heart of many machine learning endeavors, but is notoriously difficult due to the curse of dimensionality. We introduce a new framework for learning *normalized* energy (log probability) models that is inspired by diffusion generative models, which rely on networks optimized to estimate the score. We modify a score network architecture to compute an energy while preserving its inductive biases. The gradient of this energy network with respect to its input image is the score of the learned density, which can be optimized using a denoising objective. Importantly, the gradient with respect to the noise level provides an additional score that can be optimized with a novel secondary objective, ensuring consistent and normalized energies across noise levels. We train an energy network with this *dual* score matching objective on the ImageNet64 dataset, and obtain a cross-entropy (negative log likelihood) value comparable to the state of the art. We further validate our approach by showing that our energy model *strongly generalizes*: log probabilities estimated with two networks trained on non-overlapping data subsets are nearly identical. Finally, we demonstrate that both image probability and dimensionality of local neighborhoods vary substantially depending on image content, in contrast with conventional assumptions such as concentration of measure or support on a low-dimensional manifold.

1 Introduction

Many problems in image processing and computer vision rely, explicitly or implicitly, on prior probability models. However, learning such models by maximizing the likelihood of a set of training images is difficult. The dimensionality of the space (i.e., the number of image pixels) is large, and worst-case data requirements for estimation grow exponentially (the "curse of dimensionality"). The machine learning community has developed a variety of methods to train a parametric network to estimate $\log p(x)$, known as an "energy model" (Hinton et al., 1986; LeCun et al., 2006), relying on the inductive biases of the network to alleviate the data requirements. For all but the simplest of models, this approach is frustrated by the intractability of estimating the normalization constant (Hinton, 2002; LeCun and Huang, 2005; Yedidia et al., 2005; Gutmann and Hyvärinen, 2010; Dinh et al., 2014; Rezende and Mohamed, 2015; Dinh et al., 2017; Song and Kingma, 2021).

A clever means of escaping this conundrum is to estimate the gradient of the energy with respect to the image (known as the "score"), which eliminates the normalization constant, and can be learned

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

from data with a "score-matching" objective (Hyvärinen and Dayan, 2005). The recent development of "diffusion" generative models (Sohl-Dickstein et al., 2015; Song and Ermon, 2019; Ho et al., 2020; Kadkhodaie and Simoncelli, 2020) builds on this concept, by estimating a *family* of score functions for images corrupted by Gaussian white noise at different amplitudes. These scores may then be used to draw samples from the corresponding estimated density, using an iterative reverse diffusion procedure. These methods have enabled dramatic improvements in both the quality and diversity of generated image samples, but the learned density is implicit. An explicit and normalized energy model (and density) can be obtained through integration of the divergence of the score vector field along a trajectory (Song et al., 2021a), at tractable but substantial computational cost.

Here, we leverage the power of diffusion models to develop a robust and efficient framework for directly learning a normalized energy model from image data. We approximate the energy with a deep network that takes as input both a noisy image and the corresponding noise variance, and derive two separate objectives. The first, obtained by differentiating with respect to the noisy image, is a denoising objective (as used in diffusion models). The second, obtained by differentiating with respect to the noise variance, ensures consistency of the energy estimates across noise levels, which we show to be critical for obtaining accurate and normalized energies. We optimize the sum of the two, in a procedure that we refer to as "*dual* score matching". We also propose a novel architecture for energy, computed as the inner product of the input and output images of a score network. This preserves the inductive biases of the base score network, leading to equal or superior denoising performance (and thus, sample quality).

We train our energy model on ImageNet64 (Russakovsky et al., 2015; Chrabaszcz et al., 2017), and show that the estimated energies lead to negative log likelihood (or cross-entropy) values comparable to the state of the art. We further demonstrate that the energy model strongly generalizes in the sense of Kadkhodaie et al. (2024): two separate models trained on *non-overlapping* subsets of the training data assign essentially the *same* probabilities to each image. This convergence is observed for training set sizes far smaller than the worst case prediction of the curse of dimensionality. We find that the distribution of log probabilities over ImageNet images covers a broad range, with densely textured images at the lower end, and sparse images at the upper end. The probability is relatively insensitive to changes in image luminance, but decreases with dynamic range. Finally, we highlight two geometrical properties of the learned image distribution. The first is an extremely tight inverse relationship between volume and density that leads to an absence of concentration of energy values. They furthermore follow a Gumbel distribution, revealing an unexpected form of statistical regularity. The second is that the dimensionality of the energy landscape in the neighborhood of an image varies greatly depending on image content and neighborhood size. We observe images with full-dimensional neighborhoods of non-negligible size and images with lower-dimensional neighborhoods even at sub-quantization scales. These results challenge traditional presuppositions regarding high-dimensional distributions, such as the concentration of measure phenomenon (Vershynin, 2018; Wainwright, 2019) and the manifold hypothesis (Tenenbaum et al., 2000; Bengio et al., 2013). Pre-trained models and software for running all experiments are available at `https://github.com/FlorentinGuth/DualScoreMatching`.

2 Learning normalized energy models with dual score matching

Estimating a high-dimensional probability density from samples faces two significant challenges as a result of the curse of dimensionality. The first is *statistical*: realistically-sized datasets do not contain enough information to learn an unknown high-dimensional distribution. One thus needs powerful inductive biases, typically in the form of a parametric network architecture, to hope to recover the data distribution. The second challenge is *computational*: the traditional objective aims to maximize the likelihood of the model over the data, which is intractable due to the need to compute the normalization constant. Nevertheless, recently developed generative models seem to succeed despite these problems. Drawing inspiration from diffusion models (Section 2.1), we develop a more direct solution by deriving a novel objective (Section 2.2) and architecture (Section 2.3) for learning *normalized* log probabilities (energies) from data. We validate our approach in Section 2.4.

2.1 Motivation

Traditionally, energy models are defined in terms of a parametric function $U_\theta(x)$ that approximates the unnormalized log density over $x \in \mathbb{R}^d$: $p_\theta(x) = \frac{1}{Z_\theta} \mathrm{e}^{-U_\theta(x)}$, with a normalizing constant

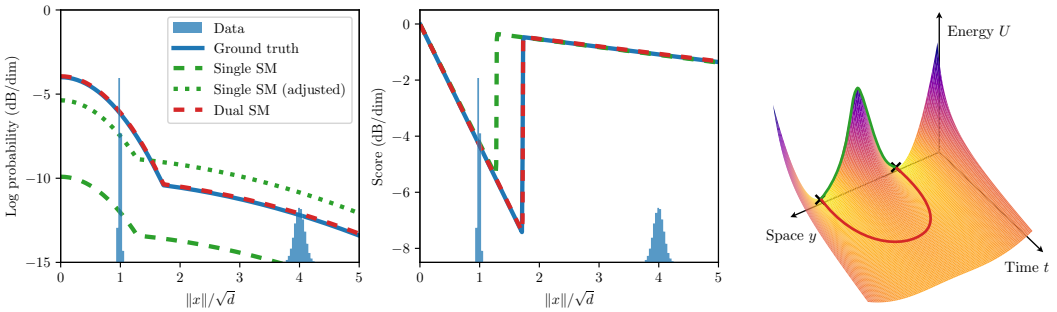

Figure 1: Comparison of single and dual score matching on recovering the energy of a scale mixture of two Gaussians in $d = 1000$ dimensions. Experimental details are provided in Appendix C.3. **Left:** Radial slices of the log probability. The single score matching estimate (green dashed curve) fails to recover the true energy (blue solid curve), even after global normalization (green dotted curve), while dual score matching (red dashed curve) succeeds. **Middle:** Radial components of the scores. Single score matching learns an accurate score over the support of the data (blue bar plot) but not outside of it. **Right:** Energy landscape across space and time (noise level) for a mixture of two Gaussians in one dimension. The direct path between the modes at $t = 0$ crosses a large energy barrier (green curve), which is alleviated on a path that is not restricted to $t = 0$ (red curve).

$Z_\theta = \int \mathrm{e}^{-U_\theta(x)} \mathrm{d}x$. The parameters θ are estimated by minimizing the expected negative log likelihood (NLL), $\mathbb{E}_x[-\log p_\theta(x)] = \mathbb{E}_x[U_\theta(x)] + \log Z_\theta$, which is equivalent to minimizing the KL divergence between $p_\theta(x)$ and the data distribution. The normalizing constant Z_θ plays a critical role in learning, representing the total energy over $\mathbb{R}^d$ which trades off against the energy of the data. Unfortunately, direct estimation (i.e., computing the integral) is typically intractable.

The normalization constant can be eliminated from the NLL by differentiating w.r.t. x, yielding a quantity known as the (negative) score: $-\nabla_x \log p_\theta(x) = \nabla_x U_\theta(x)$. As a result, a score model can be efficiently fitted to data via "score matching" (Hyvärinen and Dayan, 2005), which minimizes the Fisher (as opposed to KL) divergence between the model and the data. For data corrupted by Gaussian white noise, this amounts to solving a denoising problem (Vincent, 2011; Raphan and Simoncelli, 2011). But this computational advantage comes at a statistical cost: a good approximation of the score does not always lead to a good model of the energy (Koehler et al., 2023), as we now illustrate. Consider the case of an equal mixture of two multivariate Gaussian distributions with zero mean and variances σ_1^2 and σ_2^2. In high dimensions, this distribution concentrates near the two hyper-spheres with radii $\sigma_1\sqrt{d}$ and $\sigma_2\sqrt{d}$, leading to data scarcity in the rest of the space. We show in Figure 1 energies estimated from samples by optimizing their gradient via (single) score matching. This fails to recover the true energy, even after adjusting a normalization constant. Indeed, evaluating the energy difference between the two Gaussians requires good estimation of the score *along an integration path* between them. If there is no data in-between modes to constrain the score, due to an energy barrier or concentration phenomena, this leads to inconsistent energy values across modes. In other words, single score matching estimates energy values up to a *mode-dependent* additive constant.

Diffusion models expand on score matching by learning scores of data corrupted by white Gaussian noise for a range of different noise amplitudes, $-\nabla_y U(y, t)$, with $y = x + \mathcal{N}(0, t\mathrm{Id})$. The evolution of the density $p(y|t)$ as "time" (noise variance) t increases is a diffusion process, and thus the corresponding energies increase in smoothness with t. This *multiscale* family of scores can be used to draw high-quality samples from $p(x)$ using a reverse diffusion algorithm, which follows a trajectory of partial denoising steps (Song and Ermon, 2019; Song et al., 2021a; Ho et al., 2020; Kadkhodaie and Simoncelli, 2020). In fact, the diffusion scores implicitly capture a density model of the data (Song et al., 2021b; Kingma et al., 2021): the relative energy levels between modes, to which score matching at $t = 0$ is blind (Zhang et al., 2022), are encoded in the score at the time t when they merge (Raya and Ambrogioni, 2023; Biroli et al., 2024). This implicit density model can be evaluated in various ways, which all involve a tractable but costly integration of score divergences (or denoising errors) *as a function of time* (Song et al., 2021a; Kong et al., 2023; Skreta et al., 2024; Karczewski et al., 2025). See Appendix A.2 for a more in-depth exposition.

Intuitively, diffusion models deal with multimodal distributions by providing a high-probability path between modes through space *and time*, as visualized in the right panel of Figure 1. In other words,

the *joint* distribution $p(y,t)$ qualitatively has a connected support. This suggests that we may learn the joint "space-time" energy $U(y,t)$ by score matching on (y,t). Specifically, the "space score" $-\nabla_y U_\theta(y,t)$ can be learned using a denoising objective, and the "time score" $-\partial_t U_\theta(y,t)$ (Choi et al., 2022) can be learned using an analogous score matching objective. Matching *both* space and time scores correctly constrains the energy levels across modes: the energy difference between two modes can be explicitly recovered by integrating the energy derivative along the red path in Figure 1 (although we will not need to do so explicitly).

2.2 Dual score matching: Objective function

Space and time score matching. We aim to learn a time-dependent energy function, $U_\theta(y,t)$, to approximate the NLL of the noisy image distribution at all noise levels t:

$$U(y,t) = -\log\left(\int p(x)\mathrm{e}^{-\frac{1}{2t}\|x-y\|^2-\frac{d}{2}\log(2\pi t)}\mathrm{d}x\right). \tag{1}$$

Differentiating this NLL with respect to y gives the (negative) "space score", which can be expressed using the Miyasawa-Tweedie identity (Robbins, 1956; Miyasawa, 1961) as

$$\nabla_y U(y,t) = \mathbb{E}_x\left[\frac{y-x}{t}\,\middle|\, y\right]. \tag{2}$$

This leads to the denoising score matching objective (Vincent, 2011; Raphan and Simoncelli, 2011; Saremi et al., 2018) used to train diffusion models:

$$\ell_{\mathrm{DSM}}(\theta,t) = \mathbb{E}_{x,y}\left[\left\|\nabla_y U_\theta(y,t) - \frac{y-x}{t}\right\|^2\right]. \tag{3}$$

Differentiating the energy with respect to t yields a (negative) "time score" (Choi et al., 2022), for which we can derive a similar identity (see Appendix B.1):

$$\partial_t U(y,t) = \mathbb{E}_x\left[\frac{d}{2t} - \frac{\|y-x\|^2}{2t^2}\,\middle|\, y\right]. \tag{4}$$

This leads to an analogous "time score matching" objective:

$$\ell_{\mathrm{TSM}}(\theta,t) = \mathbb{E}_{x,y}\left[\left(\partial_t U_\theta(y,t) - \frac{d}{2t} + \frac{\|y-x\|^2}{2t^2}\right)^2\right]. \tag{5}$$

Combining objectives across noise levels. The two objectives in eqs. (3) and (5) are defined for a given noise level t. We form an overall objective by integrating these over t, with an appropriate weighting. For the denoising score matching objective, a natural choice is the so-called *maximum-likelihood weighting* (Song et al., 2021b; Kingma et al., 2021), which provides a bound on the KL divergence with the data distribution:

$$\mathrm{KL}(p\,\|\,\tilde{p}_\theta) \le \frac{1}{2}\int_0^\infty \ell_{\mathrm{DSM}}(\theta,t)\,\mathrm{d}t, \tag{6}$$

where $\tilde{p}_\theta$ is the implicit density of the generative diffusion model. This integral can be approximated by Monte-Carlo sampling from a distribution of noise levels. In practice, we have found it best to sample $\log t$ uniformly over a finite interval (specifically, $p(t) \propto 1/t, t \in [t_{\min}, t_{\max}]$), which corresponds to the following implementation of the integral:

$$\int_{t_{\min}}^{t_{\max}} \ell_{\mathrm{DSM}}(\theta,t)\,\mathrm{d}t = \mathbb{E}_t[t\,\ell_{\mathrm{DSM}}(\theta,t)]. \tag{7}$$

Note that the resulting term in the expected value is unit-less: it is invariant to simultaneous rescaling of the data and noise level. We thus choose to weight the time score matching objective in eq. (5) by t^2, so that it is also unit-less, and to evaluate it over the same distribution $p(t)$. Finally, after appropriate normalization by the dimensionality d that ensures that the two objectives have comparable orders of magnitude, we simply add them (we found no significant improvement from tuning a tradeoff hyperparameter). In summary, our *dual score matching objective* is:

$$\ell(\theta) = \mathbb{E}_t\left[\frac{t}{d}\,\ell_{\mathrm{DSM}}(\theta,t) + \left(\frac{t}{d}\right)^2 \ell_{\mathrm{TSM}}(\theta,t)\right]. \tag{8}$$

Normalization. Since we match only the partial derivatives of $U_\theta(y,t)$ to those of the true energy $U(y,t)$, we only recover the energy up to a *global* constant: at the end of training, $U_\theta(y,t) \approx U(y,t) + \text{const}$. Note again that this crucially relies on $p(y,t)$ having a connected support. An important aspect of our framework is that it enables estimation of this constant, which determines the normalization of $\mathrm{e}^{-U_\theta(y,t)}$. Indeed, the time score objective ensures that this normalizing constant does not depend on time: *mass is conserved through the diffusion*. Since the true distribution is approximately Gaussian at large noise levels, $p(y|t_{\max}) \approx \mathcal{N}(0, t_{\max}\mathrm{Id})$, we can set this constant to the entropy of this Gaussian distribution:

$$U_\theta(y,t) \longrightarrow U_\theta(y,t) - \mathbb{E}_y[U_\theta(y,t) \,|\, t = t_{\max}] + \frac{d}{2}\log(2\pi\mathrm{e}t_{\max}). \tag{9}$$

Figure 1 provides a high-dimensional numerical example verifying that dual score matching indeed provides an accurate estimate of *normalized* energy values (in particular, for $t = 0$), in contrast to single score matching.

Related approaches. Similar combinations of space and time scores were considered by Choi et al. (2022) (who referred to it as the "pathwise" method) and Kobler and Pock (2023). These time score objectives, however, relied on a second derivative in time instead of a regression objective. Our time score objective is also a special case of the conditional time score matching objective developed concurrently in Yu et al. (2025). Finally, Yadin et al. (2024) train an energy model with an objective combining score matching with a classification cross-entropy loss for estimating a discretized version of the noise level t.

2.3 Dual score matching: Architecture

How should one choose an architecture to compute the energy $U_\theta(y,t)$? Rather than designing one from scratch (Salimans and Ho, 2021; Cohen et al., 2021; Thiry and Guth, 2024), we construct one by modifying a score-based denoising architecture that is known to have appropriate inductive biases. Let $s_\theta \colon \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}^d$ be such an architecture (e.g., a UNet). We wish to define a new energy architecture $U_\theta \colon \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}$ such that $\nabla_y U_\theta \approx s_\theta$, preserving the inductive biases of s_θ. To achieve this, we set

$$U_\theta(y,t) = \frac{1}{2}\langle y, s_\theta(y,t)\rangle. \tag{10}$$

This inner product parameterization has been concurrently proposed by Thornton et al. (2025). We show in Appendix B.2 that $\nabla_y U_\theta(y,t) = s_\theta(y,t)$ if the score network s_θ is both conservative and *homogeneous* (Romano et al., 2017; Reehorst and Schniter, 2018). Homogeneity has been shown to hold approximately in the related setting of blind denoisers (Mohan* et al., 2020; Herbreteau et al., 2023), and can be enforced architecturally. Note that the seemingly similar choice of a squared norm architecture $U_\theta(y,t) = \frac{1}{2}\|s_\theta(y,t)\|^2$ used in some previous energy models (Salimans and Ho, 2021; Hurault et al., 2021; Du et al., 2023; Yadin et al., 2024; Thiry and Guth, 2024) leads to the same desirable homogeneity properties in y *but not in* s_θ, and thus fails to preserve the optimization properties of the original score network. Further architectural details are provided in Appendix C.1.

2.4 Performance evaluation

Denoising performance. We first verify that the gradient of the proposed energy network, $\nabla_y U_\theta$, provides as good a denoiser as the score network s_θ on which it is based. We train the two networks separately, U_θ with the dual score matching objective (eq. (8) using double back-propagation), and s_θ with the standard (denoising) score matching objective (eq. (3)). Further details are provided in Appendix C.2. Table 1 compares denoising performance across noise levels. For all but the smallest noise levels, the energy-based model achieves (slightly) better denoising performance than the score model. Thus, there is no penalty in modeling the energy rather than the score. This is contrary to the results of Salimans and Ho (2021), and is likely due to our use of an architecture that is homogeneous, and for which non-conservativeness of the score is not a large source of denoising error (Mohan* et al., 2020; Chao et al., 2023). Finally, these results demonstrate that the two components of our dual score matching objective are not trading off against each other; rather, they complement and even *reinforce* each other (i.e. their minima coincide).

Table 1: Denoising performance at several noise levels for corresponding score and energy networks, averaged over images in ImageNet64, with intensities scaled to lie in $[0, 1]$. All quantities are expressed as peak signal-to-noise ratio (PSNR) in dB: $\mathrm{PSNR} = -10 \log_{10}(\mathrm{MSE})$.

Noise variance	90	75	60	45	30	15	0	−15	−30
Score network	**90.20**	**75.47**	60.43	47.19	35.58	25.92	18.84	13.31	−0.11
Energy network	90.09	75.17	**60.45**	**47.25**	**35.67**	**26.01**	**18.88**	**13.48**	**2.53**

Table 2: Negative log likelihood (in bits/dimension) on ImageNet64 test set. See Appendix A.1 for more details.

Method	Anti-aliasing	Augmentation	Discreteness	Type	Single NFE	NLL
Glow (Kingma and Dhariwal, 2018)	✗	None	Continuous	Normalized	✓	3.81
PixelCNN (Van den Oord et al., 2016)	✗	None	Discrete	Normalized	✗	3.57
I-DDPM (Nichol and Dhariwal, 2021)	✗	None	Continuous	Upper bound	✗	3.54
VDM (Kingma et al., 2021)	✗	None	Discrete	Upper bound	✗	3.40
FM (Lipman et al., 2023)	✓	None	Uniform*	Normalized	✗	3.31
NFDM (Bartosh et al., 2024)	✗	Horizontal flips	Uniform*	Normalized	✗	3.20
TarFlow (Zhai et al., 2024)	✗	Horizontal flips	Uniform	Normalized	✓	2.99
Ours	✓	Horizontal flips	Continuous	Estimate	✓	3.36

Negative log likelihood. How accurate are our energy estimates? A standard evaluation of probabilistic models consists of estimating the cross-entropy $\mathbb{E}_x[-\log p_\theta(x)]$ between the data distribution and the model, also known as negative log likelihood (NLL). In practice, NLL is computed by computing the probability the model assigns to a held-out test set. There are however several subtleties that make direct model comparisons challenging. Specifically, there are 3 factors that can cause variations in NLL of ± 0.5 bits/dimension or more: details of data pre-processing (e.g., downsampling or data augmentation), conversion method from continuous to discrete probability, and the estimator type (exactly normalized, variational bound, or approximately normalized). We expand on these issues in Appendix A.1.

We compare NLLs of our method and a variety of recent energy models in Table 2. This evaluation demonstrates that our model is comparable to the best-performing models in the literature, within the variability arising from the three factors of variation mentioned in the previous paragraph. Two unique advantages of our method are that it provides (1) direct (one-shot) estimates of energy (as opposed to other density estimation approaches used with diffusion models, which we review in Appendix A.2), and (2) access to energy across all noise levels, providing a window into larger-scale features of the energy landscape. For instance, our energy network can compute the probability of 50k images in 12s on an A100 GPU, whereas the score network requires upwards of 3h20, even with as few as 100 times steps and 10 noise samples to compute the energy (see Appendix A.2). This justifies the longer training time due to the cost of double-backpropagation (training for 1M steps on ImageNet64 on a single A100 GPU respectively took 120 hours, compared to 32 hours for the score network), which may be alleviated by the use of sliced score matching (Song et al., 2020).

3 Analysis of the learned energy-based model

3.1 Generalization

The previous section demonstrated that our energy-based model achieves near state-of-the-art NLL on ImageNet64. That is, the model *on average* assigns high probability to a set of held-out test images. Next, we establish that the energies of the *individual* images are reliable. In particular, we verify that the model's energy assignment is stable under change of the training data.

To this end, we use the strong generalization test developed in Kadkhodaie et al. (2024). We partition the training data into two non-overlapping sets, train a separate energy-based model on each set, and then compare the energies computed by these two models on images from both training subsets. We gradually increase the size of each training set until the two models assign approximately equal log probability across all images. Figure 2 shows the results of this experiment. The two models assign very different probabilities to the same image when the training set size, N, is small. But

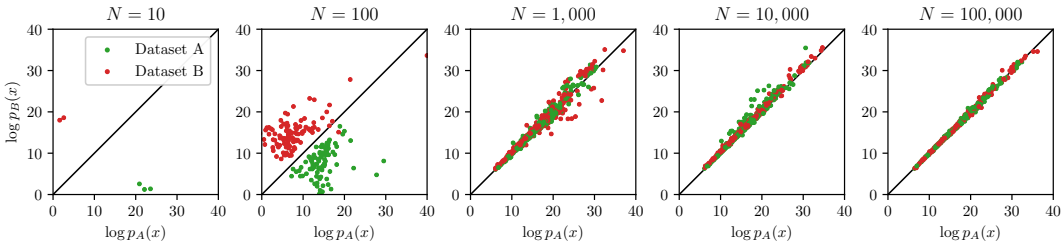

Figure 2: Convergence of energy estimates. The data set is split into two halves (denoted A and B), and separate energy models are trained on N samples drawn from each half. Each scatterplot compares the energy estimates of the two models at $t = 0$, over all $2N$ training images. As N increases, the energy estimates of the two models converges for all images.

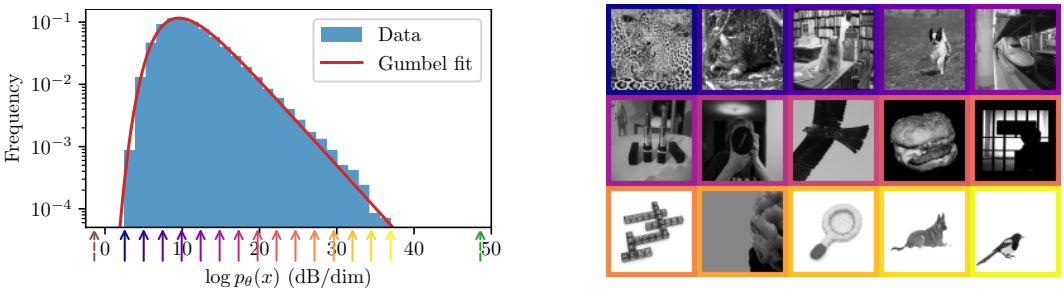

Figure 3: Histogram of log probabilities of test images in the ImageNet dataset. Color-coded arrows indicate values for the example images on the right, and the leftmost (brown) and rightmost (green) arrows indicate values for a uniform noise image in $[0, 1]$ and a constant image of intensity 0.5, respectively. The distribution is well-fit by a Gumbel distribution (red line). Additional example images organized by probability are provided in Figures 6 to 8 (Appendix D.1).

they converge gradually and compute nearly the same values at $N = 10^5$. Note that the rate of convergence depends on image probability: more data is needed before the two models agree on the high-probability images. It is also worth noting that the transition from memorization to generalization of energy models is marked by a large increase of variance in energy over the training set (starting at $N = 100$).

This result establishes that the model variance vanishes with a feasible training set size. However, it does not guarantee that the values are accurate—the models could be biased. Direct calculation of model bias requires access to the true density, which is only available for synthetic data (as in Figure 1). However, the small NLL values over test data (Table 2) do suggest a small model bias.

3.2 Distribution of energies and relationship to image content

We now study properties of the learned energy model. What is the entropy (average energy) of the image dataset? Do all images have the same probability? If not, what determines the probability of an image? To investigate these questions, we compute the log probability of all $50,000$ images in the ImageNet64 test set. We express log probability in units of decibels per dimension (dB/dim), computed as $\frac{1}{d} 10 \log_{10} p_\theta(x)$, with $d = 64 \times 64 = 4096$. In this scale, an increase of 10 dB/dim corresponds to a multiplicative rescaling of the probability density by a factor of 10^d.

Entropy. The average value of $-\log p_\theta(x)$ provides an estimate of -11.4 dB/dim for the differential entropy of ImageNet. Note that the uniform distribution over the hypercube $[0, 1]^d$ has an entropy of 0. This indicates that natural images occupy a fractional volume of about $10^{-1.14d}$. This can be converted to an estimate of discrete entropy by assuming that log probability is constant within quantization bins. For 8-bit images (with 256 possible intensity values), this corresponds to an entropy of 4.20 bits/dimension (the deviation from the value in Table 2 is due to the use of grayscale images here). In other words, there are $\sim 10^{5,180}$ quantized ImageNet images out of $10^{9,860}$ possible images at this resolution.

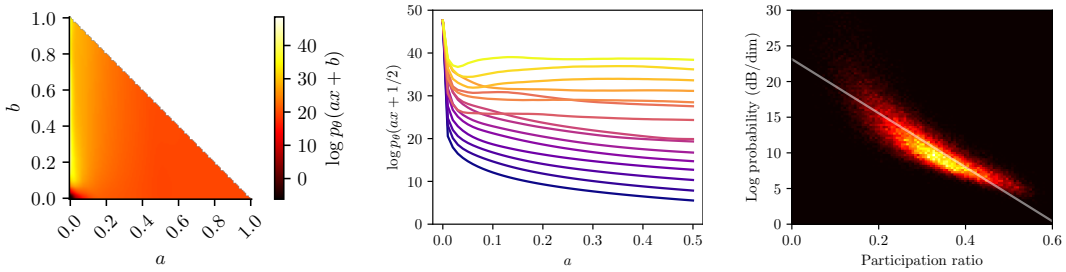

Figure 4: Influence of image statistics on probability. **Left.** $\log p_\theta(ax + b)$ as a function of a and b. **Middle.** Horizontal slice ($b = \frac{1}{2}$) of the left panel for the example images of Figure 3. **Right.** Log probability as a function of sparsity, measured as the participation ratio of wavelet coefficients.

Lack of concentration. Many high-dimensional probability distributions exhibit a *concentration* phenomenon: typical realizations have nearly equal energy. For instance, in simple image probability models such as a Gaussian model or a sparse wavelet model (where wavelet coefficients are independent), the energy of independent components add, and the law of large numbers imply a concentration of the energy around its mean. In contrast, our energy network reveals enormous diversity in the log probabilities of individual ImageNet images (Figure 3, left). Over the entire test set, they span a range of 34.4 dB/dim, corresponding to a probability ratio of $\approx 10^{14,000}$. Given this enormous ratio, it is surprising that any low probability image appears in a finite dataset. To make sense of this apparent contradiction, it is important to realize that the total probability mass of images with a given value of $\log p$ in a dataset is equal to this probability value *multiplied by the volume of the corresponding probability level set*. Observing a range of probability values of $10^{14,000}$ thus reveals that these enormous variations in probability density must be nearly compensated for by inverse variations in the volume of their corresponding level sets.

Shape of the distribution of log probabilities. The distribution of log probability values is highly skewed (Figure 3, left): there is a heavy tail of high-probability images and a much lighter tail of low-probability images. We also observed these qualitative features on the CelebA and CIFAR-10 datasets, shown in Appendix D.2. Surprisingly, this distribution is well approximated by a Gumbel distribution (we report parameter values in Appendix C.3), which arises as a limit distribution for the maximum of many i.i.d. random variables. This surprising observation is, to the best of our knowledge, novel, and calls for an explanation. In independent component models, the log probabilities are sums of i.i.d. random variables, and are therefore Gaussian-distributed in high dimensions. In contrast, a simple model that leads to a Gumbel distribution is a high-dimensional spherically-symmetric distribution with an exponentially-distributed radial marginal (Lyu and Simoncelli, 2009).

Image content. We also show in Figure 3 (right) example images with a variety of log probabilities. High-probability images contain small objects, on a blank (often white) background. Conversely, low-probability images are filled with dense detailed texture. Since the energy of an image, when expressed in bits, corresponds to the size of its optimally compressed representation, this implies that higher-probability images have shorter codes and thus less "information content". We show in Appendix D.4 images that have been obtained by locally maximizing or minimizing probability.

Intensity range and sparsity. Following these visual observations, we further examine the influence of intensity range and sparsity on image probability (Figure 4). First, we evaluate the log probability of reference images as we manipulate their brightness and contrast through an affine operation. We find that the brightness has minimal effect on the probability, while higher-contrast images have lower probability. This reveals that the distribution support is *star-shaped*: any pair of images are connected by a high-probability path passing through a constant image. In contrast, we show in Appendix D.3 that linear interpolation between pairs of images typically increase probability: the distribution support is not convex. Next, we verify that log probability is correlated with a simple measure of sparsity of images. We compute a multi-scale wavelet decomposition of the images, and measure the ℓ^1-norm divided by the ℓ^2-norm of the coefficients. The square of this quantity, $\|x\|_1^2/d\|x\|_2^2 \in (0, 1]$ is known as the participation ratio, with smaller values indicating higher sparsity. The right panel of Figure 4 shows that this simple measure captures a significant portion of the variance of $\log p_\theta(x)$.

3.3 Effective dimensionality of the energy landscape

Beyond estimating log probability for a given image, we aim to characterize the local behavior of the density in the vicinity of that image. For instance, we would like to assess whether the probability is locally concentrated near a low-dimensional "tangent" subspace, and if so, to estimate its dimensionality. In this section, we explain how these quantities may be computed from our energy model. We then show that the local dimensionality around an image is often much lower than the ambient dimensionality of the space, but that this dimensionality depends on both the particular image and the scale that is used to define the local neighborhood.

Multi-scale dimensionality. Consider hypothetical distributions supported on the blue regions in the left two panels of Figure 5. For the left example, the effective dimensionality of a local neighborhood decreases with the size of that neighborhood. The opposite behavior is also possible, as shown in the right example. These examples demonstrate that dimensionality measures which aim to describe these geometrical structures need to depend on both the location and the scale of the neighborhood (Mohan* et al., 2020; Tempczyk et al., 2022).

Effective dimensionality. We now introduce an effective dimensionality measure which can be equivalently defined from the optimal denoiser or the evolution of the probability landscape as it diffuses. Given a noisy observation y of a clean image x with noise variance t, the optimal denoiser estimates x with the conditional expectation $\mathbb{E}[x \mid y]$. Its average deviation from x capture the local support of the data distribution around x at scale t. Intuitively, from observing y, the optimal denoiser identifies that x is located on a d_{eff}-dimensional "tangent" space and projects y onto it. This preserves the components of the noise that lie along the subspace, incurring a denoising error td_{eff}. Thus, we define the local effective dimensionality around x at scale t as

$$d_{\text{eff}}(x,t) = \frac{1}{t}\mathbb{E}_y\Big[\|x - \mathbb{E}_x[x \mid y]\|^2 \,\Big|\, x\Big]. \tag{11}$$

Effective dimensionality can be equivalently defined directly from the energy. Consider the forward diffusion which progressively adds more noise to an image x, blurring the probability landscape. At time t, we can define an effective energy $\mathbb{E}_y[U(y,t) \mid x]$. Its rate of change with t captures the effective dimensionality around x. Intuitively, if this landscape is locally a d_{eff}-dimensional subspace, then adding more noise causes probability to diffuse in the $d - d_{\text{eff}}$ normal "off-manifold" directions, spreading over a volume $\sim t^{(d-d_{\text{eff}})/2}$. Taking the logarithm, we obtain $\mathbb{E}_y[U(y,t) \mid x] \sim (d - d_{\text{eff}})\frac{1}{2}\log t$. Thus, we can equivalently define

$$d_{\text{eff}}(x,t) = d - \partial_{\frac{1}{2}\log t}\mathbb{E}_y[U(y,t) \mid x]. \tag{12}$$

We show in Appendix B.3 that the definitions in eqs. (11) and (12) are equivalent, unifying two seemingly distinct points of view underlying these dimensionality measures (Wu and Verdú, 2011; Mohan* et al., 2020; Tempczyk et al., 2022; Stanczuk et al., 2022; Kamkari et al., 2024; Horvat and Pfister, 2024).

Numerical results. We show in Figure 5 the behavior of the effective energy and dimensionality as a function of the noise level. As the noise level increases, the log probability initially remains constant, indicating that probability is uniformly spread in a ball around x. The probability eventually decreases as the diffusion radius becomes larger than the local support size, and all lines converge to the mean (the negative entropy), consistent with the asymptotic Gaussian behavior of the energy. The effective dimensionality (fig. 5, right) vanishes at large noise levels (images have a compact support, which behaves like a point at sufficiently large noise levels) but increases as the noise level is reduced, eventually reaching the ambient dimensionality d (our model has a continuous non-zero density everywhere by construction). In-between these two extremes, there exists a sizable range of scales $t \in [10^{-9}, 10^{-2}]$ where almost the entire range of dimensionalities coexist. In particular, higher-probability images have lower-dimensional neighborhoods, even at relatively small scales, while lower-probability images have nearly full-dimensional neighborhoods, even at relatively large scales. This result empirically confirms the point raised in Section 3.2: the local probability mass around high and low probability images are of the same order, despite the enormous gap between their probability values. The local high dimensionality of the low-probability images corresponds to a significantly larger volume occupied by their local neighborhood. We note two potential limits to these estimates. First, the quantization of pixel values into bins of size $1/256$ limits the resolution

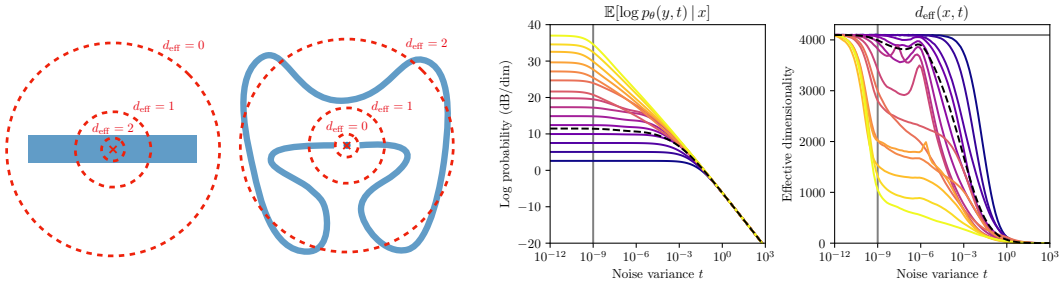

Figure 5: **Left:** Two hypothetical examples illustrating how local effective dimensionality depends on the scale of the neighborhood. For both examples, support of density corresponds to the blue regions. In the left example, the dimensionality around the red point decreases with scale (from 2 down to 0), while the opposite is true for the right example. **Right:** Log probability and effective dimensionality as a function of noise level. Colored lines correspond to different example images x (shown in right panel of Figure 3), while the dashed black line shows the average over the ImageNet test set. The vertical gray line indicates the minimum noise level presented during training ($t = 10^{-9}$), and the horizontal gray line the ambient dimensionality of the dataset ($d = 4096$).

to $t \sim 10^{-5}$, possibly explaining the momentary decrease in dimensionality as t decreases. Second, our energy model was only trained on noise levels down to $t_{\min} = 10^{-9}$. Thus, energies at smaller values of t are solely determined by model inductive biases, which favor a constant energy.

4 Discussion

We have developed a novel framework for estimating the log probability (energy) of a distribution of images from observed samples. The estimation method is simple and robust, and converges to a stable solution with relatively small amounts of data (in our examples, $100,000$ images of size 64×64 suffices). The framework leverages the tremendous power of generative diffusion models, relying on networks trained to estimate the score of the data distribution by minimizing a denoising objective. We augment this with a secondary objective that ensures consistency of the model across noise levels, and an architecture constructed from an existing denoiser so as to preserve its inductive biases. We validate our model by verifying that it achieves denoising performance equal to or surpassing the original denoiser, and NLL values comparable to the state of the art. We note that it is straightforward to extend our approach to conditional models: given conditioning information, the network needs only to compute the conditional energy function. The expression of losses ℓ_{DSM} and ℓ_{TSM} would be unchanged, as well as the energy architecture.

We mention two important limitations of our work. First, our energy model has a roughly quadrupled training time compared to the base score network, due to the use of double back-propagation. We believe this can be improved through architecture improvements, specialized auto-differentiation functions, or through the use of sliced score matching (Song et al., 2020). But in any case, the additional training time is offset by the dramatic efficiency gains in computing energy values. Another limitation of our approach is that we cannot provide any theoretical guarantees that minimizing our objective leads to a good approximation of the true energy. We expand on this question in Appendix A.3.

Our learned energy model provides new insights into the geometrical properties of the space of photographic images. Notably, we observe a lack of concentration of measure, with log probability values varying over a wide range and following a Gumbel distribution. As a limit distribution for the maximum, rather than the sum, of i.i.d. random variables, we expect that this might arise in a broad class of high-dimensional distributions. We also introduced a novel image- and scale-dependent measure of effective dimensionality which unifies two separate points of view developed in the literature. We demonstrate that different image neighborhoods display both high and low dimensionality over a wide range of scales. These results challenge simple interpretations of the manifold hypothesis. Moreover, the estimated dimensionalities remain too high to explain how the curse of dimensionality is avoided. This suggests that the energy landscape of natural images has additional undocumented geometrical regularity.

Acknowledgments

FG thanks Louis Thiry for starting this research direction (Thiry and Guth, 2024). We also thank Joan Bruna and Pierre-Étienne Fiquet for inspiring discussions. We gratefully acknowledge the support and computing resources of the Flatiron Institute (a research division of the Simons Foundation).

References

Grigory Bartosh, Dmitry P Vetrov, and Christian Andersson Naesseth. Neural flow diffusion models: Learnable forward process for improved diffusion modelling. *Advances in Neural Information Processing Systems*, 37:73952–73985, 2024.

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

Sagnik Bhattacharya, Abhiram R Gorle, Ahmed Mohsin, Ahsan Bilal, Connor Ding, Amit Kumar Singh Yadav, and Tsachy Weissman. ItDPDM: Information-theoretic discrete poisson diffusion model. *arXiv preprint arXiv:2505.05082*, 2025.

Giulio Biroli, Tony Bonnaire, Valentin De Bortoli, and Marc Mézard. Dynamical regimes of diffusion models. *Nature Communications*, 15(1):9957, 2024.

Joan Bruna and Jiequn Han. Provable posterior sampling with denoising oracles via tilted transport. *Advances in Neural Information Processing Systems*, 37:82863–82894, 2024.

Chen-Hao Chao, Wei-Fang Sun, Bo-Wun Cheng, and Chun-Yi Lee. On investigating the conservative property of score-based generative models. *International Conference on Machine Learning*, 2023.

Kristy Choi, Chenlin Meng, Yang Song, and Stefano Ermon. Density ratio estimation via infinitesimal classification. In *International Conference on Artificial Intelligence and Statistics*, pages 2552–2573. PMLR, 2022.

Patryk Chrabaszcz, Ilya Loshchilov, and Frank Hutter. A downsampled variant of imagenet as an alternative to the CIFAR datasets. *arXiv preprint arXiv:1707.08819*, 2017.

R Cohen, Y Blau, D Freedman, and E Rivlin. It has potential: Gradient-driven denoisers for convergent solutions to inverse problems. *Adv Neural Information Processing Systems (NeurIPS)*, 34, 2021.

Laurent Dinh, David Krueger, and Yoshua Bengio. NICE: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014.

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

Yilun Du, Conor Durkan, Robin Strudel, Joshua B Tenenbaum, Sander Dieleman, Rob Fergus, Jascha Sohl-Dickstein, Arnaud Doucet, and Will Sussman Grathwohl. Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and mcmc. In *International conference on machine learning*, pages 8489–8510. PMLR, 2023.

Lawrence C Evans. *Partial differential equations*, volume 19. American Mathematical Society, 1993.

Dongning Guo, Shlomo Shamai, Sergio Verdú, et al. The interplay between information and estimation measures. *Foundations and Trends® in Signal Processing*, 6(4):243–429, 2013.

Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 297–304. JMLR Workshop and Conference Proceedings, 2010.

Sébastien Herbreteau, Emmanuel Moebel, and Charles Kervrann. Normalization-equivariant neural networks with application to image denoising. *Advances in Neural Information Processing Systems*, 36:5706–5728, 2023.

Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

Geoffrey E Hinton, Terrence J Sejnowski, et al. Learning and relearning in Boltzmann machines. *Parallel distributed processing: Explorations in the microstructure of cognition*, 1(282-317):2, 1986.

J Ho, A Jain, and P Abbeel. Denoising diffusion probabilistic models. *Adv Neural Information Processing Systems (NeurIPS)*, 33, 2020.

Jonathan Ho, Xi Chen, Aravind Srinivas, Yan Duan, and Pieter Abbeel. Flow++: Improving flow-based generative models with variational dequantization and architecture design. In *International conference on machine learning*, pages 2722–2730. PMLR, 2019.

Christian Horvat and Jean-Pascal Pfister. On gauge freedom, conservativity and intrinsic dimensionality estimation in diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024.

Samuel Hurault, Arthur Leclaire, and Nicolas Papadakis. Gradient step denoiser for convergent plug-and-play. *arXiv preprint arXiv:2110.03220*, 2021.

Michael F Hutchinson. A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Communications in Statistics-Simulation and Computation*, 18(3):1059–1076, 1989.

Aapo Hyvärinen and Peter Dayan. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4), 2005.

Z Kadkhodaie and E P Simoncelli. Solving linear inverse problems using the prior implicit in a denoiser. *arXiv preprint arXiv:2007.13640*, Jul 2020.

Zahra Kadkhodaie, Florentin Guth, Eero P Simoncelli, and Stéphane Mallat. Generalization in diffusion models arises from geometry-adaptive harmonic representations. In *The Twelfth International Conference on Learning Representations*, 2024.

Hamid Kamkari, Brendan Ross, Rasa Hosseinzadeh, Jesse Cresswell, and Gabriel Loaiza-Ganem. A geometric view of data complexity: Efficient local intrinsic dimension estimation with diffusion models. *Advances in Neural Information Processing Systems*, 37:38307–38354, 2024.

Rafał Karczewski, Markus Heinonen, and Vikas Garg. Diffusion models as cartoonists! The curious case of high density regions. In *The Thirteenth International Conference on Learning Representations*, 2025.

Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *Advances in neural information processing systems*, 34:21696–21707, 2021.

Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018.

Erich Kobler and Thomas Pock. Learning gradually non-convex image priors using score matching. *arXiv preprint arXiv:2302.10502*, 2023.

Frederic Koehler, Alexander Heckett, and Andrej Risteski. Statistical efficiency of score matching: The view from isoperimetry. *ICLR*, 2023.

Xianghao Kong, Rob Brekelmans, and Greg Ver Steeg. Information-theoretic diffusion. In *The Eleventh International Conference on Learning Representations*, 2023.

Chieh-Hsin Lai, Yuhta Takida, Naoki Murata, Toshimitsu Uesaka, Yuki Mitsufuji, and Stefano Ermon. FP-diffusion: Improving score-based diffusion models by enforcing the underlying score fokker-planck equation. In *International Conference on Machine Learning*, pages 18365–18398. PMLR, 2023.

Yann LeCun and Fu Jie Huang. Loss functions for discriminative training of energy-based models. In *International workshop on artificial intelligence and statistics*, pages 206–213. PMLR, 2005.

Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, Fujie Huang, et al. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023.

S Lyu and E P Simoncelli. Nonlinear extraction of 'independent components' of natural images using radial Gaussianization. *Neural Computation*, 21(6):1485–1519, Jun 2009. doi: 10.1162/neco.2009.04-08-773.

K Miyasawa. An empirical Bayes estimator of the mean of a normal population. *Bull. Inst. Internat. Statist.*, 38:181–188, 1961.

S Mohan*, Z Kadkhodaie*, E P Simoncelli, and C Fernandez-Granda. Robust and interpretable blind image denoising via bias-free convolutional neural networks. In *Int'l Conf on Learning Representations (ICLR)*, Addis Ababa, Ethiopia, Apr 2020.

Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Gorur, and Balaji Lakshminarayanan. Do deep generative models know what they don't know? In *International Conference on Learning Representations*, 2019.

Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR, 2021.

M Raphan and E P Simoncelli. Least squares estimation without priors or supervision. *Neural Computation*, 23(2):374–420, Feb 2011. doi: 10.1162/NECO_a_00076.

Gabriel Raya and Luca Ambrogioni. Spontaneous symmetry breaking in generative diffusion models. *Advances in Neural Information Processing Systems*, 36:66377–66389, 2023.

Edward T Reehorst and Philip Schniter. Regularization by denoising: Clarifications and new interpretations. *IEEE transactions on computational imaging*, 5(1):52–67, 2018.

D Rezende and S Mohamed. Variational inference with normalizing flows. In *Int'l Conf on Machine Learning (ICML)*, pages 1530–1538. PMLR, 2015.

H Robbins. An empirical bayes approach to statistics. In *Proc Third Berkeley Symposium on Mathematical Statistics and Probability*, volume I, pages 157–163. University of CA Press, 1956.

Yaniv Romano, Michael Elad, and Peyman Milanfar. The little engine that could: Regularization by denoising (red). *SIAM Journal on Imaging Sciences*, 10(4):1804–1844, 2017.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015.

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

Saeed Saremi, Arash Mehrjou, Bernhard Schölkopf, and Aapo Hyvärinen. Deep energy estimator networks. *arXiv preprint arXiv:1805.08306*, 2018.

Marta Skreta, Lazar Atanackovic, Joey Bose, Alexander Tong, and Kirill Neklyudov. The superposition of diffusion models using the Itô density estimator. In *The Thirteenth International Conference on Learning Representations*, 2024.

J Sohl-Dickstein, E Weiss, N Maheswaranathan, and S Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In Francis Bach and David Blei, editors, *Proc 32nd Int'l Conf on Machine Learning (ICML)*, volume 37 of *Proceedings of Machine Learning Research*, pages 2256–2265, Lille, France, 07–09 Jul 2015. PMLR.

Y Song and S Ermon. Generative modeling by estimating gradients of the data distribution. *Adv Neural Information Processing Systems (NeurIPS)*, 32, 2019.

Y Song, J Sohl-Dickstein, D P Kingma, A Kumar, S Ermon, and B Poole. Score-based generative modeling through stochastic differential equations. In *Int'l Conf on Learning Representations (ICLR)*, 2021a.

Yang Song and Diederik P Kingma. How to train your energy-based models. *arXiv preprint arXiv:2101.03288*, 2021.

Yang Song, Sahaj Garg, Jiaxin Shi, and Stefano Ermon. Sliced score matching: A scalable approach to density and score estimation. In *Uncertainty in artificial intelligence*, pages 574–584. PMLR, 2020.

Yang Song, Conor Durkan, Iain Murray, and Stefano Ermon. Maximum likelihood training of score-based diffusion models. *Advances in Neural Information Processing Systems*, 34:1415–1428, 2021b.

Jan Stanczuk, Georgios Batzolis, Teo Deveney, and Carola-Bibiane Schönlieb. Your diffusion model secretly knows the dimension of the data manifold. *arXiv preprint arXiv:2212.12611*, 2022.

Piotr Tempczyk, Rafał Michaluk, Lukasz Garncarek, Przemysław Spurek, Jacek Tabor, and Adam Golinski. LIDL: Local intrinsic dimension estimation using approximate likelihood. In *International Conference on Machine Learning*, pages 21205–21231. PMLR, 2022.

Joshua B Tenenbaum, Vin de Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *science*, 290(5500):2319–2323, 2000.

L Theis, A van den Oord, and M Bethge. A note on the evaluation of generative models. In *International Conference on Learning Representations (ICLR 2016)*, pages 1–10, 2016.

Louis Thiry and Florentin Guth. Classification-denoising networks. *arXiv preprint arXiv:2410.03505*, 2024.

James Thornton, Louis Béthune, Ruixiang ZHANG, Arwen Bradley, Preetum Nakkiran, and Shuangfei Zhai. Composition and control with distilled energy diffusion models and sequential monte carlo. In *International Conference on Artificial Intelligence and Statistics*, pages 3259–3267. PMLR, 2025.

Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

Aaron Van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al. Conditional image generation with PixelCNN decoders. *Advances in neural information processing systems*, 29, 2016.

Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

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

Martin J Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge university press, 2019.

Yihong Wu and Sergio Verdú. MMSE dimension. *IEEE Transactions on Information Theory*, 57(8): 4857–4879, 2011.

Shahar Yadin, Noam Elata, and Tomer Michaeli. Classification diffusion models. *arXiv preprint arXiv:2402.10095*, 2024.

Jonathan S Yedidia, William T Freeman, and Yair Weiss. Constructing free-energy approximations and generalized belief propagation algorithms. *IEEE Transactions on information theory*, 51(7): 2282–2312, 2005.

Hanlin Yu, Arto Klami, Aapo Hyvarinen, Anna Korba, and Omar Chehab. Density ratio estimation with conditional probability paths. In *Forty-second International Conference on Machine Learning*, 2025.

Shuangfei Zhai, Ruixiang Zhang, Preetum Nakkiran, David Berthelot, Jiatao Gu, Huangjie Zheng, Tianrong Chen, Miguel Angel Bautista, Navdeep Jaitly, and Josh Susskind. Normalizing flows are capable generative models. *arXiv preprint arXiv:2412.06329*, 2024.

Mingtian Zhang, Oscar Key, Peter Hayes, David Barber, Brooks Paige, and Francois-Xavier Briol. Towards healing the blindness of score matching. In *NeurIPS 2022 Workshop on Score-Based Methods*, 2022.

Kaiwen Zheng, Cheng Lu, Jianfei Chen, and Jun Zhu. Improved techniques for maximum likelihood estimation for diffusion ODEs. In *International Conference on Machine Learning*, pages 42363–42389. PMLR, 2023.

A Discussions on density estimation

A.1 On comparison of NLL values

We identify three issues that arise in estimation of NLL values, each of which limits the precision with which they can be compared.

A first issue concerns the selection and pre-processing of the training data. There are two different versions of ImageNet at 64×64 resolution: one introduced in Van den Oord et al. (2016), no longer available, and a second which uses anti-aliasing during downsampling (reducing entropy and thus NLL), introduced in Chrabaszcz et al. (2017). This change in the dataset accounts for variations of about 0.3 bits/dimension (Zheng et al., 2023). Similarly, the data augmentations used can both increase entropy (e.g., random flips) or reduce it (e.g., center crops, or resizing operations with anti-aliasing).

A second point is that NLL estimates must be computed on a discrete probability model, and NLL estimates for continuous models are dependent on the method used to discretize the distribution. The simplest option is to assume that the probability is uniform within quantization bins. Discrete probabilities are then computed as a product of the continuous probability with the volume of the quantization bin, which corresponds to an additive shift of 8 bits in the NLL for image data. This can be enforced by adding uniform noise to the data, a technique known as "uniform dequantization" (Ho et al., 2019), which leads to an upper bound on the NLL of the corresponding discrete model (Theis et al., 2016). These differences can account for variations of 0.1 to 0.5 bits/dimension (Kong et al., 2023). A more sophisticated variational dequantization procedure can improve results by an additional 0.1 bits/dimension (Ho et al., 2019; Song et al., 2021b). In contrast, directly modeling the discrete data with appropriate methods can lead to reductions in NLL of more than 2 bits/dimension (Bhattacharya et al., 2025).

Finally, a third point is that depending on the method, the NLL should be interpreted differently. Classically, p_θ is a normalized probability distribution, typically obtained through a change-of-variable formula (as in flow-based models), so that up to an unknown additive constant (the entropy of the data), the NLL directly evaluates the KL divergence $\mathrm{KL}(p \,\|\, p_\theta)$. For some other models (such as VAEs), the NLL is not directly tied to a probabilistic model and is instead evaluated through a variational lower bound, leading to upper bounds on $-\log p(x)$ for each x. The NLL however remains an upper bound on the entropy of the data. The MSE-based formulas of Kong et al. (2023) also fall in this category when using a (necessarily suboptimal) network denoiser. Lastly, approximately normalized energy-based models such as ours or CDM (Yadin et al., 2024) compute estimates of $-\log p(x)$ that are neither lower nor upper bounds, so that NLL can only be interpreted as an estimate of the entropy of the data.

As a result of these differences, NLL values of different methods are often not directly comparable. Here, we aim to learn an accurate *continuous* probability model of image distributions rather than obtaining the best upper bound on the *discrete* entropy of the data.

When evaluating previously published NLL results, we found that these details were often not provided, and we thus had to make assumptions in compiling Table 2. The "anti-aliasing" column refers to the version of ImageNet64 used, we assumed that articles that cite Van den Oord et al. (2016) for the dataset made use of the aliased version. We assume no data augmentation if none is mentioned (for TarFlow (Zhai et al., 2024), the authors mention performing center crops of the data, but an inspection of their code indicates that it has been disabled for ImageNet64). The "discreteness" column refers to the nature of the probability model and the potential conversion from continuous to discrete probabilities ("discrete" for discrete probability models, "continuous" for an additive shift of 8 bits, and "uniform" for uniform dequantization by adding uniform noise). We assume a continuous model if no dequantization is mentioned (FM (Lipman et al., 2023) and NFDM (Bartosh et al., 2024) use a slightly different notion of uniform dequantization which improves NLL by 0.04 bits/dimension). The "type" column refers to the nature of the LLM estimate ("normalized" for an exactly normalized probability model coming from a flow-based model, "upper bound" for variational lower bounds on log probability, and "estimate" for other approaches). Finally, we indicate those methods that use a single neural function evaluation (NFE) to compute log probability in the "single NFE" column.

A.2 Background on density computation in diffusion models

We review several methods of estimating log probabilities from diffusion models that have appeared in the literature.

In early publications on diffusion methods, probabilities are computed with the so-called probability flow ODE (Song et al., 2021a). This corresponds to the distribution of samples generated by the backward ODE. Given a large noise level $t_{\max}$, the backward ODE solves the equation

$$-\frac{\mathrm{d}x_t}{\mathrm{d}t} = -\frac{1}{2}\nabla_y U_\theta(x_t, t), \tag{13}$$

backwards in time from $x_{t_{\max}} \sim \mathcal{N}(0, t_{\max}\mathrm{Id})$ and produces an approximate sample $x = x_0 \sim p_{\mathrm{ODE}}$. The log probability of this sample can be calculated as (Song et al., 2021a)

$$-\log p_{\mathrm{ODE}}(x) = \frac{\|x_{t_{\max}}\|^2}{2t_{\max}} + \frac{d}{2}\log(2\pi t_{\max}) - \frac{1}{2}\int_0^{t_{\max}} \Delta_y U_\theta(x_t, t)\mathrm{d}t. \tag{14}$$

Note that eq. (14) is also valid for arbitrary test points x, in which case the ODE (13) needs to be solved *forward* in time from $x_0 = x$ at $t = 0$ to $t = t_{\max}$. Equation (14) requires estimating the divergence of the score (the Laplacian of the energy), which is typically approximated with the Hutchinson trace estimator (Hutchinson, 1989).

Another approach is to use a variational bound as in Song et al. (2021b); Kingma et al. (2021); Kong et al. (2023). This variational bound arises from the exact identity (Kong et al., 2023)

$$-\log p(x) = \mathbb{E}_y[U(y, t_{\max}) \,|\, x] - \int_0^{t_{\max}} \left(td - \mathbb{E}_y\left[\|x - \mathbb{E}_x[x \,|\, y]\|^2 \,\middle|\, x\right]\right)\frac{\mathrm{d}t}{2t^2}. \tag{15}$$

The first term is the effective energy at $t = t_{\max}$, which is equivalent to $\frac{d}{2}\log(2\pi e t_{\max})$ as $t_{\max} \to \infty$. The second term features the optimal mean squared error over noise levels $t \in [0, t_{\max}]$. Note that the integrand can be rewritten as $(d - d_{\mathrm{eff}}(x,t))\mathrm{d}\left(\frac{1}{2}\log t\right)$, and eq. (15) can thus be derived by integrating the two equivalent definitions of effective dimensionality (11) and (12) (see Appendix B.3). Equation (15) naturally leads to an upper-bound on the negative log probability of the data when replacing the optimal denoiser $\mathbb{E}_x[x \,|\, y, t]$ with the denoiser derived from the Miyasawa-Tweedie expression, $y - t\nabla_y U_\theta(y, t)$:

$$-\log p_{\mathrm{MSE}}(x) = \frac{d}{2}\log(2\pi e t_{\max}) - \int_0^{t_{\max}} \left(td - \mathbb{E}_y\left[\|x - y + t\nabla_y U_\theta(y, t)\|^2 \,\middle|\, x\right]\right)\frac{\mathrm{d}t}{2t^2}. \tag{16}$$

This framework can be generalized to other noise distributions (Guo et al., 2013) such as Poisson noise, which is more adapted to discrete distributions (Bhattacharya et al., 2025).

Finally, it was recently observed in Skreta et al. (2024); Karczewski et al. (2025) that a cheaper unbiased stochastic estimator of this bound can be obtained with the Itô formula. Given a realization $(x_t)_{t\in\mathbb{R}_+}$ of the forward SDE (Brownian motion) $\mathrm{d}x_t = \mathrm{d}w_t$ started at $x_0 = x \sim p(x)$, then

$$-\log p(x) = U(x_{t_{\max}}, t_{\max}) - \int_0^{t_{\max}} \left(\langle \nabla_y U(x_t, t), \mathrm{d}x_t\rangle - \frac{1}{2}\|\nabla_y U(x_t, t)\|^2 \mathrm{d}t\right). \tag{17}$$

When $t_{\max} \to \infty$, one can again use $U(x_{t_{\max}}, t_{\max}) \sim \frac{\|x_{t_{\max}}\|^2}{2t_{\max}} + \frac{d}{2}\log(2\pi t_{\max})$. Replacing the unknown true energy U in the integrand with a model U_θ leads to a biased stochastic estimate of the log probability:

$$-\log p_{\mathrm{SDE}}(x) = \frac{\|x_{t_{\max}}\|^2}{2t_{\max}} + \frac{d}{2}\log(2\pi t_{\max}) - \int_0^{t_{\max}} \left(\langle \nabla_y U_\theta(x_t, t), \mathrm{d}x_t\rangle - \frac{1}{2}\|\nabla_y U_\theta(x_t, t)\|^2 \mathrm{d}t\right). \tag{18}$$

Karczewski et al. (2025) show that averaging over SDE trajectories started at the same $x_0 = x$ leads to an upper bound on the NLL, in fact recovering the one in eq. (16).

In summary, the ODE gives deterministic probabilities exactly corresponding to the corresponding generative model (eq. (14)), while for the SDE one can choose between a denoising-based upper bound (eq. (16)) or a cheap stochastic estimator of it (eq. (18)). These evidently similar approaches

are equivalent if and only if the model is consistent across noise levels (i.e., satisfies the diffusion equation). In practice, the choice of estimation method can lead to variations of up to 0.5 bits/dimension (Kong et al., 2023), see in particular Karczewski et al. (2025) for a careful study). These approaches also apply to our model, in addition to the direct evaluation of $U_\theta(x, t = 0)$. While this latter estimate does not correspond to a generative model (like the ODE) or a variational bound (like the SDE), its main advantage is its computational efficiency, as it does not require integrating over noise levels and gives deterministic values without averaging over noise realizations (for a divergence term or mean squared error).

A.3 On the theoretical justification of dual score matching

A limitation of our approach is that we offer no theoretical guarantees that minimizing our objective leads to a good approximation of the energy. It would be desirable to show that our training objective (assuming infinite training data) quantitatively controls the distance between the learned energy and the data energy. This could be achieved by showing that the joint distribution $p(y, t)$ (with $\log t$ uniformly distributed in $[\log t_{\min}, \log t_{\max}]$) satisfies a Poincaré inequality, i.e.,

$$\mathrm{Var}[U_\theta(y,t) - U(y,t)] \leq C\mathbb{E}\Big[\|\nabla_y U_\theta(y,t) - \nabla_y U(y,t)\|^2 + (\partial_t U_\theta(y,t) - \partial_t U(y,t))^2\Big], \quad (19)$$

for some constant C that is not too large. While we conjecture that such a result holds (or a variant, such as when considering the distribution of $(y/\sqrt{t}, \log t)$, to match our noise level weighting), note that this is weaker than a control in the Kullback-Leibler divergence. Replacing $\mathrm{Var}[U_\theta(y,t) - U(y,t)]$ with $\mathrm{KL}(p \,\|\, p_\theta)$ in eq. (19) would require that the *model* distribution $p_\theta(y, t)$ satisfies a log-Sobolev inequality, which could only be enforced with specific architectures. As a result, there is no control of the learned energy outside the support of the data distribution, so that out-of-distribution detection may be unreliable. It also implies our model may not be exactly normalized. Our NLL calculations are thus only approximate (although computationally cheap). These limitations are common to all current score-based and unnormalized energy-based approaches, including the related work of Yadin et al. (2024).

A.4 On frequency estimation with density models

Here, we describe a counterintuitive property of density models in high dimensions which poses a challenge for estimating frequencies. We also refer the interested reader to Theis et al. (2016), which offers related observations.

Consider a mixture of two uniform distributions on two compact sets (classes) C_1 and C_2 with respective frequencies f_1 and f_2. The probability density is then constant on each class, with value $p_i = \frac{f_i}{V_i}$ where V_i is the volume of C_i. The volume V_i typically scales exponentially with d, $V_i \sim r_i^d$ where r_i is the characteristic size of C_i. The energy values on each class are then

$$U_i = -\log p_i = d \log r_i - \log f_i. \quad (20)$$

In high dimensions $d \gg 1$, the energy values are dominated by the volume and only weakly depend on the frequencies of each class.

A concrete example of this phenomenon can be seen in the mixture of two Gaussian distributions considered in Figure 1, where the two classes correspond to the two spheres of radius $\sqrt{d}\sigma_i$ for $i \in \{1, 2\}$. The typical values of the energy $U_i = \frac{d}{2}\log\big(2\pi\mathrm{e}\sigma_i^2\big) - \log f_i$ are dominated by the entropy of the corresponding Gaussian.

The implications of this observation for high-dimensional density models are twofold. First, estimated densities are dominated by volumes of typical sets (entropies of the mixture components), more than frequencies of different categories (one-dimensional marginals). (Note that the latter is easily learned from data, while estimating the former is a much more challenging task.) This observation explains empirical reports that energy models may assign higher probability to out-of-distribution samples (Nalisnick et al., 2019; Karczewski et al., 2025). Second, estimating energy up to a small relative error is not sufficient to capture observed frequencies. For our ImageNet64 model ($d = 4096$), a relative error of ~ 0.1 dB/dimension (as estimated from the generalization experiment in Figure 2) is negligible compared to typical energy differences of ~ 10 dB/dimension, but the required precision to estimate frequencies with a relative accuracy of 10% is $1/d = 2 \times 10^{-4}$ dB/dimension.

B Proofs and derivations

B.1 Time score matching (proof of eq. (4))

We start from the expression of the energy (negative log probability) of noisy data:

$$U(y,t) = -\log p(y|t), \tag{21}$$

$$p(y|t) = \int p(x)p(y|x,t)\mathrm{d}x. \tag{22}$$

Differentiating the energy with respect to t yields

$$\partial_t U(y,t) = -\frac{\partial_t p(y|t)}{p(y|t)} \tag{23}$$

$$= -\frac{\int p(x)p(y|x,t)\partial_t \log p(y|x,t)\mathrm{d}x}{p(y|t)} \tag{24}$$

$$= \mathbb{E}_x[-\partial_t \log p(y|x,t) \,|\, y,t]. \tag{25}$$

Note that the derivation exactly matches that of the Miyasawa-Tweedie identity by replacing ∂_t with ∇_y, and does not make any assumptions about the form of $p(y|x,t)$. Restricting to additive Gaussian noise, $-\log p(y|x,t) = \frac{1}{2t}\|y-x\|^2 + \frac{d}{2}\log(2\pi t)$, and substituting into eq. (25) gives the "time score-matching" identity of eq. (4):

$$\partial_t U(y,t) = \mathbb{E}_x\left[\frac{d}{2t} - \frac{\|y-x\|^2}{2t^2}\,\middle|\, y\right]. \tag{26}$$

B.2 Energy architecture

Suppose that the energy network is defined in terms of a score network $s_\theta(y,t)$ as $U_\theta(y,t) = \frac{1}{2}\langle y, s_\theta(y,t)\rangle$, where the score network is assumed conservative and homogeneous. Conservativity means that there exists a scalar function ϕ such that $s_\theta(y,t) = \nabla_y\phi(y,t)$, which implies that the Jacobian $\nabla_y s_\theta(y,t) = \nabla_y^2\phi(y,t)$ is symmetric. The homogeneity property requires that for all $\lambda \geq 0$, $s_\theta(\lambda y,t) = \lambda s_\theta(y,t)$. Differentiating with respect to λ and setting $\lambda = 1$ yields

$$\nabla_y s_\theta(y,t)y = s_\theta(y,t). \tag{27}$$

We now calculate the gradient of the energy network:

$$\nabla_y U_\theta(y,t) = \frac{1}{2}\big(s_\theta(y,t) + \nabla_y s_\theta(y,t)^{\mathrm{T}}y\big) = s_\theta(y,t), \tag{28}$$

using the conservative and homogeneity properties to derive that $\nabla_y s_\theta(y,t)y = \nabla_y s_\theta(y,t)^{\mathrm{T}}y = s_\theta(y,t)$. Note that even if s_θ is not conservative, then it still holds that $\nabla_y U_\theta(y,t) = \frac{1}{2}\big(\nabla_y s_\theta(y,t) + \nabla_y s_\theta(y,t)^{\mathrm{T}}\big)y$, which can be interpreted as a symmetrization of the Jacobian of s_θ.

We also remark that if s_θ is homogeneous, then U_θ is quadratically homogeneous ($U_\theta(\lambda y,t) = \lambda^2 U_\theta(y,t)$). Note that this does not correspond to enforcing (asymmetric) Gaussian one-dimensional marginals $\langle y,u\rangle$. Rather, this enforces that the distribution of $\langle y,u\rangle$ conditioned on the orthogonal projection $y - \langle y,u\rangle u/\|u\|^2$ is (asymmetric) Gaussian.

B.3 Effective dimensionality (equivalence between eqs. (11) and (12))

We start by calculating the time derivative of the effective energy $\mathbb{E}_y[U(y,t) \,|\, x]$. We have

$$\partial_t \mathbb{E}_y[U(y,t) \,|\, x] = \partial_t \int p(y|x,t)U(y,t)\mathrm{d}y \tag{29}$$

$$= \int (\partial_t p(y|x,t)U(y,t) + p(y|x,t)\partial_t U(y,t))\mathrm{d}y. \tag{30}$$

The first term is the derivative with respect to variance of a Gaussian distribution $\mathcal{N}(x, t\mathrm{Id})$. It satisfies the diffusion equation:

$$\partial_t p(y|x,t) = \frac{1}{2}\Delta_y p(y|x,t). \tag{31}$$

Similarly, for the second term, we use the fact that $U(y,t) = -\log p(y|t)$ where $p(y|t)$ also satisfies the diffusion equation (as can be seen from the Fokker-Planck equation in the variance exploding case (Song et al., 2021a)):

$$\partial_t U(y,t) = -\frac{\partial_t p(y|t)}{p(y|t)} \tag{32}$$

$$= -\frac{\Delta_y p(y|t)}{2p(y|t)} \tag{33}$$

$$= \frac{1}{2p(y|t)}\nabla_y \cdot (p(y|t)\nabla_y U(y|t)) \tag{34}$$

$$= \frac{1}{2}\Delta_y U(y|t) - \frac{1}{2}\|\nabla_y U(y|t)\|^2. \tag{35}$$

This equation appeared in Lai et al. (2023); Bruna and Han (2024) and is a special case of a Hamilton-Jacobi equation (Evans, 1993).

Combining eqs. (31) and (35) into eq. (30) and integrating by parts twice, we have

$$\partial_t \mathbb{E}_y[U(y,t)\,|\,x] = \int p(y|x,t)\left(\Delta_y U(y,t) - \frac{1}{2}\|\nabla_y U(y|t)\|^2\right)\mathrm{d}y. \tag{36}$$

We recognize the expression of the mean squared error as given by combining Miyasawa-Tweedie with Stein's unbiased risk estimate (SURE). Indeed,

$$\mathbb{E}_y\left[\|x - \mathbb{E}_x[x\,|\,y]\|^2 \,\middle|\, x\right] = \mathbb{E}_y\left[\|x - y + t\nabla_y U(y,t)\|^2 \,\middle|\, x\right] \tag{37}$$

$$= \mathbb{E}_y\left[\|x-y\|^2 + 2t\langle x - y, \nabla_y U(y,t)\rangle + t^2\|\nabla_y U(y,t)\|^2 \,\middle|\, x\right] \tag{38}$$

$$= td + t^2\mathbb{E}\left[-2\Delta_y U(y,t) + \|\nabla_y U(y,t)\|^2 \,\middle|\, x\right], \tag{39}$$

where we have used Stein's lemma in the last step. We finally combine eqs. (36) and (39) to obtain

$$\frac{1}{t}\mathbb{E}_y\left[\|x - \mathbb{E}_x[x\,|\,y]\|^2 \,\middle|\, x\right] = d - 2t\partial_t \mathbb{E}_y[U(y,t)\,|\,x]. \tag{40}$$

It is convenient to rewrite $2t\partial_t = \partial_{\frac{1}{2}\log t}$ as $\mathrm{d}\left(\frac{1}{2}\log t\right) = \frac{\mathrm{d}t}{2t}$.

We define the common value in eq. (40) to be $d_{\mathrm{eff}}(x,t)$. It is related (but not equal) to dimensionality measures estimated from the singular values of the Jacobian of a denoiser (Mohan* et al., 2020; Horvat and Pfister, 2024). The limit of $d_{\mathrm{eff}}(x,t)$ when $t \to 0$ has appeared in the literature under the name of local intrinsic dimensionality using (approximate) likelihood (LIDL) (Tempczyk et al., 2022; Stanczuk et al., 2022; Kamkari et al., 2024). Its average over images x is equal to the MMSE dimension of Wu and Verdú (2011).

C Experimental details

C.1 Energy network architecture

UNet architecture. Our UNet architecture s_θ is composed of 3 encoder blocks, a middle block, and 3 decoder blocks. Each block is itself composed of 3 layers, each a sequence of bias-free convolution, normalization, and non-linearity, for a total of 21 layers. The first convolutional layer of each encoder block but the first and the middle block has a stride of 2 (downsampling, in both vertical and horizontal directions), and the last convolutional layer of each decoder block (except for the last one and the middle block) is transposed with a stride of 2 (upsampling). The output of each encoder block is concatenated to the input of the corresponding decoder block (which comes from the output of the corresponding encoder block, via a "skip connection"). The number of channels is doubled in each block, starting from a base value of 64 at the coarsest scale. We replace ReLUs with GeLUs to ensure that $\nabla_y U_\theta$ is differentiable. Thus, U_θ is only approximately quadratically homogeneous. As a result, and because training does not result in an exactly conservative s_θ, $\nabla_y U_\theta(y,t)$ should be computed directly as opposed to using s_θ.

Normalization. We also replace batch normalizations, whose behavior during the backward pass is incompatible with a second back-propagation, with a homogeneous version of instance normalization (Ulyanov et al., 2016). If the input x consists of C channels $(x_c)_{1\leq c\leq C}$, and its spatial mean is $\mu(x) \in \mathbb{R}^C$, each channel is normalized according to

$$x_c \mapsto \sqrt{\frac{\|x-\mu(x)\|^2+\varepsilon}{C\|x_c-\mu(x_c)\|^2+\varepsilon}}(x_c-\mu(x_c)). \tag{41}$$

Up to a small $\varepsilon > 0$ parameter for numerical stability, this ensures that after normalization all channels x_c have equal norms while preserving the global norm $\|x\|$. This normalization layer is also homogeneous when $\varepsilon = 0$, as the spatial mean is estimated from the input x. This normalization is followed by a learned rescaling of each channel, $x_c \mapsto \gamma_c x_c$, where $\gamma \in \mathbb{R}^C$ is learnable.

Noise level conditioning. The noise variance t is also an input of s_θ. As is standard in diffusion models (Nichol and Dhariwal, 2021), a time embedding $e(t) \in \mathbb{R}^{256}$ is computed with Fourier features $\cos(\omega_k t)$, $\sin(\omega_k t)$ (we use 32 frequencies $(\omega_k)_k$ that are linearly spaced in the log domain and ranging from $1/t_{\max}$ to $1/t_{\min}$) followed by a shallow MLP. This time embedding $e(t)$ is then used to condition the output of each normalization layer via gain control: $x_c \mapsto ((1+\langle w_c, e(t)\rangle)x_c)$ where w_c is a learned layer- and channel-dependent vector $\in \mathbb{R}^{256}$.

C.2 Training hyper-parameters

To summarize Section 2.2, the dual score matching training objective is

$$\ell(\theta) = \mathbb{E}_{x,z,t}\left[\left\|\sqrt{\frac{t}{d}}\nabla_y U_\theta(y,t) - \frac{z}{\sqrt{d}}\right\|^2 + \left(\frac{t}{d}\partial_t U_\theta(y,t) - \frac{1}{2}\left(1-\frac{\|z\|^2}{d}\right)\right)^2\right], \tag{42}$$

where $x \sim p(x)$ is the data distribution, $z \sim \mathcal{N}(0, \mathrm{Id})$ is the noise, $y = x + \sqrt{t}z$ is the noisy measurement, and $\log t \sim \mathcal{U}(\log t_{\min}, \log t_{\max})$. In our experiments we use $t_{\min} = 10^{-9}$ and $t_{\max} = 10^3$, and the training image intensities are rescaled to have values in $[0, 1]$.

We use the ImageNet64 dataset (Chrabaszcz et al., 2017), with (only) horizontal flips as data augmentation. The models used in Tables 1 and 2 and for the generalization experiment in Figure 2 are trained on color images, while the model used in Figures 3 to 5 is trained on grayscale images. Pixel values are rescaled to $[0, 1]$ by dividing by 255. All models are trained for 1M steps, with a batch size of 128. We use the Adam optimizer with default parameters and an initial learning rate of 0.0005 (except for the generalization experiments which used a learning rate of 0.0002) that is halved every $100,000$ steps. All models are trained on a single NVIDIA H100 GPU, which takes about 5 days for ImageNet64.

C.3 Additional details

Gaussian scale mixture example (Figure 1). We generate $n = 100,000$ samples from a mixture of two Gaussian distributions, $\frac{1}{2}\mathcal{N}(0, \sigma_1^2\mathrm{Id}) + \frac{1}{2}\mathcal{N}(0, \sigma_2^2\mathrm{Id})$, with $\sigma_1 = 1$ and $\sigma_2 = 4$, in dimension $d = 1,000$. The true (normalized) energy is given by

$$U(y,t) = -\log\left(\sum_{i=1}^{2} \mathrm{e}^{-\frac{\|y\|^2}{2(\sigma_i^2+t)} - \frac{d}{2}\log\left(2\pi(\sigma_i^2+t)\right) - \log 2}\right). \tag{43}$$

We parameterize the energy as a mixture of quadratics:

$$U_\theta(y,t) = -\log\left(\sum_{i=1}^{2} \mathrm{e}^{-a_i(t)\|y\|^2 - b_i(t)}\right), \tag{44}$$

where the functions a_i, b_i are computed by a 5-layer MLP with a hidden dimension of 256 that takes $\log(t + t_{\min})$ as input. This network is trained either with single (space) score matching or with dual score matching, both across noise levels $t \in [t_{\min}, t_{\max}]$. Training is otherwise similar to the ImageNet64 experiments (see Appendix C.2), for $20,000$ training steps with a batch size of 512 and an initial learning rate of 0.0001, over noise levels from $t_{\min} = 10^{-2}$ to $t_{\max} = 10^2$. We note that the energy learned after single score matching training is stochastic: as the relative energy between the two mixture components is not constrained by the data, its value is determined by the random initialization, so that rerunning the experiment will lead to a different value.

Histogram of log probabilities (Figure 3). The Gumbel fit is calculated by maximizing likelihood. The obtained parameters (in decibels/dimension) are 9.57 for the location and 3.17 for the scale. Equivalently, $p(x)$ follows a Fréchet distribution, with a scale parameter equal to 9.05^d and shape parameter equal to $1.37/d$, where $d = 64 \times 64 = 4096$ is the dimension of ImageNet64 grayscale images.

Computing dimensionality (Figure 5). Equations (11) and (12) provide two ways to estimate effective dimensionality from a learned energy model. If the model is exact, $U_\theta(y, t) = U(y, t)$, then they coincide (more generally, they coincide when the model satisfies the diffusion equation), but in general they only approximately coincide. For numerical stability, we found it preferable to use the version defined from the mean squared error of the underlying denoiser (eq. (11)). This guarantees non-negative dimensionalities, and also has the advantage of being an upper bound on the true dimensionality (as any denoiser yields an upper bound on the minimum MSE).

D Additional results

D.1 ImageNet images according to their probability

Figure 6: Highest probability images in ImageNet64 (test set).

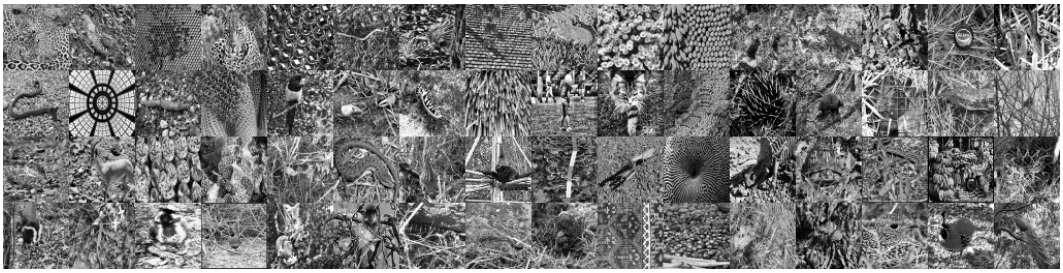

Figure 7: Lowest probability images in ImageNet64 (test set).

Figure 8: Example images from ImageNet64 (test set) with linearly-spaced log probabilities, ordered from low to high.

D.2 Distribution of log probabilities on other datasets

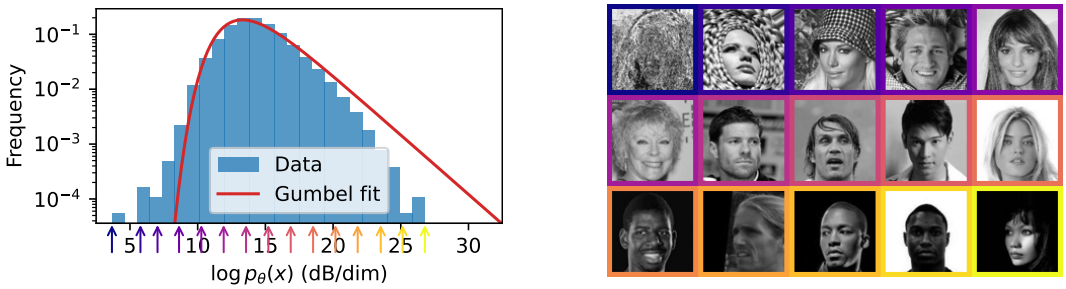

Figure 9: Histogram of log probabilities of test images in the CelebA dataset (grayscale). Color-coded arrows indicate values for the example images on the right. The maximum-likelihood fit by a Gumbel distribution (red line) is perturbed by the low-probability outliers, some of which correspond to corrupted images.

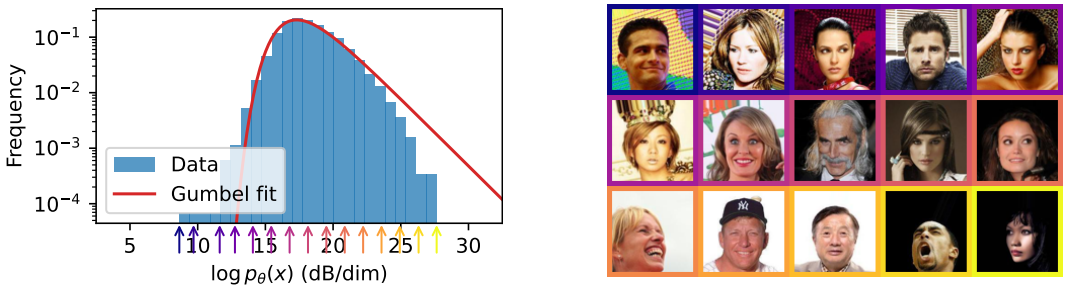

Figure 10: Histogram of log probabilities of test images in the CelebA dataset (color). Color-coded arrows indicate values for the example images on the right. The maximum-likelihood fit by a Gumbel distribution (red line) is perturbed by the low-probability outliers.

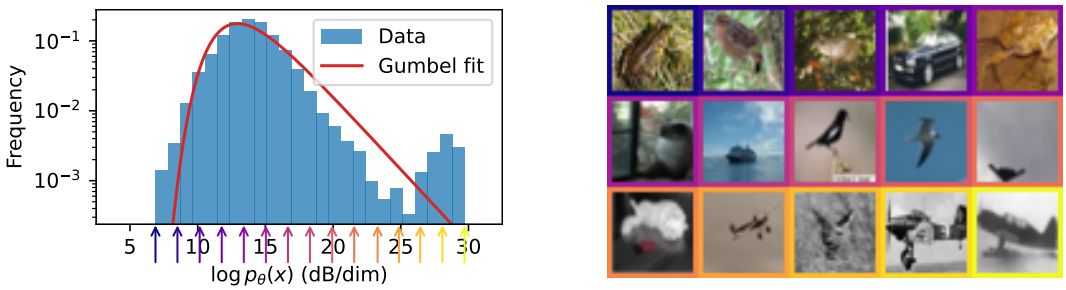

Figure 11: Histogram of log probabilities of test images in the CIFAR-10 dataset. Color-coded arrows indicate values for the example images on the right. The maximum-likelihood fit by a Gumbel distribution (red line) is perturbed by high-probability outliers, which correspond to grayscale images.

D.3 Log probabilities of interpolated images

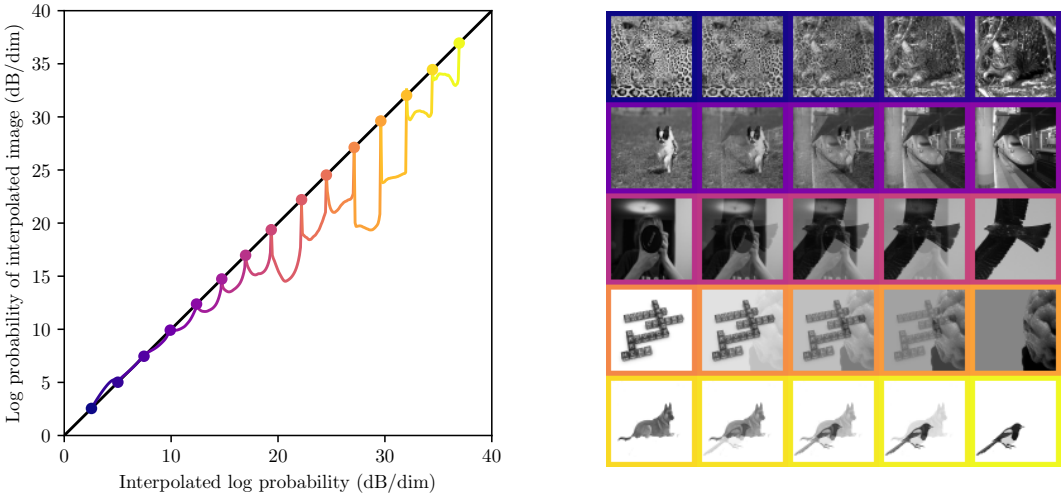

Figure 12: Log probability of interpolated images. For each pair (x_i, x_{i+1}) of consecutive example images (see Figure 3), we consider a linear path between them, $\alpha \in [0, 1] \mapsto (1 - \alpha)x_i + \alpha x_{i+1}$ (shown in the right panel). In the left panel, we plot the log probability of the resulting images $\log p_\theta((1 - \alpha)x_i + \alpha x_{i+1})$ as a function of the interpolated log probability $\alpha \log p_\theta(x_i) + (1 - \alpha) \log p_\theta(x_{i+1})$. For low-probability images, this leads to a slight increase in probability: at the extreme, the interpolation between two noise images is another noise image of slightly lower contrast. However, for medium- and high-probability images, this leads to a sharp decrease in probability even for α close to 0 or 1: the superposition of two natural images is not a natural image.

D.4 Optimizing log probability

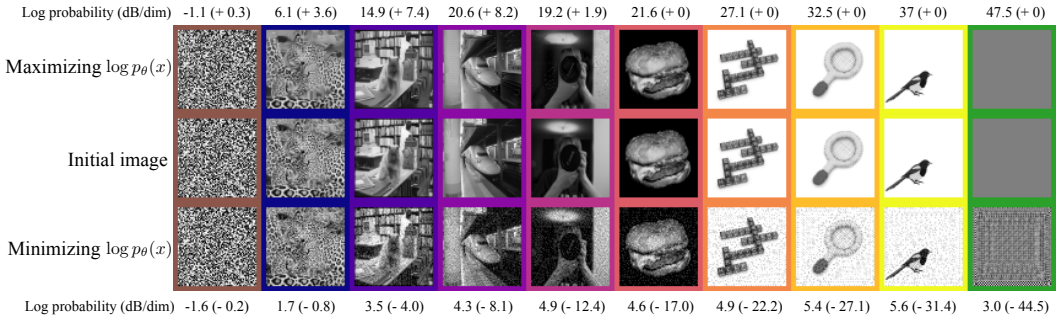

Figure 13: For each initial image (middle row), we perform $1,000$ steps of gradient ascent (top row) or descent (middle row) on the log probability with a fixed step size of $10^{-2}/d$. At each step, the pixel intensity values are clipped to $[0, 1]$, and at the end of the optimization we show the iterate with the highest or lowest probability. Resulting log probability values, and their difference with the starting value, are shown above or below the final images. Maximizing log probability leads to generally smoother images (although with some visual artifacts), except for sufficiently high-probability images, where gradient ascent fails to increase the probability. Minimizing log probability leads to noisier images, with surprising patterns for high-probability images.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Yes, we describe the scope and motivations for the work clearly in the abstract and introduction sections.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss limitations and assumptions extensively throughout the paper, particularly in the discussion.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Assumptions are stated clearly and proofs are provided either in the appendix or by citation.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Experimental details are provided in Appendix C.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We have only used publicly available datasets (ImageNet64). We release code to reproduce all experiments and pre-trained models at https://github.com/FlorentinGuth/DualScoreMatching.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Experimental details are described in Appendix C.2.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Statistical error is negligible for the reported results. Figure 2 verifies that the trained energy model is not sensitive to its initialization or subset of the training data.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: These details are reported in Section 2.4 and Appendix C.2.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: This work is not about applications but about foundations of energy estimation, which makes the short-term societal impact negligible.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Both the original and downsampled ImageNet papers are cited.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: We did not use LLMs in this work.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.