We introduce a simple modification to the standard maximum likelihood estimation (MLE) framework. Rather than maximizing a single unconditional likelihood of the data under the model, we maximize a family of noise conditional likelihoods consisting of the data perturbed by a continuum of noise levels. We find that models trained this way are more robust to noise, obtain higher test likelihoods, and generate higher quality images. They can also be sampled from via a novel score-based sampling scheme which combats the classical covariate shift problem that occurs during sample generation in autoregressive models. Applying this augmentation to autoregressive image models, we obtain 3.32 bits per dimension on the ImageNet 64x64 dataset, and substantially improve the quality of generated samples in terms of the Frechet Inception distance (FID) — from 37.50 to 12.09 on the CIFAR-10 dataset.

1 Introduction

Likelihood maximization models, i.e., models trained by maximizing log-likelihood, are a leading class of modern generative models. Of these, autoregressive models boast state-of-the-art performance in many domains, including images Salimans et al. [2017], Child et al. [2019], text Vaswani et al. [2017], and audio Cord et al. [2016]. These architectures also show great promise for modeling long range dependencies Tay et al. [2020], Gu et al. [2021].

However, while log-likelihood is broadly agreed upon as one of the most rigorous metrics for goodness-of-fit in statistical and generative modeling, models with high likelihoods do not necessarily produce samples of high visual quality. This phenomenon has been discussed at length by Theis et al. [2015], Huszár [2015], and corroborated in empirical studies Grover et al. [2018], Kim et al. [2022].

Autoregressive models have an additional affliction: they have notoriously unstable dynamics during sample generation Bengio et al. [2015] due to their sequential sampling algorithm, which can cause errors to compound across time steps. Such errors cannot usually be corrected ex post facto due to the autoregressive structure of the model, and can substantially affect downstream steps as we find that their likelihoods are sensitive to even the most minor of perturbations.

Score-based diffusion models Song et al. [2020], Ho et al. [2020] offer a different perspective on the matter. Even though sampling is also sequential, diffusion models are more robust to perturbations because, in essence, they are trained as denoising functions Ho et al. [2020]. Moreover, the update direction in each step is unconstrained (unlike token-wise autoregressive models, which can only update one token at a time, and only once), meaning the model can correct errors from previous steps. However, diffusion models are poor likelihood models, as they cannot be trained via maximum likelihood, and density evaluations are inexact and require solving ODEs involving hundreds to thousands of function evaluations. Thus we wonder: is there a conceptual middle ground?
Figure 1: Generated samples on CelebA 64x64 (above) and CIFAR-10 (below). Autoregressive models trained via vanilla maximum likelihood (left) are brittle to sampling errors and can quickly diverge, producing nonsensical results. Those trained via our proposed algorithm (right) are more robust, which can significantly increase the coherence of the generated images.

In this paper, we offer such a framework. We further analyze the likelihood-sample quality mismatch in autoregressive models, and propose techniques inspired by diffusion models to alleviate it. In particular, we leverage the fact that the score function is naturally learned as a byproduct of maximum likelihood estimation. This allows a novel two-part sampling strategy with noisy sampling and score-based refinement.

Our contributions are threefold. 1) We investigate the pitfalls of training and inference under the log-likelihood maximization scheme, particularly regarding sensitivity to noise corruptions. 2) We propose a simple sanity test for checking the noise-robustness of likelihood models. 3) We introduce a novel framework for the training and sampling of likelihood maximization models that improves noise-robustness and substantially boosts the sample quality of the resulting model. Ultimately, we obtain a model that can generate samples at a quality approaching that of diffusion models, without losing the maximum likelihood framework and $O(1)$ likelihood evaluation speed of likelihood maximization models.

2 The Pitfalls of Maximum Likelihood

We first show that density models trained to maximize the standard log-likelihood are surprisingly sensitive to minor perturbations. We then discuss why this is bad for generative modeling performance.

2.1 A Simple Sanity Test

Consider the class of minimally corrupted probability densities we call $p_\pi$, where

$$p_\pi = p_{\text{data}} \ast p_{\text{mult}_{\{-1,0,1\}}(\pi/2,1-\pi/2)}, \quad \pi \in [0, 1].$$

(1)

Here, $\ast$ denotes the convolution operator, and $p_{\text{mult}_{\{a,b,c\}}(\alpha,\beta,\gamma)}$ is the density a $d$-dimensional multinomial distribution taking on $a$, $b$, and $c$ with probabilities $\alpha$, $\beta$, and $\gamma$, respectively. $p_\pi$ is minimally corrupted in the sense that, if $p_{\text{data}}$ is an integer-discretized distribution (say, 8-bit images),
\( p_\pi \) describes the distribution of points \( p_{\text{data}} \) that have had their least significant bit incremented or decremented with probability \( \pi \).

To the human eye, the difference between samples drawn from \( p_\pi \) and \( p_{\text{data}} \) is almost imperceptible, even for \( \pi = 1 \) (see Fig 2). However, for likelihood models, this perturbation drastically increases the negative log-likelihood of the data under the model (see Table 1), to the point that it significantly undermines (if not outright nullifies) any recent advances in density estimation. This basic inconsistency suggests that the learned density of many standard likelihood models is brittle and overly emphasizes bit-level statistics that have little influence on the inherent content of the image.

2.2 Why We Should Care

We provide two reasons for why failing this test is problematic, especially for autoregressive models. First, noise is natural — and being less robust to noise also means being a poorer fit to natural data. Outside of the log-likelihood, measures of generative success in generative models fall under two categories: qualitative assessments (i.e., the no-reference perceptual quality assessment [Wang et al., 2002] or ‘eyeballing’ it) and quantitative heuristics (i.e., computing statistics of hidden activations of pretrained CNNs [Salimans et al., 2016, Heusel et al., 2017, Sajjadi et al., 2018]). Both strategies either rely directly on the human visual system, or are known to be closely related to it [Giuliani and van Gerven, 2015, Yamins et al., 2014, Khaligh-Razavi and Kriegeskorte, 2014, Eickenberg et al., 2017, Cichy et al., 2016]. Therefore, implicit in the use of these criteria is the existence of a human (or human-like) model of images \( q_{\text{human}} \), where \( q_{\text{human}} \approx p_{\text{data}} \) [Huszár, 2015]. The fact that we find samples from \( p_\pi \) nearly indistinguishable from \( p_{\text{data}} \), whereas \( p_\theta \) finds them very different suggests that \( p_{\text{data}} \approx q_{\text{human}} \neq p_\theta \).

Second, sample quality suffers. This holds for general likelihood models, given what we argue in the first point — namely \( p_\theta \neq q_{\text{human}} \). However, noise-sensitivity is doubly problematic in autoregressive models. Due to the sequential nature of autoregressive sampling and the fact that models are trained entirely on data from the true distribution, any sampling error can drastically affect the sampling trajectory. This is related to the well-known covariate shift phenomenon [Bengio et al., 2013, Shimodaira, 2000]. Moreover, such errors compound quickly. Table 1 shows that mis-sampling pixels by even a single bit can cause drastic changes to the overall likelihood. This can explain why standard autoregressive models commonly produce nonsensical results (Fig 1).

3 Noise Conditional Maximum Likelihood

To alleviate the problems discussed in Section 2, we propose a simple modification to the standard objective in maximum likelihood estimation. Rather than evaluating a single likelihood as in the vanilla formulation, we consider a family of noise conditional likelihoods

\[
E_{\sigma \sim \mu} E_{x \sim p_x} \log p_{\theta, \sigma}(x),
\]

where \( p_x \) is a stochastic process indexed by noise scales \( \sigma \) describing a noise-corrupted version of \( p_{\text{data}} \), and \( \mu \) is a distribution over such scales. We call this the noise conditional maximum likelihood (NCML) framework. In general, (2) is an all-purpose plug-in objective that can be used with any likelihood model adapted to accept a noise conditioning vector, though a continuous likelihood (e.g. Salimans et al., 2017, Li and Klüger, 2022) is necessary for computation of the score function.

Letting \( \sigma \) be the time index of a diffusion process, our approach becomes closely related to score-based diffusion models [Song et al., 2020], albeit with two crucial differences.

First, instead of merely estimating the noise conditional score function of the perturbed data density \( p_\sigma \) for \( \sigma \in [0, T] \), we directly estimate \( p_\sigma \) itself. However, we still learn the noise conditional score function as a by-product of NCML. Moreover, we may access the score function by simply differentiating the log likelihood. Therefore, we can refine sampled points via Langevin dynamics. This provides an alternative strategy for sampling from \( p_{\theta, \sigma} \), which we explore in 3.1.

Second, we need not design our diffusion so that \( p_{\theta, \sigma} \) approximates the limiting stationary distribution of the process. This is necessary in diffusion models as the limiting prior is the only tractable distribution to initialize the sampling algorithm with. Since we have learned the density itself for all \( \sigma \in [0, T] \), we may initialize from any point of the diffusion, which increases the flexibility of the sampling strategy, and can drastically reduce the steps required to solve the reverse diffusion.
<table>
<thead>
<tr>
<th>Model</th>
<th>FID</th>
<th>NLL $\pi = 0^*$</th>
<th>NLL $\pi = 0.5$</th>
<th>NLL $\pi = 1$</th>
<th>NLL $\pi = 0^*$</th>
<th>NLL $\pi = 0.5$</th>
<th>NLL $\pi = 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ELBO</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VDM</td>
<td>7.41</td>
<td>2.49</td>
<td>3.75</td>
<td>3.97</td>
<td>3.40</td>
<td>3.76</td>
<td>3.88</td>
</tr>
<tr>
<td>ScoreFlow</td>
<td>5.40</td>
<td>2.90</td>
<td>3.82</td>
<td>3.99</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>VDVAE</td>
<td>-</td>
<td>2.84</td>
<td>3.90</td>
<td>4.10</td>
<td>3.52</td>
<td>3.66</td>
<td>3.82</td>
</tr>
<tr>
<td><strong>Likelihood</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Flow++</td>
<td>-</td>
<td>3.09</td>
<td>3.86</td>
<td>4.08</td>
<td>3.69</td>
<td>3.82</td>
<td>3.99</td>
</tr>
<tr>
<td>DenseFlow</td>
<td>48.15</td>
<td>2.98</td>
<td>3.80</td>
<td>4.02</td>
<td>3.35</td>
<td>3.68</td>
<td>3.85</td>
</tr>
<tr>
<td>PixelCNN++</td>
<td>55.72</td>
<td>2.92</td>
<td>3.84</td>
<td>4.01</td>
<td>3.52</td>
<td>3.84</td>
<td>4.00</td>
</tr>
<tr>
<td>PixelSNAIL</td>
<td>36.62</td>
<td>2.85</td>
<td>3.83</td>
<td>3.99</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Sparse Transformer</td>
<td>37.50</td>
<td>2.80</td>
<td>3.82</td>
<td>3.98</td>
<td>3.44</td>
<td>3.73</td>
<td>3.89</td>
</tr>
<tr>
<td>NCNPN (ML)</td>
<td>46.72</td>
<td>2.91</td>
<td>3.83</td>
<td>3.99</td>
<td>3.49</td>
<td>3.68</td>
<td>3.88</td>
</tr>
<tr>
<td>NCNPN (NCML-VE)</td>
<td>32.71</td>
<td>2.87</td>
<td>3.75</td>
<td>3.95</td>
<td><strong>3.32</strong></td>
<td><strong>3.67</strong></td>
<td><strong>3.85</strong></td>
</tr>
<tr>
<td>NCNPN (NCML-subVP)</td>
<td>23.42</td>
<td>2.95</td>
<td>3.69</td>
<td>3.94</td>
<td>3.36</td>
<td>3.66</td>
<td>3.82</td>
</tr>
<tr>
<td>NCNPN (NCML-VP)</td>
<td><strong>12.09</strong></td>
<td>3.20</td>
<td><strong>3.62</strong></td>
<td><strong>3.91</strong></td>
<td>3.43</td>
<td><strong>3.63</strong></td>
<td><strong>3.79</strong></td>
</tr>
</tbody>
</table>

Table 1: Results on CIFAR-10 and ImageNet 64x64. Negative log-likelihood (NLL) is in bits per dimension. Lower is better. *NLL with $\pi = 0$ is equivalent to NLL of the original data.

3.1 Sampling with Autoregressive NCML Models

The NCML framework allows for two sampling strategies. The first is to draw directly from the noise-free distribution $p_{\theta,0}$, in which case the conditional likelihood simplifies to a standard (unconditional) likelihood, and sampling is identical to that for a standard autoregressive model.

However, as discussed in Section 2, this strategy is unstable and tends to quickly accumulate errors. This motivates an alternative sampling strategy, which involves drawing from $p_{\theta,\sigma}$ for $\sigma > 0$, then solving a reverse diffusion process back to $\sigma = 0$. The latter is possible due to the fact that the reverse diffusion is itself a diffusion process that depends on the score function Anderson [1982], which we have access to. This is identical to the sampling procedure in score-based diffusion models Song et al. [2020], except for the key difference that we need not initialize with the prior distribution.

4 Experiments and Discussion

In all experiments in Table 1, we choose $p_{\pi}$ to be the variance exploding (VE), variance preserving (VP), and sub-variance preserving (sub-VP) SDEs, respectively. Due to space constraints, we refer to Song et al. [2020] for more details. For our architecture, we introduce the noise conditional pixel-wise network (NCPN), which consists of a PixelCNN backbone with added attention layers. We evaluate all models on minimally perturbed transformations (see 2.1) of CIFAR-10 and ImageNet 64x64 for $\pi \in \{0, \frac{1}{2}, 1\}$, where we note that $p_{\pi=0} = p_{\text{data}}$. All noise conditional models, i.e., ours, VDM Kingma et al. [2021], and ScoreFlow Song et al. [2021], are evaluated at $t = 0$.

While it is clear that all models have reduced likelihoods when evaluated on the perturbed distribution $p_{\pi}, \pi \in \{\frac{1}{2}, 1\}$, we note that our models are more robust to such transformations, even though they are evaluated under the noiseless condition, and trained on a different class of noise, i.e., the marginal likelihoods of diffusion processes. Furthermore, sample quality across all models correlates better with likelihoods on the perturbed distributions than likelihoods on the base distribution.

5 Conclusion

We proposed a simple sanity test for checking the robustness of likelihoods to minor perturbations. We found that most likelihood models are not robust under this test, and developed a new framework that improves performance in this setting, with substantial improvements in training and sampling.
References


Table 2: Results on CelebA 64x64. Negative log-likelihood (NLL) is in bits per dimension. Lower is better. *NLL with \( \pi = 0 \) is equivalent to NLL of the original data.

<table>
<thead>
<tr>
<th>Model</th>
<th>NLL  ( \pi = 0 )*</th>
<th>NLL  ( \pi = 0.5 )</th>
<th>NLL  ( \pi = 1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>NCPN (ML)</td>
<td>2.25</td>
<td>3.72</td>
<td>4.35</td>
</tr>
<tr>
<td>NCPN (NCML VE)</td>
<td>2.22</td>
<td>3.63</td>
<td>4.21</td>
</tr>
<tr>
<td>NPCN (NCML sub-VP)</td>
<td>2.31</td>
<td>3.44</td>
<td>3.98</td>
</tr>
<tr>
<td>NCPN (NCML VP)</td>
<td>2.48</td>
<td>3.14</td>
<td>3.67</td>
</tr>
</tbody>
</table>

A Appendix

A.1 Additional Experimental Details

For experiments on CIFAR-10 and ImageNet 64x64, we compare against Kingma et al. [2021], Song et al. [2021], Child [2020], Ho et al. [2019a], Grcić et al. [2021], Salimans et al. [2017], Chen et al. [2018], Child et al. [2019]. Some results could not be included due to the irreproducibility of the techniques. There is limited existing work on likelihood-based modeling on CelebA 64x64, so we do not provide comparisons here.

Our proposed NCPN architecture consists of the PixelCNN++ backbone Salimans et al. [2017] with axial attention layers Ho et al. [2019b] after each residual block. We retain the hyperparameters of PixelCNN++, changing only the dropout on the CIFAR-10 dataset (from 0.5 to 0.25), which we reduced due to the regularization properties of NCML. For the axial attention layers, we use 8 heads and skip connection rescaling as in Song et al. [2020]. Finally, we add noise conditioning to each residual block via a Gaussian Fourier Projection layer, much like Ho et al. [2020], Song et al. [2020].

For our NCML-trained models, the diffusion times of the VE, VP, and sub-VP SDEs were chosen to be \( T = 0.5 \), \( T = 0.1 \), and \( T = 0.025 \), respectively. The values are somewhat arbitrary, but selected such that the standard deviation of the per-pixel differences between samples in \( p_{\text{data}} \) and their noised counterparts in \( p_T \) was \( \approx 10 \). We suspect that further improvements can be made to the empirical results if these numbers were chosen more judiciously.

All NCPN models were trained on RTX 2080 Ti GPUs for 500,000 iterations. This is approximately 1.5 weeks of training. We use the same NCPN architecture and hyperparameters across all datasets (except for dropout, which is set to 0.25 on CIFAR-10 and 0.00 on ImageNet 64x64 and CelebA 64x64). All NCPN models have 73M parameters.

A.2 Additional Figures and Tables
Figure 2: Samples from NCPN trained on ImageNet 64x64, with $p_t$ as a variance preserving (VP) diffusion process.
Figure 3: Samples from NCPN trained on CelebA 64x64, with $p_t$ as a variance preserving (VP) diffusion process.
Figure 4: Samples from NCPN trained on CIFAR-10, with $p_t$ as a variance preserving (VP) diffusion process.
Figure 5: Samples from NCPN trained on CIFAR-10, with $p_t$ as a sub-variance preserving (sub-VP) diffusion process.
Figure 6: Samples from NCPN trained on CIFAR-10, with \( p_t \) as a variance exploding (VE) diffusion process.