Abstract

Due to Van den Oord et al. (2018), probability distillation has recently been of interest to deep learning practitioners, where, as a practical workaround for deploying autoregressive models in real-time applications, a student network is used to obtain quality samples in parallel. We identify a pathological optimization issue with the adopted stochastic minimization of the reverse-KL divergence: the curse of dimensionality results in a skewed gradient distribution that renders training inefficient. This means that KL-based “evaluative” training can be susceptible to poor exploration if the target distribution is highly structured. We then explore alternative principles for distillation, including one with an “instructive” signal, and show that it is possible to achieve qualitatively better results than with KL minimization.

1 INTRODUCTION

Deep autoregressive models are currently among the best choices for unsupervised density modeling tasks (Van den Oord et al., 2016b[a]). However, due to the factorization induced by such models on the data space by the chain rule of probability, sampling from such models requires $O(T)$ sequential computation steps, where $T$ is the dimension/sequence-length of the data. This makes sampling slow and inefficient for most practical purposes. A recent solution to alleviating this bottleneck was proposed by Van den Oord et al. (2018), who show that an autoregressive WaveNet model (Van den Oord et al., 2016a) can be distilled into a student network, which is significantly faster to sample from due to parallel computation, and therefore much more suitable for deployment in real-time applications.

In this paper, we identify a fundamental issue with the approach taken by Van den Oord et al. (2018), and provide alternative perspectives for distilling the sampling process of a teacher model.

The solution proposed by Van den Oord et al. (2018) is to minimize the reverse Kullback-Leibler divergence (KL) between the student and the teacher distribution. Generally, this relies on two essential components: (1) the gradient signal from the teacher network (the WaveNet model) and (2) invertibility of the student network (the Parallel WaveNet model). This allows samples drawn from the student to be evaluated under the likelihood of both models, and the student can then be updated via stochastic backpropagation from the teacher. The student is implemented with an inverse autoregressive transformation (Kingma et al., 2016) which admits fast sampling at the cost of slow likelihood estimation. On the other hand, since the autoregressive teacher allows fast evaluation of likelihoods but is slow to sample from, the approach takes advantage of the best of both worlds.

In theory, it seems that minimizing the KL should be sufficient for distilling a teacher into a student. However, in practice, it has been necessary to implement additional heuristics in order to achieve samples with similar realism as that of the teacher (Van den Oord et al., 2018; Ping et al., 2018). In particular, a power loss that biases the power across frequency bands of the student sampler to match that of human speech patterns has been crucial in recent work for generating synthesized speech without the student collapsing to unrealistic “whispering”.

Our contributions are as follows.

- We show how the reverse-KL loss is ill-suited for the task of probability distillation. Roughly speaking, this is because the teacher distribution typically possesses a low-rank nature for high-dimensional structured data, making the expansion signal (defined in Section 3.1), which is necessary for successfully distilling such a teacher, a rare event.
• We then explore different alternatives for distillation, by recasting the problem of distillation into learning a transformation of probability density from a prior space to the data space. This view connects decoder-based generative models with probability density distillation.

• Finally, we run experiments with images and speech data, demonstrating that it is possible to learn fast samplers with our proposed alternatives, which do not suffer from the pathologies of reverse-KL training.

2 AUTOREGRESSIVE MODELS

Given a joint probability distribution $p(x)$, where $x$ denotes a $T$-dimensional vector, one can factorize the distribution according to the chain rule of probability, for arbitrary ordering of dimensions:

$$p(x) = p(x_1) p(x_2 | x_1) \ldots p(x_T | x_1, \ldots, x_{T-1}) = \prod_{t=1}^{T} p(x_t | x_{1:t-1})$$  \hspace{1cm} (1)

In the tabular case, i.e., when $x_t$ can take $V$ different possible values, the joint probability can be represented by a table of $O(V^T)$ entries. When the event set of $x_t$ is uncountable, the joint density is not even tractable. This motivates the use of a parametric model to compress the conditional probability $p(x_t | x_{1:t-1})$, where one has $p_\theta(x) = \prod_{t=1}^{T} p_\theta(x_t | x_{1:t-1})$. This is referred to as an autoregressive model. Parameters are usually shared across the dimensions of $x$, since $T$ may vary, for example in the case of recurrent neural networks or convolutional neural networks, which have been empirically demonstrated to possess good inductive biases for tasks involving images (Van den Oord et al., 2016b) and speech data (Van den Oord et al., 2016a). However, sampling is sequential, requiring $T$ passes, which is why autoregressive models are slow to sample from, making them impractical for tasks requiring sampling, such as speech generation. This has motivated the work of Van den Oord et al. (2018), who propose probability density distillation to learn a student network with a structure that allows for parallel sampling, by distilling a state-of-the-art autoregressive teacher into it.

3 PROBABILITY DISTILLATION WITH NORMALIZING FLOWS

Van den Oord et al. (2018) propose to distill the probability distribution parameterized by a WaveNet model (the teacher network, denoted by $T$) by minimizing its reverse-KL with a student network (denoted by $S$):

$$D_{KL}(p_S \parallel p_T) = \mathbb{E}_{x \sim p_S} [\log p_S(x) - \log p_T(x)]$$ \hspace{1cm} (2)

The idea is to leverage recent advances in change of variable models (also known as normalizing flows) (Rezende & Mohamed, 2015; Kingma et al., 2016; Huang et al., 2018a; Berg et al., 2018) to parallelize the computation of the sampling process.

First, the student distribution is constructed by transforming an initial distribution $p_S(z)$ (e.g. normal or uniform distribution) in a way such that each dimension $x_t$ in the output $x$ depends only on up to $t$ preceding variables (according to a chosen ordering) in the input $z$:

$$z \sim p_S(z), \quad x_t \leftarrow f_t(z_t; \pi_t(z_1, ..., z_{t-1})), \forall t, \hspace{1cm} (3)$$

where $f_t$ is an invertible map between $x_t$ and $x_t$. Unlike the sampling process of an autoregressive model, where one needs to accumulate all $x_{1:t-1}$ to sample $x_t$ from $p_T(x_t | x_{1:t-1})$, which scales $O(T)$, the transformations $f_t$ can be carried out independently of $t$, allowing for $O(1)$ time sampling due to the parallel computation.

Second, the entropy term of $p_S$ can be estimated using the change of variable formula:

$$\mathbb{E}_{x \sim p_S(x)} [\log p_S(x)] = \mathbb{E}_{z \sim p_S} \left[ \log p_S(z) \left| \begin{array}{c}
\frac{\partial f(z)}{\partial z}^{-1}
\end{array} \right. \right]$$ \hspace{1cm} (4)

where $f$ is the multivariate transformation $f(z) \!=\! f(z_1; z_<t)$. Furthermore, owing to the partial dependency of $f_t$, $f$ has a triangular Jacobian matrix, reducing the computation of the log-determinant of the Jacobian in Eq. (4) to linear time:

$$\left| \frac{\partial f(z)}{\partial z} \right| = \prod_t \frac{df_t(z_t; z_<t)}{dz_t}. \hspace{1cm} (5)$$

Finally, when the teacher network has a tractable explicit density, one can evaluate the likelihood of samples drawn from $p_S$ under $p_T$ efficiently (in the case of autoregressive models such as WaveNets, one can use teacher forcing, or equivalently, change of variable formula, to compute log-likelihood in parallel).

3.1 ANALYSIS OF THE CURSE OF DIMENSIONALITY

Assume the student density $p_S$ lies within a certain family $Q$. Ideally, if $p_T \in Q$, the solution to the minimization problem in Equation (2) would be $p_S = p_T$. However, this is not trivial in practice when an oracle solver
does not exist, forcing us to resort to stochastic optimization to update $p_S$ iteratively. In this section, we identify a failure mode of distillation when stochastically minimizing the reverse-KL with gradient descent.

Empirically, there are two stages during stochastic minimization of the reverse-KL:

(i) $p_S$ starts to fit to the mode of $p_T$, and
(ii) $p_S$ gradually expands from the mode of $p_T$ to fit the shape of the distribution.

Stage (i) is fast due to the well-known zero forcing property of the reverse-KL (Minka et al., 2005). Figure 1.3 of Turner & Sahani (2011) shows that $p_S$ tends to be more concentrated when it is assumed to be fully factorial (independent). We show that even when $Q$ contains $p_T$, stochastic optimization can be slow in stage (ii) and thus result in a more concentrated, suboptimal $p_S$. This implies it is not only a matter of the assumption on the family of $p_S$, but on how we optimize it.

Intuitively, for each sample $x$ drawn from $p_S$ using reparameterization, the negative gradient of the integrand in Equation (2) with respect to $x$ is made up of two counteracting factors: one that pushes $x$ away from the mode of the student density $p_S$ (max entropy) and one that pulls $x$ towards the mode of the teacher density $p_T$ (min energy). These two counteracting terms form the “error” component of the path derivative (Roeder et al., 2017):

$$
\nabla_x (\log p_{S}(x) - \log p_{T}(x)) \nabla_{\phi} f_{\phi}(z), \ x = f_{\phi}(z), \quad (6)
$$

when the student is updated (see Appendix B for more details). Hence, there is an intrinsic exploration-exploitation tradeoff in the nature of this method. We show below that this learning algorithm tends to exploit more: the gradient of a sample $x$ is much more likely to point towards the high density region under the teacher, which means $-\log p_T$ dominates, collapsing the student.

To analyze the efficiency of training with reverse-KL, we consider the case when both the student and teacher are multivariate normal centered at the origin, as a model of the problem. Assume without loss of generality\(^1\) that $p_S = \mathcal{N}(0, I)$ and $p_T = \mathcal{N}(0, \Sigma_T)$ (both centered at 0 to analyze the efficiency of stage (ii)), where $I$ is an $n$-by-$n$ identity matrix and $\Sigma_T \in S^+_n$ is a positive definite matrix.

Let $g_x \doteq \nabla_x (\log p_T(x) - \log p_S(x))$ be the negative gradient \(\text{wrt}\) the random variable $x \sim p_S$. We are interested in the event $\{x^T g_x > 0\}$, which we call the \textit{expansion signal}, as it denotes the event of the gradient pointing away from the teacher’s mode, thereby helping the student expand its probability mass. The following proposition establishes the connection between the eigenvalues of the covariance matrix $\Sigma_T$ and the probability of the expansion signal.

\textbf{Proposition 1.} Let $p_S = \mathcal{N}(0, I)$ and $p_T(0, \Sigma_T)$. Draw $x \sim p_S$. Let $A_U$ be the surface area of the unit sphere $U = \{x: \|x\| = 1\}$, and $A_{U_{U^T}}$ be the surface area

\[^1\text{When the covariance matrix of the student distribution } \Sigma_S \text{ is not an identity matrix, one can transform both } p_S \text{ and } p_T \text{ via the change of variable: } x' = U^T x \text{ where } \Sigma_S = U^T U \text{ is the Cholesky decomposition of the covariance matrix; such that } p_S(x') \text{ is standardized, } p_T(x') \text{ has a “relative” covariance (due to the rotation under } U^{-1}), \text{ and our analysis carries on.}\]
of \( \{ x \in U : \sum_i \rho_i x_i^2 > 0 \} \). Then the probability of 
\( \{ x^T g_x > 0 \} \) is given by 
\[
\frac{A_{U \cap p}}{A_U} \quad (7)
\]
where \( \rho_i = 1 - \frac{1}{d^2_i} \) and \( d_i^2 \) is the \( i \)-th eigenvalue of the 
covariance matrix.

**Proof.** By definition, we have \( g_x = -\Sigma_T^{-1} x + x \). Let \( \Sigma_T = \Lambda D \Lambda^{-1} \) be the eigen-decomposition of the 
covariance, where \( D_{ii} = d_i^2 \) is the \( i \)-th eigenvalue and the 
columns of \( \Lambda \) are the eigenvectors. Due to the rotational 
invariance and the uniformity of the density of the standard 
normal \( p_S \) on the level set \( \{ x : \| x \| = 1 \} \), we can decompose 
the probability mass outward from the mode as 
\[
P \left( \left\{ x^T g_x > 0 \right\} \right) = P \left( \left\{ x^T \Lambda (\mathbb{I} - D^{-1}) \Lambda^{-1} x > 0 \right\} \right) = \frac{A_{U \cap p}}{A_U}.
\]

**Illustration of the proposition.** What the proposition implies is that the chances of receiving a gradient signal 
that points outward depend on the eigenvalues of the 
covariance matrix of the teacher: the greater the number of eigenvalues that are smaller than 1 (more ill-conditioned), the lower the chances. This means that the expansion signal via the path derivative can be increasingly unlikely when the dimensionality in \( x \) grows and \( p_T \) is highly structured. Consider Figure 1a for example, where the solid contour plot and dashed contour plot represent the densities of \( p_T \) and \( p_S \), respectively. For a random sample drawn from \( p_S \), marked by the yellow star, the gradients of \( \log p_T \) and \( -\log p_S \) with respect to \( x \), parallel to the red and blue arrows. The net \( g_x \) here can be decomposed into two parts: one that is 
perpendicular to \( x \) and \( g_x \), and one that is parallel with \( x \), \( g_x \). In this example, \( x \) and \( g_x \) point in opposite directions, meaning the back-propagated signal would pull \( x \) towards the mode of \( p_T \).

**Asymptotics of teacher density.** On average, the chances of getting a stochastic gradient signal that push the points away from the mode is the fraction of the area of the unit sphere intersecting with the hypercone, represented by the shaded area in Figure 1b. In practice, such a condition coefficient can be very small, as it is well known that a high dimensional distribution over 
structured data is effectively low-rank. This makes it harder 
for \( p_S \) to expand its probability mass along the high density 
manifold under \( p_T \). In fact, the smallest eigenvalue of a normalized (scaled by \( 1/T \)) Wishart distribution converges almost surely to zero as the dimensionality \( T \) approaches infinity. For a more general depiction of the asymptotic distribution of the eigenvalues, see the Marchenko-Pastur Law (Marchenko 
& Pastur, 1967).

Although our analysis here deals with the Gaussian case for ease of analysis, the theme extends to any degenerate 
distribution. For realistic data distributions, such as natural images, the data usually lies within low-dimensional 
manifolds (Carlsson et al., 2008; Fefferman et al., 2016; Narayanan 
& Mitter, 2010), and our arguments apply generally for such cases.

### 3.2 EMPIRICAL DEMONSTRATION

We showed above that with increasing dimensionality, and for a structured teacher, there is a diminishing probability of gradient signals that can push the student to expand around the mode of the teacher. We speculate that the distilled density of the student will therefore be collapsed around the mode of the teacher density, resulting in student samples having higher likelihood under the teacher than a “typical” sample drawn from the teacher would normally have. We validate this hypothesis in the following experiment.

We take both \( p_T \) and \( p_S \) to be multivariate Gaussian distributions, with the sampling process defined as \( x \leftarrow \mu + R \cdot z \) where \( \mu \in \mathbb{R}^T \), \( R \in \mathbb{R}^{T \times T} \) and \( z \sim \mathcal{N}(0, I) \). We randomly initialize each element of \( R \) for \( T \) independently according to the standard Gaussian, set \( \mu = [2, ..., 2]^T \) to be a vector of \( T/2 \)‘s, and fix them while training \( S \) to distill \( T \). For \( T \in \{4, 16, 32, 64\} \), training proceeds as follows: we sample \( x \) from the student, estimate \( \log p_S(x) \) using the change of variable formula, and evaluate \( x \) under \( \log p_T \). We use a minibatch size of 64 and learning rate of 0.005 with the Adam optimizer (Kingma 
& Baj, 2014), and make 5000 updates. For evaluation, we draw 1000 samples from both \( p_T \) and \( p_S \), and display the empirical distribution of \( \log p_T(x) \) in Figure 2a.

First, we observe that the log-likelihood of the teacher samples deviate from 0 as dimensionality grows. In fact, 
assuming \( x_1 \) is sampled i.i.d. (for simplicity) from a distribution whose second moment \( \mu_2 \) exists, the \( l_2 \) norm of \( \tilde{x} \) would almost surely converge to \( \mu_2 \), by the strong law of large numbers. This is a phenomenon known as the concentration of measure. To see this, observe that:

\[\text{Wishart is the conjugate prior of the precision matrix (inverse of covariance) of a multivariate Gaussian}\]
Figure 2: We distill a Gaussian teacher with a Gaussian student. $x$-axis: likelihood under the teacher; $y$-axis: count of samples drawn from the teacher (real samples) and the learned student (generated samples). (a-d) in the subfigures correspond to \{4, 16, 32, 64\} - dimensional multivariate Gaussians.

\[
\|\bar{x}\|^2 = \bar{x}^T \bar{x} = \sum_{t=1}^T x_t^2 = \sum_{t=1}^T \left( \frac{x_t}{\sqrt{T}} \right)^2 \tag{8}
\]

\[
= \frac{1}{T} \sum_{t=1}^T x_t^2 \to m_2, \text{ a.s. as } T \to \infty. \tag{9}
\]

The concentration is due to the compromise between density and volume of space (which vanishes exponentially as dimensionality grows). The consequence is that when one samples from a high dimensional Gaussian, the norm of the sample can be well described a constant, which means one is effectively sampling from the shell of the Gaussian ball.

Second, with an increasing number of dimensions, we observe that $p_S$ does indeed concentrate more on the high density region of $p_T$. This suggests that the imbalanced gradient signal poses an optimization problem for distilling. To validate this, we repeat the experiment 8 times, and estimate the probability of the gradient signal pointing away from the mode of the teacher throughout the training of the student by drawing 128 samples at each time step (averaging out all 1,024 binary values).

The resulting plots are shown in Figure 3. The sudden increase in the probability at the initial stage indicates that the student quickly fits to and concentrates around the mode of the teacher, as per stage (i). After the student expands to a reasonable size and shape, per stage (ii), the probability drops and getting a signal along the thin manifold under the teacher density becomes increasingly unlikely as dimensionality grows, which is consistent with Proposition 1.

Additionally, the mismatch of norm (after centering, i.e. likelihood) in Figure 2a implies that the use of KL would result in a mismatch of certain important statistics (such as norm of the samples, which is a perceivable feature in images and audio frames) even when $p_S$ is fairly close to $p_T$.

Figure 3: Estimate of probability of expansion signal throughout training of the student (with reverse-KL loss). The legend indicates the dimensionality of the problem.

Finally, in the above study, we only identify this optimization difficulty in the case of Gaussian teacher and Gaussian student. However, it is also well known that the reverse-KL tends to be mode-seeking (see Figure 4b,4c for example), and is not well-suited for learning multimodal densities (Turner & Sahani, 2011; Huang et al., 2018b).

4 PROBABILITY DISTILLATION WITH INVERSE MATCHING

In this section, we discuss possible alternatives for distilling a teacher. We assume there exists an invertible mapping from a prior space $Z$ to the data space $X$, such that one can trivially sample from a prior distribution $z \sim p_T(z)$ and pass the sample through this invertible map such that the sample is distributed according to $p_T(x)$. For Gaussian conditional autoregressive models, for example, one would sequentially pass scalar standard Gaussian noise $z_t$ through the following recursive function $x_t = \mu_t(x_{1:t-1}) + \sigma_t(x_{1:t-1}) \cdot z_t$. For notational convenience, we denote the “inverse” of this transforma-
Proposition 2. Interestingly, activity of \( d \) = \( \text{distillation with oracle prediction} \). However, preparing such a dataset of \( T^{-1}(z) \) samples would typically be time-consuming. We present the following two alternatives.

1. Distillation with \( z \)-reconstruction. We consider minimizing \( d(z, \mathcal{T} \circ S(z)) \), which is a reconstruction loss and the student network and teacher network are viewed as the encoder and decoder, respectively. In this case, \( \mathcal{T} \) is invertible and fixed, the only functional form of \( S \) that gives zero reconstruction would be \( T^{-1} \), which means the random variable \( S(Z) \) should also be distributed according to \( p_T \). In fact, minimizing the \( z \)-reconstruction loss corresponds to a parametric distance induced by the teacher network. Define \( d_T(a, b) = d(\mathcal{T}(a), \mathcal{T}(b)) \), where \( d \) is a distance metric. Then

\[
d_T(T^{-1}(z), S(z)) = d(\mathcal{T} \circ T^{-1}(z), \mathcal{T} \circ S(z)) = d(z, \mathcal{T} \circ S(z)).
\]

Interestingly, \( d_T \) is also a metric:

**Proposition 2.** \( d_T \) is a metric if and only if \( \mathcal{T} \) is injective.

**Proof.** Trivially, positive-definiteness and symmetry are inherited from \( d \) if and only if \( \mathcal{T} \) is an injection. To see that subadditivity is also preserved, for some \( a, b \) and \( c \), let \( \mathcal{T}_a = \mathcal{T}(a) \), \( \mathcal{T}_b = \mathcal{T}(b) \) and \( \mathcal{T}_c = \mathcal{T}(c) \). Since \( d(\mathcal{T}_a, \mathcal{T}_b) \leq d(\mathcal{T}_a, \mathcal{T}_c) + d(\mathcal{T}_b, \mathcal{T}_c) \), due to the subadditivity of \( d \), for any \( \mathcal{T}_a, \mathcal{T}_b \) and \( \mathcal{T}_c \), we have \( d_T(a, b) \leq d_T(a, c) + d_T(b, c) \) for any \( a, b \) and \( c \).

This means that the \( z \)-reconstruction loss behaves like a distance between \( T^{-1}(z) \) and \( S(z) \). So when \( z \)-reconstruction is minimized, it implies \( S \) gets closer to \( T^{-1} \) in the sense of the induced (parametric) metric \( d_T \). However, this parametric metric is not necessarily good, which we will demonstrate empirically in Section 4.1. A potential failure mode for it is when the teacher has extremely high uncertainty, e.g. large standard deviation for the conditional gaussian distribution, which would result in an inverse mapping (scaled by \( 1/\sigma \)) with an extremely small slope.

2. Distillation with \( x \)-reconstruction. Finally, we consider minimizing the reconstruction loss \( d(x, \mathcal{S} \circ \mathcal{T}(x)) \), where \( x \sim p_D \), the (empirical) data distribution, treating the teacher network as the encoder, and the student network as the decoder. When the teacher density coincides with the underlying data distribution, this would be equivalent to training with oracle prediction, as \( \mathcal{T}(X) \) would be distributed according to \( p_T \). This is a reasonable assumption when \( p_T \) approximates \( p_D \) well, and this is in fact true as \( p_T \) is usually trained with maximum likelihood under \( p_D \). This training criterion is more “instructive” in the sense that the student network is shown what a typical sample looks like, whereas both reverse-KL and \( z \)-reconstruction rely on “evaluating” the samples drawn from the student network, and then correcting the student based on the (possibly imperfectly calibrated) score assigned by the teacher. Hence, \( x \)-reconstruction does not suffer from the exploration problem intrinsic to the evaluative training of reverse-KL minimization and \( z \)-reconstruction.

Now we revisit the two essential components required for distillation with reverse-KL.

1. Invertibility: None of the three training criteria we explored involves estimating the entropy of \( p_S \), so in principle, we do not require invertibility of the student. In fact the entropy of \( p_S \) is implicitly maximized since \( \mathcal{T} \) is bijective. To prevent degenerate \( p_S \), one simply needs to avoid using hidden units of dimensionality smaller than the input size without skip connectivity, which compresses the noise.

2. Differentiability: For distillation with oracle prediction and \( x \)-reconstruction, we only require the translation between \( X \) and \( Z \), via \( \mathcal{T} \) and \( T^{-1} \), which is readily accessible for many standard distributions, e.g. linear map between Gaussians (both \( \mathcal{T} \) and \( T^{-1} \)), logistic-linear map from mixture of logistics to uniform (\( \mathcal{T} \) only, but sampling is achievable by sampling the mixture component first), and neural transformation (Huang et al., 2018a) (\( \mathcal{T} \) only). We also note that it is possible to recover uniform density from discrete data, by injecting noise proportional to the probability per class to break ties when passing the data through the cumulative sum of the probability (CDF).

4.1 EXPERIMENTS

4.1.1 Linear model with increasing dimensionality

We replicate the experiment in Section 3.2 with \( z \)-reconstruction and \( x \)-reconstruction loss (equivalent to oracle prediction in this case). Mapping \( \mathcal{T} \) from \( X \) to \( S \) is simply inverse of the sampling transformation. In Figure 2 we observe that both models outperform distil-
4.1.2 Distillation with different losses

In this section, we distill PixelCNN++ (Salimans et al., 2017) teacher networks trained on the MNIST handwritten digits dataset (LeCun et al., 1998) and the Fashion-MNIST dataset (Xiao et al., 2017). We trained the teacher model for 100 epochs and distilled it into the student with another 100 epochs of updates, using minibatch size of 64, learning rate of 0.0005 for the Adam optimizer with a decay rate of 0.95 per epoch, 3 ResNet blocks per downsampling and upsampling convolution, 32 hidden channels, and a single Gaussian conditional. The data is preprocessed with uniform noise between pixel values and rescaled using the logit function. We use $l_2$ loss for the reconstruction and prediction methods.

First, we observe that when trained with the reverse-KL loss, the students collapse on undesirable modes. As shown in (Figure 4b), the inductive bias of the causal convolution leads to higher density of the samples with striped textures. When trained with the $z$-reconstruction loss, the MNIST student samples all collapse to the same

Figure 4: Density distillation of teacher models trained on MNIST (first row) and Fashion-MNIST (second row). Column 1: samples from teacher network $T$. Column 2: samples from student trained with the KL loss $S_{KL}$. Column 3: samples from student trained with the $z$-reconstruction loss $S_z$. Column 4: samples from student trained with the $x$-reconstruction loss $S_x$. Column 5: samples from student trained with the $x$-reconstruction loss where $x$ is sampled from the teacher $S_O$.

Figure 5: Experiments with a ResNet student using (a) $l_2 x$-reconstruction loss, (b) $l_1 x$-reconstruction loss, (c) adding $N(0, 0.5)$ noise to the encodings $z$ and using $l_2 x$-reconstruction loss, (d) adding $N(0, 0.5)$ noise to $z$ and using $l_1 x$-reconstruction loss, and (e) adding $N(0, 0.5)$ noise to $z$ and using a mixed loss $l_2 + l_1$. In general, we observe that adding noise significantly improves sample quality, and training with $l_1$ losses lead to sharper samples.
digit. Interestingly, when we visualize the corresponding $T^{-1}(z)$ as we slightly perturb the norm of $z$ (see Appendix A), we observe that the digits abruptly change identity. This suggests that when moving onto a different sublevel set of norm in $z$-space, the corresponding $x$ jumps from one digit manifold to another, and the direction of $z$ does not preserve digit identity. This might explain why the student collapses to a digit: this is due to the bad local minimum that corresponds to relatively low reconstruction cost in the $z$-space.

Next, we observe that the student trained with $x$-reconstruction loss (Figure 4d) does not have good quality samples while the reconstructions are visually perfect. We hypothesize this is due to the well-known problem of mismatch between the empirical distribution of the encodings $T(x)$ and the prior distribution $p_S(z)$ of training decoder based generative models (Kingma et al., 2016). We contrast this with a student trained on oracle predictions (Figure 4e) and observe that the latter's samples match the teacher’s samples better.

Finally, we see that the samples from the teacher trained on Fashion-MNIST with the $x$-reconstruction loss (Figure 4i) have a smoother texture than the one trained with oracle samples (Figure 4j), which again, are perceptually closer to samples from the teacher. We elaborate more on this discrepancy in the next section.

4.1.3 Learning to distill and learning to generate

Since we are not constrained in our modeling choice for the student, we experiment with a ResNet student which is trained to directly map an encoded datapoint from the teacher back to the datapoint. The ResNet is deep enough so that the receptive field at the output is sufficient to span all of the encoded $z$, so that far-away influence is still exploitable.

Using the $l_1$ reconstruction cost leads to sharper samples from the student (contrast 5a with 5b). It appears to us that the $l_2$ loss tends to maintain global details, while the $l_1$ loss can sometimes sacrifice global coherence for local structure, potentially due to the sparsity induced by it. In 5e, we use an average of both losses in an attempt to maintain both these characteristics, but we notice evidence of the failings manifesting to some extent as well.

A significant improvement in sample quality is observed upon adding Gaussian noise to the encodings before training the student (contrast 5a,5b with 5c,5d). Our intuition for this is as follows: providing a decoder network with pairs of points in the data space and the corresponding encoded (or latent) space would typically result in $z$-space not being sampled almost everywhere, since image data (and therefore the corresponding encoding in $z$-space) usually lies on a lower-dimensional manifold, as discussed earlier. Adding noise enhances the support of the distribution, effectively spreading the “responsibility” of an encoding to cover more volume, which smoothens the mapping learned by a decoder. When the goal is to sample from a prior, training methods that encourage such $z$-space-filling strategies and smoother mappings improve sample quality, which is reminiscent of decoder-based sampling models such as variational auto-encoders (Kingma & Welling, 2014).
This leads us to an important point about these experiments: since the student is trained on noised (encoded) points from the data distribution, this is no longer purely density distillation. The student no longer aims to reproduce the sampling behavior of the teacher (as in [4]), but rather uses the teacher to provide structural information through its encodings. This information, when “spread out” through noise-injection and used by a student to learn decodings into real data (through a reconstruction penalty or more sophisticated losses) results in a network that can now be considered as a stand-alone generator, with the teacher acting as an inference machine that preserves information in the latent space. This can potentially allow a student to outperform its teacher in terms of sample quality, by enabling the learning of a smoother mapping from z-space to data space.

As a demonstration on more realistic data, we train a PixelCNN on CIFAR-10 and distill it with a ResNet student using $l_1$-reconstruction with the $l_1$-loss, and noise injection (Figure 6). We observe that the student learns the desired sampling behaviour, and additionally, owing to the more instructive training with data samples, there is greater global coherence in the samples as compared to the teacher. There is an obvious relative lack of low-level precision, but we believe this comes primarily from the pixel-position uncertainty induced by the pixelwise loss, and could potentially be fixed by augmenting the loss with sophisticated image priors ([Ulyanov et al., 2018]).

4.1.4 Neural vocoder

Finally, we compare distillation with $x$-reconstruction and distillation with reverse-KL on the neural vocoder task for speech synthesis ([Sotelo et al., 2017]). The neural vocoder has been an essential component of many text-to-speech models proposed recently ([Wang et al., 2017]; [Shen et al., 2018]; [Arik et al., 2017]; [Ping et al., 2018]). We train our teacher to map vocoders ([Morise et al., 2016]) to raw audio using the SampleRNN model ([Mehri et al., 2016]). We model the conditional distribution of the teacher with a unimodal Gaussian distribution, which makes it easy to compute the corresponding $z$. We specifically compare against closed form regularized KL with Gaussian conditionals as proposed in [Ping et al., 2018]. We design our student network with a WaveNet architecture consisting of six flows, and perform sampling as in Parallel WaveNet ([Van den Oord et al., 2018]). Each flow is a dilated residual block of 10 layers with a convolution kernel width of 2, and 64 output channels. With the $x$-reconstruction method under an $l_1$-loss, we find that our student produces samples without the characteristic whispering of the reverse-KL trained student. We have uploaded samples for comparison at [https://soundcloud.com/inverse-matching/](https://soundcloud.com/inverse-matching/).

5 CONCLUSION

In this paper, we investigated problems with distilling an autoregressive generative model under a reverse-KL cost between the student and the teacher, where the student can perform efficient parallel generation. Specifically, we showed that distillation with the reverse-KL can suffer from imbalanced gradient signals due to the curse of dimensionality, making the expansion signal necessary for efficient exploration unlikely. Further, we explored different alternatives which work qualitatively better when compared against distillation with reverse-KL.

ACKNOWLEDGEMENTS

Chin-Wei would like to thank Shawn Tan, and Kris Sankaran for discussion and feedback, and David Krueger for contributing to the idea of minimizing $z$-reconstruction for distillation.

This work was enabled by the computational resources provided by Compute Canada, Element AI, and Lyrebird.

References


Figure 7: We randomly sample $z$ from $\mathcal{N}(0, I_{784})$ (for each row) and rescale the vector such that it has norm $r \cdot \sqrt{784}$, where $r \in [0.700, 0.750, 0.800, 0.850, 0.900, 0.920, 0.940, 0.960, 0.980, 0.990, 0.995, 1.000, 1.005, 1.010, 1.020, 1.040, 1.060, 1.080, 1.100, 1.150, 1.200, 1.250, 1.300]$, which correspond to the change along the horizontal axis. We observe that directional information does not preserve digit identity in the data space, and the manifold per digit can be stretched around the origin on different level sets of norm.
B PATH DERIVATIVE

When $x$ is real-valued and when reparameterization of the sample drawn from $p_S(x)$ allows for separation of a random variable $z \sim p(z)$ independent of the parameters of $p_S$ and a deterministic transformation $x = S_\phi(z)$, we can decompose the gradient of the KL divergence (2) with the parameters $\phi$ (of $S$) as (Roeder et al., 2017):

$$\nabla_\phi E \sim_{p_{S_\phi}(x)} [\log p_{S_\phi}(x) - \log p_T(x)]$$

$$= \nabla_\phi E \sim_{p(z)} [\log p_{S_\phi}(x) - \log p_T(x)]$$

$$= E \sim_{p(z)} [\nabla_\phi (\log p_{S_\phi}(x) - \log p_T(x))] ;$$

$$= E \sim_{p(z)} [\nabla_x (\log p_{S_\phi}(x) - \log p_T(x)) \nabla_\phi S_\phi(z) + \nabla_\phi \log p_{S_\phi}(x)]$$

where the last equality is the total derivative because the term $p_{S_\phi}(x)$ depends on $\phi$ through both the sample being evaluated, $x = S_\phi(z)$, and the evaluating log-likelihood function, $\log p_{S_\phi}$.

The score function is in expectation zero, since

$$\int p_{S_\phi}(x) \nabla_\phi \log p_{S_\phi}(x) dx = \int p_{S_\phi}(x) \frac{\nabla_\phi p_{S_\phi}(x)}{p_{S_\phi}(x)} dx = \nabla_\phi \int p_{S_\phi}(x) dx = \nabla_\phi 1 = 0,$$

so it can be thought of as a control variate: an unbiased term used to reduce variance in gradient estimate. The path derivative term measures the dependency on the parameters $\phi$ through the reparameterized sample $x = S_\phi(z)$. As a result, the gradient direction wrt the sample $x$ directly affects how the parameters of the distribution will be updated to change the shape of $p_{S_\phi}$. 