TOWARDS WELL-DISTRIBUTED GENERATIVE NETWORKS USING ADVERSARIAL AUTOENCODERS

Anonymous authors
Paper under double-blind review

1 A DISCUSSION OF EVALUATION METRICS

We observe that the FID does not correlate well to the reduced chi-square. This is consistent with the analysis by Kynkäänniemi et al. (2019), which showed that when varying the truncation parameter in image generation using StyleGAN, diversity is negatively correlated with quality, and suitably truncated models, which explicitly sacrifice diversity for a gain in quality, can produce a smaller FID than the untruncated model. This shows that FID is sensitive to both the distribution of features and image quality, which makes the value difficult to interpret.

They propose precision and recall instead. They construct a manifold from a set of points as follows: for each point, find the distance to the k-th nearest neighbour within the set. Take that distance as the radius of a hypersphere centered at this point. The manifold defined by a set of points is the union of the interior of these spheres. They chose $k = 3$.

Then, the set of training images and the set of generated images are mapped into the feature space of some pretrained network (VGG16). The manifold for the two sets of feature vectors are constructed. The precision is the proportion of generated feature vectors contained in the training manifold, and conversely for recall.

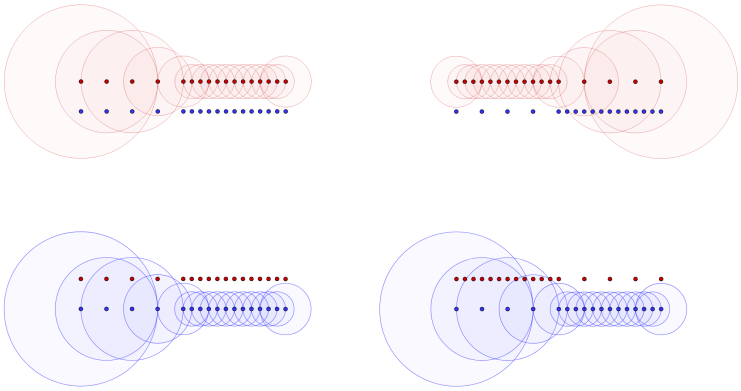
The behaviour of this pair of metrics is pathological on our dataset, as all models scored zero. We offer our explanation of this result. We think that this is because the VGG16 feature space used for computing the precision and recall is very high-dimensional (4096) while the distributions of the training and generated images are low-dimensional. In GANs, popular choice for the dimensionality of the latent space generally does not exceed 512, which is much smaller than 4096. Thus it is in general virtually impossible for the support of the generated image distribution to exactly align with that of the training images, as there is a high-dimensional subspace orthogonal to the data manifold at each point, and this means a lot of freedom for minor artifacts to move a image away from the data manifold. The 3DShapesHD dataset is particularly low-dimensional (5), and the samples will be densely packed in a thin slice in the VGG16 feature space, which yields small radii for the hyperspheres used for defining the training and generated manifolds, enough to cause the two manifolds to be disjoint, resulting in a precision and recall of zero.

Indeed, we can see that when two subsets of training images are tested against each other so that there is no artifact and the two manifolds are exactly aligned, the precision and recall are well-behaved.

We also show that when the training and generated manifolds are misaligned, it is possible for precision and recall to favor a distribution with a less accurate distribution. We show an example in figure 1. The training and generated manifolds are not exactly aligned. In (a), the generated distribution is offset from the training distribution but otherwise has the same density. The precision and recall are $\frac{4}{17}$. In (b), the density is less accurate, but the precision and recall increase to $\frac{11}{17}$.

In general, the radii of the hyperspheres is larger where the distribution is sparser. So, a generator that produces a low density where the training distribution is dense has a better chance of including more training samples further away, increasing the recall, while a generator that concentrates its samples near where the training distribution is sparse has a better chance of hitting the training manifold, increasing its precision. The result is that a generator that matches the density of the training distribution less accurately can actually score a higher precision and/or recall.

We argue that metrics that depend on mapping images into a high-dimensional deep feature space in general suffer from similar problems, which cause them to be over-sensitive to small details. On the other hand, the high-level features that are meaningful in human terms for defining the distribution of



(a) (b)

Figure 1: An example of a problematic case for the precision and recall metrics. Blue points are training samples, red points are generated samples, and circles represent the hypersphere centered at each point with radius equal to the distance to its third nearest neighbor.

images are dataset-specific, and a non-specific pre-trained network is unlikely to be able to capture them in general.

2 Dynamic Weight Adjustment

During training, the weight of the adversarial terms, the λ's, are adjusted dynamically using the following method: First, a target discriminator loss $\mathcal{L}^*$ is chosen. When the aggregate posterior and the prior are indistinguishable, which is the goal, the optimal discriminator assigns to every input a probability of being "real" of 0.5, and its loss is $-2 \ln 0.5 = 2 \ln 2$. We take $\mathcal{L}^* = 1.98 \ln 2$. Let the initial value of λ be $\lambda^{(0)}$, whose value is not very important.

Let $\mathcal{L}^{(t)}$ be the discriminator's loss at iteration t, and $\lambda^{(t)}$ the adversarial weight at iteration t. λ is updated according to

$$\lambda^{(t)} \leftarrow \lambda^{(t-1)} \cdot \min\left\{ \max\left\{ \frac{\mathcal{L}^*}{\mathcal{L}^{(t)}}, 2^{-\Delta} \right\}, 2^{\Delta} \right\}^{\frac{1}{T}} \tag{1}$$

T controls the long-term speed of change is set to 100. Δ controls the maximum magnitude of change per iteration and is set to 0.1.

Basically, we adjust the weight of the adversarial loss so that the discriminator's loss is pegged at a desired value. The adversarial weight increases when the discriminator loss is too low, and decreases when it is too high.

The advantage of this method is that the discriminator's loss is much easier to interpret than the raw value of the adversarial weight, and controlling the former offers a reasoned way to adjust the latter.

Note that this method works only if the encoder is able to make the aggregate posterior arbitrarily close to the prior, for otherwise the discriminator's loss may not reach the target however large the adversarial weight is. This is why we use the parametrized posterior in the encoder: so that it can easily make the posterior identical to the prior by setting the mean to 0 and variance to 1.

3 More Qualitative Results

All models compared perform well enough that visual difference between samples gernerated by the different methods are minor. However, we observe that our method offers better coverage over the part of data distribution that has certain uncommon features: for example, people wearing headwear in the FFHQ dataset and nighttime images in LSUN Church. We project test images with these features into the latent space of the generators, and compare our results with plain StyleGAN2, as

Figure 2: Comparison of projected images of input images of people wearing headwear.

Figure 3: Comparison of projected images of input images of churches at night.

shown in figures 2 and 3. The plain StyleGAN2 has a strong tendency to interpret the headwear as hair and reproduce a nighttime image with unnatural lighting, while our method handles these cases better.

REFERENCES

Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *Advances in Neural Information Processing Systems*, 32, 2019.