Precision Recall Cover: A Method for Assessing Generative Models

Fasil Cheema
Lassonde School of Engineering
EECS Department
York University, Toronto, Canada

Ruth Urner
Lassonde School of Engineering
EECS Department
York University, Toronto, Canada

Abstract

Generative modelling has seen enormous practical advances over the past few years. Evaluating the quality of a generative system however is often still based on subjective human inspection. To overcome this, very recently the research community has turned to exploring formal evaluation metrics and methods. In this work, we propose a novel evaluation paradigm based on a two way nearest neighbor test. We define a novel measure of mutual coverage for two probability distributions. From this, we derive an empirical analogue and show analytically that it exhibits favorable theoretical properties while it is also straightforward to compute. We show that, while algorithmically simple, our derived method is also statistically sound. In contrast to previously employed distance measures, our measure naturally stems from a notion of local discrepancy, which can be accessed separately. This provides more detailed information to practitioners for diagnosing where their generative models will perform well, or conversely where their models fail. We complement our analysis with a systematic experimental evaluation and comparison to other recently proposed measures. Using a wide array of experiments we demonstrate our algorithm's strengths over other existing methods and and confirm our results from the theoretical analysis.

1 INTRODUCTION

Generative modelling is a rapidly expanding field of machine learning, with increased interest since the introduction of Generative Adversarial Networks (GANs)(Goodfellow et al., 2014; Arjovsky et al., 2017; Zhang et al., 2018; Karras et al., 2019; Brock et al., 2019). As is typical for unsupervised machine learning tasks, evaluating the quality of a generative system is often still based on subjective human inspection rather than formal quantitative measures. To better guide the development of new models recent research has proposed novel evaluation metrics to quantitatively grade a model's performance (Sajjadi et al., 2018; Kynkäänniemi et al., 2019; Naeem et al., 2020; Djolonga et al., 2020; Borji, 2022). A promising line of work introduced the idea of measuring mutual coverage analogous to the statistical notions of precision and recall (Sajjadi et al., 2018). The main idea is to evaluate both the degree to which a generative model produces instances that are actually realistic (is the generative distribution covered by the true distribution?) and the degree to which it manages to reflect the full diversity of the true distribution that it is aiming to mimic (is the true distribution covered by the generative distribution?).

These approaches are appealing since they assess a generative model's performance through general algorithmic tools that are not tied to a specific parametric form of the generative distribution nor would they require direct access to the probability measure. While some generative models such as kernel density estimators (KDE) (Rosenblatt, 1956) or normalizing flows (Tabak and Vanden-Eijnden, 2010) construct explicit probability distributions, other types of generative models, such as GANs (Goodfellow et al., 2014) and variational auto encoders (VAEs) (Kingma and Welling, 2014) only implicitly correspond to a probability distribution. Not having direct access to this distribution makes grading such models in a non-subjective manner more challenging. The recent line of research on precision and recall type of metrics has thus aimed at developing methods to grade generative models directly based on the samples created (Sajjadi et al., 2018; Kynkäänniemi et al., 2019; Djolonga et al., 2020; Naeem et al., 2020).

The problem then becomes a way to assess the similarity two probability distributions, in a quantitative yet transparent way, with access to samples from the two distributions only. While the originally proposed notions of precision and recall for generative models was based on an intrigu-

Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS) 2023, Valencia, Spain. PMLR: Volume 206. Copyright 2023 by the author(s).

ing formal foundation (Sajjadi et al., 2018; Djolonga et al., 2020), it lacked a clear algorithmic way to instantiate the formalism. This again lead to the possibility of arbitrary choices affecting the overall outcome, and thus lacked transparency. Follow up studies have proposed capturing the idea of measuring precision and recall through nearest neighbor type tests, and thus improved the framework on the front of algorithmic clarity (Kynkäänniemi et al., 2019; Naeem et al., 2020). However, these latter frameworks lacked a concise formal backup with a provable relation to population level properties.

Our work aims to combine the algorithmic appeal of the nearest neighbor based methods with a statistically sound framework. We develop a novel version of the nearest neighbor type precision and recall measures, which we term *Precision Recall Cover (PRC)*. Our measure stems from a natural population level notion that captures the degree to which relevant areas of one distribution receive sufficient coverage by the other distribution. "Relevant" and "sufficient" are quantified by two parameters β and α that allow for fine-tuning the coverage requirements. Its empirical version then becomes a k, k'-nearest neighbor test, where coverage is defined by a k'-nearest neighbor ball from one distribution obtaining at least k sample points from the other distribution. We provide some basic formal analysis of both the population level and empirical version of the precision recall cover measure and thereby argue for its statistical soundness. Additionally we provide an experimental evaluation of our measure, comparing its performance as well as failure and success cases with the previous precision recall type measures. Finally, we provide a proof of concept of how our measure would assess the behavior of a GAN.

1.1 Related Work

In recent years the topic of metrics for generative models has received immense attention and there is now a significant amount of publications dedicated to this topic (Borji, 2022; Kynkäänniemi et al., 2019; Djolonga et al., 2020; Sajjadi et al., 2018; Naeem et al., 2020; Salimans et al., 2016; Heusel et al., 2017). We will focus our discussion here on evaluation measures most relevant to ours. One of the first scores to assess GANs were the Frechet Inception Distance (FID) (Heusel et al., 2017) and the Inception Score (IS) (Salimans et al., 2016).

FID models both distributions as having been generated by multivariate normal distributions (Gaussians). It then fits the sample sets to the respective Gaussians and measures the distance of the Gaussians based on their parameters. While it is still a popular metric, the quality of such a measure is inherently dependent on how appropriate the assumption of modelling the distributions as Gaussians was. For typical generative models designed to produce a variety of outputs, such modelling is likely not adequate.

A recent line of work has aimed at assessing generative models through concepts analogous to precision and recall. The first study in this line introduced a metric aimed at measuring how much each distribution covered the support of the other (Sajjadi et al., 2018). The work provides an elegant framework of measuring mutual coverage by means of representing the two distribution as mixtures of joint and exclusive components, the feasible sets of mixture coefficients then yielding a Precision Recall curve. They coin their work *Precision Recall* (PR). While formally very elegant, the framework does not come with a direct algorithmic analogue. The implementation of this framework thus provided a backdoor to algorithm choices influencing the actual behaviour of the measure.

Follow up studies have aimed at providing a remedy by basing the precision recall evaluation on nearest neighbor computations. A metric called *Improved Precision Recall (IPR)* (Kynkäänniemi et al., 2019) maintains the motivations of the original paper on precision-recall type metrics (Sajjadi et al., 2018) by seeking out regions where one distribution is covered by the other and vice versa. As a nearest neighbor based measure, IPR is naturally based on samples from both the true distribution P, and the generative distribution Q. However, IPR does not come with a population level analogue, and thereby does not allow for any analysis of statistical consistency.

One of the latest additions to this line of work models notions of *Density and Coverage (DC)*, again by means of a nearest neighbor based computation Naeem et al. (2020). The aim here was to improve over the previously proposed IPR, in particular IPR's sensitivity to outliers, by basing the measure on a weighted k-nearest neighbors computation. However, as IPR, DC also lacks a population level analogue.

1.2 Overview and Summary of Contributions

We formally define our novel notion of mutual coverage of two probability distributions based on the idea of assessing precision and recall of a generative model in Section 2. We provide some initial analysis of properties of this notion to motivate and justify our formalism in Section 2.2. More specifically, we show how suitable choices of the parameters α and β in our measure will allow identification of identical and disjoint supports as well as approximate measures of how much mass each distribution assigns to the intersection of their supports, their "joint support". We then provide an empirical analogue of our notion of precision recall cover and outline how to compute it in Section 2.3. We provide statistical guarantees for our empirical version in Section 3. Further, our analysis reveals an attractive local property. We show that the method will locally correctly identify regions of sufficient coverage. This may be used as a diagnostic tool when assessing the performance of a generative model. Finally, in Section 4 we present a variety empirical evalu-

ations and tests of our measure. Comparing to previously established k-nearest neighbor based methods, we establish that our measure exhibits favorable convergence properties and more robustness with respect to the choice of k thank the prior k-nn based measures. Further, we empirically demonstrate (again contrasting with the earlier k-nn based methods) that our measure yields correct assessments of real life (image) data and generative models (VAEs and GANs). For these, we systematically vary the quality of diversity of generated samples through a variety of means and show how our measure correctly assesses the changes. All proofs as well as details and additional analysis and empirical evaluations are in the appendix to save space.

2 PRECISION RECALL COVER

In this section we introduce some preliminaries and notation. Then we discuss motivation and consequently defined our precision recall cover measure for two distributions on the population level. In the next subsection, we then present an empirical analogue to our measure.

2.1 Setup and Definitions

We consider a space $X \subseteq \mathbb{R}^d$ and two distributions P and Q over X. We will think of these as the true data generating distribution P, and the generated distribution Q that is induced by some learned generative model. We use the notation $x \sim P$ to mean a point x sampled from distribution P. We use the notation $\hat{P} = (x_1, x_2, \ldots, x_n)$ and $\hat{Q} = (y_1, y_2, \ldots, y_m)$ for i.i.d. samples from P and Q respectively. The empirical distributions induced by the samples (uniform distributions over the sample points) are then denoted by P_n and Q_m. For simplicity of presentation, we will assume that both P and Q admit continuous density functions $d_P : X \to \mathbb{R}$ and $d_Q : X \to \mathbb{R}$ respectively. We will use the notation $\mathrm{supp}(P)$ and $\mathrm{supp}(Q)$ to denote the supports of the distributions P and Q respectively.

We will also use notation such as $Q(A)$ to denote the probability of event A occurring with respect to probability distribution Q. For our proposed evaluation measure, the events of importance will be balls in $\mathbb{R}^d$. We will let $B_P(x, \beta)$ denote the smallest ball of probability mass at least β with respect to probability distribution P centered at point x. Since we assume that our distributions have a continuous density function, there always exists balls $B_P(x, \beta)$ of mass exactly β. For the empirical distributions, $B_{\hat{P}}(x, \frac{k}{n})$ will denote a k-nearest neighbor ball with respect to sample set $\hat{P}$ around domain point x.

Several recent studies attempt to highlight the fidelity and diversity of generated samples (Naeem et al., 2020; Sajjadi et al., 2018; Kynkäänniemi et al., 2019). This typically entails measuring how much each distribution covers the support of the other. Usually, *precision* is defined as the portion of the support of Q that is in the support of P as well. Conversely, *recall* is defined as the portion of the support of P that is in the support of Q as well.

Our notions of *precision cover* and *recall cover* aim to capture the intuition that we should only care about a region being covered (by the other distribution) if it has a at least certain probability mass. In the context of a learned generative model, if an area has a negligibly small probability in terms of the true distribution, then we may not care about the generative model not being able to generate instances in that area. Conversely, if the generative model has a negligibly small probability of generating a certain type of instances, then we may not be too concerned about how likely such instances are generated by the true process.

These considerations are accounted for by the two parameters α and β in the definition (below) of our precision-recall cover measure. The parameters α and β provide "knobs" on the cut-off for an area to be considered "negligibly small" and correspondingly for coverage to be considered "sufficient".

Definition 1 (Precision Recall Cover (PRC))**.** *Let P and Q be two probability distributions over some space X, and let $\alpha, \beta \in [0, 1]$ with $\alpha \leq \beta$ be given. Then we define the* (α, β)-precision coverage of P by Q *by*

$$\mathrm{PC}_{\alpha,\beta}(P, Q) = \mathbb{P}_{y \sim Q}[P(B_Q(y, \beta))) \geq \alpha] \qquad (1)$$

Conversely, we define the (α, β)-recall coverage of P by Q *by*

$$\mathrm{RC}_{\alpha,\beta}(P, Q) = \mathbb{P}_{x \sim P}[Q(B_P(x, \beta))) \geq \alpha] \qquad (2)$$

The precision version of this measure takes two distributions, and for each point y from Q considers a ball of probability mass β. The measure reflects what portion of these balls have probability mass at least α with respect to P. The choice of α and β allow for generality and for the user to fine tune the sensitivity of the evaluation measure. For instance, perhaps our distribution P is completely disjoint from Q however the points from each distribution are very close in $\mathbb{R}^d$, we may not want to punish this behaviour as badly as for points that are completely disjoint and very far from each other. Recall is the analogous measure with the role of the two distributions swapped.

2.2 Population Level Properties of Precision Recall Cover

To further motivate our proposed measures, we start by stating some simple properties of our proposed measure.

Observation 1. *Let $X \subseteq \mathbb{R}^d$ be some domain and P and Q be two distributions over X. Then:*

1. $\mathrm{PC}_{\alpha,\beta}(P, P) = \mathrm{RC}_{\alpha,\beta}(P, P) = 1$ *for all* $0 < \alpha \leq \beta \leq 1$

2. $\mathrm{PC}_{\alpha,\beta}(P,Q) = \mathrm{RC}_{\alpha,\beta}(Q,P)$
3. $0 \leq \mathrm{PC}_{\alpha,\beta}(P,Q) \leq 1$ *and* $0 \leq \mathrm{RC}_{\alpha,\beta}(P,Q) \leq 1$ *for all* $0 < \alpha \leq \beta \leq 1$
4. *If* $\mathrm{supp}(P) \cap \mathrm{supp}(Q) = \emptyset$*, then there exist* $\alpha \leq \beta$ *sufficiently small such that* $\mathrm{PC}_{\alpha,\beta}(P,Q) = 0$ *and* $\mathrm{RC}_{\alpha,\beta}(P,Q) = 0$

The first three points above follow directly from Definition 1. The last statement shows that, as the parameters α and β get smaller (the measure thus more refined) PRC will correctly identify that distributions have disjoint support. Moreover, as we will show next, for sufficiently small α and β, PRC will also arbitrarily well approximate how much mass Q assigns to the support of P and vice versa.

Theorem 2. *Let P and Q be distributions over a space X. For any $\epsilon > 0$ there exist (sufficiently small) values $0 \leq \alpha < \beta \leq 1$, such that*

$$|\mathrm{PC}_{\alpha,\beta}(P,Q) - Q(\mathrm{supp}(P))| \;\leq\; \epsilon$$

and

$$|\mathrm{RC}_{\alpha,\beta}(P,Q) - P(\mathrm{supp}(Q))| \;\leq\; \epsilon.$$

2.3 Empirical PRC (and Algorithm)

In practice, we generally do not have access to the actual distributions P and Q, but rather have samples $\hat{P} \sim P^n$ and $\hat{Q} \sim Q^m$ from them. We now define an empirical analogue of the PR-cover measure.

Given n samples $\hat{P} = (x_1, \ldots x_n)$ and m samples $\hat{Q} = (y_1, \ldots, y_m)$, we can apply our measure over the discrete distributions that are induced by sample sets. If we use $\alpha = \frac{k}{m}$ and $\beta = \frac{k'}{n}$ where $k' = Ck$ for some $C \in \mathbb{N}$, the measure will be based on k- and k'-nearest neighbor balls over the sample sets.

Definition 2 (Empirical (k,k')-PRC)**.** *Let $\hat{P} = (x_1, \ldots x_n)$ and $\hat{Q} = (y_1, \ldots, y_m)$ be two sample sets, and let $k, k' \in \mathbb{N}$ with $k \leq k'$ be given. Then we define the* (k,k')-Precision Coverage ((k,k')-PC) of $\hat{P}$ by $\hat{Q}$ *by*

$$\mathrm{PC}_{k,k'}(\hat{P},\hat{Q}) = \frac{1}{m}\sum_{j=1}^{m} \mathbb{1}\left[P_n(B_{\hat{Q}}(y_j, \frac{k'}{m})) \geq \frac{k}{n}\right] \quad (3)$$

Conversely, we define the (k,k')-Recall Coverage ((k,k')-RC) of $\hat{P}$ by $\hat{Q}$ *by*

$$\mathrm{RC}_{k,k'}(\hat{P},\hat{Q}) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}\left[Q_m(B_{\hat{P}}(x_i, \frac{k'}{n})) \geq \frac{k}{m}\right] \quad (4)$$

In the next section, we will analyze the relationship between the true (population level) and empirical (sample based) version of our proposed measure.

Algorithm The above definition of the empirical measure gives rise to a straightforward algorithm that computes the (k,k')-coverage between two data sets $\hat{P}$ and $\hat{Q}$. For the precision version, it constructs k'-nearest neighbor balls over the sample set $\hat{Q}$ and then computes the total number of k'-nearest neighbor balls that contain at least k points from $\hat{P}$, and divides this number by the total number of sample points in $\hat{Q}$. For the recall version the roles of $\hat{P}$ and $\hat{Q}$ are swapped.

3 ANALYSIS

We now outline some of the key formal performance guarantees for the empirical (algorithmic) (k,k')-version of the PRC measure in relation to the population level version. We will phrase all our results in terms of the precision cover measure. By symmetry, they will hold for recall cover when the arguments are switched. All proofs have been moved to the appendix. The results are obtained by techniques of applying VC-analysis to k-nearest neighbour balls (Vapnik and Chervonenkis, 1971; Kpotufe, 2011; Dasgupta and Kpotufe, 2014; Berlind and Urner, 2015).

3.1 Identical Distributions

We start by presenting a type of sanity check analysis for the cases where the two distributions are identical. In that case, we have $\mathrm{PC}_{\alpha,\beta}(P,Q) = \mathrm{PC}_{\alpha,\beta}(P,P) = 1$ for all α, β, see Observation 1. We show that, given sufficiently large samples, the empirical k,k'-cover measure will also attain value 1 with high probability.

Theorem 3. *Let $\delta > 0$, $X \subseteq \mathbb{R}^d$ a domain and $\hat{P}$ and $\hat{Q}$ denote samples from distributions P and Q over X of sizes n and m respectively. Then there exist finite sample sizes N and M, depending only on the dimension of the space and the confidence parameter δ, and appropriate choices of k and k' such that the following holds: if $P = Q$, with probability at least $1 - 2\delta$, over the sampling of $\hat{P}$ of size $n \geq N$ and $\hat{Q}$ of size $m \geq M$, we have*

$$\mathrm{PC}_{k,k'}(\hat{P},\hat{Q}) = 1$$

3.2 Local Consistency

Of course, in general, we don't expect the two input distributions to be exactly identical. We now show that our k,k' mechanism will correctly identify local regions that are sufficiently covered by the other distribution: For a given local density ratio $\frac{P(B_{\hat{Q}}(x,\frac{k'}{m}))}{Q(B_{\hat{Q}}(x,\frac{k'}{m}))} > \omega$ (according to the true distributions) there is a sample size such that all such regions are correctly identified as covered by our empirical measure.

Theorem 4. *Let $\delta > 0$, $C > 1$, $X \subseteq \mathbb{R}^d$ a domain and $\hat{P}$ and $\hat{Q}$ be samples from distributions P and Q over X*

of sizes n and m respectively. Let $\omega > 0$, $k' = Ck$ and $k > 9((d+1)\ln(2m) + \ln(\frac{8}{\delta}))$, and

$$n \geq \frac{72\ln(8/\delta)}{C\omega}\ln(\frac{9m}{C\omega}).$$

Then with probability $1-2\delta$ all points $x \in \hat{Q}$ with $\frac{P(B_{\hat{Q}}(x,\frac{k'}{m}))}{Q(B_{\hat{Q}}(x,\frac{k'}{m}))} > \omega$ will be identified as "covered"; that is their k' nearest neighbor balls in $\hat{Q}$ will contain at least k points from $\hat{P}$.

The above theorem shows that our empirical measure satisfies an attractive local property. It will locally correctly identify sample points as "covered" if they fall into regions that are actually covered by the other distribution. Such a local property naturally leads to a diagnostic tool: If a sample point is not identified as covered by the empirical measure, the above theorem allows a user to infer a deficiency on the population level in terms of coverage. This might be used to identify regions where a generative model generates instances that are unrealistic (generative distribution locally not covered by the true distribution) or true instances that can not be mimicked by the generative model (true distribution locally not covered by the generative distribution).

3.3 Support Consistency

Next we show that there is a number of samples N from distribution P that will fully cover the samples of a certain size from Q that fall into the joint support. The following theorem states this more precisely:

Theorem 5. *For $\delta > 0$, and let m, k, and k' satisfy the conditions of Theorem 4. Then there exists a value N for the number of samples such that for all $n > N$, with probability $1-3\delta$ over the generated samples, $\hat{P}$ will cover all points from $\hat{Q}$ that fall into* $\mathrm{supp}(P) \cap \mathrm{supp}(Q)$.

As for the other results in this section, note that due to symmetry the theorem above holds when true and generative distribution are swapped. For that case the above statement provides an attractive guarantee: For a given dataset size from the true distribution, if we generate sufficiently many points from the generative model's indued distribution, then all points in the intersection of the two supports will be covered that is, our measure will eventually correctly identify all those points from the true distribution that can also be produced with the learned model.

4 EXPERIMENTS

In this section we present and discuss a variety of experiments that showcase our measure's behaviour alongside other metrics. This experimental section aims to supplement our theoretical analysis. We present two types of experimental setups: First, through carefully designed tests on synthetic data, we compare our algorithm's (ie. empirical measure's) behavior to previously established measures (Subsection 4.1). In particular, we highlight some failure cases of the earlier measures where our metric succeeds as well as compare the approximation quality in terms of convergence to ground truth. Second, we establish that our measure provides correct assessments in real world settings (Subsection 4.2). Through a variety of experimental setups where we systematically vary the diversity or quality of one of the distributions, we show that our measure adequately follows our design. We do this (1) by tracing the training stages of a VAE, (2) by dropping digits from MNIST (Deng, 2012), and (3) by truncating a GAN or blurring images. In addition, we also verify that our (k, k')-measure is robust to varying the choice of k as long as the ratio between k and k' remains constant $1/3$. This is in contrast to earlier k-nn based measures whose values we show to be highly sensitive to the choice of k.

4.1 Experiments on Synthetic Data

Sanity checks We devised several simple toy experiments as sanity checks to compare the results of our and earlier measures in a simple and interpretable setting. For a variety of pairs of (true and generative) distributions we assessed their similarity through PR, IPR, DC, as well as our measure PRC. For this, we used a combination of normal and uniform distributions with varying the degree of overlap and dimensions ranging from 1 to 3. In the most benign settings, we see mostly consistent behaviour among all measures in terms of the degree of overlap, ie matching distributions get perfect scores and as the degree of overlap decreases scores get progressively worse reflecting this. However, the metrics exhibit drastically different behavior in select experiments. For some distributions with disjoint supports, IPR is particularly prone to being overly accepting of samples outside the distribution's support. We also provide an example where the disjoint supports are not recognized by any metric other than PRC. Details and illustration of these experiments have been moved to the appendix for space reasons.

Convergence quality We now present comparisons of the three nearest neighbor based measures, namely the improved precision/recall (IPR), density/coverage (DC), and precision/recall cover (PRC) in terms of the convergence behavior and quality of approximation to ground truth. By varying the degree of overlap of two uniform distributions, we establish settings of clear ground truth for precision and recall (namely the probability masses of the joint support (Naeem et al., 2020)(Kynkäänniemi et al., 2019)(Sajjadi et al., 2018)(Djolonga et al., 2020)). We use the same value of k for each measure (and the same k-nn computation), and we set the number of samples from the "true distribution" and "generative distribution" to be equal throughout

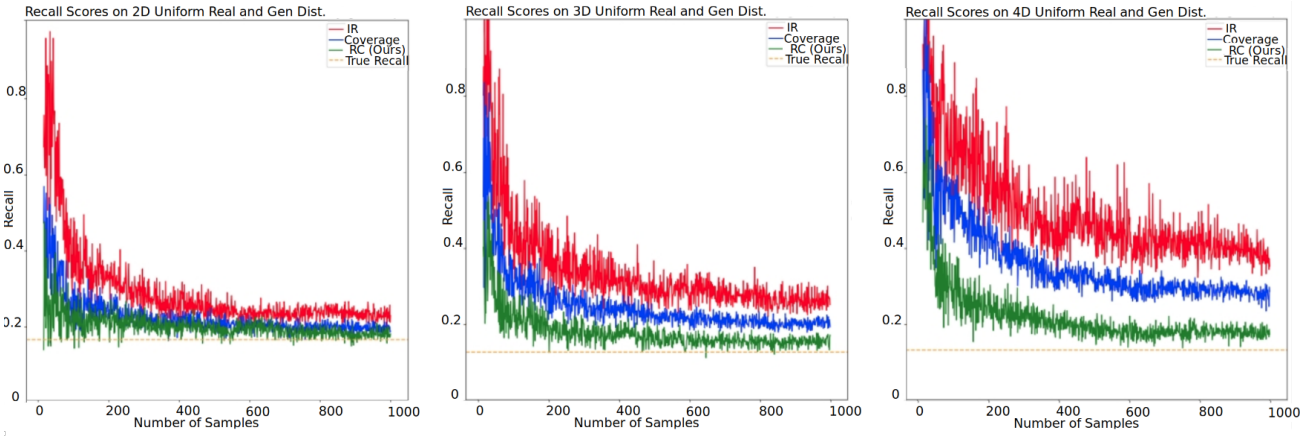

Figure 1: Recall convergence experiments for well defined uniform distributions. The orange dashed line is the true recall, the green line is our metric RC (from PRC), the blue line is Coverage (from DC), and the red line is improved recall (IR from IPR). From left to right the dimension increases from 2 to 4. These plots show the recall-type scores with changing number of samples. All measures converge to an approximation of the true recall (orange-dashed line), but as dimension increases our measure converges faster and to a better approximation.

these experiments. For PRC we use a value of k such that $k' = 3k$, as suggested by our analysis. Also, as guided by the analysis, we let the value of k (or k' in the case of PRC) change and grow logarithmically with the sample size; order $(\log(n))$ see appendix for more technical details.

The results for the recall version are shown in Figure 1. All measures shown converge to an approximation of the ground truth (orange line). However, as dimension increases our measure converges faster and clearly to a better approximation of the ground truth.

4.2 Experiments in Real World Settings

Application to VAEs In this section we present experiments on a popular generative model that, unlike GANs, gives rise to an explicit generative probability distribution. VAEs are trained by minimizing KL divergence and, if training succeeds, the quality of generated samples gets better as the KL divergence decreases. This suggests an intuitive set of experiments for the precision type measures: training on MNIST image data (Deng, 2012), we track the precision-based scores per training epoch as image quality gets better. Figure 2 shows that, among the k-nn based measures, our measure PC is the only one effectively reflecting the increase in image quality.

Application to image data directly We first discuss another set of experiments over the MNIST data: to track the change in recall-type scores with respect to a loss in diversity of generated samples, we successively dropped a digit from the image collection. We simply start off with original MNIST data as the "real set" and the "generated set". Then, a certain label (a specific digit) is dropped from the "generated set", while randomly taking out samples from the "real set" to ensure the overall sample sizes remain balanced. Figure 3 shows that, as we progressively drop digits, our recall cover measure (RC) correctly reflects the loss in diversity of the "generated set". The RC score drops approximately a 10% for each dropped digit (one out of ten digits dropped at each step).

For a second set of experiments we used the FFHQ dataset, a high resolution 1024x1024 image dataset of human faces (Karras et al., 2019). Here we progressively added noise to an image in a fashion similar to that of the forward process of diffusion models (Sohl-Dickstein et al., 2015). This allows us to control the gradual degradation of quality of the images from a high quality image of a person to pure noise. We expect to see the scores of precision type measures decrease as we add noise and the image quality decreases. The first plot in Figure 4 shows results of this setup. As the noise increases, the scores of all measures decreases. We again see the IP has a tendency to be overaccepting and that Density is not normalized (the score starts off with a value larger than 1). As a second means to induce degradation of image quality and and its effect on precision scores, we downsample the images ("generated set") and compare to the original images ("real set"). A downsampling factor of 2 takes an image of resolution 1024x1024 and averages the pixel values of every 2x2 square over the image and replaces the pixel values of each pixel in that square with the average pixel value. The center plot of Figure 4 shows the scores of PC, IP and Density as we repeatedly downsample. Again we observe that Density is not normalized and IP has a tendency to be overaccepting (while all measures indicate the decrease in quality eventually).

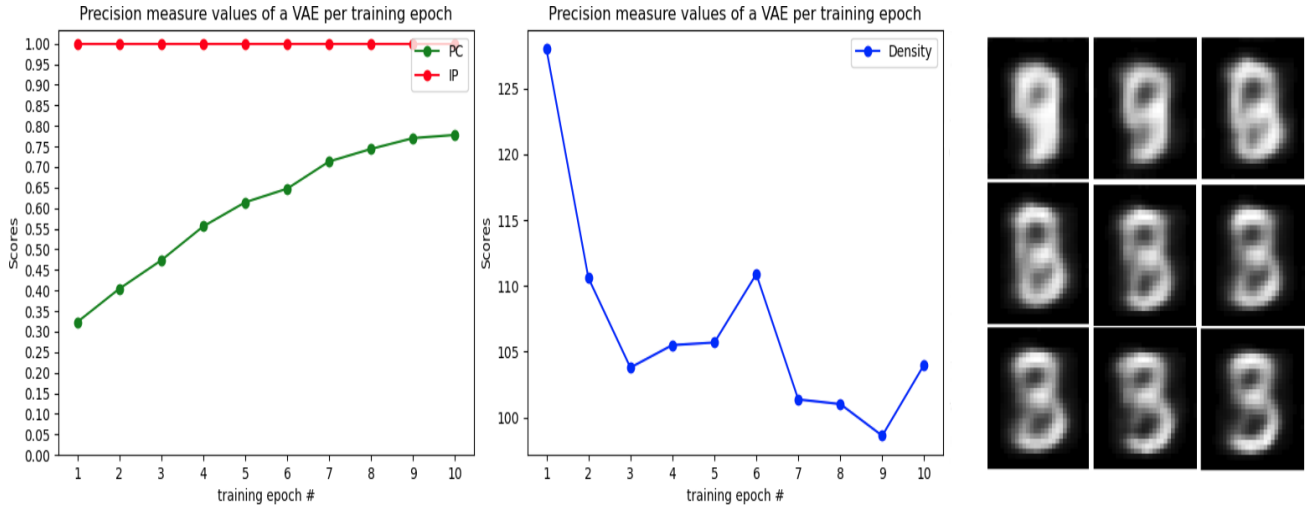

Figure 2: Training a VAE, we track the precision type scores per epoch. The plot on the left shows PC (our measure, in green) and IP (in red). The plot in the center shows Density (blue) (from DC). The image collection on the right shows one generated sample image from epoch 1 to epoch 9 (top left to bottom right). These image samples confirm that quality improves as training progresses. Only our measure PC accurately reflects this. Density gives scores of 125 (the measure is not normalized) and decreases non-monotonically and IP stays constant 1 due to its tendency for over-acceptance.

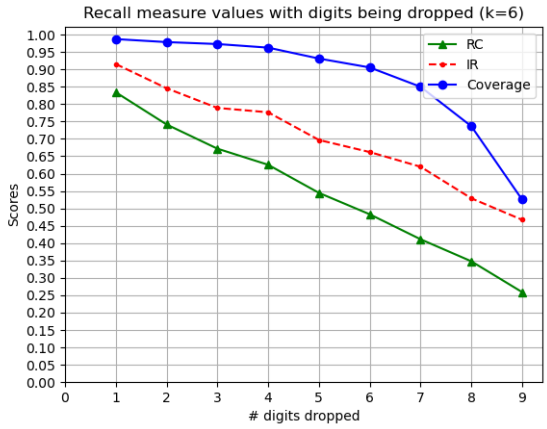

Figure 3: Tracking RC, IR and Coverage, as we successively drop a digit from MNIST data (comparing to an equal size sample from all digits in MNIST). RC and IR correctly reflect the loss in diversity in the "generated set", while Coverage remains almost constant for quite a while (until more than two thirds of the digits are dropped).

Application to GANs Finally, we explore our measure's performance on evaluating generative adversarial networks (GANs). Prior work (Kynkäänniemi et al., 2019; Brock et al., 2019; Naeem et al., 2020) discusses an artificial way of controlling diversity and fidelity via truncation. It is accepted in the literature (Brock et al., 2019; Kynkäänniemi et al., 2019) that the more truncation that is applied the higher the quality of the generated images; however this comes at the cost of less diversity in generated samples. Conversely, with less truncation, it is expected to see more diversity in samples, but worse quality. We design experiments to show that our measure will accurately reflect this diversity-quality trade off with StyleGAN (Karras et al., 2019) (see Figure 5).

Robustness w.r.t choice of k Finally we compare the k-nn based measures on popular datasets while varying k, to test their sensitivity to the choice of k. We again use FFHQ for our real dataset and samples from StyleGAN2 (Karras et al., 2020). Figure 6 shows the results. While varying the choice of k, (the same bottleneck computation of k nearest distances is used for all measures) our PC and RC measure results in consistent scores. DC and IPR, on the other hand are shown to be sensitive to the choice of k. Our measure enjoys better statistical consistency through the interplay between k and k' (whose ratio remains constant $1/3$, as suggested by our analysis). We believe that a measure not showing brittleness with respect to hyper-parameter choices (such as k in k-nn based measures) provides an important reassurance for practitioners.

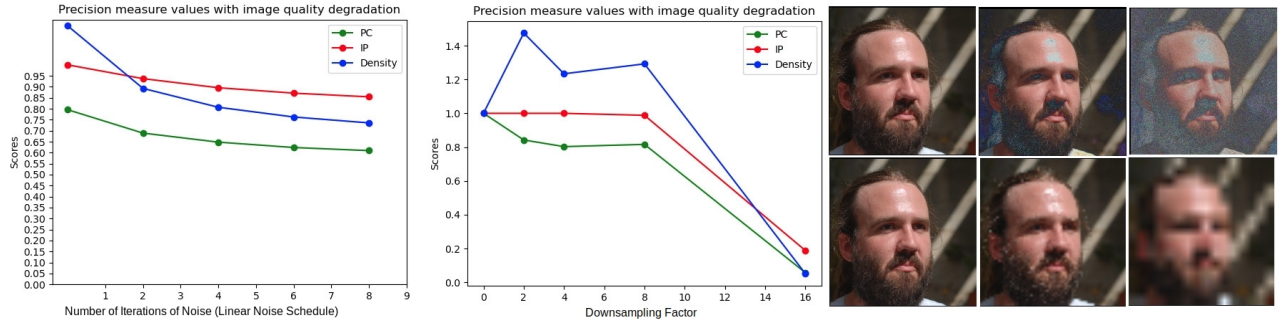

Figure 4: We track PC (green), IP (red), and Density (blue) as we decrease image quality of the "generated set" by increasing noise levels (left plot) or repeated downsampling (center plot). All measures indicate the degradation in image quality, but Density is not normalized and IP has a tendency to be overaccepting. Note that when downsampling, IP gives perfect scores Density remains above 1 until a downsampling factor of 16. The image samples on the right illustrate the gradual degradation in quality by adding noise (top row) or downsampling (bottom row).

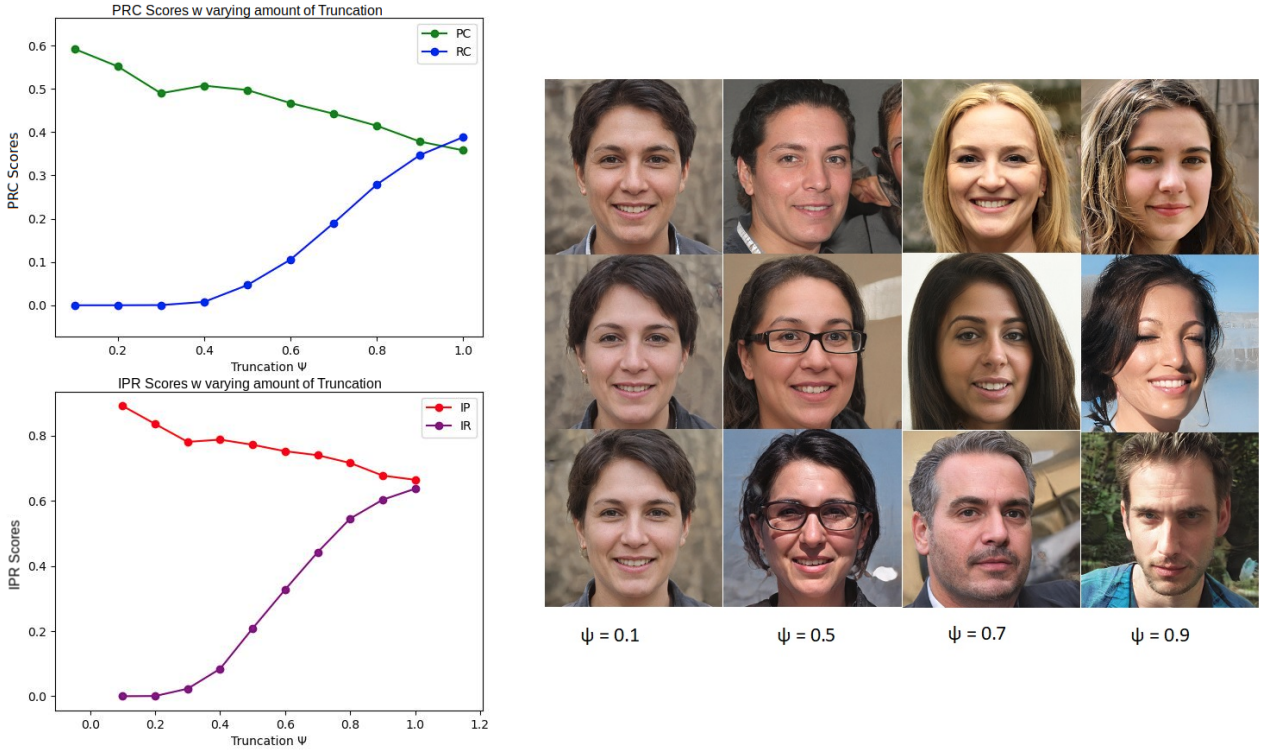

Figure 5: Some samples generated by StyleGAN Karras et al. (2019) with varying degrees of truncation. We plot our measures RC and PC as a function of the amount of truncation. As desired, we observe tat as truncation decreases (maximum truncation at $\psi = 0$, minimal at $\psi = 1$) PC (green) decreases and RC (blue) increases. We also conduct this experiment while tracking IPR (top left), this also gives expected behavior.

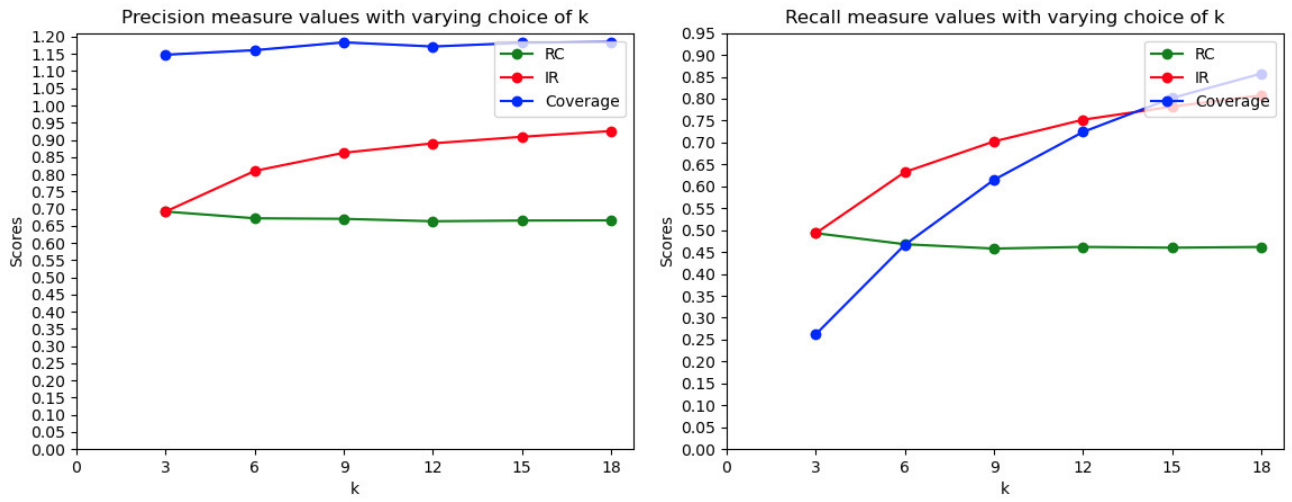

Figure 6: The plot on the left shows precision-type measures' results on the FFHQ dataset and StyleGAN2 samples with varying k. The plot on the right shows recall-type measures' results with a varying k on the same dataset and GAN. While DC and IPR are sensitive to the choice of k, the scores of PRC remain constant when varying k. Our measure enjoys better statistical consistency through the interplay between k and k' (their ratio remains $1/3$, as suggested by our analysis).

5 CONCLUDING REMARKS

We have introduced a new evaluation measure for computing and diagnosing differences and similarities between two distributions with access to samples only. Our population level (α, β)-PRC measure and its empirical (k, k')-PRC counterpart are statistically sound while the empirical measure is also algorithmically simple to evaluate. We believe that this combination, namely a sound framework that directly corresponds to the algorithmic implementation, is crucial and prior measures have lacked in offering both of these aspects simultaneously. On top, our measure is based on a natural local test of coverage that can naturally be used as a diagnostic tool, eg to improve a generative model's performance. In addition to the development, theoretical motivation and analysis, we have here presented a variety of promising empirical results from applying and comparing our measure in synthetic and real world settings.

Acknowledgements

Ruth Urner is also a Faculty Affiliate Member at Toronto's Vector Institute. This work was supported by an NSERC discovery grant.

References

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 214–223. PMLR, 2017.

Christopher Berlind and Ruth Urner. Active nearest neighbors in changing environments. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1870–1879. PMLR, 2015.

Ali Borji. Pros and cons of GAN evaluation measures: New developments. *Comput. Vis. Image Underst.*, 215:103329, 2022.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *7th International Conference on Learning Representations, ICLR*. OpenReview.net, 2019.

Sanjoy Dasgupta and Samory Kpotufe. Optimal rates for k-nn density and mode estimation. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

Josip Djolonga, Mario Lucic, Marco Cuturi, Olivier Bachem, Olivier Bousquet, and Sylvain Gelly. Precision-recall curves using information divergence frontiers. In *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020*, pages 2550–2559, 2020.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*. Computer Vision Foundation / IEEE, 2019.

Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 12104–12114. Curran Associates, Inc., 2020.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR*, 2014.

Samory Kpotufe. k-nn regression adapts to local intrinsic dimension. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 24, 2011.

Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32, 2019.

Muhammad Ferjad Naeem, Seong Joon Oh, Youngjung Uh, Yunjey Choi, and Jaejun Yoo. Reliable fidelity and diversity metrics for generative models. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020*, volume 119 of *Proceedings of Machine Learning Research*, pages 7176–7185. PMLR, 2020.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation, 2021.

Murray Rosenblatt. Remarks on Some Nonparametric Estimates of a Density Function. *The Annals of Mathematical Statistics*, 27:832 – 837, 1956.

Mehdi S. M. Sajjadi, Olivier Bachem, Mario Lucic, Olivier Bousquet, and Sylvain Gelly. Assessing generative models via precision and recall. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31, 2018.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, and Xi Chen. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning - From Theory to Algorithms.* Cambridge University Press, 2014. ISBN 978-1-10-705713-5.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, Proceedings of Machine Learning Research, Lille, France, 2015. PMLR.

Esteban G. Tabak and Eric Vanden-Eijnden. Density estimation by dual ascent of the log-likelihood. *Communications in Mathematical Sciences*, 8:217–233, 2010.

V. N. Vapnik and A. Ya. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability & Its Applications*, 16(2):264–280, 1971. doi: 10.1137/1116025.

Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks, 2018.

A Proofs of Results

A.1 Preliminaries

We start by listing some technical lemmas and tools that we use in our analysis. The following lemma appears in (Kpotufe, 2011).

Lemma 6 ((Kpotufe, 2011)). *Let B denote the class of balls on $\mathcal{X}$, with VC-dimension $\mathcal{V}_B$. Let $0 < \delta < 1$, and define $\alpha_n = (\mathcal{V}_B \ln 2n + \ln(\frac{8}{\delta}))/n$. The following holds with probability at least $1-\delta$ (over i.i.d. samples of size n) for all balls in B: Pick any $a \geq \alpha_n$. Then $\mu(B) \geq 3a \implies \mu_n(B) \geq a$ and $\mu_n(B) \geq 3a \implies \mu(B) \geq a$.*

Note that, for d-dimensional euclidean spaces the VC-dimension of the class of balls is $d+1$.

The following is a commonly used inequality. It can be found in the appendix of (Shalev-Shwartz and Ben-David, 2014).

Lemma 7. *Let $a \geq 1$ and $b > 0$. Then $x \geq 4a\log(2a) + 2b$ implies $x \geq a\log(x) + b$.*

A.2 Proofs

Proof of Observation 1. The first three claims follow directly from the definitions. For the last item, we will argue why it is true for the recall cover version. The claims for the precision cover then follow analogously. Let $(\beta_i)_{i\in\mathbb{N}}$ be a monotonically decreasing sequence of values for β in $(0,1)$ converging to 0. Note that for any point $x \in \text{supp}(P)$ we have

$$\bigcap_{i=1}^{\infty} B_P(x,\beta_i) = \{x\}$$

and thus we get

$$\bigcap_{i=1}^{\infty} \bigcup_{x\in\text{supp}(P)} B_P(x,\beta_i) = \overline{\text{supp}(P)},$$

where $\overline{\text{supp}(P)}$ denotes the (topological) closure of $\text{supp}P$. Since the Lebesgue measure of $\overline{\text{supp}(P)} \setminus \text{supp}(P)$ is 0 and we are assuming our distributions to admit density functions, we get that for any $\alpha_0 > 0$, since the support of P and of Q are disjoint, there exists an index $i \in \mathbb{N}$ such that

$$Q\Big(\bigcup_{x\in\text{supp}(P)} B_P(x,\beta_i)\Big) < \alpha_0.$$

This implies that, for any $\beta < \beta_i$ and $\alpha < \min\{\alpha_0, \beta\}$, we have

$$\text{RC}_{\alpha,\beta}(P,Q) = \mathbb{P}_{x\sim P}[Q(B_P(x,\beta))) \geq \alpha] = 0$$

and the statements follow. □

Proof of Theorem 2. As in the proof of Observation 1, one can choose α_0 and β_0 sufficiently small so that

$$Q(\bigcup_{x\in(\text{supp}(P)\setminus\text{supp}(Q))} B_P(x,\beta_0)) < \alpha_0$$

and thus, for any $\beta \leq \beta_0$ and $\alpha \leq \alpha_0$ only balls around points in the support of Q will be α-covered by Q. However note, that not all of these will necessarily be α-covered by Q. It remains to show that we can choose α sufficiently small so that the P-mass of balls that are not α-covered by Q is at most ϵ.

Let some $\epsilon > 0$ be given and let's fix some $\beta_1 < \beta_0$. For every $x \in \text{supp}(Q)$ we have $Q(B_P(x,\beta_0))) > 0$. We consider a sequence of values $(\alpha_i)_{i\in\mathbb{N}}$ that converges to 0. Then we get that the sets

$$A_i := \{x \in \text{supp}(Q) \mid Q(B_P(x,\beta_1)) > \alpha_i\}$$

are ordered by inclusion, namely $A_i \subseteq A_j$ for $j \geq i$ and

$$\bigcup_{i=1}^{\infty} \{x \in \text{supp}(Q) \mid Q(B_P(x,\beta_1)) > \alpha_i\} = \text{supp}(Q).$$

Thus, there exists a $j \in \mathbb{N}$ such that for any $i \geq j$, we get

$$P(A_i) = P(\{x \in \text{supp}(Q) \mid Q(B_P(x, \beta_1)) > \alpha_i\}) \geq P(\text{supp}(Q)) - \epsilon.$$

This completes the proof. □

Proof of Theorem 3

Proof:

First fix $\hat{Q} \sim Q$ with M samples such that: for its k' nearest neighbors we have:

$$k' \geq 81(\mathcal{V}_B \ln(2M) + \ln(\frac{8}{\delta}))$$

Choose $k' = Ck$, such that $C \geq 9$. With a choice of k such that:

$$k \geq 9(\mathcal{V}_B \ln(2M) + \ln(\frac{8}{\delta}))$$

Now recall our measure the PR-Cover is defined as:

$$\text{Precision} = \frac{1}{M} \sum_{i=1}^{M} \mathbb{1}_{y_i \sim Q}[k \leq |\hat{P} \cap B_Q(y_i, \frac{k'}{M})|]$$

And recall is defined with P and Q swapped.
Now we first fixed our M samples called set $\hat{Q}$, these balls define an empirical probability distribution:

$$\hat{Q}_M(B_{\hat{Q}}(y, \frac{k'}{M})) = \frac{k'}{M}$$

For each of our k'-nn balls the above is true. With a sufficient choice of k' which is given above. Then applying Lemma 6 since we know $\hat{Q}_M(B_{\hat{Q}}(y, \frac{k'}{M})) = \frac{k'}{M} \geq 81(\frac{\mathcal{V}_B \ln(2M) + \ln(\frac{8}{\delta})}{M})$ we readily obtain (with probability $1 - \delta$):

$$Q(B_{\hat{Q}}(y, \frac{k'}{M})) \geq 27(\frac{\mathcal{V}_B \ln(2M) + \ln(\frac{8}{\delta})}{M})$$

Now let us sample a new set of N samples $\hat{P} \sim P = Q$. Now our metric for this sample would be:

$$\text{Precision} = \frac{1}{M} \sum_{i=1}^{M} \mathbb{1}_{y_i \sim Q}[k \leq |\hat{P} \cap B_{\hat{Q}}(y_i, \frac{k'}{M})|]$$

$$\text{Precision} = \frac{1}{M} \sum_{i=1}^{M} \mathbb{1}_{y_i \sim Q}[\hat{P}_N(B_{\hat{Q}}(y_i, \frac{k'}{M})) \geq \frac{k}{N}]$$

Note that since $P = Q$ we have $P(B_{\hat{Q}}(y, \frac{k'}{M})) = Q(B_{\hat{Q}}(y, \frac{k'}{M})) \geq 27(\frac{\mathcal{V}_B \ln(2M) + \ln(\frac{8}{\delta})}{M})$

Now since we know $P(B_{\hat{Q}}(y, \frac{k'}{M})) \geq 27(\frac{\mathcal{V}_B \ln(2M) + \ln(\frac{8}{\delta})}{M})$ By applying Lemma 6 we have with probability $1 - 2\delta$:

$$P(B_{\hat{Q}}(y, \frac{k'}{M})) \geq 27(\frac{\mathcal{V}_B \ln(2M) + \ln(\frac{8}{\delta})}{M}) \Rightarrow \hat{P}_N(B_{\hat{Q}}(y, \frac{k'}{M})) \geq 9(\frac{\mathcal{V}_B \ln(2M) + \ln(\frac{8}{\delta})}{M})$$

Which implies with probability $1 - 2\delta$ that there are s points from sample $\hat{P} \sim P = Q$ such that:

$$\hat{P}_N(B_{\hat{Q}}(y, \frac{k'}{M})) = \frac{s}{N} \geq \frac{9(\mathcal{V}_B \ln(2M) + \ln(\frac{8}{\delta}))}{M}$$

$$s \geq \left(\frac{9(\mathcal{V}_B \ln(2M) + \ln(\frac{8}{\delta}))}{M}\right) N$$

So if we choose n such that $n \geq N$ so that we have: $s \geq k \geq 9(\mathcal{V}_B \ln(2M) + \ln(\frac{8}{\delta}))$

Therefore, we have $\hat{P}_n(B_{\hat{Q}}(y, \frac{k'}{M})) \geq \frac{k}{n}$ then with prob $1 - 2\delta$ we have more than k points from distribution P in the k'-nn balls of the generated samples.

A.2.1 Proof of Theorem 4

This proof has been adapted from Berlind and Urner (2015). Note that

$$n \geq \frac{72\ln(\frac{8}{\delta})}{C\omega}\ln(\frac{9m}{C\omega})$$

this implies

$$\Rightarrow n \geq \frac{9m}{C\omega} \tag{5}$$

$$\Rightarrow n \geq \frac{18m\ln(\frac{8}{\delta})}{Ck\omega} \tag{6}$$

$$\Rightarrow n \geq \frac{72m\mathcal{V}_B}{Ck\omega}\ln(\frac{9m\mathcal{V}_B}{Ck\omega}) \tag{7}$$

using Lemma 7 on the inequality 7:

$$n \geq 2\Big(4 \times \Big(\frac{9m\mathcal{V}_B}{Ck\omega}\ln(\frac{9m\mathcal{V}_B}{Ck\omega})\Big)\Big)$$

$$\Rightarrow n \geq 2\Big(\frac{9m\mathcal{V}_B}{Ck\omega}\ln(2n)\Big) \tag{8}$$

adding inequalities 6 and 8 we get:

$$2n \geq \frac{18m\mathcal{V}_B}{Ck\omega}\ln(2n) + \frac{18m\ln(\frac{8}{\delta})}{Ck\omega}$$

$$\Rightarrow n \geq \frac{9m\mathcal{V}_B}{Ck\omega}\ln(2n) + \frac{9m\ln\frac{8}{\delta}}{Ck\omega}$$

$$\Rightarrow n\frac{Ck\omega}{3m} \geq 3(\mathcal{V}_B\ln(2n) + \ln(\frac{8}{\delta}))$$

$$\Rightarrow \frac{Ck\omega}{3m} \geq \frac{3(\mathcal{V}_B\ln(2n) + \ln(\frac{8}{\delta}))}{n} \tag{9}$$

Now note that by the definition of the k' nearest neighbor balls over the generated samples we have (and recall that $k' = Ck$):

$$\hat{Q}_m(B_{\hat{Q}}(y, \frac{k'}{m})) = \frac{Ck}{m}$$

Note that we choose k such that:

$$k \geq 9(\mathcal{V}_B\ln(2m) + \ln(\frac{8}{\delta}))$$

plugging this choice of k into the definition of k'-nn ball over the generated samples:

$$\hat{Q}_m(B_{\hat{Q}}(y, \frac{k'}{m})) = \frac{Ck}{m} \geq \frac{9C(\mathcal{V}_B\ln(2m) + \ln(\frac{8}{\delta}))}{m}$$

by Lemma 6 we have:

$$\Rightarrow Q(B_{\hat{Q}}(y, \frac{k'}{m})) \geq \frac{Ck}{3m}$$

using the density ratio we obtain:

$$\Rightarrow P(B_{\hat{Q}}(y, \frac{k'}{m})) \geq \frac{Ck\omega}{3m}$$

using equation 9 into the above, we get:

$$\Rightarrow P(B_{\hat{Q}}(y, \frac{k'}{m})) \geq \frac{Ck\omega}{3m} \geq \frac{3(\mathcal{V}_B\ln(2n) + \ln(\frac{8}{\delta}))}{n}$$

using Lemma 6 again, we get:

$$\Rightarrow \hat{P}_n(B_{\hat{Q}}(y, \frac{k'}{m})) \geq \frac{Ck\omega}{9m} \geq \frac{(\mathcal{V}_B\ln(2n) + \ln(\frac{8}{\delta}))}{n}$$

using equation 5 on the above, we readily obtain:

$$\hat{P}_n(B_{\hat{Q}}(y, \frac{k'}{m})) \geq \frac{k}{n}$$

so with probability $1 - 2\delta$ we have that there are more than k points in the k'-nn ball centered around a generated sample $y \in Q$. This then ensures w.h.p. that the precision measure will recognize this k'-nn ball and thus give it a perfect score; ie. the indicator function in the definition of the precision measure is 1.

A.2.2 Proof of Theorem 5

We will first prove a lemma necessary for the following proof:

Lemma 8.

$$\lim_{\omega \to 0} Q(\mathcal{X}_Q(\epsilon, \omega) \cap \mathcal{X}_P) = 0$$

Proof. First call the set of samples from the generated distribution that have density less than ω: $\mathcal{X}_Q(\epsilon, \omega) = \{y \in \mathcal{X}_Q | \frac{P(B_Q(y,\epsilon)}{Q_Q(B(y,\epsilon)} < \omega\}$, where $B_Q(y, \epsilon)$ is a ball centered at point $y \in Q$ with mass ϵ w.r.t. distribution Q ie: $(Q(B_Q(y, \epsilon)) = \epsilon)$. Now consider an ordered sequence of densities in decreasing order called $[\omega_i]_{i \in \mathbb{N}}$, let this sequence converge to 0. We clearly get:

$$\lim_{n \to \infty} \omega_n = 0$$

Now putting this sequence into the set we defined earlier we have:

$$\lim_{n \to \infty} \mathcal{X}_Q(\epsilon, \omega_n) = \bigcap_{n=1}^{\infty} \mathcal{X}_Q(\epsilon, \omega_n)$$

$$\bigcap_{n=1}^{\infty} \mathcal{X}_Q(\epsilon, \omega_n) = \mathcal{X}_Q \backslash \mathcal{X}_P$$

That is, the intersection of the set of all points from Q that have no neighboring points within radius ϵ of them for progressively smaller values of ω converges to the points in support Q that are not in support P.

Taking the probability of the limit sequence w.r.t. probability distribution Q:

$$\lim_{n \to \infty} Q(\mathcal{X}_Q(\epsilon, \omega_n)) = Q(\bigcap_{n=1}^{\infty} \mathcal{X}_Q(\epsilon, \omega_n))$$

$$Q(\bigcap_{n=1}^{\infty} \mathcal{X}_Q(\epsilon, \omega_n)) \leq Q(\mathcal{X}_Q \backslash \mathcal{X}_P)$$

Note that if we take the intersection with $\mathcal{X}_P$ we obtain the limit we are trying to prove:

$$\lim_{n \to \infty} Q(\mathcal{X}_Q(\epsilon, \omega_n) \cap \mathcal{X}_P) \leq Q((\mathcal{X}_Q \backslash \mathcal{X}_P) \cap \mathcal{X}_P) = 0$$

□

Proof of Theorem 5

Proof. Recall that we define the set $\mathcal{X}_Q(\epsilon, \omega) = \{y \in \mathcal{X}_Q | \frac{P(B_Q(y,\epsilon)}{Q(B_Q(y,\epsilon)} < \omega\}$ (where $Q(B_Q(y, \epsilon)) = \epsilon$). Recall that in our setup we assumed that both P and Q have continuous density functions d_P and d_Q respectively. This implies that for any sample from either distribution we can pick a ball of mass exactly ϵ, for $\epsilon \in (0, 1]$.

Using Lemma 8 there exists a choice of ω small enough such that there is a generated sample size m, so no generated samples y will be in $\mathcal{X}_Q(\epsilon, \omega) \cap \mathcal{X}_P$, in similar fashion to Theorem 4. Pick $\epsilon = \frac{Ck}{3m}$. also note that:

$$Q(B_Q(y, \epsilon)) = \frac{Ck}{3m}$$

There exists a sample size, N, from P we can pick such that this sample set will have perfect precision on $\mathcal{X}_P \backslash \mathcal{X}_Q(\epsilon, \omega)$. Recall our choice of k from prior: $k \geq 9(\mathcal{V}_B \ln(2m) + \ln(\frac{8}{\delta}))$. This then implies:

$$\frac{Ck}{m} \geq \frac{9C(\mathcal{V}_B \ln(2m) + \ln(\frac{8}{\delta}))}{m}$$

By employing the contrapositive of Lemma 6 and the above, we have:

$$\hat{Q}_m(B_Q(y, \epsilon)) \leq \frac{Ck}{m}$$

This implies that there are at most Ck points in the ball $B_Q(y, \epsilon)$ from Q.

Using Theorem 4 (assuming we satisfy the conditions to use), for our choice of ω, C, δ, and m we have a value of N such that:

$$n \geq N = \frac{72m \ln(\frac{8}{\delta})}{C\omega} \ln(\frac{9m}{C\omega})$$

by Theorem 4, this sample size implies that the ball $B_Q(y, \epsilon)$ will have at least k points from P: that is: $\hat{P}_n(B_Q(y, \epsilon)) \geq \frac{k}{n}$. This thus implies (with probability $1 - 3\delta$) that $\hat{P}$ will cover all points from $\hat{Q}$ that have joint support. Our precision measure will give these generated points a perfect score. Thus concluding the proof.

□

A.3 The Algorithm

Algorithm 1: Precision-Recall Cover

Input Sample Sets $\hat{P}$, $\hat{Q}$ as well as choice of k, and C
$\hat{Q}_\beta \leftarrow \emptyset$;
$k' \leftarrow C * k$;
$\hat{r}_{\hat{Q}} \leftarrow$ k'-NearestNeighborDistances$(\hat{Q}, k')$;
for $y \in \hat{Q}$ **do**
  val $\leftarrow$ PR-Cover-Indicator$(y, \hat{r}_y, \hat{P})$
  if $val = 1$ **then**
    $\hat{Q}_\beta \leftarrow \hat{Q}_\beta \cup y$
  end
end
PC $\leftarrow \frac{|\hat{Q}_\beta|}{|\hat{Q}|}$
return PC

Algorithm 2: Precision-Recall Cover Indicator subroutine

Input Sample Set $\hat{P}$, sample point $y \in \hat{Q}$, (k')-nearest neighbor distance $\hat{r}_y$, and k
val $\leftarrow 0$;
$i \leftarrow 0$;
for $x \in \hat{P}$ **do**
  if $||y - x|| \leq \hat{r}_y$ **then**
    $i \leftarrow i + 1$;
  end
end
if $i \geq k$ **then**
  val $\leftarrow 1$;
end
return val

B Experiments

For all experiments we intend to make the code publicly available on GitHub. In the vein of reproducibility we choose to provide a detailed summary of experiments here in addition to extra images and results.

B.1 Toy Models

As stated in the main paper, we devised several toy experiments as a sanity check for our measure. In these settings, we start off with two synthetic distributions as stand-ins for the "real distribution" and the "generated distribution". The real distribution and generated distributions are either both uniform distributions or Gaussian distributions in most of our setups. We use numpy to randomly sample points from their respective distributions. We create the distributions in 1 to 3 dimensions for visualization purposes and use the same random seed throughout the experiments. We have conducted a wide set of experiments where we vary many factors in a controlled setting; we vary the amount of overlap between the two distributions, the number of samples for each distribution, the dimension, and the shape of the distributions. We will show several figures to illustrate the setup of the experiment and some of the results. In most of these toy models we show a visualization of the data (on the left) and on the right the Precision-Recall Curve from (Sajjadi et al., 2018) with the scores from the other measures on the top right of the figure. For the toy models shown here we have opted to use a choice of $k = 9$ for DC and IPR, and for PRC we use a choice of $k' = 9$ and $k = 3$. See Figure 7 for a 1 dimensional case of matching distributions. See Figure 8 for a case of 2-dimensional distributions that are partially overlapping, in this example we also have the true distribution have 3 times as many samples as the generated distribution has. In fig 9 we have 3 dimensional uniform distributions that are essentially disjoint.

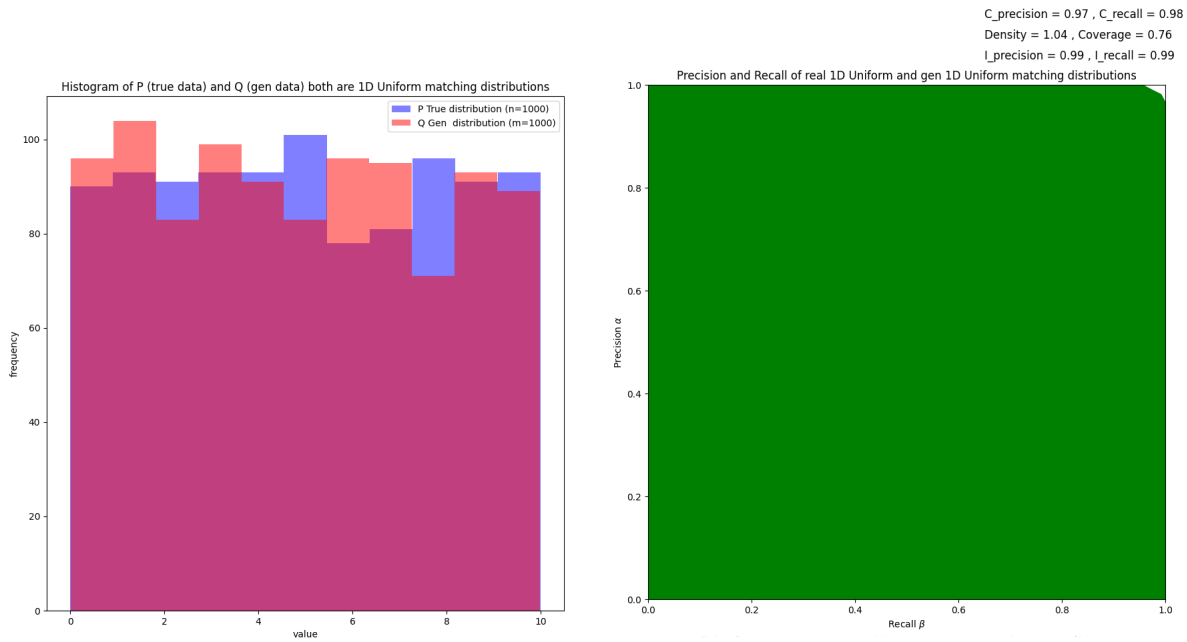

Figure 7: The true distribution (blue) and the generated distribution (salmon) are both matching (uniform from 0 to 10). See visualization of the 1-dimensional data (in form of a histogram of the samples) on the left. Both sample sets have the same size (1,000 samples). We use $k = 9$ for DC and IPR and $k' = 9$ and $k = 3$ for PRC. The PR curve (right) is filled in implying perfect precision and recall. All other measures also reflect the matching distributions (see top right for scores). We observe that density and converge (DC) is not normalized, while density takes on a value larger than 1 (even if only slightly), coverage is significantly below 1. This shows that the scores of this measure are difficult to interpret, and thus might be misleading.

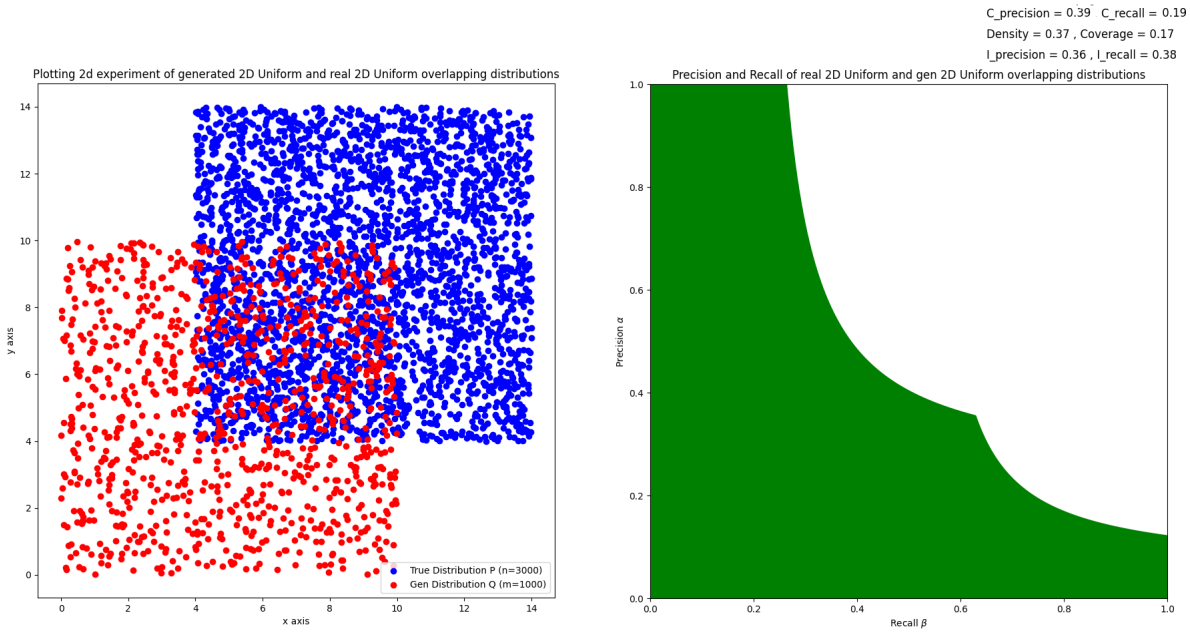

Figure 8: In this example we sample from two distributions; a 2 dimensional uniform square from 0 to 10 for both x and y axis (gen distribution in red), and a uniform square from 4 to 14 on both x and y axis (real distribution in blue). The scores mostly all reflect the partial overlap (see the PR curve right). Notice that in this example we have far more samples from the true distribution than the real (3000 samples from real dist. vs 1000 samples from gen dist.). This is reflected in the precision type scores of all measures (see top right). Note that the two distributions have equal area and have an overlapping area of 36%, as supported in the theory, since there are more true samples than gen samples we see that the precision scores are closer to the overlapping area amount than the recall-type scores.

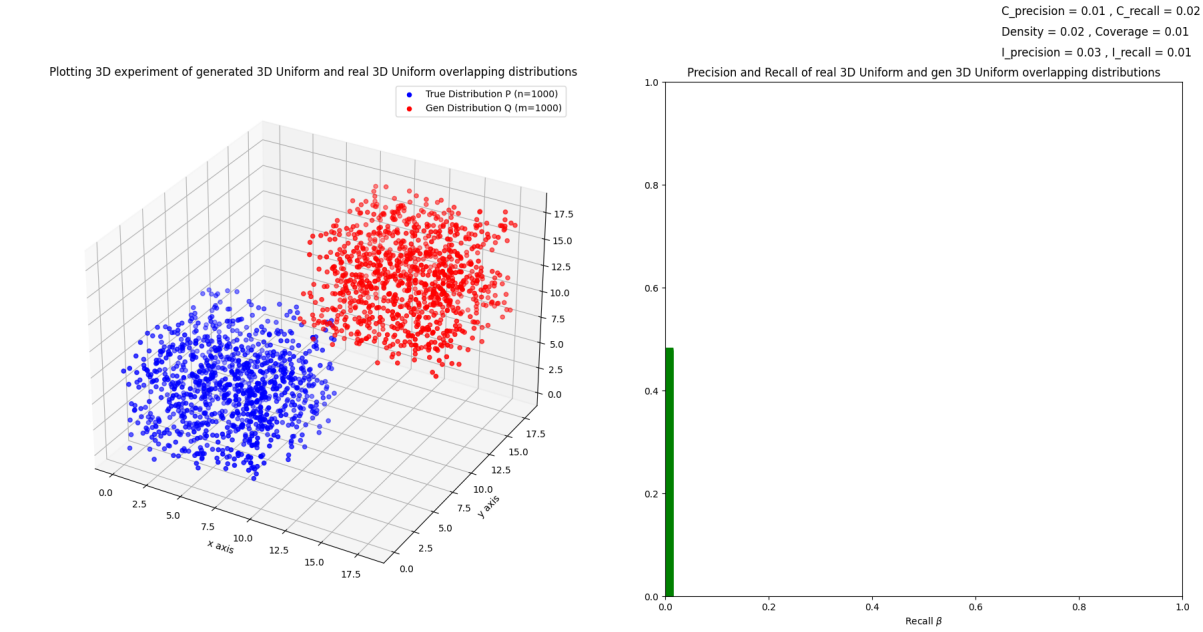

Figure 9: In this example we have two 3-dimensional cubes (see data on the left); a true uniform distribution from 0 to 10 on the three axes (in blue) and a generated uniform distribution from 10 to 20 on the three axes (in red). In this case the 2 distributions are essentially disjoint (one shared boundary points exists), and all measures reflect this. The PR curve (right) is empty implying no precision or recall, and the other measures almost all give scores of 0 (top right).

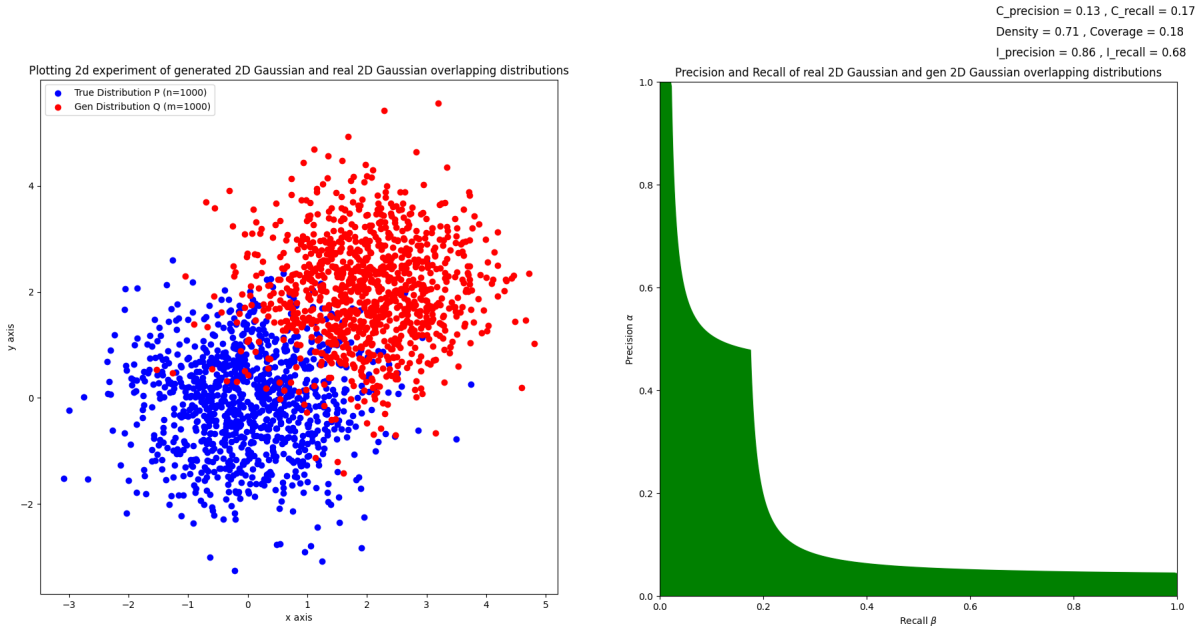

Figure 10: In this scenario we have a 2 dimensional Gaussian with mean 0 and std of 1 for both x and y axes as the true distribution (blue) and a Gaussian with mean 2 and std of 1 for both axes as the gen distribution (red). As we can see in the visualization of the data, although the data is overlapping most points from both distributions do not overlap and there is a clear distinction between the two distributions. On the right the PR curve shows that there is a discrepancy between the two distributions (seen in the lack of the PR curve being filled in), our measure PRC keeps a lower score (about 0.15) reflecting the lack of overlap between the distributions. In contrast, DC has a high density score yet retains a lower coverage score (top right). Also IPR has scores above 65% which is not appropriate for these two distributions. This is another example of the over-accepting nature of this measure.

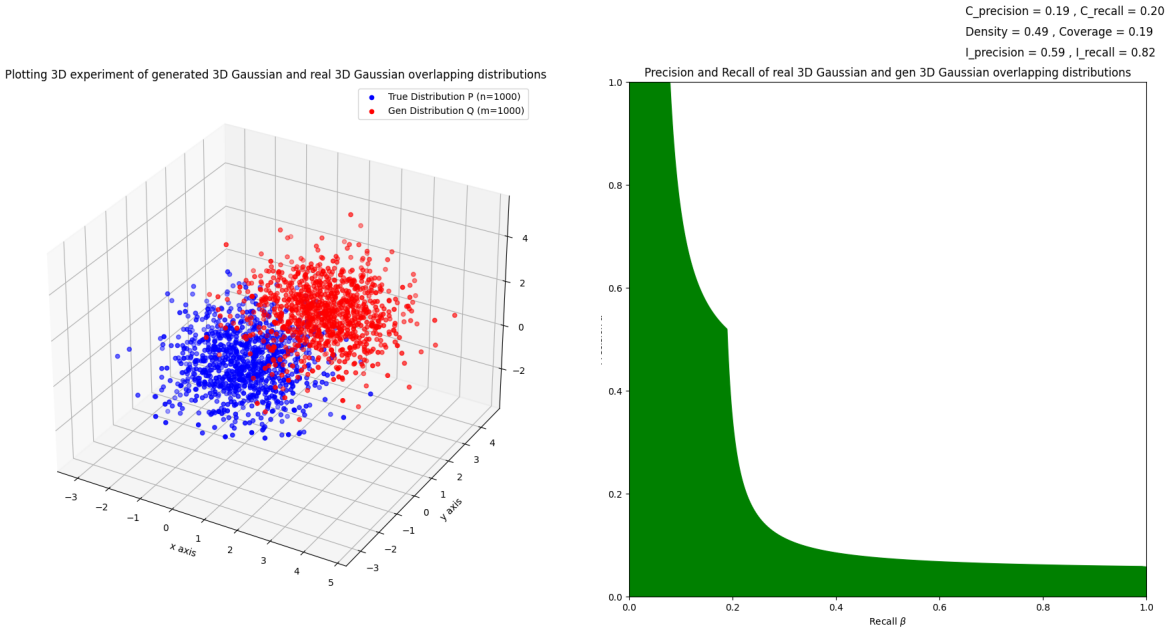

Figure 11: In this scenario we have a 3-dimensional Gaussian with mean 0 and std of 1 all three axes as the true distribution (blue) and a Gaussian with mean 2 and std of 1 for all three axes as the gen distribution (red). As we can see in the visualization of the data, although there is overlap in the data from these two distributions, most points are located in distinct areas, clearly belonging one of the two distributions. On the right the PR curve shows that there is a discrepancy between the two distributions (seen in the lack of the PR curve being filled in), our measure PRC keeps a lower score (about 0.20) reflecting the lack of overlap between the distributions. In contrast, DC has a high density score yet retains a lower coverage score (top right). Also IPR has the highest scores above 50% which is not the nature of the two distributions. This is yet another example of the over-accepting nature of this measure.

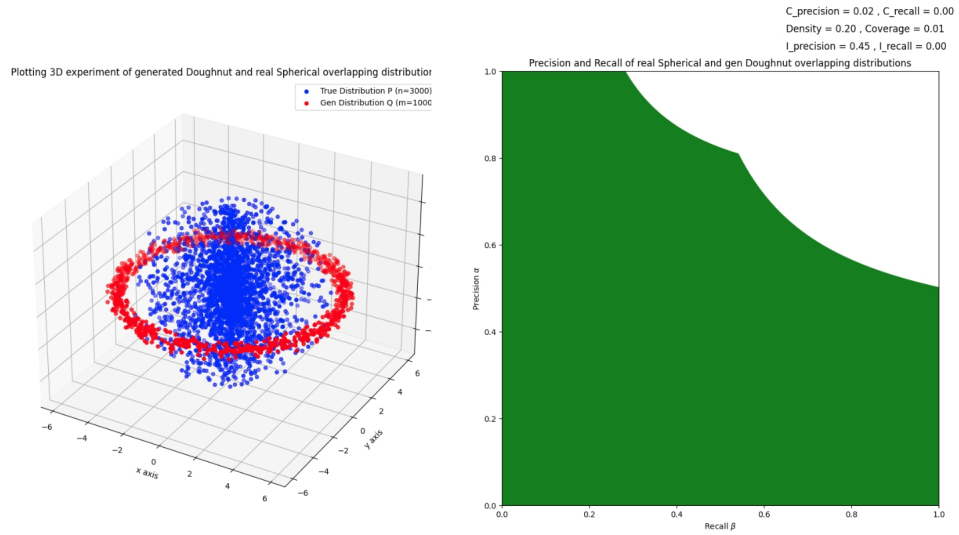

Figure 12: The PR Curve (right) gives a very high scores on both precision and recall (indicating similar distributions with large overlap) while the two distributions are in fact disjoint (data on left). The PR curve is almost fully filled in implying both high precision and recall. The doughnut (red; generated samples) does not overlap with the sphere (blue: real samples). Other metrics have the same issue: Density = 0.20, Coverage = 0.06, Improved Precision = 0.45, Improved Recall = 0.80. Only PRC correctly scores these sets as disjoint with PC = 0.02, and RC = 0.00

We observe many interesting behaviors that support our theory and highlight issues of other measures. Some interesting

observations we observe is that if we sample from the true distribution more than the gen distribution or vice versa we can see scores be skewed (this supports some of the theory we have developed see theorem 4), this behaviour of our measure is empirically reflected in fig 8. Another behaviour we have noted in this simplistic setting is that DC and IPR are prone to be more over-accepting than our measure. This is also true in settings where the distributions have significant discrepancy between them, but DC and IPR incorrectly say they are similar distributions (see fig 10, 11, 12). In fig 12 we can see a case of where two distributions are completely disjoint yet all other measures identify these distributions as similar. This again adds to the point of PR, IPR and DC being prone to over-accepting. Only our measure accurately reflects the disjoint nature of the two distributions. This experiment highlights this worrying behavior that will be seen again in later experiments in real world data.

B.2 Convergence Experiments

In these sets of experiments we use simple uniform distributions in increasing dimensions to define true values of precision and recall. We construct 2 uniform hypercube distributions for the real and generated distributions and vary the degree of overlap. In this case true precision becomes the overlapping volume over the generated distribution's volume, and conversely true recall is the overlapping volume over the true distribution's volume. The setup of this experiment is as follows: we fix a configuration for a true distribution and generated distribution with a certain amount of overlap. We set the number of samples from the true and gen distribution to be the equal $n = m$. We start at 15 samples for both distributions, obtain $n = m$ samples randomly for both distributions given a particular configuration of true and generated distributions (we use a random seed for reproducibility). Then, we compute all k-nn based measures (IPR, DC, and PRC) for the current sample size. We compute k-nn based measures only because they are directly comparable as the same bottleneck computation is done for all 3 measures (the distances between sets and within sets). Then we increment the sample size and sample from both distributions again for the given configuration of real and gen distribution and compute measures. We repeat this process until the terminal number of samples from both distributions (we choose to stop the experiment when $n = m = 1000$). as $n = m$ increases we adjust our value of k. We use a value of k inspired from the theory of order log(n). This value of k is the same for DC and IPR and PRC uses this same value for k' while keeping the ratio fixed of $C = 3$ for our (k, k')-based measure. More specifically, k ranges from 9 to 13 in our set of experiments. Figure 1 in the main part of the paper shows that our measure converges faster to the true values of precision and recall, it can also be seen that some of the other measures do not seem to converge to the true values at all (see higher dimensional setting of experiment). We also have included the experiments for the precision type experiments from increasing dimensions below.

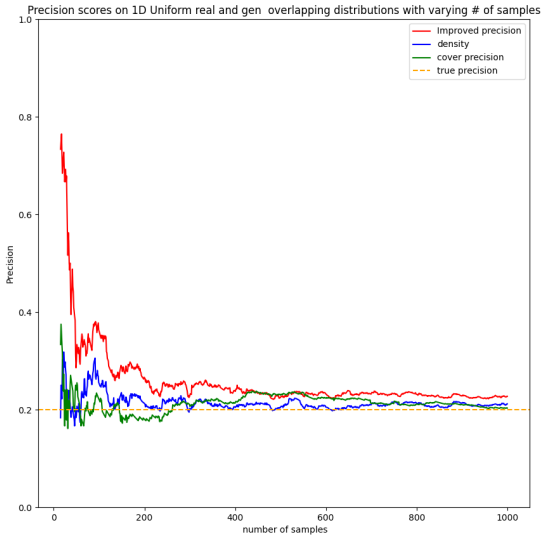

Figure 13: We compute improved precision, precision cover, and density for a fixed configuration of 1-dimensional uniform distributions. The true distribution is a uniform distribution over the interval from 0 to 10. The gen distribution is a uniform distribution from 8 to 18. This corresponds to a true precision of 0.2 for this configuration of distributions. We set $n = m$ start at 15 samples and go to 1000 samples. We can see IP (red), Density (blue), and PC (green) all converge to the true value of precision (dashed orange line).

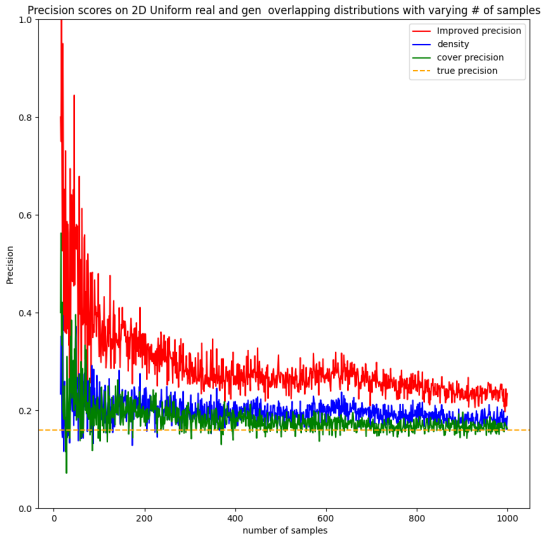

Figure 14: We compute improved precision, precision cover, and density for a fixed configuration of 2-dimensional uniform distributions. The true distribution is a uniform distribution over the interval from 0 to 10 in both x and y axes. The gen distribution is a uniform distribution from 6 to 16 for both x and y axes. This corresponds to a true precision of 0.16 for this configuration of distributions. We set $n = m$ start at 15 samples and go to 1000 samples. We can see IP (red), Density (blue), and PC (green) all converge to the true value of precision (dashed orange line).

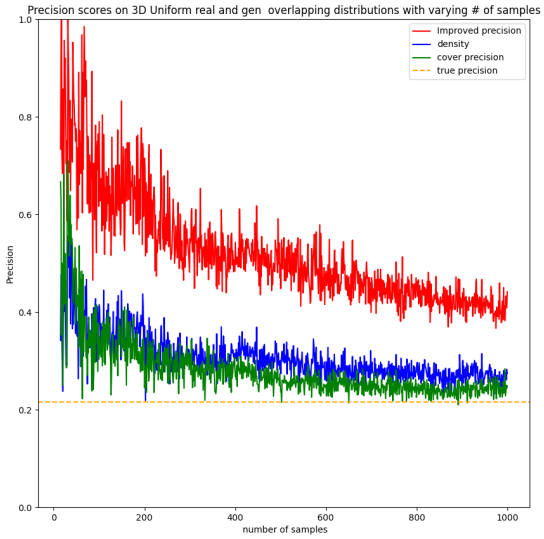

Figure 15: We compute improved precision, precision cover, and density for a fixed configuration of 3-dimensional uniform distributions. The true distribution is a uniform distribution over the interval from 0 to 10 in x, y and z axes. The gen distribution is a uniform distribution from 4 to 14 for x, y and z axes. This corresponds to a true precision of 0.216 for this configuration of distributions. We set $n = m$ start at 15 samples and go to 1000 samples. We can see IP (red), Density (blue), and PC (green) all converge to the true value of precision (dashed orange line).

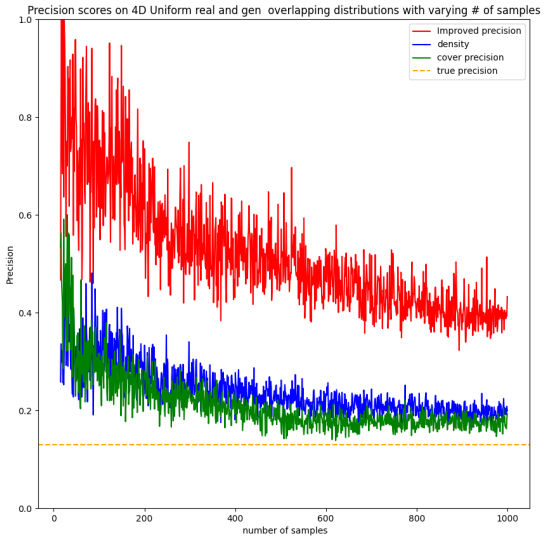

Figure 16: We compute improved precision, precision cover, and density for a fixed configuration of 4-dimensional uniform distributions. The true distribution is a uniform distribution over the interval from 0 to 10 in all 4 axes. The gen distribution is a uniform distribution from 4 to 14 for all axes. This corresponds to a true precision of 0.1296 for this configuration of distributions. We set $n = m$ start at 15 samples and go to 1000 samples. We can see IP (red), Density (blue), and PC (green) all converge to the true value of precision (dashed orange line).

B.3 VAE Experiments

For the VAE training experiments we use the MNIST data set (Deng, 2012). We use a simple VAE that uses the MNIST dataset for training (60,000 samples) and testing (10,000 samples). We use the adam optimizer and train the VAE for 10 epochs. The test set is used as the real distribution and we generate 10,000 samples from the model (to match the testing set size) as the generated distribution. We convert these images to gray-scale, so they are of dimension 28x28x1. In these sets of experiments we expect the measures to reflect the increase of quality as epochs increase. From fig 2 it is clear only our measure shows the desired behavior of increasing precision (quality) scores.

B.4 Digit Dropping

Since we have a nice set of controlled experiments to see an increase in quality in a real world setting with the VAE training experiment, we wish to do the same for our diversity based measures (recall-type scores). In this setting we design an experiment to artificially control for diversity of a dataset. We first start off with the complete MNIST dataset (70,000 samples). Then we choose a number of samples for the real and generated dataset ($n = m$), in this case 5,000 samples, for both real and gen distributions. From the complete dataset we randomly select 5,000 samples and call this the real sample set. We then randomly pick 1 digit to drop, and use a function to eliminate all samples with that digit's label from the potential generated dataset pool. From the MNIST dataset (now excluding the dropped digit) we sample 5,000 samples and call this the generated sample set. We compute improved recall, coverage, and recall cover over the real and generated sample sets. We then drop another digit and repeat the process until 9 digits (out of 10) are dropped. For this experiment every time prior to random sampling we use and record the value of the random seed used.

B.5 Experiments with GANs

For experiments with GANs we used the well developed pipeline of methods for assessing high resolution image data for quality metrics. Many other prior works have used this pipeline to test out measures for grading generative models (Sajjadi et al., 2018; Kynkäänniemi et al., 2019; Naeem et al., 2020). The pipeline consists of a network or method to project the high dimensional (usually image) data into a lower dimensional latent space. In this latent space we then have our measure compare the data and obtain a final score for PRC. We use the pipeline available from (Karras et al., 2020), which uses VGG16 to obtain image embeddings. Our approach here follows consistent methodology in the literature (Naeem et al., 2020; Sajjadi et al., 2018; Kynkäänniemi et al., 2019). We use the pretrained GAN and compute several other popular measures on the generated and real data. In our experiments we use StyleGAN and StyleGAN2.

For the noise experiments we add noise iteratively as done in diffusion model (Sohl-Dickstein et al., 2015; Ramesh et al., 2021). We use a linear noise schedule starting from 0.000001 ending at 0.0000075 with a 300 steps. We use small amounts due to the sensitive nature of the pipeline; ie enough noise to degrade image quality to human perception but not enough such that the feature extractors can no longer properly operate on the image. We notice that by adding a relatively significant amount of noise all measures give scores of 0.

Another set of experiments to control for quality or lack thereof is the downsampling set of experiments. In this set of experiments we use a value for the stride to downsample called s. We move an sxs square over the image (usually a high resolution 1024x1024 image) and average the pixel value over the square and assign each pixel in that square the value of the average. For instance, if we have a stride of 2, each 2x2 square is averaged and that average pixel value is given to each pixel in that square. This is akin to losing information by resizing an image of size 1024x1024 to size 512x512 and then resizing back to 1024x1024. This loss of pixel information can be seen to have a blurring effect on the quality of an image, see 4.

B.6 Limitations

Our measure along with the IPR, and DC uses the k-nn computations as a core element. It is well known that the k-nn computation suffers from the curse of dimensionality and thus with higher dimensions it becomes a bottleneck in the computation of each measure. Several researchers have suggested to use projections to lower dimension latent spaces for high dimensional data such as image data . These image embeddings are used in almost all previous works for generative modelling metrics, and we stay consistent with the embedding method used by other papers (VGG16) (Naeem et al., 2020; Kynkäänniemi et al., 2019; Sajjadi et al., 2018; Heusel et al., 2017; Salimans et al., 2016). Both FID and IS do not use a k-nn algorithm and as seen in other works can be faster than other metrics in computing scores to grade generative models (Karras et al., 2020).