# Revisiting Precision and Recall Definition for Generative Model Evaluation

**Loïc Simon** [1] **Ryan Webster** [1] **Julien Rabin** [1]

## Abstract

In this article we revisit the definition of Precision-Recall (PR) curves for generative models proposed by (Sajjadi et al., 2018). Rather than providing a scalar for generative quality, PR curves distinguish mode-collapse (poor recall) and bad quality (poor precision). We first generalize their formulation to arbitrary measures, hence removing any restriction to finite support. We also expose a bridge between PR curves and type I and type II error rates of likelihood ratio classifiers on the task of discriminating between samples of the two distributions. Building upon this new perspective, we propose a novel algorithm to approximate precision-recall curves, that shares some interesting methodological properties with the hypothesis testing technique from (Lopez-Paz & Oquab, 2017). We demonstrate the interest of the proposed formulation over the original approach on controlled multi-modal datasets.

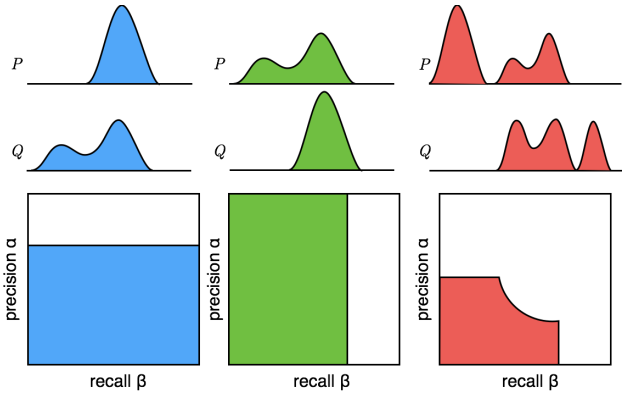


*Figure 1.* Illustration of precision-recall curves for multi-modal continuous distributions. Left: mode invention (precision is only partial but full recall). Middle: mode dropping (partial recall) but do not produce outliers (full precision but partial recall). Right: mode dropping / invention plus mode reweighting.

## 1. Introduction

This work addresses the question of the evaluation of generative models, such as Generative Adversarial Networks (GAN) (Goodfellow et al., 2014) or Variational Auto-Encoders (Kingma & Welling, 2014), that have attracted a lot of attention in the last years. These approaches aim at training a model to generate new samples from an unknown target distribution $P$, for which one has only access to a (sufficiently large) sample set $X_i \sim P$. While this class of methods have given state-of-the-art results in many applications (see *e.g.* (Brock et al., 2019) for image generation, (Iizuka et al., 2017) for inpainting, *etc*), there is still a need for evaluation techniques than can automatically assess and compare the quality of generated samples $Y_i \sim Q$ from different models with the target distribution $P$, for which the likelihood $P(Y_i)$ is unknown. Most of the time, such a comparison is just reduced to a simple visual inspection of the samples $Y_i$, but very recently several techniques have been proposed to address this problem that boils down to the comparison of two empirical distributions in high dimension. While generative models have seen successful applications far beyond just image data (such as speech enhancement (Pascual et al., 2017), text to image synthesis (Reed et al., 2016) or text translation (Lample et al., 2018)), we will focus on image generation, as is popular in the recent literature.

**Previous Work** When it comes to evaluating generative models of images, visual inspection, that is observing how "realistic" the images appear, remains the most important decider of the model's success. Indeed, state of the art methods, such a Progressive GANs (Karras et al., 2018) on face images or BigGAN (Brock et al., 2019) trained conditionally on ImageNet classes, include large grids of generated samples wherein the success of the method over previous approaches is visually obvious. Nonetheless, automatic evaluation of such models is extremely important, for example when conducting large scale empirical comparisons (Lucic et al., 2018), in cases where model failure is more subtle than simply poor image quality (*e.g.* mode collapse) such as in (Sajjadi et al., 2018), or presumably in domains in which

[1]Normandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC. Correspondence to: Loïc Simon <loic.simon@ensicaen.fr>.

*Proceedings of the $36^{th}$ International Conference on Machine Learning*, Long Beach, California, PMLR 97, 2019. 

humans are less attuned to discern quality of samples.

Attempts to provide automatic assessment of image quality can be traced back to the first GAN methods (Radford et al., 2015), where the authors assessed quality of generated samples with a nearest neighbor classifier. In (Salimans et al., 2016), the so-called *Inception Score* was introduced, which analyzes the entropy of image classes at the output of the Inception Network (Szegedy et al., 2016), which reflects if samples cover all classes and each clearly belongs to a particular class. In (Metz et al., 2017; Webster et al., 2019) test set samples (*i.e.* those unseen during training), are recovered via optimization. Successful generators are better at recovering all images from the training distribution, which in a controlled setting can be viewed as a notion of recall (Lucic et al., 2018; Sajjadi et al., 2018). In (Heusel et al., 2017), the Fréchet Inception Distance was introduced (FID), which estimates the Fréchet distance between inception features of real and generated samples modeled as multivariate normal distributions. The FID has been widely adopted because of its consistency with human inspection and sensitivity to small changes in the real distribution (*e.g.* slight blurring or small artifacts in generated images). A few recent approaches involve training a binary classifier to separate fake (*i.e.* generated) samples $Y_i$ from real data samples $X_i$. In (Lopez-Paz & Oquab, 2017), a score is defined from a two-sample statistical test of the hypothesis $P = Q$. Finally, in (Im et al., 2018), classifiers trained with various divergences (normally used as objectives for discriminators during GAN training) are used to define a metric between $Q$ and $P$. Surprisingly, successful models such as WGAN (Arjovsky et al., 2017a) have the smallest distance even on those metrics which were not used for training (*e.g.* a WGAN trained with the Wasserstein-1 distance evaluated with a least squares discriminator).

Unfortunately as pointed out by (Sajjadi et al., 2018), the popular FID only provides a scalar value that cannot distinguish a model $Q$ failing to cover all of $P$ (referred to henceforth as *low recall*) from a model $Q$ which has poor sample quality (referred to as low precision). For example, when modeling a distribution of face images, a $Q$ containing only male faces with high quality versus a $Q$ containing both genders with blurry faces may have equal FID. Following the lead of (Sajjadi et al., 2018), we will consider another category where one wants not only to assess if the samples are of good quality (high precision) but also to measure if the generated distribution $Q$ captures the variability of the target one (high recall). The reader may refer to Figure 1 to gain a crude understanding of the intended purpose of precision and recall. (Sajjadi et al., 2018) proposed an elegant definition of precision and recall for discrete distributions. They challenge their definition on image generation by discretizing the probability distributions $P$ and $Q$ over Inception features via K-means clustering. Note that a similar notion was proposed by the authors of PACGAN (Lin et al., 2018) under the name of mode collapse region (denoted as $MCR(P, Q)$). Their motivation was to develop a theoretical tool to analyze how using multi-element samples in the discriminator can mitigate mode dropping.

**Contributions and outline** The paper is organized as follows. First, Section 2 recalls usual notations and some definitions from measure theory. Then, we expose the main contributions of this paper:

- A first limit of (Sajjadi et al., 2018) is the restriction to discrete probability distributions (*i.e.* considering that samples live in a finite state space $\Omega$). In Section 3, this assumption is dispensed by defining Precision-Recall curves from *arbitrary* probability distributions for which some properties are then given;
- In the original work of (Sajjadi et al., 2018) the Precision-Recall curves approach was opposed to the hypothesis testing techniques from (Lopez-Paz & Oquab, 2017); we demonstrate in Section 4 that precision and recall are actually linear combinations of type I and type II errors of optimal likelihood ratio classifiers, and give as well some upper-bound guarantee for the estimation of Precision-Recall curves with non-optimal classifiers; Besides, our formulation also exhibits a relationship with the MCR notion proposed by (Lin et al., 2018) which turns out to be the ROC curves ($1-$type I versus type II errors) for optimal classifiers;
- Section 5 details the proposed algorithm to estimate Precision-Recall curves more accurately; the clustering optimization step used in the original method is now simply replaced by the training of a classifier which learns to separate samples from the two datasets;
- The experimental Section 6 demonstrates the advantage of the proposed formulation in a controlled setting using labelled datasets (CIFAR10 and ImageNet categories), and then shows its practical interest for evaluating state-of-the art generative image models.

## 2. Notions from standard measure theory

We start these notes by recalling some standard notations, definitions, and results of measure theory. For the remainder, $(\Omega, \mathcal{A})$ represents a common measurable space, and we will denote $\mathcal{M}(\Omega)$ the set of signed measures, $\mathcal{M}^+(\Omega)$ the set of positive measures and $\mathcal{M}_p(\Omega)$ the set of probability distributions over that measurable space.

**Definition 1.** *Let $\mu, \nu$ two signed measures. We denote by*

- $\text{supp}(\mu)$*, the support of $\mu$;*

- $\frac{d\mu}{d\nu}$, *the Radon-Nykodim derivative of* $\mu$ *w.r.t.* $\nu$;
- $|\mu|$, *the total variation measure of* $\mu$;
- $\mu \wedge \nu = \min(\mu, \nu) := \frac{1}{2}(\mu + \nu - |\mu - \nu|)$ *(a.k.a the measure of largest common mass between* $\mu$ *and* $\nu$ *(Piccoli et al., 2017)).*

The extended half real-line is denoted by $\overline{\mathbb{R}^+} = \mathbb{R}^+ \cup \{\infty\}$.

**Theorem 1** (Hahn decomposition)**.** *Let* $\mu \in \mathcal{M}(\Omega)$. *Then there exists an essentially unique partition* $\Omega = \Omega_\mu^+ \sqcup \Omega_\mu^-$ (i.e. *where* $\Omega_\mu^+ \cap \Omega_\mu^- = \varnothing$) *such that* $\forall A \in \mathcal{A}$:

$$\begin{aligned} A \subset \Omega_\mu^+ &\Rightarrow \mu(A) \geq 0 \\ A \subset \Omega_\mu^- &\Rightarrow \mu(A) \leq 0 \end{aligned} .$$

**Corollary 1.** *Let* $\mu, \nu \in \mathcal{M}^+(\Omega)$. *Then,* $\forall A \in \mathcal{A}$, *we have:*

$$(\mu \wedge \nu)(A) = \mu(A \cap \Omega_{\mu-\nu}^-) + \nu(A \cap \Omega_{\mu-\nu}^+).$$

## 3. Precision-Recall set and curve

We follow (Sajjadi et al., 2018) for the definition of the Precision-Recall (PR) set that we extent to any arbitrary pair of probability distributions $P$ and $Q$, up to two additional minor changes. First, we have tried to adapt their definition in a shorter form. Second, we include the left and lower boundaries in the PR set.

**Definition 2.** *Let* $P, Q$ *two distributions from* $\mathcal{M}_p(\Omega)$. *We refer to the Precision-Recall set* PRD$(P, Q)$ *as the set of Precision-Recall pairs* $(\alpha, \beta) \in \mathbb{R}^+ \times \mathbb{R}^+$ *such that*

$$\exists\, \mu \in \mathcal{M}_p(\Omega), P \geq \beta\mu, Q \geq \alpha\mu\ . \tag{1}$$

The *precision* value $\alpha$ is related to the proportion of the generated distribution $Q$ that match the true data $P$, while conversely the *recall* value $\beta$ is the amount of the distribution $P$ that can be reconstructed from $Q$. Therefore, in the context of generative models, one would like to have admissible precision-recall pairs that are as close to $(1, 1)$ as possible. One can then easily show the following properties:

**Theorem 2.** *Let* $P, Q$ *two distributions from* $\mathcal{M}_p(\Omega)$. *Then,*

1. $(0, 0) \in$ PRD$(P, Q) \subset [0, 1] \times [0, 1]$;
2. $P = Q \Leftrightarrow (1, 1) \in$ PRD$(P, Q)$;
3. $(\alpha, \beta) \in$ PRD$(P, Q)$ *and* $\alpha' \leq \alpha, \beta' \leq \beta$ *implies that* $(\alpha', \beta') \in$ PRD$(P, Q)$.

Because of the lack of natural order on $[0, 1] \times [0, 1]$, no point of PRD$(P, Q)$ is strictly better than all the others. Yet, the singular importance of $(1, 1)$ should draw our attention to the Pareto front of PRD$(P, Q)$ defined as follows.

**Definition 3.** *The precision recall-curve* $\partial$PRD$(P, Q)$ *is the set of* $(\alpha, \beta) \in$ PRD$(P, Q)$ *such that*

$$\forall (\alpha', \beta') \in \text{PRD}(P, Q), \alpha \geq \alpha' \text{ or } \beta \geq \beta'.$$

In fact, this frontier is a curve for which (Sajjadi et al., 2018) have exposed a parameterization. We generalize their result here (dropping any restriction to discrete probabilities).

**Theorem 3.** *Let* $P, Q$ *two distributions from* $\mathcal{M}_p(\Omega)$ *and* $(\alpha, \beta)$ *positive. Then, denoting*[1]

$$\forall \lambda \in \overline{\mathbb{R}^+}, \begin{cases} \alpha_\lambda := ((\lambda P) \wedge Q)\,(\Omega) \\ \beta_\lambda := \left(P \wedge \frac{1}{\lambda} Q\right)(\Omega) \end{cases} \tag{2}$$

1. $(\alpha, \beta) \in$ PRD$(P, Q)$ *iff* $\alpha \leq \alpha_\lambda$ *and* $\beta \leq \beta_\lambda$ *where* $\lambda := \frac{\alpha}{\beta} \in \overline{\mathbb{R}^+}$.
2. *As a result, the PR curve can be parameterized as:*

$$\partial\text{PRD}(P, Q) = \{(\alpha_\lambda, \beta_\lambda) / \lambda \in \overline{\mathbb{R}^+}\}\,. \tag{3}$$

*Proof.* The second point derives easily from the first which we demonstrate now. Let $(\alpha, \beta)$ positive and $\lambda := \frac{\alpha}{\beta}$. By definition $(\alpha, \beta) \in \text{PRD}(P, Q)$ iff $\exists\, \mu \in \mathcal{M}_p(\Omega)$

$$P \geq \beta\mu = \frac{\alpha}{\lambda}\mu \text{ and } Q \geq \alpha\mu$$

iff

$$\mu \leq \frac{1}{\alpha}(\lambda P \wedge Q)(\Omega) = \frac{1}{\beta}(P \wedge \frac{Q}{\lambda})$$

which yields the expected criteria given that $\mu(\Omega) = 1$. □

## 4. Link with binary classification

Let us consider samples $(X_i, Y_i) \sim P \times Q$ and as many Bernoulli variables $U_i \sim \mathcal{B}_{\frac{1}{2}}$. And let $Z_i = U_i X_i + (1 - U_i) Y_i$. Then $Z_i \sim \mathbb{P}_Z$ follows a mixture of $P$ and $Q$, namely $\mathbb{P}_Z = \frac{1}{2}(P + Q)$. Then, let us consider the binary classification task where from $Z_i$, one should decide whether $U_i = 1$ (often referred to as the null hypothesis). We show that the precision-recall curve can be reinterpreted as mixed error rates of binary classifiers obtained as likelihood ratio tests (hence the most powerful classifiers according to the celebrated Neyman-Pearson lemma).

**Theorem 4.** *Let* $\lambda \geq 0$. *Let* $Z = UX + (1 - U)Y$ *where* $(X, Y, U) \sim P \times Q \times \mathcal{B}_{\frac{1}{2}}$. *Defining the likelihood ratio classifier* $\tilde{U}$ *as the following indicator function*

$$\tilde{U}(Z) := \mathbb{1}_{\lambda \frac{dP}{d\mathbb{P}_Z}(Z) \geq \frac{dQ}{d\mathbb{P}_Z}(Z)}\,, \tag{4}$$

*then,* $\alpha_\lambda = \lambda \mathbb{P}(\tilde{U} = 0 | U = 1) + \mathbb{P}(\tilde{U} = 1 | U = 0)\,.$

[1] As is conventionally surmised in measure theory $0 \times \infty = 0$ so that $\alpha_\infty = Q(\text{supp}(P))$ and $\beta_0 = P(\text{supp}(Q))$.

*Proof.* Note that we can reformulate $\tilde{U}$ as $\tilde{U}(Z) = \mathbb{1}_{\Omega^+_{\lambda P - Q}}(Z)$. Then,

$$\begin{aligned}\mathbb{P}(\tilde{U}=1|U=0) &= \int_\Omega \mathbb{1}_{\Omega^+_{\lambda P-Q}}(z) d\mathbb{P}_Z(z|U=0) \\ &= \int_\Omega \mathbb{1}_{\Omega^+_{\lambda P-Q}}(z) dQ(z) = Q(\Omega^+_{\lambda P-Q})\end{aligned}.$$

Now, using $\mathbb{1}_{\tilde{U}=0} = \mathbb{1}_{\Omega^-_{\lambda P-Q}}$, we have similarly $\mathbb{P}(\tilde{U} = 0|U=1) = P(\Omega^-_{\lambda P-Q})$. Combining the two errors, we get

$$\lambda\mathbb{P}(\tilde{U}=0|U=1)+\mathbb{P}(\tilde{U}=1|U=0) = (\lambda P \wedge Q)(\Omega) = \alpha_\lambda$$

where we have used Corollary 1. □

The previous protocol demonstrates that points on the PR curve are actually a linear combination of type I error rate (probability of rejection of the true null hypothesis $\mathbb{P}(\tilde{U}=0|U=1)$) with type II error rate ($\mathbb{P}(\tilde{U}=1|U=0)$). It also shows that if one is able to compute the likelihood ratio classifier, then one could virtually obtain the precision-recall curve $\partial\text{PRD}(P,Q)$. Unfortunately, in practice the likelihoods are unknown. To alleviate this set-back, one can argue like (Menon & Ong, 2016) that optimizing standard classification losses is *in fine* equivalent to minimize a Bregman divergence to the likelihood ratio. Besides, we are going to show that using Eq. (4) with any other classifier always yields an over-estimation of $\alpha_\lambda$ and $\beta_\lambda$. To do so, we will need the following lemma, which is merely a quantitative version of the Neyman-Pearson Lemma.

**Lemma 1.** *Let $\tilde{U}(Z)$ the likelihood ratio classifier defined in Eq. (4), associated with the ratio $\lambda$. Then, any classifier $U'(Z)$ with a lower type II error, that is such that*

$$\mathbb{P}(U'=1|U=0) \leq P(\tilde{U}=1|U=0)\,,$$

*undergoes an increase of the type I error such that*

$$\begin{cases} \alpha'_\lambda := \lambda P(U'=0|U=1) + P(U'=1|U=0) & \geq \alpha_\lambda \\ \beta'_\lambda := P(U'=0|U=1) + \frac{1}{\lambda}P(U'=1|U=0) & \geq \beta_\lambda \end{cases}$$

*Proof.* The proof is similar to the classical proof of the Neyman-Pearson lemma (see Appendix A). □

**Theorem 5.** *Let $\tilde{U}$ the likelihood ratio classifier from Eq. (4) associated with the ratio $\lambda$, and let $U'$ be any other classifier. Using precision-recall pair $(\alpha'_\lambda, \beta'_\lambda)$ defined in Lemma 1, we have that*

$$\alpha'_\lambda \geq \alpha_\lambda \text{ and } \beta'_\lambda \geq \beta_\lambda$$

*Proof.* The proof uses Lemma 1 and its symmetric version (obtained by swapping the role of type-I and type-II errors). Three cases may arise:

1. If $\mathbb{P}(U'=0|U=1) \geq \mathbb{P}(\tilde{U}=0|U=1)$ and $\mathbb{P}(U'=1|U=0) \geq \mathbb{P}(\tilde{U}=1|U-0)$ then the conclusion of the theorem is trivially true;
2. If $\mathbb{P}(U'=1|U=0) \leq \mathbb{P}(\tilde{U}=1|U=0)$, then the conclusion is ensured by Lemma 1;
3. If $\mathbb{P}(U'=0|U=1) \leq \mathbb{P}(\tilde{U}=0|U=1)$, then one should use the symmetric version of Lemma 1.

□

## 5. Algorithm

Based on the above analysis, we propose the Algorithm 1 to estimate (via the function `estimatePRCurve`) the Precision-Recall curve of two probability distributions known through their respective sample sets.

**Binary Classification** We know from Theorem 4 that the Precision-Recall curve can be exactly inferred from the likelihood ratio classifier denoted as $\tilde{U}$. However, as explained earlier, since both the generated and target distributions ($Q$ and $P$ respectively) are unknown, one could not compute in practice this optimal classifier. Instead, we propose to *train* a binary classifier $U'$. Recall that from Theorem 5 the estimated PR curve, being computed with a sub-optimal classifier, lies therefore above the optimal one. We only assume in the following that the classifier –denoted to as $f$ in the algorithm description– which is returned by the function `learnClassifier` after the training, ranges in a continuous interval (e.g. $[0,1]$), so that the binary classifier $U'$ is actually obtained by thresholding: $U'(Z) = \mathbb{1}_{f(Z)\geq t}$.

As a result, since the classifier needs some training data, the $N$ sample pairs $\mathcal{D} = \{(X_i, Y_i), 1 \leq i \leq N, X_i \sim P, Y_i \sim Q\}$ in the input dataset are first split into two sets $\mathcal{D}^{\text{train}}$ and $\mathcal{D}^{\text{test}}$ (function `createTrainTest`). For each image pair $(X_i, Y_i)$, a Bernoulli random variable $U_i$ with probability $\frac{1}{2}$ is drawn to decide whether a true sample $X_i$ (when $U_i = 1$) or a fake one $Y_i$ (when $U_i = 0$) is used for the training set $\mathcal{D}^{\text{train}}$. The other sample is then collected in the test set $\mathcal{D}^{\text{test}}$ to compute the PR curve.

**Precision and Recall estimation** Recall from Theorem 3 that the PR curve $\partial\text{PRD} = \{(\alpha_\lambda, \beta_\lambda), \lambda \in \overline{\mathbb{R}^+}\}$ is parametrized by the ratio $\lambda = \frac{\alpha}{\beta}$ between precision $\alpha$ and recall $\beta$. We denote by $\partial\text{PRD}_\Lambda$ the approximated PR curve when this parameter takes values in the set $\Lambda$.

Given a test dataset $\mathcal{D}^{\text{test}}$, the function `estimatePRD` computes the PR values $(\alpha_\lambda, \beta_\lambda)$ from the *false positive rate* fpr and the *false negative rate* fnr of the trained classifier $f$:

- fpr corresponds to the empirical type I error rate, that is here (arbitrarily) the proportion of real samples $z = X_i$

**Inputs:** Dataset of target/source sample pairs: $\mathcal{D} = \{(X_i, Y_i) \sim P \times Q \text{ i.i.d}/i \in \{1, \dots, N\}\}$, Parameterization of the PR curve: $\Lambda = \{\lambda_1, \dots, \lambda_L\}$
**Output:** $\partial\text{PRD}_\Lambda \simeq \{(\alpha_\lambda, \beta_\lambda)/\lambda \in \Lambda\}$
**Algorithm** `estimatePRCurve` $(\mathcal{D}, \Lambda)$

$\mathcal{D}^{train}, \mathcal{D}^{test}$ =`createTrainTest` $(\mathcal{D})$
$f$ =`learnClassifier` $(\mathcal{D}^{train})$
$\partial\text{PRD}_\Lambda$ =`estimatePRD` $(f, \mathcal{D}^{test}, \Lambda)$
**return** $\partial\text{PRD}_\Lambda$

**Procedure** `createTrainTest` $(\mathcal{D})$

$\mathcal{D}^{train} = \varnothing, \mathcal{D}^{test} = \varnothing$
**for** $i \in \{1, \dots, N\}$ **do**
  $U_i \sim \mathcal{B}_{\frac{1}{2}}$
  $Z_i^{train} = U_i X_i + (1 - U_i) Y_i$
  $Z_i^{test} = (1 - U_i) X_i + U_i Y_i$
  $\mathcal{D}^{train} \leftarrow \mathcal{D}^{train} \cup \{(Z_i^{train}, U_i)\}$
  $\mathcal{D}^{test} \leftarrow \mathcal{D}^{test} \cup \{(Z_i^{test}, 1 - U_i)\}$
**end**
**return** $\mathcal{D}^{train}, \mathcal{D}^{test}$

**Procedure** `estimatePRD` $(f, \mathcal{D}^{test}, \Lambda)$

$fVals = \{f(z)/(z, u) \in \mathcal{D}^{test}\}$
$errRates = \varnothing$
$N_j = |\{(z, u) \in \mathcal{D}^{test}/u = j\}|$, for $j \in \{0, 1\}$
**for** $t \in fVals$ **do**
  $\text{fpr} = \frac{1}{N_1} \left|\{(z, u) \in \mathcal{D}^{test}/f(z) < t, u = 1\}\right|$
  $\text{fnr} = \frac{1}{N_0} \left|\{(z, u) \in \mathcal{D}^{test}/f(z) \geq t, u = 0\}\right|$
  $errRates \leftarrow errRates \cup \{(fpr, fnr)\}$
**end**
$\partial\text{PRD}_\Lambda = \varnothing$
**for** $\lambda \in \Lambda$ **do**
  $\alpha_\lambda = \min(\{\lambda \text{fpr} + \text{fnr}/(\text{fpr}, \text{fnr}) \in errRates\})$
  $\partial\text{PRD}_\Lambda \leftarrow \partial\text{PRD}_\Lambda \cup \{(\alpha_\lambda, \frac{\alpha_\lambda}{\lambda})\}$
**end**
**return** $\partial\text{PRD}_\Lambda$

**Algorithm 1:** Classification-based estimation of the Precision-Recall curve.

(for which $u = 1$) that are misclassified as generated samples (*i.e.* when $f(z) < t$);

- conversely, fnr is the empirical type II error rate, that is the proportion of generated samples $z = Y_i$ (for which $u = 0$) that are misclassified as real samples (*i.e.* $f(z) \geq t$);

Now, this raises the question of setting the threshold $t$ that defines the binary classifier $U'(z) = \mathbb{1}_{f(z)>t}$. Since Theorem 5 states that the computed precision and recall values $(\alpha_\lambda, \beta_\lambda)$ are actually upper-bound estimates, we use the minimum of these estimates when spanning the threshold value in the range of $f$. Note that it is sufficient to consider the finite set $fVals$ of classification scores over $\mathcal{D}^{test}$.

**Comparison with ROC curves** Using a ROC curve (for Receiver Operating Characteristic) to evaluate a binary classifier is very common in machine learning. Let us recall that it is the curve of the true positive rate $(1 - \text{fnr})$ against the false positive rate (fpr) obtained for different classification thresholds. Considering again the likelihood ratio test classifiers for all possible ratios would then provide the Pareto optimal ROC curve and could be used to assess if $P$ and $Q$ are similar or not. It turns out that the the frontier of the Mode Collapse Region proposed by (Lin et al., 2018) provides exactly this optimal ROC curve. For the recall, this notion is originally defined as follows:

$$MCR(P, Q) = \{(\epsilon, \delta)/0 \leq \epsilon < \delta \leq 1, \exists A \in \mathcal{A}, P(A) \geq \delta, Q(A) \leq \epsilon\}$$

From this definition, one can see that the MCR exhibits mode dropping by analyzing if part of the mass of $P$ is absent from $Q$. The notion differs from PRD at least in two ways. First MCR is not symmetric in $P$ and $Q$. Then it uses the mass of a subset $A$ instead of an auxiliary measure $\mu$ to characterize the shared / unshared mass between $P$ and $Q$. Despite those differences, the two notions serve a similar purpose. Given their respective interpretation as optimal type I vs type II errors, they mostly differ in terms of visual characterization of mode dropping.

## 6. Experiments

In this section we demonstrate that Algorithm 1 is consistent with the expected notion of precision and recall on controlled datasets such as CIFAR-10 and Imagenet. The results even compare favorably to (Sajjadi et al., 2018) for such datasets. The situation is more complex when one distribution is made of generated samples, because the expected gold-standard precision-recall curve cannot be predicted in a trivial way.

In all our experiments, we compute the precision-recall curve between the distribution of features of the Inception Network (Szegedy et al., 2016) (or some other network when specified) instead of using raw images (this choice will be discussed later on). In simple words, it means that we first extract inception features before training / evaluating the classifier. The classifier itself is an ensemble of 10 linear classifiers. The consensus between the linear classifiers is computed by evaluating the median of their predictions. Besides, each linear classifier is trained independently with the ADAM algorithm. We progressively decrease the learning rate starting from $10^{-3}$ for 50 epochs and use a

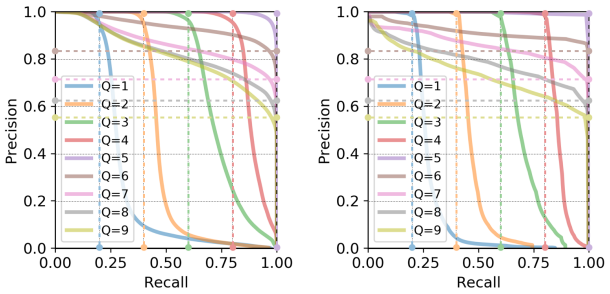


*Figure 2.* Precision-recall curves for $P$ made of the five first classes of CIFAR-10 versus $Q$ made of $q \in \{1, \dots, 9\}$ first classes. Left estimate from (Sajjadi et al., 2018) and right our implementation.

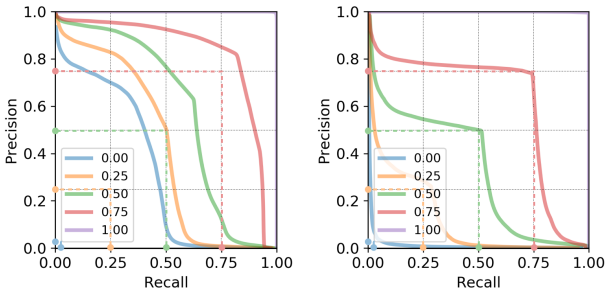


*Figure 3.* PR curves for $P$ and $Q$ made of 80 classes from ImageNet. The ratio of common classes varies from 0 to 100%. Left: from (Sajjadi et al., 2018). Right: our implementation.

fixed weight decay of $0.1$. Any sophisticated classification method could be used to achieve our goal (deeper neural network, non-linear SVM, *etc*), but this simplistic ensemble network turned out to be sufficient in practice. Observe that this training procedure is replacing the pre-processing (K-means clustering) in the original approach of (Sajjadi et al., 2018), which relies also on inception features so that both methods share a similar time complexity.

Figure 2 reproduces an experiment proposed by (Sajjadi et al., 2018). It presents the estimated precision-recall curves on distributions made from CIFAR-10 samples. The reference distribution $P$ is always the same and it gathers samples from the first $5$ classes. On the other hand, $Q$ is composed of the first $q$ classes. When $q \leq 5$ we should expect a rectangular curve with a maximum precision of $1$ and maximum recall of $q/5$ (as illustrated in middle of Fig. 1). Similarly, when $q > 5$ the expected curve is also rectangular one, but this time the maximum precision is $5/q$ and the maximum recall is $1$ (Fig. 1, left). These expected theoretical curves are shown in dash. The original implementation from (Sajjadi et al., 2018) is shown on the left and ours on the right. It is clear that both methods capture the intended behaviour of precision and recall. Besides, two subtle differences can be observed. First, as implied by Theorem 5 our implementation is always overestimating the theoretical curve (up to the variance due to finite samples). On the contrary, the clustering approach does not provide similar guarantee (as observed experimentally). Second, our implementation is slightly more accurate around the horizontal and vertical transitions.

One particular difficulty with the clustering approach lies in choosing the number of clusters. While the original choice of $20$ is reasonable for simple distributions, it can fail to capture the complexity of strongly multi-modal distributions. To highlight this phenomenon, we present in Figure 3 another controlled experiment with Imagenet samples. In this case, $P$ and $Q$ are both composed of samples from $80$ classes, with a fixed ratio $\rho$ of common classes. In this case, the expected curves can be predicted (see dash curves). They correspond to rectangular curves with both maximal precision and recall equal to $\rho$. As can be seen on the experimental curves, the clustering approach is prone to mixing the two datasets in the same clusters. It therefore produces histograms that share a much heavier mass than the non discretized distributions, resulting in PR curves that depart strongly from the expected ones. Of course, such a drawback could be partially fixed by adapting the number of clusters. However even then the clustering approach may fail, as is demonstrated in Figure 4. In this experimient, the distribution $P$ is obtained approximately 60% female faces and 40% male faces from the CelebA dataset, while $Q$ is composed of female only. The theoretical curve is a sharp transition arising at recall $0.6$. This is well captured by our estimate (right curve) while varying the number of clusters always leads to an oversmooth estimate, with either under estimated precision or over-estimated recall.

**Experiments on Generated Images** Figure 5 illustrates the proposed approach to three GANs trained on the Celeba-HQ (Karras et al., 2018) dataset. We highlight the two recent approaches of progressive GANs in (Karras et al., 2018) and the 0 centered gradient penalty ResNets found in (Mescheder et al., 2018) as they produce realistic images. For comparison, we also include DCGAN (Radford et al., 2015). For analysis with Algorithm 1, we choose the first $N = 1000$ images of Celeba-HQ, and generate as many images with each GAN. For training the classifiers, we split each set (real and fake images) into 900 training images and 100 test images. In light of the previous experiments, we choose to train our architecture on top of vision relevant features. Because we are dealing with faces, we choose the convolutional part of the VGG-Face network (Parkhi et al., 2015). One advantage on using VGG-Face is that artifacts present in generated images, such as unrealistic backgrounds, are mitigated by the VGG-face network, so that classification can focus on the realism of facial features. Of course, small artifacts can be present in even high quality

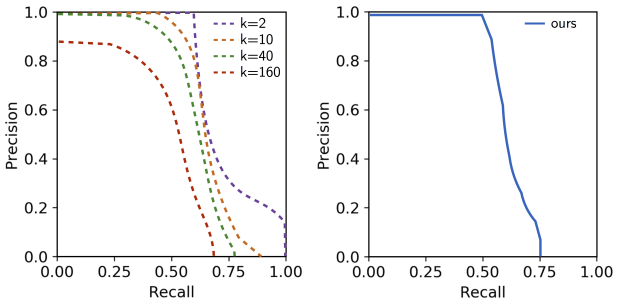


*Figure 4.* PR curves when $P$ is composed of faces from CelebA (60% females) and $Q$ is composed of females only.

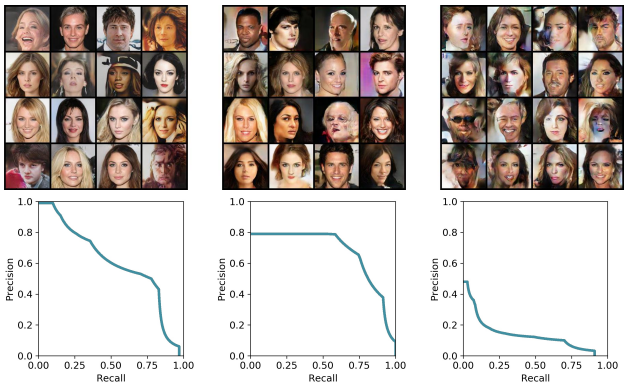


*Figure 5.* Precision-recall curves and generated images for various popular GANs on CelebA-HQ dataset (Karras et al., 2018). From left to right: PGGAN (Karras et al., 2018), ResNet (Mescheder et al., 2018), and DCGAN (Radford et al., 2015).

generators and a perfect classifier could "cheat" by only using seeing such artifacts. Fig. 5, shows the computed PR curves for the three generators. Intuitively, networks with high precision should generate realistic images consistently. Progressive GANs achieve a maximum precision of 1.0, and overall high precision, which is visually consistent. DCGAN is producing unrealistic images which is reflected by it's overall low precision. In some sense, recall reflects the diversity of the generated images with respect to the dataset and it is interesting to note all networks achieved higher recall than precision. Finally, for the sake of comparison with the FID (Heusel et al., 2017), the networks in Fig. 5 achieved FIDs of 25.23, 27.61 and 67.84 respectively from left to right (lower is better).

Next, we analyze BigGAN (Brock et al., 2019) on ImageNet for our classification approach and the clustering approach presented in (Sajjadi et al., 2018). Both approaches use inception features as before. We take 80 images from the first 40 classes of ImageNet, and then 20 images from the first 40 classes for test images. We use 20 clusters for the K-means approach and a single linear layer for the classification approach. Fig. 6 highlights a large difference between the approaches; the clustering approach overestimates the similarities between the distributions and the classification approach easily separates the two distributions. As was demonstrated in Fig. 3, there are more classes than clusters, which could explain why images from both distributions may fall into the same clusters, in which (Sajjadi et al., 2018) will fail to discern the two distributions. It is interesting to note that the classifier easily separates the distributions despite the inception features being sparse for image samples. One can observe a lack of intra-class diversity in the BigGAN samples, which may be how the classifier discerns the samples. We leave further investigation of this discrepancy for future work.

## 7. Discussion and future work

In this paper, we have revisited a recent definition of precision-recall curve for comparing two distributions. Besides extending precision and recall to arbitrary distributions, we have exhibited a dual perspective on such notions. In this new view, precision-recall curves are seen through the prism of binary classification. Our central result states that the Pareto optimal precision-recall pairs can be obtained as linear combinations of type I and type II errors of likelihood ratio classifiers. Last, we have provided a novel algorithm to evaluate the precision-recall curves from random samples drawn within the two involved distributions.

**Discussion** One achievement of our formulation is that one can directly define the precision-recall curves of distributions defined on continuous manifolds. In particular, our definition could be applied directly in the image domain, instead of first embedding the distribution in a feature space. From the strict computation perspective, there should not be any daunting obstacle in the way, as soon as we can have access to enough data to train a good classifier. This is usually the case for generative models, since the standard datasets are quite massive.

However, it is not obvious whether classifiers trained on raw-data provide useful notions of PR-curves. Indeed, given the current state of affairs of generative modeling, we think that the raw image curves may be less useful. Indeed, until now, even the best generative models produce artifacts (blurriness, structured noise, etc.). As such, the theoretical distributions (real and generated) are mutually singular. So, their theoretical precision-recall curve should be always trivial (*i.e.* reduced to the origin). It is hence a necessary evil to embed the distributions into a feature space as it allows a classifier to focus its attention on statistical disparities that

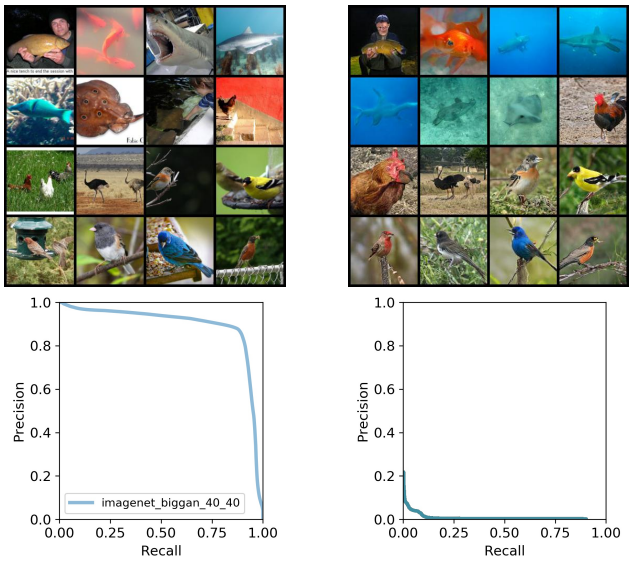


*Figure 6.* Evaluating generated samples on ImageNet. First row: Samples from various categories of ImageNet (on the left), and generated samples for the same categories from BigGAN (Brock et al., 2019) (on the right). Second Row: PR curves computed with the clustering approach of (Sajjadi et al., 2018) and ours.

are meaningful for the task at hand. For instance, when evaluating a face generator, it makes sense to use features that are representative of facial attributes. Nonetheless, future work should investigate a wider variety of pre-trained features as well as classifiers trained on raw data to determine which method is most suitable for computing PR curves.

**Perspectives** This work offers some interesting perspectives that we would like to investigate in the future. First, as opposed to the usual GAN training procedure where a scalar divergence is used to assess the similarity between generated and target distributions, one could use the proposed precision and recall definitions to control the quality of the generator while preventing mode-collapse. For instance, the discriminator could use the role of the classifier, as it has been done in (Salimans et al., 2016).

Another interesting aspect is that like most existing divergences comparing probability distributions, the proposed approach is based on likelihood ratios that only compare samples having the same values. More flexible ways do exist to compare distributions, based for instance on optimal transport, such as the Wasserstein distance (e.g 1-Wasserstein GAN (Arjovsky et al., 2017b)) and could be adapted to keep the notion of trade-off between quality and diversity.

## A. Proof of Lemma 1

Let $\varepsilon \geq 0$ such that $\mathbb{P}(U'=1|U=0) = \mathbb{P}(\tilde{U}=1|U=0) - \varepsilon$. First, we decompose $\beta'_\lambda$ into 4 terms

$$\begin{aligned}
\beta'_\lambda =& \mathbb{P}(U'=0|U=1) + \tfrac{1}{\lambda}\mathbb{P}(U'=1|U=0) \\
=& \mathbb{P}(U'=0,\tilde{U}=0|U=1) + \mathbb{P}(U'=0,\tilde{U}=1|U=1) \\
&+ \tfrac{1}{\lambda}\mathbb{P}(U'=1|U=0) \\
=& \mathbb{P}(\tilde{U}=0|U=1) - \mathbb{P}(\tilde{U}=0,U'=1|U=1) \\
&+ \mathbb{P}(U'=0,\tilde{U}=1|U=1) + \tfrac{1}{\lambda}\mathbb{P}(U'=1|U=0).
\end{aligned}$$

Considering separately each of the previous terms, we have

$$\begin{aligned}
& A = \mathbb{P}(\tilde{U}=0|U=1)\,, \\
-B =& \mathbb{P}(U'=1,\tilde{U}=0|U=1) = \int \mathbb{1}_{U'=1}\mathbb{1}_{\tilde{U}=0}dP \\
\leq& \int \mathbb{1}_{U'=1}\mathbb{1}_{\tilde{U}=0}\tfrac{1}{\lambda}dQ = \tfrac{1}{\lambda}\mathbb{P}(U'=1,\tilde{U}=0|U-0) \\
=& \tfrac{1}{\lambda}(\mathbb{P}(U'=1|U=0) - \mathbb{P}(U'=1,\tilde{U}=1|U=0)) \\
=& \tfrac{1}{\lambda}\Big(\mathbb{P}(\tilde{U}=1|U=0) - \varepsilon \\
& - (\mathbb{P}(\tilde{U}=1|U=0) - \mathbb{P}(U'=0,\tilde{U}=1|U=0))\Big) \\
=& \tfrac{1}{\lambda}(\mathbb{P}(U'=0,\tilde{U}=1|U=0) - \varepsilon).
\end{aligned}$$

Finally

$$B \geq -\tfrac{1}{\lambda}(\mathbb{P}(U'=0,\tilde{U}=1|U=0) - \varepsilon).$$

Similarly

$$\begin{aligned}
C =& \mathbb{P}(U'=0,\tilde{U}=1|U=1) = \int \mathbb{1}_{U'=0}\mathbb{1}_{\tilde{U}=1}dP \\
\geq& \int \mathbb{1}_{U'=0}\mathbb{1}_{\tilde{U}=1}\tfrac{1}{\lambda}dQ = \tfrac{1}{\lambda}\mathbb{P}(U'=0,\tilde{U}=1|U=0)\,.
\end{aligned}$$

Using both inequalities for $B$ and $C$, one gets

$$B + C \geq \frac{\varepsilon}{\lambda}.$$

Last,

$$D = \tfrac{1}{\lambda}\mathbb{P}(U'=1|U=0) = \tfrac{1}{\lambda}(\mathbb{P}(\tilde{U}=1|U=0) - \varepsilon).$$

Putting everything together, namely $\beta'_\lambda = A + B + C + D$, yields

$$\beta'_\lambda \geq \mathbb{P}(\tilde{U}=0|U=1) + \tfrac{1}{\lambda}\mathbb{P}(\tilde{U}=1|U=0) = \beta_\lambda.$$

Using (by a slight abuse of notation) $\alpha_\lambda = \lambda\beta_\lambda$ and $\alpha'_\lambda = \lambda\beta_\lambda \geq \alpha_\lambda$ concludes the proof. □

## Acknowledgments

This work was supported by fundings from *Région Normandie* under grant *RIN NormanD'eep*. The authors are grateful to the anonymous reviewers for their valuable comments and suggestions.

## References

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein generative adversarial networks. In Precup, D. and Teh, Y. W. (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 214–223, International Convention Centre, Sydney, Australia, 06–11 Aug 2017a. PMLR. URL http://proceedings.mlr.press/v70/arjovsky17a.html.

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017b.

Brock, A., Donahue, J., and Simonyan, K. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=B1xsqj09Fm.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, pp. 6626–6637, 2017.

Iizuka, S., Simo-Serra, E., and Ishikawa, H. Globally and locally consistent image completion. *ACM Transactions on Graphics (TOG)*, 36(4):107, 2017.

Im, D. J., Ma, H., Taylor, G., and Branson, K. Quantitatively evaluating gans with divergences proposed for training. *arXiv preprint arXiv:1803.01045*, 2018.

Karras, T., Aila, T., Laine, S., and Lehtinen, J. Progressive growing of GANs for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=Hk99zCeAb.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2014.

Lample, G., Conneau, A., Ranzato, M., Denoyer, L., and Jégou, H. Word translation without parallel data. In *International Conference on Learning Representations (ICLR)*, 2018. URL https://openreview.net/forum?id=H196sainb.

Lin, Z., Khetan, A., Fanti, G., and Oh, S. Pacgan: The power of two samples in generative adversarial networks. In *Advances in Neural Information Processing Systems*, pp. 1498–1507, 2018.

Lopez-Paz, D. and Oquab, M. Revisiting classifier two-sample tests. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017.

Lucic, M., Kurach, K., Michalski, M., Gelly, S., and Bousquet, O. Are gans created equal? a large-scale study. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31*, pp. 698–707. Curran Associates, Inc., 2018.

Menon, A. and Ong, C. S. Linking losses for density ratio and class-probability estimation. In *International Conference on Machine Learning*, pp. 304–313, 2016.

Mescheder, L., Geiger, A., and Nowozin, S. Which training methods for gans do actually converge? In *International Conference on Machine Learning (ICML)*, pp. 3478–3487, 2018.

Metz, L., Poole, B., Pfau, D., and Sohl-Dickstein, J. Unrolled generative adversarial networks. 2017.

Parkhi, O. M., Vedaldi, A., and Zisserman, A. Deep face recognition. In *British Machine Vision Conference*, 2015.

Pascual, S., Bonafonte, A., and Serra, J. Segan: Speech enhancement generative adversarial network. *InterSpeech*, 2017.

Piccoli, B., Rossi, F., and Tournus, M. A norm for signed measures. application to non local transport equation with source term. 2017.

Radford, A., Metz, L., and Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *CoRR*, abs/1511.06434, 2015. URL http://arxiv.org/abs/1511.06434.

Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., and Lee, H. Generative adversarial text to image synthesis. *Proceedings of the International Conference on Machine Learning (ICML)*, 2016.

Sajjadi, M. S. M., Bachem, O., Lucic, M., Bousquet, O., and Gelly, S. Assessing generative models via precision and recall. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31*, pp. 5234–5243. Curran Associates, Inc., 2018.

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X., and Chen, X. Improved techniques for training gans. In Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 29*, pp. 2234–2242. Curran Associates, Inc., 2016.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

Webster, R., Rabin, J., Simon, L., and Jurie, F. Detecting overfitting of deep generative networks via latent recovery. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.