# Report - Data Science Lab Project 2: Improving and Tuning GAN Precision and Recall

Jules Roques
Rayane Dakhlaoui
Chloé Court

November 17, 2025

# 1 Introduction

Generative Adversarial Networks (GANs) have become a cornerstone of modern generative modeling, enabling the synthesis of highly realistic data samples across a wide range of domains. However, despite their impressive capabilities, GANs often suffer from an inherent trade-off between precision (the realism of generated samples) and recall (the diversity or coverage of the data distribution). In practice, many GANs tend to favor one aspect at the expense of the other, leading to mode collapse or low-fidelity outputs.

This project focuses on exploring and improving this precision–recall balance through the lens of f-divergence–based GANs (f-GANs). Unlike the original Jensen–Shannon–based formulation, f-GANs generalize the adversarial framework by allowing the use of different divergence measures (e.g., KL, Pearson, reverse-KL), providing finer control over the generator's behavior. By systematically varying the underlying divergence, we can empirically study how different training objectives influence the model's precision and recall characteristics.

To further refine this trade-off, we incorporate soft truncation techniques and compute precision–recall (PR) curves to quantitatively evaluate generative performance. Finally, we experiment with Discriminator Rejection Sampling (DRS) as a post-processing method to selectively filter generated samples, thereby enhancing precision or recall according to the target application. Through this work, we aim to provide a clearer understanding of how divergence choices, training strategies, and post-hoc sampling methods jointly shape the performance landscape of GANs.

# 2 Evaluation Metrics

For all metrics, directly comparing raw hand-written digits between each other would not lead to relevant information, as the distance in the pixel space does not correspond to human perception of these images. We must then project our images in a feature space that carries meaningful characteristics for our specific data. Do do so, we train a simple CNN classifier on the MNIST dataset and then use its penultimate layer output for images projections.

To assess the performance of our hand-written digit generator, we use two different precision and recall adaptations designed in the late 10's for the evaluation of GAN models. The first one [4] allows us to obtain a set of "attainable" pairs of precision and recall of a distribution $Q$ w.r.t. a distribution $P$, in a sense well defined in the paper. Having this, we can confront model behaviors by comparing frontiers of such sets. Second, [2] proposes to check how distributions overlap each other in the feature space, using a k-NN coverage algorithm. It outputs a single pair $(\text{precision}, \text{recall})$ that can be used for in-training evaluation for example.

Finally, we evaluate the Fréchet Inception Distance in the same feature space as the other metrics. Indeed, Inception model is too general to capture the variations of MNIST data in its own feature space, so we use this simpler, custom one.

# 3 Improving Precision and Recall with f-GANs

Generative Adversarial Networks (GANs) can be generalized using *f-divergences*, resulting in the f-GAN framework [3]. In this approach, the choice of divergence determines the objective function for both the discriminator $D$ and generator $G$, and different divergences can emphasize different aspects of the generated distribution, such as precision or recall.

## 3.1 f-GAN as a Minimax Problem

Training an f-GAN can be formulated as a *minimax* optimization problem:

$$\min_G \max_D \mathcal{L}_f(D,G) = \mathbb{E}_{x\sim p_{data}}[D(x)] - \mathbb{E}_{z\sim p_z}[f^*(D(G(z)))], \tag{1}$$

where $f^*$ is the Fenchel conjugate of the chosen divergence function $f$, and $z$ is sampled from a latent distribution $p_z$ (typically Gaussian or uniform).

Each divergence defines a specific form of $\mathcal{L}_f(D,G)$, influencing the trade-off between precision and recall, as well as the stability of training:

- **Jensen–Shannon (JS) Divergence:** The classical GAN objective corresponds to the JS divergence, with the min-max formulation

  $$\mathcal{L}_{JS}(D,G) = \mathbb{E}_{x\sim p_{data}}[\log D(x)] + \mathbb{E}_{z\sim p_z}[\log(1 - D(G(z)))]. \tag{2}$$

  JS tends to balance precision and recall, providing stable training while encouraging realistic samples that cover the data distribution.

- **Kullback–Leibler (KL) Divergence:** The forward KL objective is

$$\mathcal{L}_{KL}(D, G) = \mathbb{E}_{x \sim p_{data}}[D(x)] - \mathbb{E}_{z \sim p_z}[\exp(D(G(z)) - 1)]. \tag{3}$$

  KL penalizes missing modes, promoting broader coverage of the data. However, the exponential term can cause large gradients and instability, especially when $D(G(z))$ is far from 1.

- **Reverse KL (RKL) Divergence:** The RKL objective is

$$\mathcal{L}_{RKL}(D, G) = \mathbb{E}_{x \sim p_{data}}[-\exp(D(x))] - \mathbb{E}_{z \sim p_z}[-1 - D(G(z))]. \tag{4}$$

  RKL focuses on high-probability regions of the real data, favoring precision over recall. This can improve sample quality but may lead to mode collapse, where the generator produces only a few modes. Extreme values of $D$ can also destabilize training due to large gradients.

## 3.2 Stability Considerations

During training, we observed that KL and RKL objectives are more prone to instability due to the exponential terms in the generator loss. Several strategies were particularly effective in improving stability:

- **Adaptive learning rates:** The choice of learning rate for the discriminator and generator can significantly affect stability. In our experiments, the KL divergence required a smaller learning rate for the discriminator to prevent it from quickly overpowering the generator. In addition, tuning the momentum parameters of the Adam optimizer helped smooth the updates and reduce oscillations during training, improving stability over time.
- **Multiple generator updates per discriminator step:** Performing two generator updates for each discriminator step helped prevent the discriminator from dominating and ensured meaningful gradients for $G$.
- **Clamping discriminator outputs:** Limiting the range of $D$'s outputs reduced extreme gradients, particularly for KL and RKL, preventing numerical explosions.

## 3.3 Models Training and Evaluation

We evaluated the f-GANs trained with KL, JS, and RKL divergences as well as the BCE vanilla, in terms of sample quality and the trade-off between precision and recall. Training evolution is presented in Figure 1 and the evaluation elements in Figure 2.

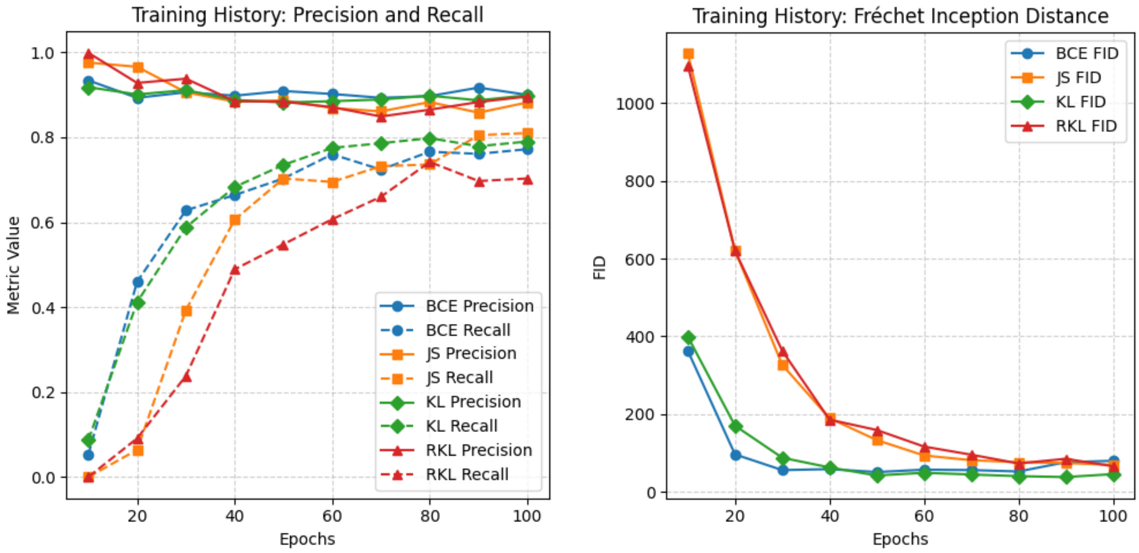
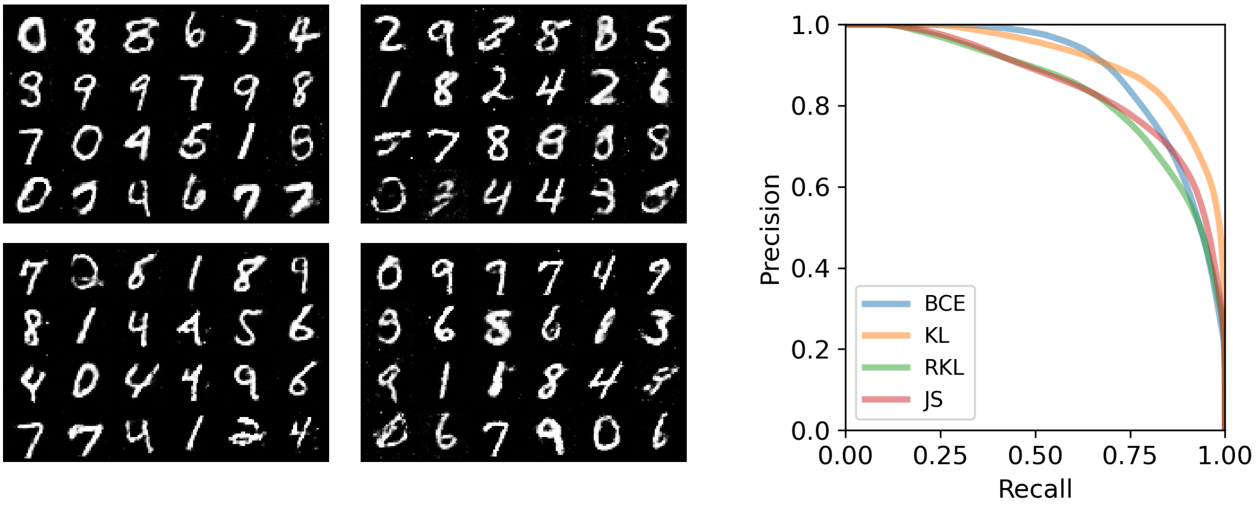


Figure 1: Precision-recall and FID curves over training for KL, JS, and RKL f-GANs, against validation data.

Figure 1 shows that the training process is stopped after convergence of our metrics and before any kind of overfitting.

From the model evaluation Figure 2, we can argue that that the baseline BCE produces the most realistic digits, but lacks diversity. According to the PR curve, KL achieves both the best precision and best recall



(a)

(b)

Figure 2: (a) Sample grids. From top left to bottom right: BCE (Vanilla), and JS, KL, RKL (f-GANs). (b) PR curves for different GAN training losses.

among the three f-GANs. However, looking at the digits generated, it is not that clear at glance. Those results are unexpected, as the theory states that KL should induce more diversity, RKL more precision, and JS find a balance between the two. More tuning on hyperparameters such as the learning rates of the discriminator and the generator could maybe change these observation.

# 4 Tuning Precision and Recall with Soft Truncation

After the training, we still have the possibility to influence on quality and diversity of generated images. We study here the impact of the soft truncation. This tuning method simply consist of sampling GAN embeddings in the latent space with different variances than one. As expected, Figure 3 highlights the fact that soft truncation is a simple yet very useful way to tune out GAN behavior after training : a higher variance helps more diversity, while a lower one helps quality. Nevertheless, we can see that it is never improving precision and recall simultaneously, this can only be done with "improvement" methods that were presented first in this report.

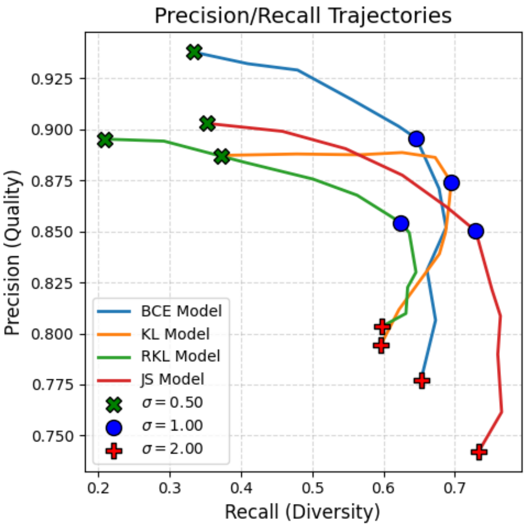
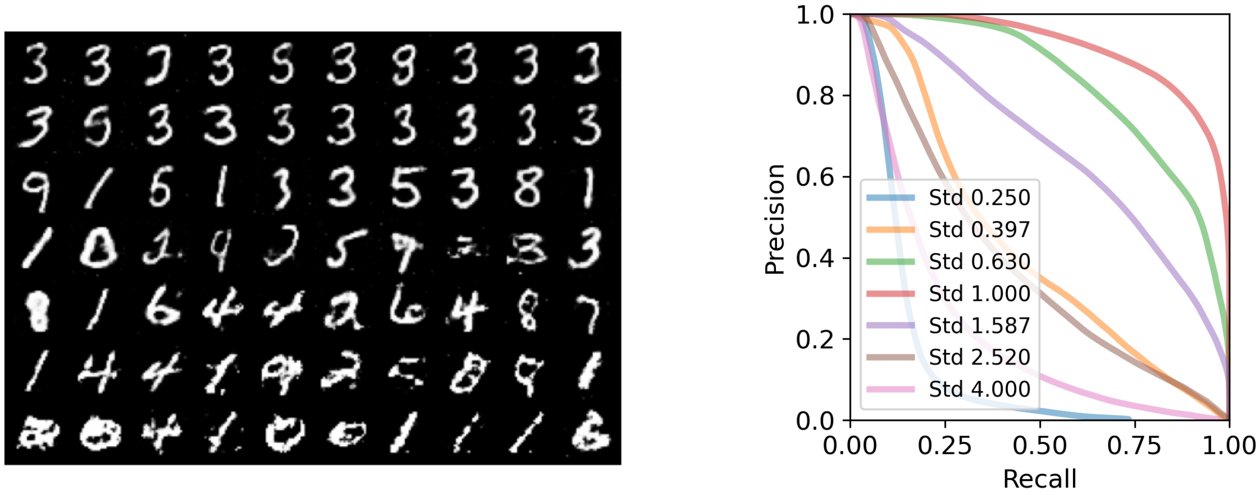


Figure 3: Trajectories of (recall, precision) pairs for different models and soft truncations $\sigma$. Truncation values vary from 0.5 to 2.0. For each model and truncation value, we compared distributions of $10^4$ real and generated images, in the feature space given by the penultimate layer of a previously trained classifier. We used the k-NN coverage algorithm presented in [2].

We highlight Figure 4 the limits of PR curves designed in [4], as already emphasized by the authors of [2]. For the same KL-trained GAN, we would expect the precision to decrease and the recall to increase as the soft truncation variance $\sigma$ increases. But the PR curves are not really acknowledging this expectation, underestimating the performances of different truncation values than one.



(a)

(b)

Figure 4: Limit of PR curves relevance for soft truncations comparison, with KL-trained GAN generated digits. (a) Truncation $\sigma$ varies from 0.25 (top) to 4.0 (bottom). (b) The PR curves for different soft truncations.

## 5 Improving Precision and Recall with Rejection Sampling

Even with careful training and divergence tuning, GANs often produce samples of mixed quality, some are sharp and realistic, while others remain blurry or structurally incorrect. Yet, the discriminator implicitly contains useful information about sample realism, as it assigns higher scores to images that better match the training distribution. We investigate *Discriminator Rejection Sampling* (DRS), introduced in [1], as a post-hoc method to improve the quality of samples generated by our f-GANs.

### 5.1 Principle of DRS

The method relies on the observation that for an optimal discriminator $D^*$,

$$\frac{p_{\text{data}}(x)}{p_G(x)} \propto \frac{D^*(x)}{1 - D^*(x)}.$$

This ratio acts as an importance weight for each candidate sample. DRS approximates these weights using the learned discriminator and performs a rejection sampling procedure:

1. Draw a latent vector $z \sim \mathcal{N}(0, I)$ and compute $x = G(z)$.
2. Evaluate the discriminator score $D(x)$ and convert it into an approximate logit.
3. Accept the sample with probability $p_{\text{accept}}(x) = \sigma\big(\ell(x) - \gamma\big)$, where $\ell(x)$ is the estimated log-likelihood ratio and $\gamma$ is a threshold chosen from a burn-in phase.

### 5.2 Burn-in and Gamma Estimation

The threshold $\gamma$ controls the strictness of the rejection rule. Following [1], we estimate it by generating a large burn-in set (e.g., 20,000 samples), computing the logits $\ell(x) = \log \hat{p}(x) - \log(1 - \hat{p}(x))$, and then setting $\gamma = \max_x \ell(x) + \text{offset}$, where the offset tunes the aggressiveness of DRS.

To assess the effect of $\gamma$, we performed a sweep over $\gamma_{offset} \in \{-3, -2, -1, 0, 1, 2, 3\}$ for KL generator, using 5,000 samples for each value and we observed that KL-trained models react strongly to $\gamma$. Negative values make the sampler highly selective (acceptance as low as 6%), while precision and recall remain stable. This selective regime yields the *best FID across all experiments*, namely 20.95 at $\gamma = -2$. The KL discriminator is steep and confident, enabling DRS to isolate a small set of high-fidelity samples without collapsing recall.

### 5.3 Results

We evaluate DRS on all trained generators (BCE, JS, KL, RKL), generating 10k samples per configuration. Table 1 reports precision, recall, FID, and acceptance for $\gamma = 2$.

| Loss | Method | FID ↓ | Precision ↑ | Recall ↑ | Acceptance (%) ↑ |
|---|---|---|---|---|---|
| JS | DRS | 50.84 | 0.9259 | 0.8315 | 85.98 |
| JS | + Soft trunc. | 62.56 | 0.9423 | 0.8031 | 85.93 |
| BCE | DRS | 61.71 | 0.9545 | 0.7687 | 87.52 |
| BCE | + Soft trunc. | 103.27 | 0.9569 | 0.7720 | 87.48 |
| KL | DRS | **24.02** | 0.9439 | 0.8176 | 89.63 |
| RKL | DRS | 58.33 | 0.9372 | 0.7398 | 88.78 |

Table 1: Evaluation of different f-GAN losses with Discriminator Rejection Sampling (DRS) and the soft truncation variant.

Across divergences, DRS maintains high acceptance rates (typically 86–90%), ensuring that the effective sample size remains large. The KL-trained generator combined with DRS achieves the best FID ($\approx 24$) while preserving a strong balance between precision and recall. RKL favours high-density regions and therefore reaches high precision but the lowest recall, consistent with its mode-seeking behaviour. BCE and JS lie in between: both exhibit solid precision but higher FID and slightly lower recall than KL.

#### 5.3.1 Effect of Soft Truncation

We also evaluate the soft truncation variant, which rescales the latent vector according to the discriminator score before generating an image. For JS, this slightly increases precision (from 0.93 to 0.94) but reduces recall (from 0.83 to 0.80) and worsens FID (from $\approx 50.8$ to $\approx 62.6$). BCE shows a similar pattern: precision increases marginally while FID deteriorates substantially (from $\approx 61.7$ to more than 100). The acceptance rate remains nearly unchanged, suggesting that soft truncation biases the latent space rather than acting as an additional filtering step.

#### 5.3.2 Takeaways

These results suggest that plain DRS already provides a favourable trade-off between sample quality and diversity, especially for the KL-based model. Soft truncation does not consistently improve performance and often worsens FID and recall. Overall, post-hoc rejection sampling offers a simple and robust way to sharpen samples without retraining, whereas more aggressive latent-space transformations require fine tuning to avoid harming distributional metrics.

## 6 Conclusion

In this project, we explored several ways to analyse and improve the precision–recall trade-off of GANs trained on MNIST. By comparing different f-divergence objectives, we showed that each loss leads to distinct behaviours, both in terms of diversity and sample quality. Soft truncation proved to be an effective post-training tool for navigating this trade-off, although it never improved precision and recall simultaneously.

Finally, Discriminator Rejection Sampling (DRS) offered a simple and robust way to enhance sample quality without retraining the model. Among all settings, the KL-based generator combined with DRS achieved the best overall results, confirming that post-hoc filtering can significantly refine the output of a trained GAN.

Overall, our experiments highlight the value of studying both training objectives and post-processing strategies to better control generative behaviour, and they illustrate how different improvement methods can be combined to target specific precision–recall goals.

# References

[1] Samaneh Azadi et al. “Discriminator Rejection Sampling”. In: *International Conference on Learning Representations* (2019).

[2] Tuomas Kynkäänniemi et al. “Improved precision and recall metric for assessing generative models”. In: *Advances in neural information processing systems* 32 (2019).

[3] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. “f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization”. In: *arXiv preprint arXiv:1606.00709* (2016). arXiv: `1606.00709 [cs, stat]`.

[4] Mehdi SM Sajjadi et al. “Assessing generative models via precision and recall”. In: *Advances in neural information processing systems* 31 (2018).