We sincerely thank Reviewers and Area Chairs for valuable feedback.

Reviewer 1 • **1. The optimization function for generator:** We really appreciate Reviewer's suggestion. We have checked Reviewer's derivation carefully. We humbly point out that there is a small sign mistake in the derivation: $\lambda_g \Phi^+(G,C^*)$ was used in Eq. 9, but the correct one is $-\lambda_g \Phi^+(G,C^*)$. We have derived generator optimization. Using $\sum_{k=1}^{K} \mathrm{KL}(P_g^{T_k}||P_d^{T_k}) = K \cdot \mathrm{KL}(P_g||P_d)$, and let $\mathcal{V}_\Phi(\mathrm{x}) = \sum_{k=1}^{K} p^{T_k}(\mathrm{x}) \log(\frac{p^{T_k}(\mathrm{x})}{\sum_{k=1}^{K} p^{T_k}(\mathrm{x})})$, we substitute Eq. 11 into Eq. 9:

$$\min_G \mathcal{V}(D^*, C^*, G) = \mathcal{V}(D^*, G) - \lambda_g \Bigg(- K \cdot \mathrm{KL}(P_g||P_d) + \mathbb{E}_{\mathrm{x} \sim P_g^T} \mathcal{V}_\Phi(\mathrm{x}) \Bigg)$$
$$= \int_x p_d(\mathrm{x}) \log \frac{p_d(\mathrm{x})}{p_d(\mathrm{x}) + p_g(\mathrm{x})} d\mathrm{x} + \int_x p_g(\mathrm{x}) \log \frac{p_g(\mathrm{x})}{p_d(\mathrm{x}) + p_g(\mathrm{x})} d\mathrm{x} + K\lambda_g \int_x p_g(\mathrm{x}) \log \frac{p_g(\mathrm{x})}{p_d(\mathrm{x})} d\mathrm{x} - \lambda_g \mathbb{E}_{\mathrm{x} \sim P_g^T} \mathcal{V}_\Phi(\mathrm{x})$$

Even we assume $K\lambda_g = 1$, it is not easy to simplify $\mathcal{V}(D^*, C^*, G)$. We will further investigate.

• 2. Reviewer's comment on analogies to conditional GAN is thought-provoking. Previous work [4] and ours have regarded self-supervised (SS) GAN as unconditional GAN. But Reviewer is correct that SS GAN can also be regarded as conditional GAN with pseudo-labels. Following Reviewer's comment, we will discuss more analogies in the paper. In particular, AC-GAN [30] with pseudo-labels is quite similar to SS-GAN proposed in [4], but deeper analysis is needed to understand the impact of replacing class labels in the original AC-GAN with pseudo-labels.

• 3. We will clarify minimax optimization and improve presentation (theorems, typos). We will definitely share code.

Reviewer 2 • **1. Self-supervised (SS) signals:** We thank Reviewer for feedback. Potentially, Contrastive Predictive Coding [*Aaron van den Oord et al. 2018*] can be SS signals for time-series.

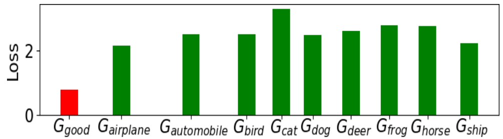

Figure A: Our proposed SS loss $-\Phi^+(G, C)$. Red: Good generator; Green: Mode-collapsed generators that correspond to different classes of CIFAR-10, as explained in Sec.4.

• 2. We follow Reviewer's suggestion to use the same toy example to show improvement of our proposed method. The setup is the same as in our paper Sec.4 L167-180, except that now we replace models/cost functions of [4] with our proposed ones, i.e., change from Fig.1a to Fig.1b. Therefore, now the loss is $-\Phi^+(G, C)$, which is shown in Fig.A for a good generator G_{good} and different mode-collapsed generators $G_{collapsed}$ (designs of G_{good} and $G_{collapsed}$ are the same as in our paper Sec.4). Comparing Fig.A and Fig.3a in our paper, the improvement using our proposed model can be observed: G_{good} has the lowest loss under our proposed model.

Datasets	**SS**	**MS**	**MS-v2**
CIFAR-10 (10K-10K FID)	12.37	11.40	11.15
CIFAR-100 (10K-5K FID)	49.40	21.39	-
ImageNet 32×32 (10K-10K FID)	26.04 (best)	13.70	-
Stacked MNIST (#mode)	878.5 ± 38.9	943.2 ± 31.4	-
Stacked MNIST (KL)	0.99 ± 0.19	0.70 ± 0.15	-

Table A: Additional results. Baseline: Dist-GAN with ResNet, as in our paper. We also follow exactly the same experiment setup. **SS**: proposed in [4]; **MS**: this work. Note that for **mode coverage** experiment (row 4), our method **MS** achieves the best results for this dataset with tiny $K/4$ architecture, **outperforming state-of-the-art by a significant margin:** [36]: #mode = 859.5 ± 68.7, KL = 1.04 ± 0.29; [R1]: #mode = 859.5 ± 36.2, KL = 1.05 ± 0.09. We will update the paper with these results.

• **3. A separate discriminator to distinguish real/fake rotated images:** We train this new model **MS-v2**. Our preliminary result shows that this is very competitive (Table A, row 1). We thank Reviewer for this good suggestion. We will further analyze. We note that this new model further corroborates our idea to leverage discrimination of *rotated* real/fake images to improve *generator* learning, as we have provided theoretical and empirical evidence in our paper.

• 4. We show results of CIFAR-100 in Table A, row 2. We follow the setup in [R2] exactly, i.e. 10K-5K FID. Note that [R2] has the state-of-the-art results for CIFAR-100. As shown in Table A, row 2, **SS** suffers mode collapse (high FID), probably due to some classes like "sunflowers", "fruit and vegetables", which original SS task does not encourage a generator to learn to produce them (*rotated sunflowers are not rare*). Our **MS** achieves good FID on this dataset and outperforms [R2] (best 10K-5K FID = 23.6) on same dataset. Imagenet (128×128 or higher resolution) experiments require very extensive computation. So far, most of Imagenet results are from big companies/institutions, eg., Google [2, 4, 41]. We will look for computing resource for this experiment. We conduct the preliminary experiments on lower resolution Imagenet (32×32) as shown in Table A, row 3. Our **MS** substantially outperforms **SS**. Note that training using our **MS** is also stable: **SS** suffers from mode collapse (FID = 56.20) at the end of training and our **MS** attains the best FID at the end of training.

Reviewer 3 • 1. We will further clarify Fig.3. Note some explanation is in L.167-180. • 2. These methods are compared under **exactly the same settings** of [26, 4]. Some results are missing because previous methods did not report for these experiments. • **3. No. of modes covered**: We thank Reviewer's suggestion. We follow exactly the same experiment setup, tiny architecture $K/4$ and evaluation protocol of [25]. Table A, row 4 shows that our performance for mode covered and KL divergence [25] is superior, significantly outperforms state-of-the-art [36, R1].

[R1] Karras et al., Progressive Growing of GANs for Improved Quality, Stability, and Variation, ICLR 2018.
[R2] Yamaguchi et al., Distributional Concavity Regularization for GANs, ICLR 2019.