Response to Reviewer #1

[Support for the claims that most divergences are global and not ideal for mode coverage]: We do provide support for our claims, both theoretically and empirically. In Appendix B, we analyze several classic generative models and show that they all reduce certain statistical distances (such as KL divergence and JS divergence) measured *globally* over the data space. In Fig. 1, we give an intuitive example illustrating why global statistical distances cannot guarantee mode coverage. This claim is further supported by our experiments in Fig. 2. We also note that while some existing generative models can avoid missing modes well in certain situations, it is unclear whether they guarantee a complete mode coverage—a condition that we explicitly define in Eq. (1) and guarantee in our approach.

[Speed and scalability]: As discussed in Section 3.3 and Appendix F.4, the number of generators needed is at most $O(\log n)$. This theoretical bound implies that the running time will be scaled by a factor of at most $O(\log n)$. In practice, this factor is small. In our experiments on synthetic data, AdaGAN takes 437 min (25 iterations) and still miss some modes, while our method takes only 134 min (9 iterations) to cover all modes. In the Stacked MNIST experiment, our method takes 320 min while AdaGAN takes 550 min. Detailed setup can be found in Appendix G.

[Weight of the minor mode]: As presented in Line 305, in Fig. 2, the ratio of the minor mode weight to the major mode weight is $1/400$. In many real-world applications, such an imbalanced mode may be caused by the difficulty of data collection, not necessarily noise. In fact, the currently active research in machine learning fairness is largely motivated by this kind of data imbalance, indicating that the imbalanced modes do exist in real-world datasets.

[KDE]: We run KDE to estimate the likelihood of our generated samples on our synthetic data experiments (using KDE bandwidth=0.1). We compute $L = 1/N \sum_i P_{model}(x_i)$, where x_i is a sample in the minor mode. For the minor mode, our method has a mean log likelihood of -1.28, while AdaGAN has only -967.64 (almost no samples from AdaGAN).

Response to Reviewer #2.

[Insight of why AdaGAN can not guarantee full mode coverage]: AdaGAN aims to fit the "residual" distribution in each iteration. As a thought experiment, consider a dataset that has a major mode and a minor mode. In one iteration, AdaGAN may cover the major mode but not perfectly fit its distribution. In the next iteration, the weight of the major mode may still be large, and thus prevents AdaGAN from learning the minor mode. In contrast, if the major mode is covered by our algorithm in an iteration, in the next iteration the relative weight of the major mode (w.r.t. the minor mode weight) will become lower. Eventually, our algorithm learns the minor mode distribution.

[Metrics used in AdaGAN]: We tested our algorithm on synthetic data using the metrics introduced by AdaGAN. We would like to emphasize that the metrics used by AdaGAN are all global measures, different from our focus on *local* mode coverage. The first metric in AdaGAN is $C := P_d(dP_{model} > t)$ with a t satisfying $P_{model}(dP_{model} > t) = 0.95$. We use KDE to estimate P_{model}. Over the entire dataset, the score for AdaGAN is 0.931, slightly better than ours 0.872. However, for the scores over the minor mode, we have $C = 1.0$ while AdaGAN has $C = 0.0$ (no samples on it). Another AdaGAN metric is the likelihood of the true data under the generated distribution, $L := 1/N \sum_i P_{model}(x_i)$. AdaGAN has $L = -7.43$, lower than ours $L = -5.03$, because the minor mode is missed in AdaGAN.

Response to Reviewer #3.

[Related work]: We will add a section to discuss the major related work in the main text of the revised paper.

[Compare with AdaGAN on synthetic dataset]: We emphasize our comparison with AdaGAN because it has the most similar setting to ours; both are boosting algorithms. Also, in the AdaGAN paper, the authors have compared AdaGAN to several baseline methods, and shown that AdaGAN outperforms other methods in both synthetic and real datasets. Thus, our rationale is to use AdaGAN as our baseline for comparison. Furthermore, it is unclear for us how to change statistical distance or hack the objective function to improve mode coverage guarantee. Such a change itself might merit further research. When training MGAN over our synthetic dataset, we found that it cannot converge and produces poor mode coverage, even after experimenting with multiple parameter configurations. Thus, we choose not to include MGAN in the synthetic data comparisons.

[Comparison with methods using different priors over the latent space]: It is indeed interesting to compare with methods that use different latent space priors. However, those methods lack the theoretical guarantee of complete mode coverage, while our method can provably cover all the modes under our metric. The use of latent space prior can help improving mode coverage, but to our knowledge, how to construct the latent space prior in general is still an open problem and may require task-specific solutions. We will discuss the differences from those methods when we revise our paper.

[Pointwise coverage without ground truth knowledge]: Indeed, it is often hard to know the mode distribution in a dataset. This is precisely the motivation of introducing Eq. (1) as our definition of complete mode coverage (see Line36-39). If the pointwise coverage Eq. (1) is satisfied, modes are guaranteed to be covered, no mater how the modes are defined and distributed. So we don't need to know the mode distribution in order to guarantee mode coverage. When evaluating our method, we need a classifier (under a certain mode definition) to determine whether a mode is covered and quantify mode coverage. For example, in stacked-MNIST, we use the standard MNIST classifier for each color channel and we combine these three classifiers as a 1000-class classifier to quantify the coverage of 1000 modes.