To Reviewer1: *1. Method simplistic, places too much constraints on activation (only ReLU-like activations).*

We believe the proposed H-regularization is novel and by no means simplistic. It is well suited for one-class learning. ReLU-like activations are widely used, e.g., Transformer, Resnet, etc. It does not affect the application of our method.

2. I would be curious to see the results more complex datasets.

In our experiments, we followed baselines and used the same datasets as them. Per your request, we conducted a new experiment using CIFAR100, which has 100 classes. The average accuracy of iteratively taking each class as the one class to do one-class learning is 68.61 which is much better than top baselines' (52.26 for OCGAN, 60.79 for ICS).

3. I'm not a fan of putting the activation function (sigmoid()) in Eq.3, instead of the probability distribution P(C|x).

Since we have only one class in the output, we believe using sigmoid() to represent the score of x belonging to the one class is reasonable. It is hard to estimate the probability distribution P(C|x) without other (negative) data.

4. page 5 can be summarized into 1 paragraph ... Thanks for the suggestion. We will follow it in revising our paper.

To Reviewer2: *1. ... if other loss functions (e.g., squared loss) were considered, which may not have Problem-I ...*

We experimented with squared loss and found it also faces the saturation problem as the output targets are always 1 during training. It gets quite poor results, 71.35 (on MNIST). A small learning rate 0.01 did not help. If we add our H-regularization and normalization method, the result gets to 97.38, which is quite close to our result (97.59) using the NLL loss. This indicates that our H-regularization and normalization method are not limited to NLL.

2. ... why the proposed method works better ... the core contribution ... is the observation of Problem-2 (feature bias).

The intuition is that our method tries to leverage features holistically to ensure the system is not biased towards certain features. This decreases the probability of abnormal (negative) data passing the system to achieve better results. It is hard to compare the value of the H-regularization term in baselines as they have completely different loss functions which make the H-regularization values incomparable.

3. Is it possible to compare baselines+2N_Inst_Norm and proposed method too? It will make the contributions clearer.

We experimented with replacing the normalization method of the top baselines ICS, TQM and OCGAN with ours but got quite poor results. This is because each baseline already has its most suitable normalization method for its approach.

4. The chosen architecture is quite small (e.g., 784-100-1), but does this make it disadvantageous ... ?- A related question is, is this expressive enough to cause the Problem-1 in page 3 to occur?

It can be disadvantageous to us. It may also mean that our method still has room for improvement. We did try a more complex CNN architecture, but the improvement is small. We will investigate more in our future work. Simple architecture also causes Problem-I as we detected saturated outputs too (hope we understood your question correctly).

5. Is grid search performed for 21 $\times$ 20 = 420 combinations? Although it is fixed for all datasets, ...

It is $21 + 20 = 41$ (we try one, fix it and then next). We should note that the hyper-parameter search is only done on MNIST. The resulting parameters are used for all the other datasets, i.e., no parameter tuning needed for each dataset.

6. A minor suggestion is to show how the performance is sensitive to n. When $10 \leq n \leq 16$, all the results are good.

7. which baselines were using the same results previously reported in original papers.

The results of OCGAN on the FMNIST data and TQM on the CIFAR10 data are obtained by running the author released code and the rest are copied from published papers. We will detail this in Appendix.

To Reviewer3: Thanks for your constructive suggestions. We will improve accordingly and cite the papers you listed.

1. iForest is an important baseline and should be included into the empirical comparison.

We run sklearn's API. iForest gets 94.74% and 94.44% (F1 score) on KDDCUP and Thyroid respectively. And the AUCs of iForest on MNIST/FMNIST/CIFAR10 are 84.43/90.51/59.70. Our method ourperforms all of them.

2. It is interesting that HRN using a MLP can perform better than ... the authors could discuss why this would happen.

We believe one of the key issues in one-class learning is how to avoid biasing some features as there is no negative data. We did not see that existing approaches explicitly deal with this problem. In our case, we identify and explicitly address the model bias problem using H-regularization, which we believe is the main reason.

3. HRN on the 2 tabular datasets is very good. ... more convincing to add more: We will add more datasets. We ran another dataset **Arrhythmia** and obtained (F1): 45.8 (OCSVM), 49.8 (DAGMM), 53.0 (TQM), and 84.5 (**HRN**).