ODIM: Outlier Detection via Likelihood of Under-Fitted Generative Models

Jul 21-27, 2024

Vienna, Austria

ICML International Conference On Machine Learning

Dongha Kim[1*], Jaesung Hwang[2*], Jongjin Lee[3], Kunwoong Kim[4], and Yongdai Kim[4]

[1]Department of Statistics and Data Science Center, Sungshin Women's University
[2]SK Telecom
[3]Samsung Research
[4]Department of Statistics, Seoul National University
* Equal contribution

Contributions

1. **A novel observation of DGM (Deep Generative Model): IM effect**
   - ✓ DGM memorizes inliers prior to outliers at early training updates.
2. **A new UOD solver: ODIM**
   - ✓ Based on IM effect of under-fitted DGM
   - ✓ Data-agnostic, simple and powerful

Introduction

Categories of OD (Outlier Detection) task

- SOD: inlier/outlier-annotated training data.
- SSOD: inlier-only training data.
- UOD: no prior information of training data.

Limitation of likelihood-based DGMs in OD task

- Generally believed that likelihood-based DGMs are not appropriate for OD tasks.

Our claim

- **Likelihood-based DGMs** can be a **powerful OD solver** when using (carefully) **under-fitted** models.

Motivation: IM effect of under-fitted DGM

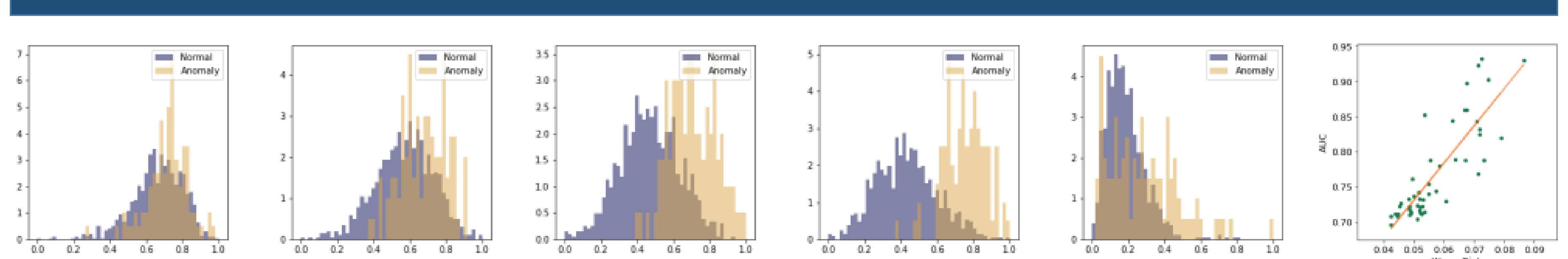

Figure 2. (**1st to 5th**) The distributions of the per-sample (normalized) VAE loss values of `Cardio` after 10, 20, 30, 40, and 500 training updates, respectively. For each panel, we depict the histograms of inliers and outliers separately. (**Last**) The positive relationship between the Wasserstein distance and identifying performance (AUC) on `Cardio`.

Inliers' loss < Outliers' loss (in early updates)

Proposed method: ODIM

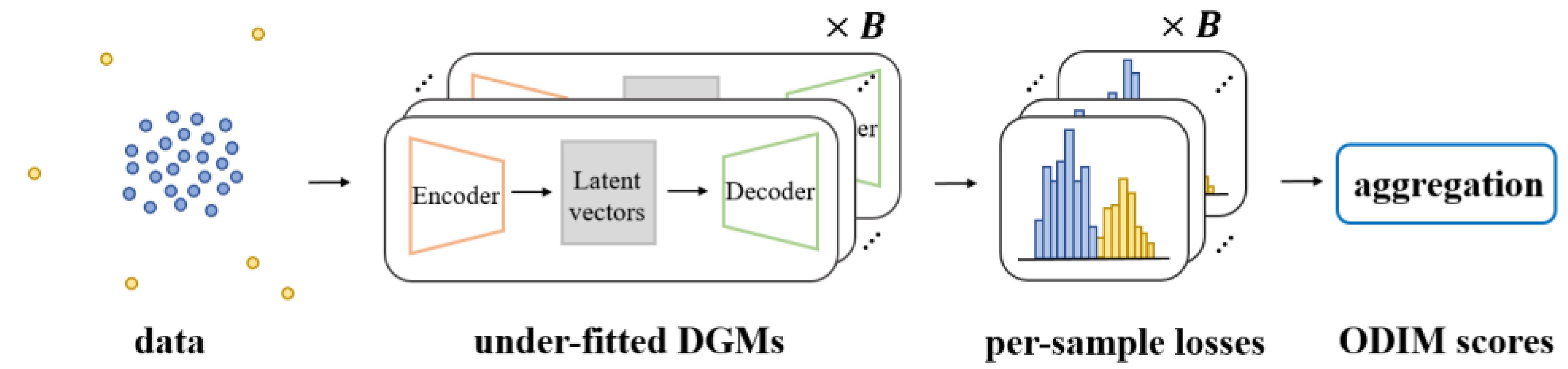

Figure 1. An illustration of the ODIM method.

Training an under-fitted DGM

- Min-max pre-processing
- Train for several updates
- The optimal update is chosen by the *loss distribution's bi-modality*

Use the ensemble to obtain improved and stable results

- ODIM score is computed using multiple under-fitted DGMs' loss values

The final ODIM score

$$l_i^* \leftarrow \frac{1}{B}\sum_{b=1}^{B} L^{\text{IWAE}}(\theta^{*(b)}, \phi^{*(b)}; \mathbf{x}_i), i = 1, \ldots, n$$

Algorithm 1 ODIM

In practice, we set $(K, N_{\text{u}}, N_{\text{pat}}) = (50, 10, 10)$.

Input: Training data set $\mathcal{U}^{tr} = \{\mathbf{x}_1, ..., \mathbf{x}_n\}$

Require: : Decoder and encoder: $p(\mathbf{x}|\mathbf{z};\theta)$ and $q(\mathbf{z}|\mathbf{x};\phi)$, GMM-2 model: $\pi_1\mathcal{N}(\mu_1, \sigma_1^2) + \pi_2\mathcal{N}(\mu_2, \sigma_2^2)$, Mini-batch size: n_{mb}, Optimizer: $\mathcal{O}$, Number of samples in IWAE: K, Update unit number: N_{u}, Maximum patience: N_{pat}

for b in $(1 : B)$ **do**
  Initialize $(\theta^{(b)}, \phi^{(b)})$ and set d_{WD}^{max} to 0.
  while $n_{\text{pat}} < N_{\text{pat}}$ **do**
    for k in $(1 : N_{\text{u}})$ **do**
      Drawn n_{mb} samples, $\{\mathbf{x}_i\}_{i=1}^{n_{\text{mb}}}$, from $\mathcal{U}^{tr}$.
      Apply the min-max scaling to $\{\mathbf{x}_i\}_{i=1}^{n_{\text{mb}}}$.
      Update $(\theta^{(b)}, \phi^{(b)})$ using the IWAE with $\{\mathbf{x}_i\}_{i=1}^{n_{\text{mb}}}$ and $\mathcal{O}$.
      $\{\tilde{l}_i\}_{i=1}^{n_{\text{mb}}} \leftarrow$ normalize$(\{L^{\text{IWAE}}(\theta^{(b)}, \phi^{(b)}; \mathbf{x}_i)\}_{i=1}^{n_{\text{mb}}}$
      Fit the parameters in GMM-2 using $\{\tilde{l}_i\}_{i=1}^{n_{\text{mb}}}$ and calculate the WD distance d_{WD}.
      if $d_{\text{WD}} > D_{\text{WD}}^{\max}$ **then**
        $d_{\text{WD}}^{\max} \leftarrow d_{\text{WD}}$
        $\theta^{*(b)}, \phi^{*(b)} \leftarrow \theta^{(b)}, \phi^{(b)}$
        $n_{\text{pat}} \leftarrow 0$
      else
        $n_{\text{pat}} \leftarrow n_{\text{pat}} + 1$
      end if
    end for
  end while
end for
Calculate ODIM scores:

$$l_i^* \leftarrow \frac{1}{B}\sum_{b=1}^{B} L^{\text{IWAE}}(\theta^{*(b)}, \phi^{*(b)}; \mathbf{x}_i), i = 1, \ldots, n$$

Output: ODIM scores $\{l_i^*\}_{i=1}^{n}$

Theory

Justification of IM effect

$$L^{\text{VAE}}(\theta, \phi; \mathbf{x}) := \mathop{\mathbb{E}}_{\mathbf{z}\sim q(\mathbf{z}|\mathbf{x};\phi)}\left[\log\left(\frac{p(\mathbf{x}|\mathbf{z};\theta)p(\mathbf{z})}{q(\mathbf{z}|\mathbf{x};\phi)}\right)\right]$$

Proposition 3.1. [1] *For an input vector* $\mathbf{x}$, *the following holds:*

$$\mathbb{E}_{\theta,\phi}\left\|\frac{\partial}{\partial\theta}L^{\text{VAE}}(\theta, \phi; \mathbf{x})\right\|_2^2 = \Theta\left(\|\mathbf{x}\|_1^4\right).$$

Min-max > Standardization

Proposition 3.4. *Let* $\mathbf{X}^{in}$ *and* $\mathbf{X}^{out}$ *be inlier and outlier random vectors with zero mean, i.e.,* $\mathbb{E}(X^{in}) = \mathbb{E}(X^{out}) = 0$. *Suppose that their respective supports are* $Supp(X^{in}) = A^{in}$ *and* $Supp(X^{out}) = A^{out}$, *where* A^{in} *is a bounded convex set and* A^{out} *is a set wrapping* A^{in}, *i.e.,* $A^{in} \cap A^{out} = \emptyset$ *and* $conv(A^{out}) \supsetneq A^{in}$. *Define* $\mathbf{X}_{mm}^{in}$ *and* $\mathbf{X}_{mm}^{out}$ *as pre-processed inlier and outlier random vectors using the min-max scaling. Similarly, we define* $\mathbf{X}_{st}^{in}$ *and* $\mathbf{X}_{st}^{out}$ *obtained by the standardization. Then, we have* $\mathbb{E}\|\mathbf{X}_{mm}^{in}\|_1 = \mathbb{E}\|\mathbf{X}_{mm}^{out}\|_1$, *while* $\mathbb{E}\|\mathbf{X}_{st}^{in}\|_1 < \mathbb{E}\|\mathbf{X}_{st}^{out}\|_1$.

Experiments

UOD performances

Table 1. Averaged AUC and PR scores over 46 tabular datasets.

Method	OCSVM	COPOD	ECOD	DeepSVDD	ICL	DDPM	DTE	ODIM
AUC	0.740	0.730	0.729	0.543	0.652	0.712	0.730	**0.757**
PR	0.360	0.339	0.349	0.182	0.201	0.332	0.321	**0.366**

Table 2. Averaged AUC and PR scores over 6 image datasets.

Method	OCSVM	COPOD	ECOD	DeepSVDD	ICL	DDPM	DTE	ODIM
AUC	0.744	0.508	0.511	0.580	0.655	0.738	0.757	**0.813**
PR	0.271	0.090	0.091	0.176	0.172	0.267	0.282	**0.429**

Table 3. Averaged AUC and PR scores over 5 text datasets.

Method	OCSVM	COPOD	ECOD	DeepSVDD	ICL	DDPM	DTE	ODIM
AUC	0.566	0.554	0.537	0.504	0.546	0.548	0.598	**0.659**
PR	0.062	0.060	0.057	0.054	0.058	0.059	0.070	**0.097**

Partially labeled case

Table 5. Averaged results of training AUC (and PR) scores with various values of l. We consider l, $l = 0.0, 0.3, 0.5$.

l	0.0	0.3	0.5
AUC (PR)	0.885 (0.647)	0.947 (0.871)	0.958 (0.891)

Differentially private ODIM

Table 6. Averaged results of training AUC (and PR) scores when applying the DP-SGD algorithm. We iterate the DP-SGD until $\epsilon = 10$ while fixing $\delta = 10^{-5}$.

Method	DeepSVDD	ODIM
AUC (PR)	0.614 (0.152)	**0.710 (0.234)**