Combining Statistical Depth and Fermat Distance for Uncertainty Quantification

Hai-Vy Nguyen[1,2,3], **Fabrice Gamboa**[2], **Reda Chhaibi**[2], **Sixin Zhang**[3], **Serge Gratton**[3], **Thierry Giaccone**[1]

[1] Ampere Software Technology
[2] Institut de mathématiques de Toulouse
[3] Institut de Recherche en Informatique de Toulouse

{thierry.giaccone, hai-vy.nguyen}@ampere.cars,
{reda.chhaibi, fabrice.gamboa}@math.univ-toulouse.fr,
{serge.gratton, sixin.zhang}@irit.fr

Abstract

We measure the out-of-domain uncertainty in the prediction of Neural Networks using a statistical notion called "Lens Depth" (LD) combined with Fermat Distance, which is able to capture precisely the "depth" of a point with respect to a distribution in feature space, without any distributional assumption. Our method also has no trainable parameter. The method is applied directly in the feature space at test time and does *not* intervene in training process. As such, it does *not* impact the performance of the original model. The proposed method gives excellent qualitative results on toy datasets and can give competitive or better uncertainty estimation on standard deep learning datasets compared to strong baseline methods.

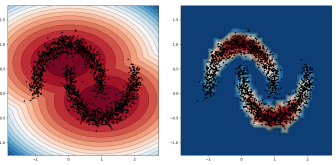
1 Introduction

We consider a multi-class classification problem with the input space $\mathcal{X}$. In general, a classification model consists of a feature extractor (backbone) Φ_{θ_1} and a classifier h_{θ_2}: $f_\theta = h_{\theta_2} \circ \Phi_{\theta_1}$, where $\theta = (\theta_1, \theta_2)$ is the set of parameters of the model. The backbone transforms inputs into fixed-dimension vectors in the so-called *feature space* $\mathcal{F}$. The classifier h then maps the features to predictions. The model f_θ is trained on i.i.d. examples drawn from *In-Distribution* (ID) P_{in}. $f_{\hat{\theta}}$ denotes the trained model.

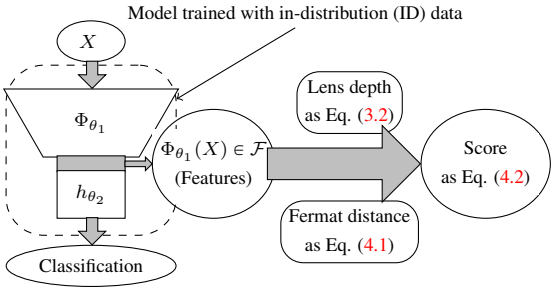
OOD detection. Classification by neural networks has proved highly effective in terms of precision. However, beside performance, in critical applications, one needs to detect out-of-distribution (OOD) data for safety reasons. Indeed, at the inference stage, the model should only predict for data coming from the ID and reject OOD samples. For this purpose, one needs to associate a confidence (or uncertainty) score S with these data so that one can reject uncertain predictions. This is referred as *Out-of-domain uncertainty* [5]. At the inference stage, x is considered as ID if $S(x) \geqslant \varepsilon$ (with some threshold $\varepsilon \in \mathbb{R}$) and OOD otherwise. We develop a method applicable directly in the feature space $\mathcal{F}$ of a trained model $f_{\hat{\theta}}$. It yields a score function $S_{\mathcal{F}}$: $S(x) := S_{\mathcal{F}}(\Phi_{\hat{\theta}_1}(x))$. The high-level idea is to measure directly "how central" a point is with respect to (w.r.t.) clusters taking into account density and geometry of each cluster in the feature space. This provides an uncertainty score. For this objective, standard methods consist in assuming some *prior* distribution such as GDA (Gaussian Discriminant Analysis) based on Gaussian fitting [13]. However, the assumption that the data in a cluster is Gaussian distributed or follow any particular distribution is quite restrictive. We will show in our experiments section that the Gaussian assumption fails even in a very simple case (Section

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

https://doi.org/10.52202/079017-0951



(a) Motivation example of two-moons. GDA (left) fails completely to capture the distribution of dataset whereas our proposed method (right) represents very well how central a point is w.r.t. clusters without *prior assumption*.



(b) General scheme of our method. Given a set of features Φ, the Fermat distance is a metric which respects and adapts to the distribution of Φ_{θ_1}. Lens depth wraps the Fermat distance into a probabilistic and interpretable score S. No additional supervised training is needed.

Figure 1.1: Motivation example and general scheme of our method.

5.1). Let us take the example of a simple frame in the plane with 2 clusters corresponding to 2 classes in form of two-moons (Fig. 1.1a). In this example, GDA fails totally to capture the distribution of clusters. This motivates us to develop a non-parametric method that can measure explicitly how "central" a point is w.r.t. a cluster without the need of additional training and prior assumption. Furthermore, the method should accurately capture distribution with complex support and shape, in order to be adapted to a variety of cases. To measure how central a point is w.r.t. a distribution, we use the so-called notion of statistical *Lens Depth* (*LD*) [15], that will be presented in Section 3.1. Furthermore, for *LD* to correctly capture the shape of the distribution, an appropriate distance must be adopted that adaptively takes into account its geometry and density. Fermat distance is a good candidate for this purpose. However, it is not directly tractable as it stands on integrals along rectifiable paths. A recent paper [6] proposes the use of an explicit sampled Fermat distance and shows its consistency property (see also [2]). In our work, we make use of their results to compute the *LD*. The general scheme is illustrated in Figure 1.1b. In our experiments, the classification model is provided by a neural network, h is a softmax layer consisting of a linear transformation and a *softmax* function, $\mathcal{F}$ is the output space of the penultimate layer right before the softmax layer.

Consistency of the uncertainty score. A consistent uncertainty score function should allow us to detect OOD. Furthermore, when more samples are rejected based on this score, the accuracy of the multi-class classification on the retained samples should increase. In other words, the fewer examples retained (based on the score), the better the accuracy. Our method measures a natural "depth" of the considered example. Consequently, the larger the depth of this example, the more typical this point is (relative to the training set), and so the easier it is for the model to classify.

In summary, our contribution is at the following three levels:

- We are bringing to machine learning the mathematical tool of LD, combined with Fermat distance. It proves particularly efficient for OOD uncertainty quantification. We also propose improvements that avoid undesirable artifacts, and simple strategies for reducing significantly the complexity in computing LD.
- The method we propose is non-parametric and non-intrusive. We do not have priors on the distribution of data nor features. We do not require modifying the training algorithms.
- The method is almost hyperparameter-free, as we show that it is rather insensitive the only parameter used to define Fermat distance.

Tables 5.1 and 5.2 give benchmarks. Our code can be found at LD-experiment-code.

2 Related Work

Intrusive approaches. One approach to construct a confidence score consists in fine-tuning the model $f_{\hat{\theta}}$ using some auxiliary OOD data so that the ID and OOD data are more separable in the feature space [16]. One may even use very particular type of models and training mechanisms for the original classification task such as *Prior Networks* in which the prior distribution is assumed on the output of the model f_θ [19]. More laborious methods to handle uncertainty in neural network is

Bayesian modeling [17, 4]. Another approach is to train additional models such as Deep Ensembles [9] or LL ratio [23]. In these approaches, one needs to carefully perform a supplementary training. Otherwise one could reflect wrongly the true underlying distribution. Moreover, the performance of the multi-class classification task could be impacted. For all these reasons, these methods can be considered *intrusive*.

Non-intrusive approaches. Independently from above methods, a *non-intrusive* approach is to work directly in the feature space $\mathcal{F}$ of the trained model $f_{\hat{\theta}}$. This is *non-intrusive* in the sense that there is no need of changing the model nor supplementary training. Besides, model performance is not impacted. One of the simplest method is to use the k-nearest neighbor distance [24]. It is simple but has the drawback of ignoring the global cluster geometry and density as it considers only the nearest neighbors. A more sophisticated approach is GDA [13] that uses minimum Mahalanobis distance [1] [18] based on *Gaussian prior*. Despite taking the distribution into account, Gaussian modeling is restrictive as it leads to an ellipsoid for shaping each cluster.

Single forward-pass uncertainty. A popular work which yields uncertainty score in a single forward pass is DUQ [27]. In this method, one needs to train particular models, namely RBF models [12], with some carefully fine-tuned penalty to encourage sensitivity. This makes the training process more difficult, hence impacting negatively the classification performance. [14] proposes SNGP using a distance-aware output layer, based on Gaussian Processes, with Spectral Normalization (SN) in training. Again, these additionnal constrains can decrease the overall performance of model (compared to standard softmax model). More recently, [20] proposes DDU, which is based on a GDA approach, but one adds SN to encourage smoothness. [7] proposes Nonparametric Uncertainty Quantification (NUQ) using a kernel-based method. Although this method is non-parametric, it is highly dependent on the choice of kernel and the kernel bandwidth.

3 Background

3.1 Lens Depth

Lens depth (LD) [15] is a specific notion of a more general quantity called Depth [26]. A depth is a score measure of the membership of a point w.r.t. a distribution in a general space. The greater the depth, the more central the point is to the distribution. LD of a point w.r.t a distribution P_X is defined as the probability that it belongs to the intersection of two random balls. These balls are centered at two independent random points X and Y, both having the distribution P_X and a radius equal to the distance between X and Y. More formally, if we work on $\mathbb{R}^d$, the LD of a point $x \in \mathbb{R}^d$ w.r.t. P_X is defined as follows,

$$LD(x, P_X) := \mathbb{P}(x \in B_1 \cap B_2) \,. \tag{3.1}$$

Here, D is a given distance on $\mathbb{R}^d$; X_1,X_2 are i.i.d with law P_X; $B(p, r)$ is the closed ball centered at p with radius r; $B_i = B(X_i, D(X_2, X_1))$, $i = 1, 2$. Let $A(X_1, X_2) = B_1 \cap B_2$. Equation (3.1) naturally gives rise to the following empirical version of LD,

$$\widehat{LD}_n(x) := \binom{n}{2}^{-1} \sum_{1 \leqslant i_1 < i_2 \leqslant n} \mathbb{1}_{A(X_{i_1}, X_{i_2})}(x) \,. \tag{3.2}$$

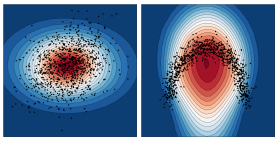
Note that for the empirical version, the intersection set can be rewritten as

$$A(X_1, X_2) = \{x : \max_{i=1,2} D(x, X_i) \leqslant D(X_1, X_2)\} \,. \tag{3.3}$$

Obviously, a crucial question is the choice of the distance D. A naive choice is the Euclidean one. Examples of $\widehat{LD}$ using Euclidean distance are depicted in Fig. 3.1. We see that in the Gaussian case, the level curves of $\widehat{LD}$ rather well capture the distribution. However, for the moon distribution they fail miserably. This is not surprising as the Euclidean distance does not take into account the data distribution P_X. This gives rise to a natural problem as stated by [6]: *How to learn a distance that can capture both the geometry of the manifold and the underlying density?* The Fermat distance allows us to solve this problem and it is presented in the following section.

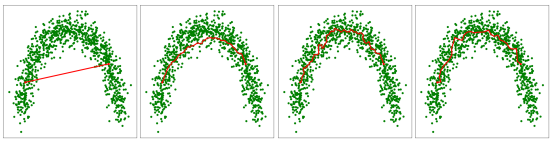
[1] The Mahalanobis distance from a point $x \in \mathbb{R}^\delta$ to a given probability Q (with mean μ and covariance matrix Σ) is defined as $d(x, Q) = \sqrt{(x-\mu)^T \Sigma^{-1} (x-\mu)}$.

https://doi.org/10.52202/079017-0951



(a) Gaussian (b) Moon

Figure 3.1: $\widehat{LD}$ using Euclidean distance. Using Euclidean distance cannot capture correctly the distribution.



(a) $\alpha = 1$ (b) $\alpha = 1.2$ (c) $\alpha = 3$ (d) $\alpha = 7$

Figure 3.2: Sample Fermat path between two randomly chosen points with different values of α.

3.2 Fermat distance

Following [6], let S be a subset of $\mathbb{R}^d$. For a continuous and positive function $f : S \to \mathbb{R}_+$, $\beta \geqslant 0$ and $x, y \in S$, the Fermat distance $\mathcal{D}_{f,\beta}(x, y)$ is defined as

$$\mathcal{D}_{f,\beta}(x, y) := \inf_\gamma \mathcal{T}_{f,\beta}(\gamma) \ , \text{ with } \mathcal{T}_{f,\beta}(\gamma) := \int_\gamma f^{-\beta} \ . \tag{3.4}$$

The infimum is taken over all continuous and rectifiable paths γ contained in $\bar{S}$, the closure of S, that start at x and end at y.

Sample Fermat Distance. Let Q be a non-empty, locally finite, subset of $\mathbb{R}^d$, serving as dataset. $|x|$ denotes Euclidean norm of x, $q_Q(x) \in Q$ is the particle closest to x in Euclidean distance – assuming uniqueness[2]. For $\alpha \geqslant 1$, and $x, y \in \mathbb{R}^d$, the sample Fermat distance is defined as

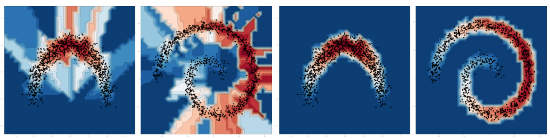
$$D_{Q,\alpha}(x, y) := \min \left\{ \sum_{j=1}^{k-1} |q_{j+1} - q_j|^\alpha : (q_1, \ldots, q_k) \in Q^k \text{ with } q_1 = q_Q(x),\ q_k = q_Q(y),\ k \geqslant 1 \right\} . \tag{3.5}$$

The paper [6] shows that the sample Fermat distance when appropriately scaled converges to the Fermat distance. For sake of brevity, we refer to Appendix A for exact statement. More theoretical insight of Fermat distance and applications to clustering can be found in [25].

Intuition behind Sample Fermat Distance. The sample Fermat distance searches for the shortest path relating the points. The length of each path is the sum of the Euclidean distances of consecutive points in the path powered by α. With $\alpha = 1$, the shorted path between x and y is simply the line relating $q_Q(x)$ and $q_Q(y)$ (Fig. 3.2a). However, with a sufficiently large α, this definition of path length discards consecutive points with a large Euclidean distance instead favoring points that are closely positioned in terms of Euclidean distance. So, this will qualify the path passing through high density areas. Moreover, as this distance depends also on the number of terms in the sum in Eq. (3.5), this enforces a path to be smooth enough. These two remarks show that Fermat distance naturally captures the density and geometry of the dataset.

In Fig. 3.2, we go back to the moon example where Q is a moon-shaped cluster of points. We randomly choose 2 points and compute the Fermat path. We see that with $\alpha = 1$, we recover the Euclidean distance and so the Fermat path is simply a line. For α larger than 1 but not large enough (for instance, $\alpha = 1.2$, Fig. 3.2b), the Fermat path still does not capture the orientation of the dataset. However, as α gets larger, the Fermat path rapidly tracks the orientation of the dataset. For instance, with $\alpha = 3$, the path follows very well the distribution shape.

[2]Of course, uniqueness is generically achieved, for example almost surely for random points sampled according to a diffuse measure.



(a) Before (b) Before (c) After (d) After

Figure 4.1: $\widehat{LD}$ with Sample Fermat Distance on moon and spiral datasets. (a) and (b) are results of using directly sample Fermat distance in Eq. (3.5). This produces undesirable artifacts where we observe zones of constant value of LD. This phenomenon is explained by Proposition 1. (c) and (d) use our modified version in Eq. (4.1): it captures perfectly the distributions.

4 Combining LD and Fermat Distance

4.1 Artifacts from classical Fermat distance

In the computation of the depth, instead of using Euclidean distance, we use sample Fermat distance. The results for the moon and spiral datasets are depicted in Fig. 4.1a and 4.1b. We see that the shape of datasets is much better captured. However, we also observe some zones having constant LD value (represented by the same color). The existence of such zones are explained by the following proposition:

Proposition 1. *For $x \in \mathbb{R}^d$, $\widehat{LD}(x) = \widehat{LD}(q_Q(x))$. In other words, the empirical lens depth is constant over the Voronoï cells[3] associated to Q.*

The proof of Proposition 1 is in **Appendix** E. The consequence of the last proposition is that, even for a point far removed from Q, the value of $\widehat{LD}$ remains the same as that of its nearest point in Q. Consequently, $\widehat{LD}$ does not vanish at infinity. This is totally undesirable, as an ideal property of any depth is to vanish at infinity. To avoid this undesirable artifact, we need to modify the sample Fermat distance so that it takes into account the distance to Q.

4.2 Modified Sample Fermat Distance

The modified distance is defined, for $y \in Q,\ x \in \mathbb{R}^d$ as follows:

$$D^{\text{modif}}_{Q,\alpha}(x,y) := \min_{q \in Q}\{|x-q|^\alpha + D_{Q,\alpha}(q,y)\}\,. \tag{4.1}$$

Here, $D_{Q,\alpha}(q,y)$ has been defined in Eq. (3.5).

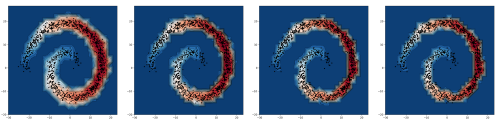
Interpretation. In the original definition in Eq. (3.5), the path always starts by the closest point in the dataset. Consequently, the distance to this closest point is totally ignored. To eliminate this drawback, the distance to a potential starting point lying in Q is added. Note that the optimization problem for calculating $D^{\text{modif}}_{Q,\alpha}$ is of the same type as that for calculating $D_{Q,\alpha}$ with only a change of starting point. Hence, the consistency of this empirical distance towards the theoretical Fermat distance remains true. Indeed, in the new formulation (4.1), the point $q \in Q$ is not fixed at $q_Q(x)$ but remains free and is a part of the optimization problem. Notice further that our modified version enjoys two nice properties. Firstly, if $x \in Q$ then Eq. (4.1) coincides with Eq. (3.5) ($D^{\text{modif}}_{Q,\alpha}(x,y) = D_{Q,\alpha}(x,y)$). Secondly, $D^{\text{modif}}_{Q,\alpha}(x,y)$ increases to infinity when x is going far away from Q. Consequently, in this case, the corresponding $\widehat{LD}$ w.r.t Q tends to 0. The $\widehat{LD}$ using this modified version of the distance is displayed on two examples in Fig. 4.1c and 4.1d. With our modification, the undesirable artifact of constant-valued zone is erased. Furthermore, for points far away from the dataset, $\widehat{LD}$ tends quickly to 0. In conclusion, our method captures the shape of distributions perfectly.

4.3 Qualitative evaluation of stability

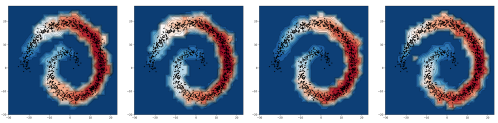
We experiment and evaluate the stability of our method on the spiral dataset. This is a tricky dataset, and a standard method like the Gaussian one cannot capture its shape.

[3]Definition of the Voronoï cells is in **Appendix** J.

https://doi.org/10.52202/079017-0951



(a) LD applying different values of hyperparameter α in Sample Fermat Distance. From left to right, $\alpha \in \{3, 5, 10, 15\}$. For different values of α, the method always captures really well the distribution.



(b) LD using only 20% of points (200 points) on simulated spiral dataset of 1000 points. The contours of LD level changes slightly between different tries, but in general, the proposed method captures well the general shape of distribution.

Figure 4.2: Stability with respect to number of training points and α.

4.3.1 Stability with respect to number of training points

When running a statistical algorithm, it is desirable to have as large a sample as possible. However, in many cases, only a very small amount of data is available. This motivates the study of the stability of our method in a small data regime. Here, we simulate the spiral dataset with 1000 points. Then, we choose randomly only 20% of the simulated points (i.e. 200 points) as the sample dataset to compute LD. We perform different runs for different random samples with $\alpha = 5$ for a visual evaluation. For the sake of brevity, only the results of four tests are shown in Fig. 4.2b. More replications are displayed in **Appendix** C. We see that in the 4 tries, our method gives slightly different estimation of LD. This small fluctuation is to be expected, as we take only 20% of the points at random each time. Nevertheless, the method captures the shape of the dataset really well (the full sample of 1000 points is displayed in the figures). Besides, we also perform an experiment where we reduce the number of points until the method fails to capture the shape of the distribution. This helps us to have a better idea how our method works at small-data regime. We refer to **Appendix** D for results.

4.3.2 Stability with respect to the hyperparameter α

In our method, only one hyperparameter (α), governing the Fermat distance needs to be chosen. It is therefore important to assess the stability of the method w.r.t. α. For this purpose, we experiment with different values of $\alpha > 1$ (recall that $\alpha = 1$ corresponds to the Euclidean distance). For each $\alpha \in \{3, 5, 10, 15\}$, we test our method on the spiral dataset. Results are shown in Fig. 4.2a. The conclusion is that our method is very stable through different values of α. Indeed, in the 4 cases, it always captures almost perfectly the dataset support, which implies a strong stability of the method. Of course this stability is only achieved in the proper range when α is large enough (See Fig. 3.2).

4.4 From LD to OOD uncertainty score

Our ultimate objective is to use LD to provide an OOD uncertainty score. To do so, we apply LD to the feature space $\mathcal{F}$ of our classification model. Let C be the number of separate clusters. Now, there are two ways for computing LD of a new point: (1) All the clusters are considered as a sole distribution to compute LD; (2) Compute LD w.r.t. the different clusters and then take the max among the LD's (i.e. LD w.r.t. the nearest cluster). It turns out that the first approach gives unsatisfying result as explained in **Appendix** F. So, we adopt the second approach in this paper. More formally, let us denote $\widehat{LD}(\Phi(x), \mathcal{C}_i)$ the empirical LD of x w.r.t. the i^{th} cluster formed by training examples of class i (in the feature space) (i.e. the clustering is given by the labeling as we have labels here). Then, the confidence score of x is defined as

$$S(x) := \max_i \widehat{LD}(\Phi(x), \mathcal{C}_i) \ . \tag{4.2}$$

4.5 Computational complexity: the main limitation of LD and how to be more efficient

From Eq. (3.2), we can deduce that the complexity of calculating LD for a given point is $O(CN^2)$ (C is the number of classes, N is number of examples in each class). It is therefore useful to reduce the number of inner points N used to calculate LD while maintaining good precision. Keeping only n inner points among the N initial ones, we then have 3 different straightforward strategies:

- **I. Random.** Randomly sample without replacement n points among N intial points.

- **II. K-mean/center.** We want the n points to cover well the support of the initial sample. Hence, we first apply a *K-mean clustering* with n centroids on the N points. Then, the n resulted centroids are used as inner points.
- **III. K-mean/center+.** Same as strategy **II**, but instead of using directly the centroids, we use the inner point closest to each centroid.

We test and discuss about these strategies in Appendix G. It turns out that *K-mean / Center* outperforms the two other strategies with a very small number of inner points n. We refer to Appendix G for more discussion and detailed experiment. So, for the rest of this paper, we use strategy **III.**

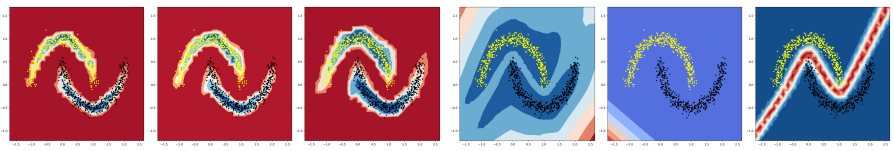
5 Experiments on Neural Networks

We first evaluate our method on the two-moon dataset. Then, we evaluate on 2 benchmarks FashionMNIST-MNIST and CIFAR10-TinyImageNet/CIFAR100/SVHN the ability of our method for the detection of OOD. Besides, we also evaluate the consistency property of our uncertainty score as presented in introduction section (shown in Fig. 5.2). Without further mention, we fix $\alpha = 7$ for all experiments. For a fair comparison, we use the same model architectures as in the previous work of [27]. More details about the models and the training schemes can be found in the **Appendix** B.

5.1 How is the input distribution represented in the feature space of *softmax* model?

We first perform experiment on the two-moon dataset consisting of 2 classes, each having a moon shape. We train a neural network with 2 hidden layers for classification (more details can be found in the **Appendix** B). After training, the model parameters are fixed and different methods for uncertainty evaluation are applied in the feature space $\mathcal{F}$ of this model. One popular way to provide an uncertainty score is to use the predictive distribution entropy[4]. It is maximized when the distribution is uniform. In this example, predictive distribution entropy is high only in a boundary zone (Fig. 5.1f). This is to be expected, as the model is trained to learn a boundary between the two classes. Nevertheless, it is desired to assign a high uncertainty to the region without training data. Indeed, it might be too risky to make decision in these zones, especially in critical applications.

Is Gaussian prior suitable? We consider the methods of Euclidean distance (Fig. 5.1d) and GDA [13] (Fig. 5.1e). For the Euclidean distance method, we compute the distance to the centroids of the different clusters (in $\mathcal{F}$) and then we take its minimum. Surprisingly, in this example, the crude use of Euclidean distance seems to better capture the input distribution than GDA (failing miserably on this dataset). This suggests that the distribution of clusters in feature space is more complicated than the Gaussian one. This remark shows the necessity to have a method able to capture better the distribution. LD can capture impressively well the zone where we have training data (Fig. 5.1a, 5.1b and 5.1c corresponding to $\alpha = 3$, 10 and 15). Hence, LD is able to pin down clusters with a complex support shape in the feature space. Furthermore, we intentionally use 3 values for α with large gaps to show the stability w.r.t. α.



(a) LD,$\alpha = 3$ (b) LD,$\alpha = 10$ (c) LD,$\alpha = 15$ (d) Euclidean (e) GDA (Gaussian) (f) Predictive entropy

Figure 5.1: Methods for uncertainty estimation applied on the same neural net trained to classify 2 classes in moon-shape (represented by yellow and black points respectively). Uncertainty estimations are computed based solely on the feature space of the model without seeing directly the inputs. Red represents high uncertainty. Our method (Fig. 5.1a, 5.1b and 5.1c) gives excellent results and much better than other methods.

5.2 FashionMNIST vs MNIST

We perform five different runs to train classification models on the dataset FashionMNIST [28]. Firstly, we evaluate our method by studying the separation capacity between the test set of FashionMNIST

[4]The entropy of a predicted probability $p \in \mathbb{R}^C$ is calculated as $H(p) = -\sum_{i=1}^{C} p_i \log(p_i)$, with $\sum_{i=1}^{C} p_i = 1$ and $0 \leqslant p_i \leqslant 1$.

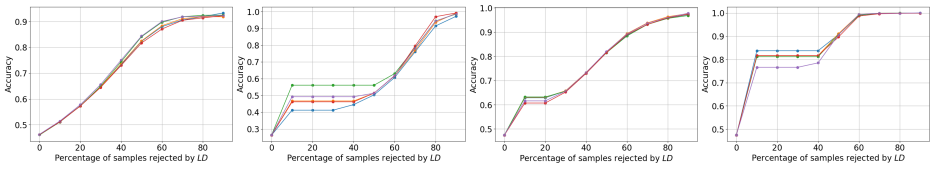
https://doi.org/10.52202/079017-0951

Table 5.1: Results on FashionMNIST, with MNIST as OOD set. Results marked by (□) are extracted from [23] and (△) are extracted from [27]. Deep Ensembles by [9], Mahalanobis Distance by [13], LL ratio by [23], DUQ by [27].

Charateristics	Method	AUROC
No impact on original model	LD (our method)	$0.971_{\pm 0.001}$
	Euclidean Distance	$0.943_{\pm 0.009}$
	Mahalanobis Distance	0.942
Use particular type of model difficult to train	DUQ (△)	0.955
Need to train many models	Deep Ensembles (5 Models) (△)	0.861
Need to train extra generative models	LL ratio (△)	0.994

and of MNIST [11] based on *AUROC* score. Results are reported in Table 5.1. We first compare our method to Euclidean and GDA method [13]. Notice that our method outperforms these two distance-based methods. A more sophisticated method called DUQ [27] stands on a devoted neural architecture (RBF network). This particular type of model is much more difficult to train and so generally does not preserve the accuracy of the main classification task (compared to standard *softmax* models). Once again, our method outperforms this method. This indicates that our method measures a natural "depth" directly in the feature space without the need of changing completely the model as in DUQ method. Another popular method is Deep Ensembles in which one trains and applies many independently-trained models for the same task. Despite its heavy demanding of resource, our method outperforms this approach in this experiment. A more advanced method for density estimation is LL ratio [23]. In this method, one needs to train two supplementary generative models to estimate distributions. This method needs an adequate noise and really careful training of these 2 generative models so that they can reflect the true underlying input density. With this complex process [5], this method gives better AUROC score than ours in this experiment.

Consistency curve. Following some previous works (e.g. [9], [27]), we compound test set of FashionMNIST and MNIST together and all the data from MNIST are considered to be incorrectly predicted by the model. Then, a certain percentage of data is rejected based on their LD. If LD is an appropriate indicator for prediction uncertainty, then accuracy on the retained data has to be an increasing function of the rejection percentage. We call the resulted curve *consistency curve*. The results for five runs are depicted in Fig. 5.2a. We see that the curves are always increasing over 5 runs. Hence, LD is a good measure for uncertainty estimation.



(a) FashionMNIST/MNIST (b) CIFAR10/SVHN (c) CIFAR10/CIFAR100 (d) CIFAR10/Tiny-ImageNet

Figure 5.2: Consistency curves for FashionMNIST/MNIST, CIFAR10/SVHN, CIFAR10/CIFAR100 and CIFAR10/Tiny-ImageNet over 5 runs (each curve corresponds to an independently trained model).

5.3 CIFAR10 vs SVHN/Tiny-ImageNet/CIFAR100

In this section, we compare LD to popular deterministic single-forward pass methods for uncertainty quantification (UQ). We also compare with Deep-Ensemble, as it stays a *de facto* method for UQ. We train the models on the training set CIFAR10 [8] and then the test set of CIFAR10 is considered as in-distribution (ID) data. We use test sets of 3 datasets SVHN [21], CIFAR100 and Tiny-ImageNet [10] as OOD data. AUROC scores are reported in Table 5.2. We first compare LD with GDA using ResNet18 (the first two lines). We see that LD consistently outperforms GDA on 3 OOD sets. Next, we compare our method with DDU [20]. This method is basically GDA but one adds spectral normalization (SN) in the model. Using ResNet18, our method outperforms on CIFAR100 and SVHN. An interesting remark is that using SN seems to improve AUROC on CIFAR100 and SVHN. This is expected as SN encourages more smoothness. However, for Tiny-ImageNet, AUROC gets worse using SN. That is, somehow SN makes the features of TinyImageNet closer to those of CIFAR10. Using WideResNet, DDU seems to be a little better than ours on CIFAR100 and SVHN but on TinyImagNet, our method is better. Next, we compare LD with DUQ. Our method consistently

[5] See Appendix H for more details.

https://doi.org/10.52202/079017-0951

outperforms DUQ on all the benchmarks. Besides, our method also outperforms energy-based method [16]. Considering the method SNGP [14], it is based on a Gaussian process layer with SN, making itself a very intrusive method. In spite of this, LD outperforms this method on the two pairs CIFAR10/Tiny-ImageNet and CIFAR10/CIFAR100. On CIFAR10/SVHN, this method performs a little better than ours. Finally, we observe that Deep Ensemble (with 5 models) seems to perform very well on the 3 pairs. This is to be expected, as it uses 5 models instead of one single model. However, surprisingly, on CIFAR10/Tiny-ImageNet, LD outperforms this latter method.

Besides, to see if our method scales well when number of classes of ID data increases, we also experiment with CIFAR100 as ID data. The results are shown in Fig. 5.3. Overall, through experiments with CIFAR10 and CIFAR100 as ID data, we see that our method performs very well and outperforms many strong baseline UQ methods. This proves that LD is a useful tool to capture the underlying distribution to provide an OOD uncertainty score.

Consistency curve. We plot the consistency curves over the 5 runs as in Section 5.2 for 3 pairs (Fig. 5.2b, 5.2c and 5.2d). Once again, accuracy is always an increasing function of the rejected percentage of the data based on LD. This confirms again that LD is an appropriate indicator for uncertainty estimation and so it is useful for decision making.

Table 5.2: Results on CIFAR10 with Tiny-ImageNet, CIFAR100 and SVHN as OOD sets. SN: Spectral Normalisation, JP: Jacobian Penalty.

Source	Method	Model	Penalty	AUROC Tiny-ImageNet	AUROC CIFAR100	AUROC SVHN
ours	LD (ours)	ResNet18	No	$\mathbf{0.965}_{\pm 0.003}$	$\mathbf{0.892}_{\pm 0.002}$	$\mathbf{0.936}_{\pm 0.006}$
ours	GDA ([13])	ResNet18	No	$0.945_{\pm 0.005}$	$0.864_{\pm 0.003}$	$0.914_{\pm 0.014}$
ours	LD (ours)	ResNet18	SN	$0.927_{\pm 0.003}$	$\mathbf{0.900}_{\pm 0.001}$	$\mathbf{0.950}_{\pm 0.008}$
ours	DDU (GDA + SN) ([20])	ResNet18	SN	$\mathbf{0.937}_{\pm 0.009}$	$0.872_{\pm 0.005}$	$0.947_{\pm 0.015}$
[27]	DUQ ([27])	ResNet18	JP	–	–	$0.927_{\pm 0.013}$
ours	LD (ours)	Wide-ResNet-28-10	SN	$\mathbf{0.926}_{\pm 0.002}$	$0.906_{\pm 0.001}$	$0.939_{\pm 0.007}$
[20]	DDU (GDA + SN) ([20])	Wide-ResNet-28-10	SN	$0.9107_{\pm 0.0005}$	$\mathbf{0.913}_{\pm 0.0004}$	$\mathbf{0.979}_{\pm 0.002}$
[20]	DUQ ([27])	Wide-ResNet-28-10	JP	$0.868_{\pm 0.001}$	$0.859_{\pm 0.003}$	$0.937_{\pm 0.006}$
[20]	SNGP ([14])	Wide-ResNet-28-10	SN	$0.899_{\pm 0.002}$	$0.911_{\pm 0.002}$	$0.940_{\pm 0.001}$
[20]	Energy-based ([16])	Wide-ResNet-28-10	No	$0.881_{\pm 0.0006}$	$0.889_{\pm 0.0007}$	$0.945_{\pm 0.005}$
[20]	5-Ensemble ([9])	Wide-ResNet-28-10	No	$0.9006_{\pm 0.0003}$	$0.921_{\pm 0.0002}$	$0.977_{\pm 0.003}$

Table 5.3: AUROC score with CIFAR100 as ID data and Tiny-Imaget as OOD data. Results of other methods are extracted from [20] where all the methods were experimented on the same *Wide-ResNet-28-10* model.

Method	AUROC
LD (ours)	$0.8310_{\pm 0.0013}$
Softmax Entropy	$0.8153_{\pm 0.0005}$
Energy-based	$0.8133_{\pm 0.0006}$
SNGP	$0.7885_{\pm 0.0004}$
DDU	$0.8313_{\pm 0.0006}$
5-Ensemble	$0.8295_{\pm 0.0009}$

5.4 On the limitations of the Gaussian assumption: LD vs GDA

In Table 5.2, we see that in some cases, GDA performs a little better than LD. Does this mean the distribution in the feature space is Gaussian? In fact, if the OOD set is sufficiently far from the ID one, then OOD data lies outside the smallest ellipsoid containing the ID data. In this case, Gaussian fitting can perfectly separate ID and OOD, even if the distribution is not Gaussian. That is, a good AUROC score by GDA does not necessarily imply that the distribution is Gaussian. However, if OOD and ID sets get closer, sharper detection boundaries between ID and OOD data become necessary.

To assess this, we perform an experiment with OOD more similar to ID data, thanks to hold-one-out experiments. For each of the two datasets MNIST and CIFAR10, we train model on the 9 classes and hold out one class as OOD data. In this way, ID set is more similar to OOD one as they come from the same dataset. Results are shown in Table 5.4. In this setup, the gap in terms of AUROC score

https://doi.org/10.52202/079017-0951

between LD and GDA is much larger than in Table 5.2. As such, LD seems to be adaptively finding a sharper boundary than the Gaussian method. This is to be expected, as the boundaries obtained from Gaussian fitting are necessarily elliptical.

Table 5.4: AUROC score for OOD data that are close to ID data

Data pair	Hold-one-out MNIST	Hold-one-out CIFAR10
LD (ours)	$\mathbf{0.969}_{\pm 0.007}$	$\mathbf{0.806}_{\pm 0.015}$
GDA	$0.898_{\pm 0.019}$	$0.759_{\pm 0.032}$

6 Discussion

As one property of Fermat distance is being able to adapt itself to the manifold, one could think of alternative methods following manifold learning literature. However typical methods in manifold learning do not have the property of yielding shorter distances in high density areas. Moreover, LD allows the choice of any distance, unlike other typical depths (Half-space depth or Simplicial depth...). As such, the choice of combining LD with Fermat distance is synergistic and not independent at all.

The main advantage of our method is that we make no prior distribution assumption. However, there are still extreme cases where our method would not work well. Indeed, let us consider the case where there is a class with 2 clusters in the feature space. From theoretical viewpoint, for the Fermat distance to be well defined, it is crucial for the density f to remain bounded from above and away from zero - see Appendix A. Hence, in between clusters, we need "very small density" but not "zero density". However, from a practical viewpoint, in such cases, one could argue that 2 clusters of the same class should not be too distinct. Indeed, if the main model in trained to classify properly, semantically similar inputs should be close to each other, leading to connected clusters for each class. But in general, the cluster of each class should be sufficiently connected to yield an ideal result. This also explains why we propose to work in the feature space instead of using directly raw data points. Indeed, feature spaces help us to extract the low dimensionality of the data more efficiently at a semantic level, and to have proper clusters in the feature space.

Besides, note that the aim of LD is to measure the Out-of-domain uncertainty, which is due to zones in feature space that are scarce in data. As the model is not trained in these zones, one should be cautious with the model's predictions on these zones as it can behave very randomly due to scarcity of training data. Consider for example the two-moons experiment where the two moons have more spread (and even overlap). In such cases, we have enough data in the zone between the 2 classes. Hence, LD should not be able to detect the uncertainty in this case. Other metrics such as predictive entropy should be a good candidate in this case.

Finally, an interesting use–case is to apply our method on pre-trained models. This is because SOTA models become often too large to retrain ourselves. If we have no idea about the data distribution, we are convinced that our method should be a useful tool, at least as a starting point, for better understanding the data scarcity in feature space.

7 Conclusion

In this work, we use Lens Depth combined with a modified version of the sample Fermat distance. This combination captures naturally the shape and density of the input distribution. This is not the case with many previously proposed methods, which assume a prior distribution or use additional models with trainable parameters, or even modify the mechanism of the training process. Our method is non-parametric and non-intrusive. Through a toy dataset as well as experiments conducted on Deep Neural Networks, we show that our method adapts very well to many cases. Hence, our work opens new venues for non-parametric methods capturing the input distribution to quantify uncertainty in the context of Deep Learning. For future work, it would be interesting both to have an efficient algorithm for computing Lens Depth with some error bound and to mathematically investigate the impact of the hyper-parameter α. Finally notice that, while we were focused on neural networks, any classification model with a feature space, e.g. kernel methods, can benefit from our framework.

Broader impact statement

Our Out-of-Distribution (OOD) uncertainty quantification holds implications for enhancing safety in critical applications. By effectively addressing uncertainty beyond the training data, our approach contributes to robust decision-making, particularly in scenarios where reliability is crucial. Moreover, our method can play a role in ensuring fairness by recognizing and mitigating potential biases that may arise when the training data lacks sufficient representation.

References

[1] F. Aurenhammer. Voronoi diagrams—a survey of a fundamental geometric data structure. *ACM Computing Surveys (CSUR)*, 23(3):345–405, 1991.

[2] A. Cholaquidis, R. Fraiman, F. Gamboa, and L. Moreno. Weighted lens depth: Some applications to supervised classification. *Canadian Journal of Statistics*, 51(2):652–673, 2023.

[3] T. DeVries and G. W. Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

[4] Y. Gal et al. Uncertainty in deep learning. 2016.

[5] J. Gawlikowski, C. R. N. Tassi, M. Ali, J. Lee, M. Humt, J. Feng, A. Kruspe, R. Triebel, P. Jung, R. Roscher, et al. A survey of uncertainty in deep neural networks. *Artificial Intelligence Review*, 56(Suppl 1):1513–1589, 2023.

[6] P. Groisman, M. Jonckheere, and F. Sapienza. Nonhomogeneous euclidean first-passage percolation and distance learning. *Bernoulli*, 28(1):255–276, 2022.

[7] N. Kotelevskii, A. Artemenkov, K. Fedyanin, F. Noskov, A. Fishkov, A. Shelmanov, A. Vazhentsev, A. Petiushko, and M. Panov. Nonparametric uncertainty quantification for single deterministic neural network. *Advances in Neural Information Processing Systems*, 35:36308–36323, 2022.

[8] A. Krizhevsky, V. Nair, and G. Hinton. Cifar-10 (canadian institute for advanced research).

[9] B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

[10] Y. Le and X. Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

[11] Y. LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

[12] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[13] K. Lee, K. Lee, H. Lee, and J. Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31, 2018.

[14] J. Liu, Z. Lin, S. Padhy, D. Tran, T. Bedrax Weiss, and B. Lakshminarayanan. Simple and principled uncertainty estimation with deterministic deep learning via distance awareness. *Advances in neural information processing systems*, 33:7498–7512, 2020.

[15] R. Y. Liu. On a notion of data depth based on random simplices. *The Annals of Statistics*, pages 405–414, 1990.

[16] W. Liu, X. Wang, J. Owens, and Y. Li. Energy-based out-of-distribution detection. *Advances in neural information processing systems*, 33:21464–21475, 2020.

[17] D. J. C. MacKay. A practical bayesian framework for backpropagation networks. *Neural Computation*, 4:448–472, 1992.

[18] P. C. Mahalanobis. On the generalized distance in statistics. *Sankhyā: The Indian Journal of Statistics, Series A (2008-)*, 80:S1–S7, 2018.

https://doi.org/10.52202/079017-0951

[19] A. Malinin and M. Gales. Predictive uncertainty estimation via prior networks. *Advances in neural information processing systems*, 31, 2018.

[20] J. Mukhoti, A. Kirsch, J. van Amersfoort, P. H. Torr, and Y. Gal. Deep deterministic uncertainty: A new simple baseline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24384–24394, 2023.

[21] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng. Reading digits in natural images with unsupervised feature learning. 2011.

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[23] J. Ren, P. J. Liu, E. Fertig, J. Snoek, R. Poplin, M. Depristo, J. Dillon, and B. Lakshminarayanan. Likelihood ratios for out-of-distribution detection. *Advances in neural information processing systems*, 32, 2019.

[24] Y. Sun, Y. Ming, X. Zhu, and Y. Li. Out-of-distribution detection with deep nearest neighbors. In *International Conference on Machine Learning*, pages 20827–20840. PMLR, 2022.

[25] N. G. Trillos, A. Little, D. McKenzie, and J. M. Murphy. Fermat distances: Metric approximation, spectral convergence, and clustering algorithms. *arXiv preprint arXiv:2307.05750*, 2023.

[26] J. W. Tukey. Mathematics and the picturing of data. In *Proceedings of the International Congress of Mathematicians, Vancouver, 1975*, volume 2, pages 523–531, 1975.

[27] J. Van Amersfoort, L. Smith, Y. W. Teh, and Y. Gal. Uncertainty estimation using a single deep deterministic neural network. In *International conference on machine learning*, pages 9690–9700. PMLR, 2020.

[28] H. Xiao, K. Rasul, and R. Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, 2017.

A Convergence of rescaled Sample Fermat Distance to Population Fermat Distance

Theorem 2.3 in [6] proves convergence of rescaled sample FD to population FD, with rescaling n^β with $\beta = (\alpha - 1)/d$. More precisely, there exists a constant μ such that

$$\lim_{n\to\infty} n^\beta D_{Q_n,\alpha}(x, y) = \mu D_{f,\beta}(x, y)$$

when the samples $Q_n = (X_1, X_2, \dots)$ are sampled either as n i.i.d. with density f or as a Poisson point process with intensity nf. It is crucial for f to remain bounded from above and away from zero.

B Experimental Details

All experiments related to neural networks are implemented in Pytorch *2.0.1+cuda*, with its default initialization.

B.1 Two-moons

For generating two-moon dataset, we use package scikit-learn [22], with noise 0.07, random state 1, and 1000 samples.

For model, we construct a simple fully connected network with 2 hidden layers, each gives output of dimension 20. First hidden layer is followed by non-linearity *ReLU*. We train model for 160 epochs using Adam optimizer, learning rate 10^{-3}, other parameters are set by default of Pytorch package.

B.2 FashionMNIST

For a fair comparison, we follow exactly the same CNN architecture proposed in [27] and the same training scheme with only one minor modification: the dimension of penultimate layer is 25 instead of 256 for efficient computation related to *LD*. We observe this modification has no impact on accuracy of model. We refer reader to [27] for details. From training set, we randomly split 80:20 to have training data and validatation. We choose the best model based on accuracy on validation set. Test accuracy after training over 5 runs is $92.35\% \pm 0.19$.

For estimating $\widehat{LD}$, we use 1500 training examples for each class based on results of the experiment in Section G. We observed that applying the method on normalized feature vectors (L2-norms) (which is the reported result) gives slightly better result than applying directly on the feature vectors. The method is applied on the test set of FashionMNIST consisting of 10,000 images and the test set of MNIST also consisting of 10,000 images).

B.3 CIFAR10

For ResNet18 without spectral normalization (SN), we use ResNet-18 model implemented by [3] with a minor modification and training scheme of the same authors. More specifically, after Global Average Pooling layer of CNN, we add a layer of output dimension of 25 instead of 256 proposed by [27] before softmax layer. For training model, we use SGD optimizer with nesterov momentum, learning rate 0.1 (decayed by factor of 5 at epochs 60,120 and 160), momentum 0.9, weight decay $5 \cdot 10^{-4}$.Model is trained for 200 epochs. We train model on the full training set (i.e. no validation set) and save the last model, i.e. the model at epoch 200. Test accuracy after training over 5 runs is 0.950 ± 0.001.

For ResNet18 and Wide-ResNet-28-10 with SN, we use exactly the same model and training scheme proposed in [20] for fair comparisons. We refer readers to the concerned paper for details.

CIFAR10/SVHN: As there are many more images to test compared to the previous experiment, we use only 1000 training images in each class for estimate LD using *K-mean/Center* strategy to have a reasonable run time. We tested the method on normalized and non-normalized feature vectors and

https://doi.org/10.52202/079017-0951

observed no significant difference. The reported result is on non-normalized vectors. Notice that the method is applied on test sets of CIFAR10 consisting of 10,000 images and of SVHN consisting of 26,032 images.

CIFAR10/Tiny-ImageNet and CIFAR10/CIFAR100: we use only 500 points for each class to evaluate LD. We observe that increasing number of points for more than 500 points gives no significant improvement.

As in [27], at test time, we use the statistics (mean and standard deviation) of the training set (i.e. FashionMNIST or CIFAR10 in our case). Indeed, these statistics are used both in the *Batch normalization* layers and in the data normalization process (both for OOD and for ID set).

B.4 Hold-one-out MNIST and CIFAR10

We training models on the hold-one-out MNIST and CIFAR10. Hence, each training set is now of 9 classes. We perform 5 runs for each set and evaluate AUROC score using LD and GDA. The model and training scheme for hold-one-out MNIST is exactly the same as in Appendix B.2. The model and training scheme for hold-one-out CIFAR10 is exactly the same as in Appendix B.3. For evaluate LD, we use 500 points for each class.

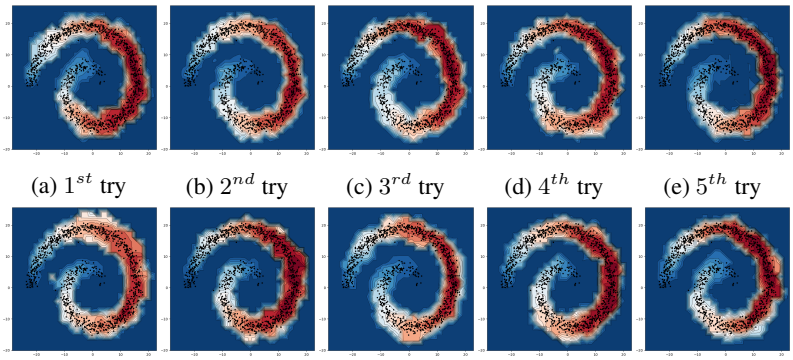
B.5 Models with Spectral Normalization

We train ResNet18 with SN (for CIFAR10) and WideResNet with SN (for CIFAR100) using the code provided by [20].

To evaluate LD on CIFAR100/TinyImageNet, we apply directly on the training set of CIFAR100, as it have only 500 training examples for each class.

C More qualitative results of Fermat Lens Depth on spiral dataset using 20% points

We use LD using only 20% of points (200 points) on simulated spiral dataset of 1000 points over 10 runs for qualitatively evaluating stable of the method w.r.t. number of points.Results in Fig.C.1.



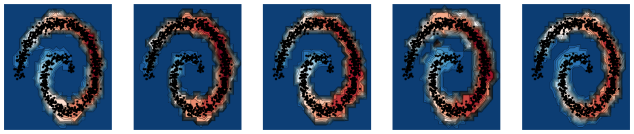
(a) 1^{st} try (b) 2^{nd} try (c) 3^{rd} try (d) 4^{th} try (e) 5^{th} try

(f) 6^{th} try (g) 7^{th} try (h) 8^{th} try (i) 9^{th} try (j) 10^{th} try

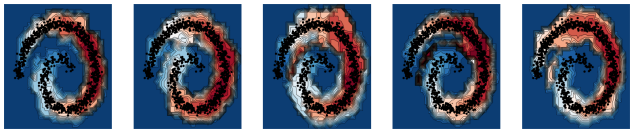
Figure C.1: LD using only 20% of points (200 points) on simulated spiral dataset of 1000 points over 10 runs. We see that the contours of LD level changes slightly between different tries, but in general, the proposed method captures well the general form of distribution. Note that the points presented in the plot are the full dataset of 1000 points.

D More qualitative results of Fermat Lens Depth on spiral dataset using a portion of point until failing

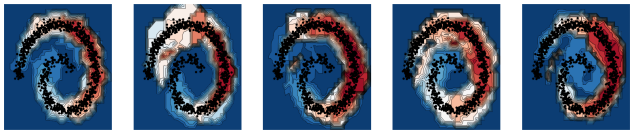
We perform an experiment where we reduce the number of points until the method fails to capture the shape of the distribution. This helps us to have a better idea how our method works at small-data regime. The results are shown in Fig. D.1.



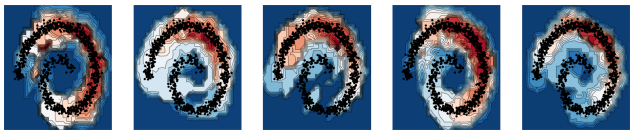
(a) 200 randomly sampled points (20% of original points)



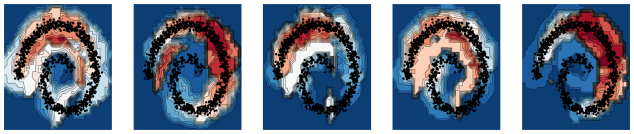
(b) 100 randomly sampled points (10% of original points)



(c) 80 randomly sampled points (8% of original points)



(d) 60 randomly sampled points (6% of original points)



(e) 50 randomly sampled points (5% of original points)

Figure D.1: LD using only a certain portion of points until LD fails to capture the shape of the distribution. Each row represents a fixed percentage,for which we performs 5 independent runs. Notice that the points displayed on the figure is the full dataset of 1000 points, so that we can observe how well $\widehat{LD}$ captures the original data distribution. Please also note that we randomly sample a small portion of the original points. Hence, the sampled points can be concentrated in a small region instead of being distributed along the spiral. So, the sampled points can represent not very correctly the original distribution. Therefore, it is not surprising that LD fails to capture the original support at around $5-6\%$ of the original size. (Note that we use 20% for results in Fig.4.2.b in the main paper.)

E Proof of Proposition 1

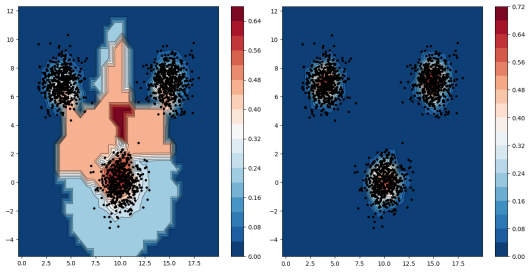
Proof. By definition, $q(q(x)) = q(x)$ and $q(y) = y$ as $q(x), y \in Q$, so the closest point to them in Q is themselves. Hence, according to Eq.3.5, it is obvious that $D_{Q,\alpha}(x, y) = D_{Q,\alpha}(q(x), y), \forall x \in \mathbb{R}^d$ and $y \in Q$. Applying this sample Fermat distance to Eq.3.2 to compute empirical LD, we obtain $\widehat{LD}(x) = \widehat{LD}(q(x))$. □

https://doi.org/10.52202/079017-0951

F One or multiple clusters?

Our ultimate objective is to use LD for finding out-of-distribution data associated with an uncertainty (or equivalently confidence) score. For this purpose, we apply LD in feature space of *softmax* model. More concretely, we apply in the activation of penultimate layer right before *softmax* layer. In this setup where we have different separate clusters, one important question is: *Can we simply consider them as one distribution for computing LD?*

To answer this question, we simulate a dataset composed of 3 separate Gaussian clusters and then we compute LD w.r.t this dataset by consider them as one distribution. The result is shown in Fig.F.1a. We see that the result is not good: value of LD is large for zones lying between clusters whereas in this case, we want LD to be large only in 3 cluster and NOT in the zones lying between them. So, the solution for this problem is quite straightforward: we compute LD of a points w.r.t different clusters, than the final LD of that point is considered as its LD w.r.t to the closest cluster, i.e. we take the *max* among computed LD's. The result is shown in Fig.F.1b. We see that the result now is much bette: LD is only large in the zones of dataset which are 3 clusters.



(a) Consider 3 clusters as one distribution

(b) *Max* of LD w.r.t to different clusters

Figure F.1: LD computed by 2 ways: (a)Consider 3 clusters as one distribution w.r.t which one compute LD and (b) Compute LD w.r.t 3 separate clusters than final LD is computed as the *max* among the 3 LD's computed

G Effectiveness of Reduced Lens Depth

In this section, we evaluate the effectiveness of the 3 strategies discussed in Section 4.5 to reduce the computing complexity of LD. We evaluate the quality of each strategy by measuring how well the ID can be separated from OOD set in term of the *AUROC* metric[6]. The pair FashionMNIST/MNIST is used to assess the suitability of the three strategies. This pair is much more difficult to handle than MNIST/NotMNIST as argued in previous works (e.g. [27]). For each strategy, we use $n \in \{500, 1000, 1500\}$ points for each class (each class contains 6000 training examples). Results are reported in Table G.1. In all cases, strategy **II** always gives the best result. Remarkably, with only 500 points, *K-mean / Center* is already better than the two other strategies with 1500 points. *K-mean / Center* has a regularization effect from averaging points (for calculating centroids). We conjecture that this effect makes the method much more stable, and also facilitates the capture of the overall shape of the cluster by avoiding certain irregular points that could have a negative impact on the estimate of LD.

Moreover, as number of points are increased from 500 to 1500, we observe no significant improvement in *AUROC* score. This reinforces our conjecture about the impact of irregular points on estimation of LD and furthermore, this remark implies that if the n chosen points cover well enough the initial space occupied by the N original points, then we only need to choose a very small percentage of points for a good estimation of LD. So, for the rest of this paper, we use strategy *K-mean / Center*.

Finally, we note that this could lead to some change in the original density. However, at the end, our objective is to measure how "central" a point is w.r.t. our data and only LD matters. So, our motivation for using reduced methods is to find a configuration of points that cover well the support of the original data. If this is the case, even if there is change in density, the change of LD is minimal

[6]AUROC is equal to 1 when the data are perfectly separable.

Table G.1: Comparing AUROC performance of strategies for reducing complexity of LD on Fashion-MNIST/MNIST

NO. OF TRAINING EXAMPLES	500	1000	1500
I. RANDOM	0.9368	0.9426	0.9436
II. K-MEAN / CENTER	**0.9543**	**0.9548**	**0.9553**
III. K-MEAN / CENTER+	0.9475	0.9536	0.9537

and the ordering of points by LD is not really impacted. That is, points that are "central" will remain with large LD and points near the the frontier of the original support should still have small value of LD.

H LL ratio method

In this method, instead of using directly the main model, one needs to train two supplementary generative models to estimate distributions. A first model is trained on ID data and a second one is trained on perturbed inputs. So that, the second model captures only the background statistics. Under suitable assumptions, authors show that the ratio between these two likelihoods cancels out the background information. Consequently, the LL ratio focuses on the semantic part of the input and so can be used as a score to distinguish ID from OOD data. This method needs an adequate noise in a way that the perturbed inputs contain only background information. This process itself is complicated as we need some supplementary dataset to choose the noise. Moreover, one needs to really carefully train these 2 generative models so that they can reflect the true underlying input density.

I Rejection curve for In-distribution Data of CIFAR10

We use only test set of of CIFAR10, i.e. in-distribution data to evaluate consistency curve. Result in Fig. I.1. As expected, the curve is increasing, which implies that LD in a consistent indicator for confidence level.

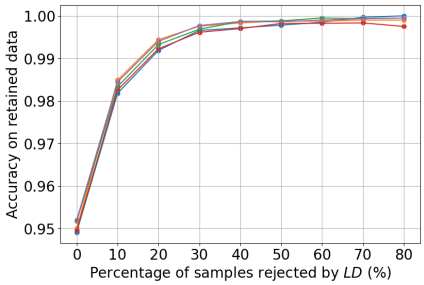


Figure I.1: Rejection curve CIFAR10 over 5 runs (each curve corresponds to an independently trained model).

J Voronoï cells

Suppose we have a finite number of distinct points in the plane, referred as *sites*, *seeds* or *generators*. Each seed has a corresponding region, called a Voronoï cell, made up of all the points in the plane closer to that seed than to any other. We refer to [1] for more details.

https://doi.org/10.52202/079017-0951

NeurIPS Paper Checklist

1. **Claims**
   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?
   Answer: [Yes]
   Justification: The abstract and introduction clearly state the claims made, including the contributions made in the paper and important assumptions.
   Guidelines:
   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.
2. **Limitations**
   Question: Does the paper discuss the limitations of the work performed by the authors?
   Answer: [Yes]
   Justification: the main limitation of our method is the computational complexity compared to standard methods as clearly shown in Section 4.5. However, we proposed a method to reduce the complexity.
   Guidelines:
   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.
3. **Theory Assumptions and Proofs**
   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?
   Answer: [Yes]
   Justification: For each theoretical result, the paper provides the full set of assumptions and a complete proof.
   Guidelines:
   - The answer NA means that the paper does not include theoretical results.

- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: the paper fully discloses all the information needed to reproduce the main experimental results of the paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: the paper provide open access to code.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.

https://doi.org/10.52202/079017-0951

- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**
   Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?
   Answer: [Yes]
   Justification: the paper specifies all the training and test details.
   Guidelines:
   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.
7. **Experiment Statistical Significance**
   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?
   Answer: [Yes]
   Justification: the results are accompanied by confidence intervals over different independent runs.
   Guidelines:
   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.
8. **Experiments Compute Resources**
   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [No]
Justification: we have provided theoretical complexity of our method. Hence, the execution time can be easily deduced from the performance parameters of devices.
Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**
Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?
Answer: [Yes]
Justification: we have reviewed the NeurIPS Code of Ethics.
Guidelines:
   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**
Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?
Answer: [Yes]
Justification: we have discussed about the broader impact of our paper.
Guidelines:
   - The answer NA means that there is no societal impact of the work performed.
   - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
   - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
   - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
   - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
   - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**
Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?
Answer: [NA]
Justification: the paper poses no such risks.
Guidelines:
   - The answer NA means that the paper poses no such risks.

https://doi.org/10.52202/079017-0951

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**
    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?
    Answer: [NA]
    Justification: the paper does not use existing assets.
    Guidelines:
    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.
13. **New Assets**
    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?
    Answer: [NA]
    Justification: the paper does not release new assets.
    Guidelines:
    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.
14. **Crowdsourcing and Research with Human Subjects**
    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?
    Answer: [NA]
    Justification: the paper does not involve crowdsourcing nor research with human subjects.
    Guidelines:
    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.

- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: the paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

https://doi.org/10.52202/079017-0951