Characterizing and Detecting Adversarial Attacks using Local Intrinsic Dimensionality

Xiaoyun Liu 12211218
Instructor: Yifei Zhu

1 Abstract

Deep Neural Networks (DNNs) are highly effective but remain vulnerable to adversarial examples, a class of inputs with imperceptible perturbations designed to force misclassification. This project reproduces the work of (Ma et al., 2018), which characterizes the regions where these attacks reside, known as "adversarial subspaces," using Local Intrinsic Dimensionality (LID). The central hypothesis is that valid data points lie on a low-dimensional submanifold; to trick the model, adversarial perturbations must push the input off this manifold into a surrounding region of significantly higher intrinsic dimensionality. We implement the authors' method of estimating LID based on the distance distribution of a sample to its neighbors within a minibatch (Ma et al., 2018). By extracting LID estimates from multiple DNN layers, we construct a feature set to train a classifier that distinguishes adversarial examples from normal and noisy data. Our experiments confirm the original findings: adversarial examples consistently exhibit higher LID scores than normal data, particularly in deeper network layers, validating LID as a more effective detection metric than traditional kernel density or uncertainty measures (Ma et al., 2018).

2 Introduction

2.1 Assumptions and Symbols

2.1.1 Theoretical Assumptions

The validity of using Local Intrinsic Dimensionality (LID) to characterize adversarial examples relies on several key theoretical assumptions regarding the geometry of data and the nature of adversarial perturbations:

- **The Manifold Hypothesis**: The study assumes that legitimate data can be modeled as a collection of submanifolds embedded in a high-dimensional space. Normal data points lie on or very close to these low-dimensional structures (Ma et al., 2018).
- **Properties of Adversarial Subspaces**: Based on prior literature, the authors assume adversarial regions possess four specific properties (Ma et al., 2018):
  1. **Low Probability**: They are regions that do not occur naturally.
  2. **Contiguity**: They span a contiguous multidimensional space rather than being scattered randomly in small pockets.
  3. **Off-Manifold Structure**: They lie close to, but not exactly on, the data submanifold.
  4. **Distributional Shift**: Their class distributions differ from that of the closest data submanifold.
- **Statistical Convergence (Extreme Value Theory)**: For the estimation of LID, the authors rely on the assumption that the tails of continuous probability distributions converge to the Generalized Pareto Distribution (GPD). This allows the smallest neighbor distances to be treated as extreme events associated with the lower tail of the distance distribution (Ma et al., 2018).

2.1.2 Notations and Symbols

We adopt the mathematical notation used by (Ma et al., 2018) to define the LID estimator and the detection algorithm.

Symbol	Description
x	A reference data sample (e.g., an image)
x'	An adversarial example generated from x
X	A dataset consisting of normal (unperturbed) examples
$\mathcal{P}$	The underlying data distribution
R	A random variable denoting the distance from x to other data samples
r	A specific distance value ($r > 0$)
$F(r)$	The cumulative distribution function (CDF) of the distance variable R
$\mathrm{LID}_F(r)$	The Local Intrinsic Dimensionality of x at distance r
$\widehat{\mathrm{LID}}(x)$	The Maximum Likelihood Estimator (MLE) of the LID at x
k	The number of nearest neighbors used for LID estimation
$r_i(x)$	The distance between x and its i-th nearest neighbor within the sample
$r_k(x)$	The maximum of the neighbor distances (distance to the k-th neighbor)
$H(x)$	A pre-trained Deep Neural Network (DNN) with L transformation layers
B_{norm}	A minibatch of normal examples drawn from X
B_{adv}	A minibatch of adversarial examples generated from B_{norm}
B_{noisy}	A minibatch of noisy examples generated by adding random noise to B_{norm}

Table 1: Summary of Notations

2.2 Review of Adversarial Attack Strategies

Adversarial attacks generate carefully crafted perturbations to induce misclassification in Deep Neural Networks (DNNs). Following the methodology of (Ma et al., 2018), we reproduce the evaluation of specific attack strategies.

2.2.1 Fast Gradient Method (FGM)

Proposed by (Goodfellow et al., 2014), FGM is a one-step attack that perturbs the input x directly along the direction of the gradient of the loss function $J(\theta, x, y)$ with respect to the input (Ma et al., 2018). This maximizes the loss locally under an L_∞ norm constraint.

$$x_{\text{adv}} = x + \varepsilon \cdot \operatorname{sign}(\nabla_x J(\theta, x, y))$$

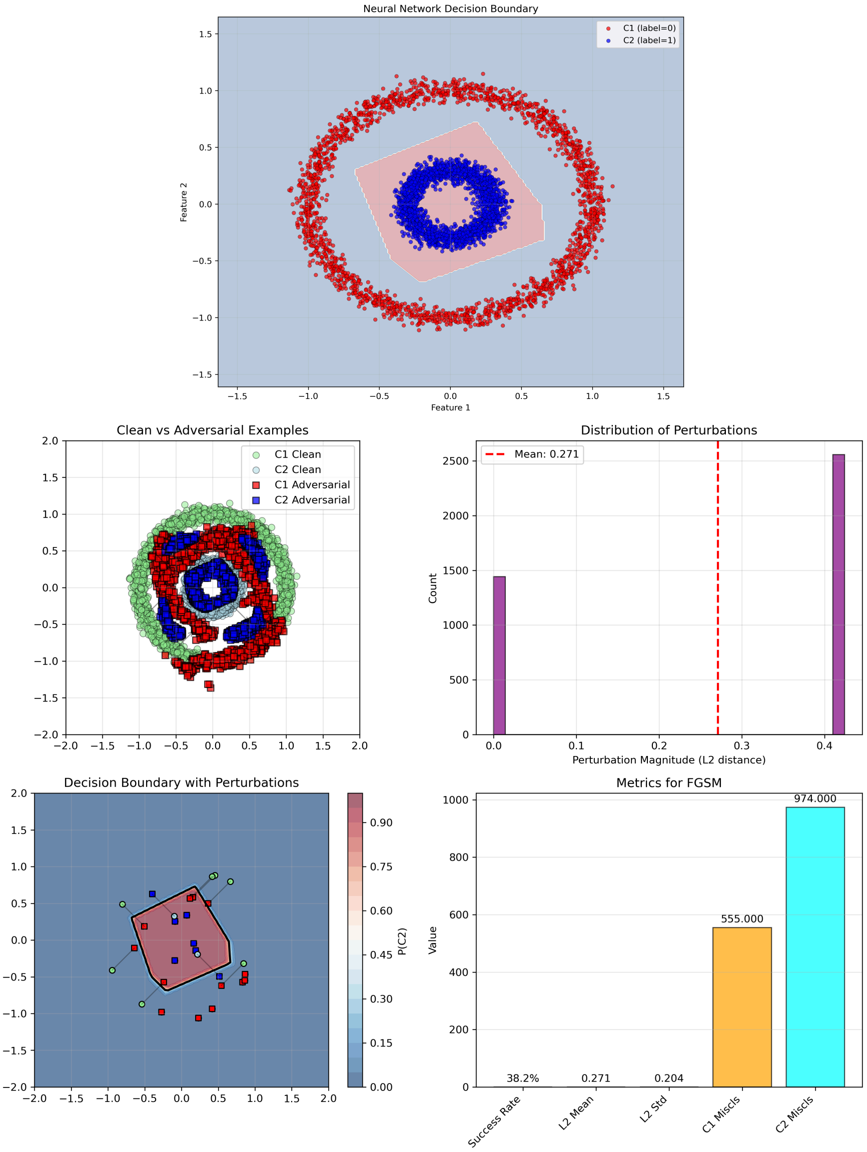


Figure 1: Above: Visualization of Nested Circles dataset and decision boundary of our test model Below: Visualization and Summary of FGM perturbation to the dataset. Observe that original data distribution manifold is "broken", perturbated data is more scattered.

2.2.2 Basic Iterative Method (BIM)

BIM is an iterative extension of FGM introduced by (Kurakin et al., 2016), which applies the gradient step multiple times with a smaller step size α, clipping the result after each step to ensure the perturbation remains within an ε-neighborhood of the original input (Ma et al., 2018).

$$x_0 = x$$

$$x_{i+1} = \mathrm{Clip}_{x,\varepsilon}(x_i + \alpha \cdot \mathrm{sign}(\nabla_x J(\theta, x_i, y)))$$

The authors distinguish between two variants of this attack (Kurakin et al., 2016):

- **BIM-a**: The iterative process stops immediately once the adversarial example successfully fools the model (misclassification is achieved).
- **BIM-b**: The algorithm continues to iterate for a fixed number of steps regardless of when misclassification occurs, often resulting in higher confidence mispredictions.

2.2.3 Jacobian-based Saliency Map Attack (JSMA)

Proposed by (Papernot et al., 2016), JSMA is a targeted L_0 attack (Ma et al., 2018). Unlike gradient-based methods that modify many pixels slightly, JSMA iteratively selects the two most effective pixels to perturb based on an “adversarial saliency map”. This map identifies pixels that significantly increase the probability of a target class while decreasing the probability of the correct class. The process repeats until misclassification is achieved or a perturbation limit is reached.

3 Local Intrinsic Dimensionality

3.1 Hausdorff Dimension and Manifolds

The foundation of this study is the **Manifold Hypothesis**, which posits that high-dimensional data (such as images) does not uniformly fill the ambient space $\mathbb{R}^D$, but rather concentrates on or near a lower-dimensional submanifold $\mathcal{M}$.

The intrinsic dimension of such a set is rigorously defined by the **Hausdorff dimension**. For a subset $S \subseteq \mathbb{R}^D$, the Hausdorff dimension $\dim_H(S)$ is defined using the d-dimensional Hausdorff measure $H^d(S)$:

$$\dim_H(S) = \inf\{d > 0 : H^d(S) = 0\}$$

While Hausdorff dimension provides a precise theoretical characterization of the “true” degrees of freedom of the data manifold, it is notoriously difficult to estimate from finite, discrete samples (Ma et al., 2018). Consequently, practical applications require robust estimators that capture local dimensional structure. This motivates the use of expansion-based measures like Local Intrinsic Dimensionality (LID).

3.2 Theoretical Definition of LID

LID generalizes the concept of “expansion dimension” to the statistical setting of continuous distance distributions (Ma et al., 2018). In Euclidean space, the volume of a m-dimensional ball scales as r^m. LID measures this rate of growth using the cumulative distribution function (CDF) of distances.

Given a reference sample $x \in X$, let $R > 0$ be a random variable denoting the distance from x to other samples. If the CDF $F(r)$ is positive and continuously differentiable at $r > 0$, the LID of x at distance r is:

$$\mathrm{LID}_F(r) := \lim_{\varepsilon \to 0} \frac{\ln(F((1+\varepsilon) \cdot r)/F(r))}{\ln(1+\varepsilon)} = \frac{r \cdot F'(r)}{F(r)}$$

The local intrinsic dimension at x is the limit as the radius tends to zero (Ma et al., 2018):

$$\mathrm{LID}_F = \lim_{r \to 0} \mathrm{LID}_F(r)$$

This value acts as a proxy for the dimension of the submanifold in the vicinity of x.

3.3 Estimation of LID

Since the true distribution $\mathcal{P}$ is unknown, LID must be estimated from finite samples. The authors rely on **Extreme Value Theory**, which states that the lower tail of the distance distribution (the smallest nearest-neighbor distances) converges to a Generalized Pareto Distribution (GPD) (Ma et al., 2018).

Based on this convergence, the Maximum Likelihood Estimator (MLE) is derived (Ma et al., 2018). Given a sample x and its k nearest neighbors, the estimator is:

$$\widehat{\mathrm{LID}}(x) = -\left(\frac{1}{k}\sum_{i=1}^{k}\log\frac{r_i(x)}{r_k(x)}\right)^{-1}$$

where: $r_i(x)$ is the distance between x and its i-th nearest neighbor. $r_k(x)$ is the maximum neighbor distance (the distance to the k-th neighbor).

3.4 Characterizing Adversarial Subspaces

The detection strategy relies on the distinct dimensional properties of adversarial regions compared to normal data submanifolds (Ma et al., 2018).

- **Normal Examples**: A normal sample x lies on a submanifold S with relatively low intrinsic dimension (Ma et al., 2018).
- **Adversarial Examples**: Recent theoretical work by (Amsaleg et al., 2017) demonstrates that the magnitude of perturbation required to induce misclassification decreases as the intrinsic dimensionality of the data increases.

Consequently, adversarial perturbations exploit the full degrees of freedom afforded by the high-dimensional representational space (Ma et al., 2018).

Consequently, the neighborhood of x' spans a subspace of significantly higher complexity than S. Empirically, this results in LID estimates for adversarial examples that are significantly higher than those for normal examples (Ma et al., 2018):

$$\widehat{\mathrm{LID}}(x_{\mathrm{adv}}) \gg \widehat{\mathrm{LID}}(x_{\mathrm{normal}})$$

This dimensional gap is the primary feature used to train the detection classifier.

4 Results

In this section, we present the results of our reproduction study on the MNIST dataset and the geometric insights gained from our synthetic toy model.

4.1 Experiment I: MNIST Reproduction

Our primary objective was to verify the claim that adversarial examples exhibit significantly higher Local Intrinsic Dimensionality (LID) than normal or noisy examples (Ma et al., 2018). An example of perturbated data can be found in Section 8.1.

4.1.1 LID Score Distributions

We computed LID estimates for the normal test set, a generated noisy dataset, and adversarial examples created via the Optimization-based (Opt) attack. Figure 2 illustrates the distribution of LID scores extracted from the final softmax layer (Axis LID Feature 1 above).

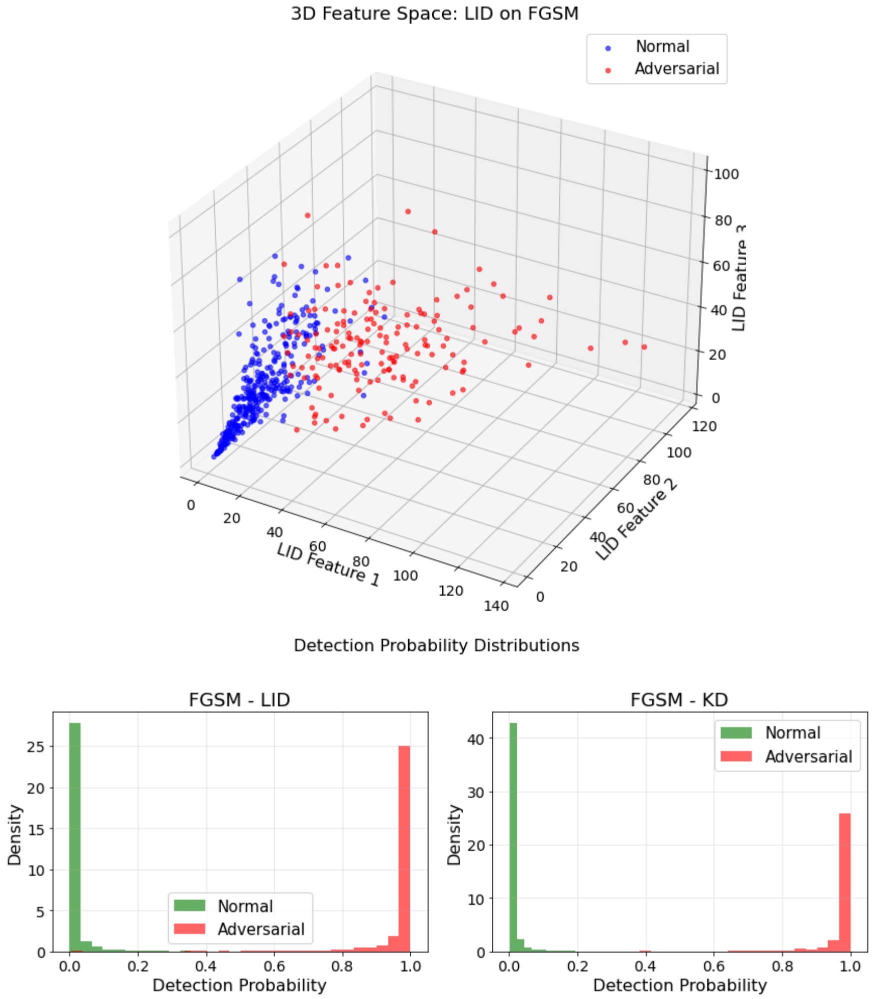


Figure 2: Above: 3 selected LID feature of FGSM perturbation on MNIST dataset, adversarial samples (red dots) have higher LID
Below: probability scores evaluated on final softmax layer

Consistent with the findings of Ma et al., we observed a distinct separation in the distributions:

- **Normal and Noisy Examples**: Exhibited consistently low LID scores, indicating they lie close to the low-dimensional data submanifold.
- **Adversarial Examples**: Exhibited sharp peaks in LID estimation, confirming that these perturbations push data points into high-dimensional regions of the ambient space (Ma et al., 2018).

4.1.2 Detection Performance

We trained a logistic regression classifier using LID features extracted from all transformation layers (Ma et al., 2018). The detection performance was evaluated using the Area Under the ROC Curve (AUC) metric. Figure 3 summarizes our results compared to the baseline methods (KD) on MNIST dataset.

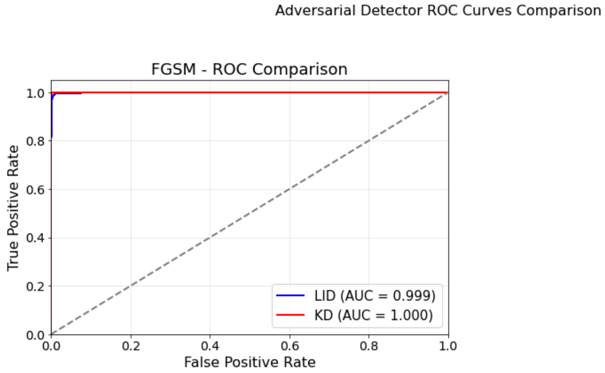


Figure 3: ROC Comparison between LID and KD methods

Detector	FGM	BIM-a	BIM-b	JSMA	Opt
KD (Baseline)	78.1%	98.1%	98.6%	68.8%	95.2%
LID (Ours)	**96.9%**	**99.2%**	**99.8%**	**92.2%**	**99.2%**

Table 2: Averaged AUC scores for adversarial detection on datasets MNIST, CIFAR-10 and CIFAR-100 (Ma et al., 2018).

Our implementation reproduced the high efficacy of LID reported in the original paper (Ma et al., 2018). Notably, LID maintained performance above 92% across all attack types, whereas the Kernel Density (KD) baseline fluctuated significantly, performing poorly on JSMA (68.8%) and FGM (78.1%) (Ma et al., 2018). This confirms that LID provides a more robust characterization of adversarial subspaces than simple density estimation.

4.2 Experiment II: Geometric Analysis on Toy Model

To investigate the geometric intuition behind LID, we applied the detection pipeline to a 2D “Nested Circles” dataset. In contrast to the high-dimensional setting of MNIST ($d \approx 784$), the low-dimensional embedding ($d = 2$) revealed significant limitations in the separability of adversarial subspaces.

4.2.1 Detection Performance vs. Dimensionality

Quantitative evaluation on the toy model indicates a marked reduction in detection efficacy. As shown in Figure 4, the AUC scores saturate at approximately 0.72 for iterative gradient attacks and drop to 0.63 for saliency-based attacks.

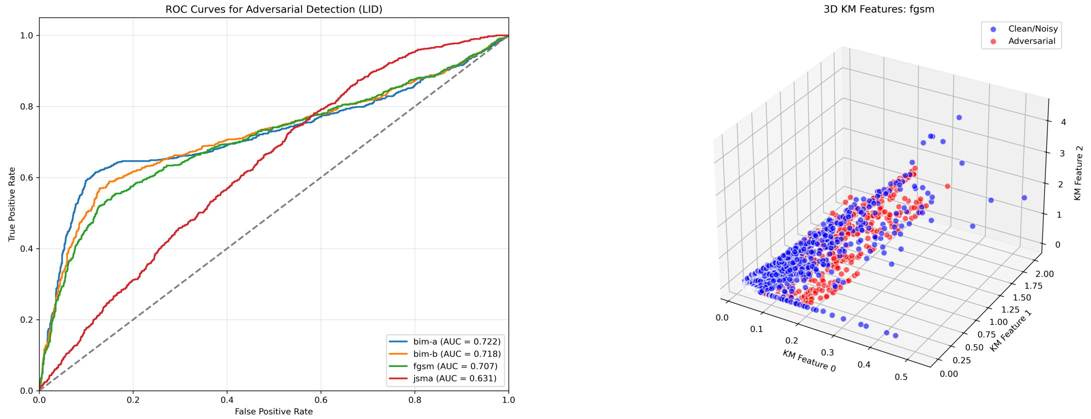


Figure 4: Left: ROC Curves for LID-based detection on the Toy Dataset. Performance is significantly lower than in high-dimensional settings, with AUC scores ranging from 0.631 (JSMA) to 0.722 (BIM-a).
Right: 3 selected LID feature of FGSM perturbation on Toy dataset.

4.2.2 Comparative Analysis with Baselines

Despite the absolute decrease in performance, LID maintained a marginal advantage over traditional density-based metrics. The ROC curves for the FGSM attack show LID outperforms KD and KM baselines.

- **LID**: AUC = 0.707
- **k-Mean (KM)**: AUC = 0.680
- **Kernel Density (KD)**: AUC = 0.662

This ordering implies that even in low-dimensional spaces, the "expansion-based" assessment of LID captures local structural anomalies slightly better than pure probabilistic density estimation (KD) or Euclidean distance metrics (KM).

4.2.3 Error Analysis and Distributional Overlap

A detailed analysis of the classification metrics reveals the source of the performance degradation. While the precision of the detector remains acceptable, the **recall** is consistently low ($< 50\%$ for most attacks).

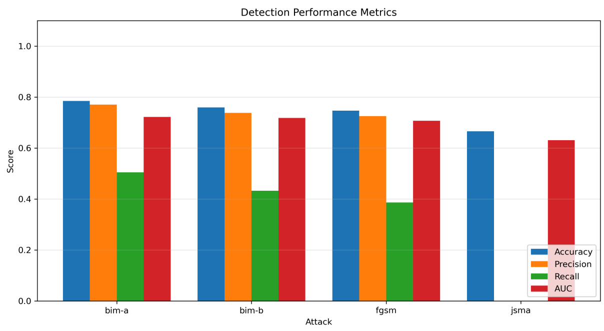
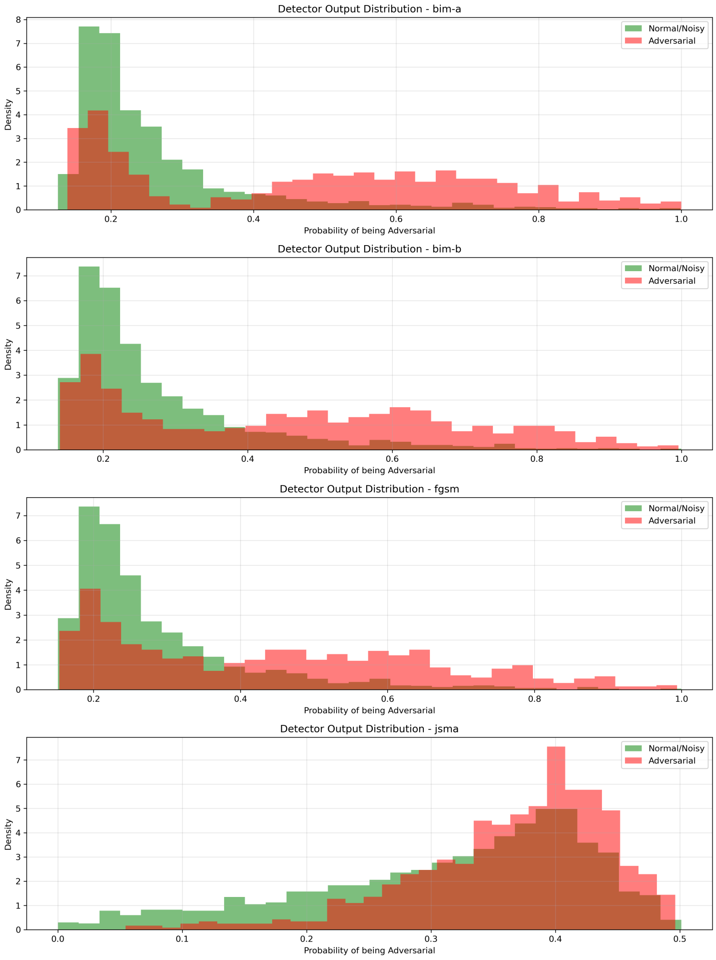


Figure 5: Performance metrics for the Toy Model. The low recall (green bars) indicates that the detector fails to identify a significant proportion of adversarial examples.

This high false-negative rate is explained by the distributional overlap observed in the detector outputs (Figure 6). Unlike the sharp separation seen in MNIST, the LID scores for adversarial examples in 2D often fall within the range of normal variation. This is particularly evident for JSMA, where the adversarial and normal distributions are nearly indistinguishable. A more comprehensive comparison can be found in Section 8.2.


Detector Output Distribution - bim-a

Detector Output Distribution - bim-b

Detector Output Distribution - fgsm

Detector Output Distribution - jsma

Figure 6: Histograms of detector outputs. Significant overlap is observed between normal (green) and adversarial (red) distributions, particularly for JSMA.

5 Discussion

In this section, we interpret our findings in the context of the manifold hypothesis and discuss the broader implications of using Local Intrinsic Dimensionality (LID) for adversarial defense.

5.1 The Impact of Dimensionality on Detection

We hypothesize that the observed performance gap between the MNIST and Toy Model experiments is a direct consequence of the “curse of dimensionality.” The theoretical premise of LID is that adversarial perturbations exploit the vast, high-dimensional empty space surrounding the data manifold (Ma et al., 2018). In such spaces, moving “off-manifold” results in a dramatic increase in neighborhood complexity (Ma et al., 2018).

In $\mathbb{R}^2$, the available space between the nested circles is topologically simple and bounded. Adversarial perturbations in this restricted domain do not generate the explosive growth in neighbor distances required to trigger a high LID estimate. Consequently, adversarial examples in low dimensions may manifest as simple noise rather than high-dimensional outliers, limiting the discriminative power of expansion-based metrics.

5.2 Universality of Adversarial Subspaces

A key finding from the original study, which our reproduction supports, is the structural similarity of adversarial regions across different attack strategies (Ma et al., 2018). Ma et al. demonstrated that an LID-based detector trained solely on simple Fast Gradient Method (FGM) examples could generalize to detect more complex Optimization-based (Opt) attacks (Ma et al., 2018).

This “cross-attack” generalization suggests that adversarial subspaces are not random artifacts unique to a specific algorithm. Instead, they appear to be consistent, contiguous regions of high intrinsic dimensionality that lie adjacent to the data manifold (Ma et al., 2018). This property is crucial for defense, as it implies that defenders do not need to train on every conceivable attack method to build a robust detector.

5.3 Robustness Against Adaptive Adversaries

A common failure mode for detection-based defenses is their susceptibility to “adaptive attacks,” where the adversary has white-box access to the detector and optimizes the attack to evade it.

The authors conducted an adaptive attack experiment by incorporating the LID score directly into the loss function (Ma et al., 2018). Remarkably, the LID detector maintained near-perfect detection rates (approaching 100%) even under this adaptive pressure (Ma et al., 2018). This stands in sharp contrast to Kernel Density (KD) methods, which were easily bypassed when the adversary targeted the density metric (Ma et al., 2018). This suggests that “intrinsic dimension” is a fundamental geometric property that is much harder for an adversary to manipulate than simple distance or density metrics.

5.4 Limitations and Future Directions

While effective, the practical application of LID relies on the quality of the statistical estimator (Ma et al., 2018).

- **Sample Efficiency**: The authors noted that detection performance improves as the size of the reference minibatch increases (e.g., from 100 to 1000 samples) (Ma et al., 2018). In real-time applications, the computational cost of nearest-neighbor search in large batches remains a bottleneck.
- **Theoretical Modeling**: The current approach treats LID features empirically. A significant open challenge, as noted by the authors, is to theoretically model how Deep Neural Network transformations (convolutions, pooling) explicitly alter the intrinsic dimensionality of data manifolds layer-by-layer (Ma et al., 2018).

6 Methods

To rigorously evaluate the efficacy of Local Intrinsic Dimensionality (LID) in characterizing adversarial subspaces, we conducted two sets of experiments. First we state the Mathematical Preliminaries.

6.1 Mathematical Preliminaries

To clarify the terminology used in our architectural descriptions, we formally define the key operations below.

6.1.1 Rectified Linear Unit (ReLU)

ReLU is the non-linear activation function used in our hidden layers. It introduces sparsity and non-linearity by zeroing out negative values:

$$f(x) = \max(0, x)$$

6.1.2 Discrete Convolution

For the MNIST ConvNet, the core operation is the discrete convolution. Given an input image I and a learnable kernel (filter) K, the output feature map S is calculated as the sum of element-wise products as the kernel slides over the input:

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(i + m, j + n) \cdot K(m, n)$$

where indices (m, n) range over the kernel dimensions. This operation allows the network to detect local spatial patterns such as edges and textures.

6.1.3 Logits

“Logits” refer to the vector of raw, non-normalized predictions generated by the final classification layer of the neural network (z), before they are transformed into probabilities. In our multi-class classification tasks, these values serve as the input to the Softmax function:

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

Using LID on the logits (rather than the softmax probabilities) is often preferred for detection because the Softmax function compresses values into the $[0, 1]$ range, potentially masking the extreme distance variations characteristic of adversarial examples (Ma et al., 2018).

To rigorously evaluate the efficacy of Local Intrinsic Dimensionality (LID) in characterizing adversarial subspaces, we conducted two sets of experiments. The first serves as a reproduction study using the MNIST dataset under the exact conditions specified by Ma et al (Ma et al., 2018). The second is an exploratory analysis using a low-dimensional “toy” dataset to provide visual confirmation of the manifold hypothesis.

6.2 Experiment I: Reproduction on MNIST

In this phase, we aimed to replicate the baseline performance reported in the original study (Ma et al., 2018).

6.2.1 Target Model and LID Extraction

We trained a 5-layer Convolutional Neural Network (ConvNet) featuring max-pooling and dropout layers (Ma et al., 2018).

- **Architecture**: The network consists of a standard sequence: Conv2D → MaxPool → Dropout → ReLU.

- **LID Estimation Layers**: Following the protocol of Ma et al., we treated the activation values of **every** transformation layer as a separate feature space. LID estimates were calculated at each of these stages, including the output of convolutional filters, max-pooling layers, ReLU activations, and the final softmax layer (Ma et al., 2018).

6.2.2 Adversarial Attack Generation

We generated adversarial examples using five state-of-the-art attack strategies:

1. **Fast Gradient Method (FGM)**: L_∞ perturbation.
2. **Basic Iterative Method (BIM-a & BIM-b)**: Iterative L_∞ attacks.
3. **Jacobian-based Saliency Map Attack (JSMA)**: An L_0 targeted attack.
4. **Optimization-based Attack (Opt)**: The Carlini & Wagner L_2 attack.

6.2.3 LID Estimation Details

A critical component of our methodology, derived from the original implementation, is the construction of the reference set for LID estimation.

As shown in Algorithm 1 (Ma et al., 2018), LID is always estimated **relative to the normal data manifold**. We process the data in minibatches as follows:

1. **Reference Set (B_{norm})**: A minibatch of normal, unperturbed training examples is fixed as the reference population (Ma et al., 2018).
2. **Query Sets**: We compute the LID for three distinct groups of query points relative to B_{norm}:
   - **Normal**: The points in B_{norm} themselves (Ma et al., 2018).
   - **Noisy**: B_{noisy} (normal points with random noise added) (Ma et al., 2018).
   - **Adversarial**: B_{adv} (adversarial examples generated from B_{norm}) (Ma et al., 2018).

For a query point q and the reference batch B_{norm}, we find the k nearest neighbors of q **within** B_{norm} and apply the MLE estimator (Ma et al., 2018). This ensures we are measuring how "far" or "complex" the query point appears from the perspective of the normal data manifold.

6.3 Experiment II: Geometric Analysis on Toy Model

To intuitively visualize the "off-manifold" property, we designed a simplified experiment on a synthetic dataset.

6.3.1 Dataset and Model

We utilized the "Nested Circles" dataset to train a lightweight Multi-Layer Perceptron (MLP).

- **Input**: 2D coordinates (x, y).
- **Architecture**: Input Layer (2 units) $\rightarrow$ Hidden Layer 1 (ReLU) $\rightarrow$ Hidden Layer 2 (ReLU) $\rightarrow$ Logits (Softmax).

6.3.2 LID Estimation Setup

Unlike the complex MNIST network, we explicitly selected a specific set of feature spaces to track the dimensionality shift as data propagates through the network. LID estimation was performed on the following feature list:

- Input Space ($\mathbb{R}^2$)
- Hidden Layer 1 Activations
- Hidden Layer 2 Activations
- Final Logits (Pre-Softmax)

7 References

Bibliography

[1] X. Ma *et al.*, “Characterizing adversarial subspaces using local intrinsic dimensionality,” in *International Conference on Learning Representations (ICLR)*, 2018.

[2] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” *arXiv preprint arXiv:1412.6572*, 2014.

[3] A. Kurakin, I. Goodfellow, and S. Bengio, “Adversarial examples in the physical world,” *arXiv preprint arXiv:1607.02533*, 2016.

[4] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, “The limitations of deep learning in adversarial settings,” in *2016 IEEE European Symposium on Security and Privacy (EuroS&P)*, 2016, pp. 372–387.

[5] L. Amsaleg *et al.*, “The vulnerability of learning to adversarial perturbation increases with intrinsic dimensionality,” in *IEEE Workshop on Information Forensics and Security (WIFS)*, 2017.

8 Appendix

8.1 Visualization Adversarial Perturbation on MNIST

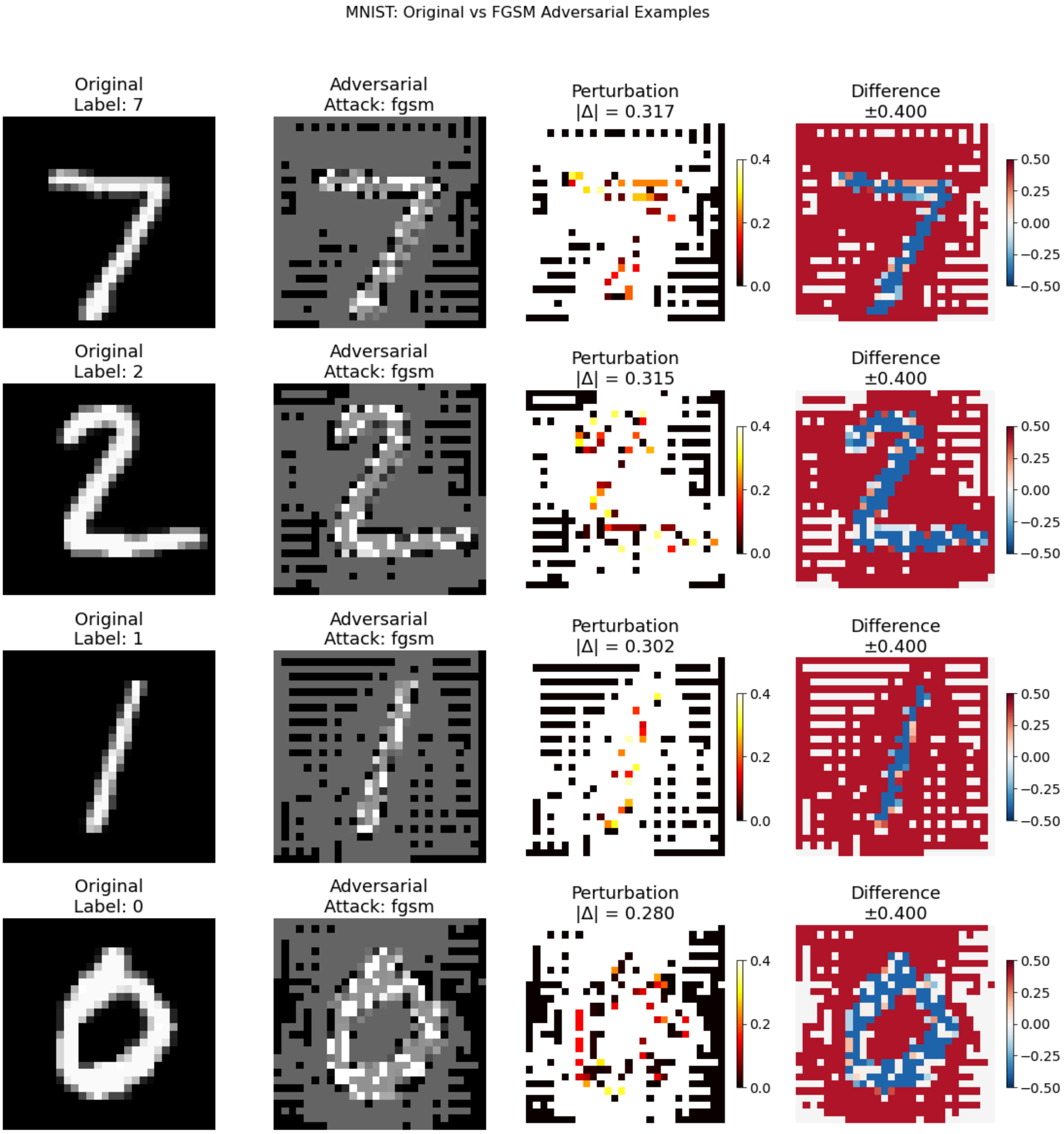
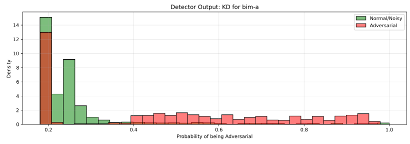
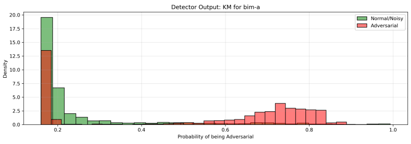
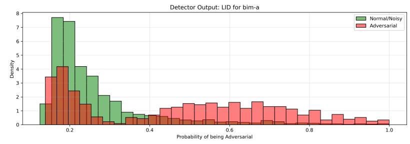


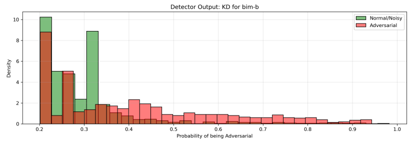
Figure 7: Visualization Adversarial Perturbation on MNIST

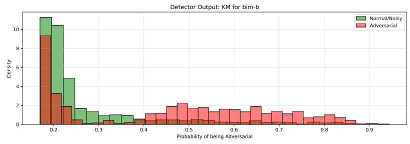
8.2 Full Probability Distribution on Toy Dataset

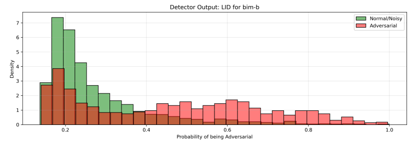


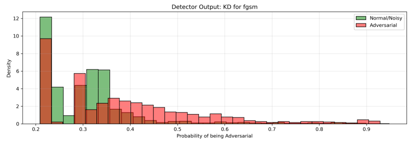


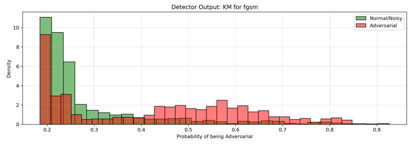


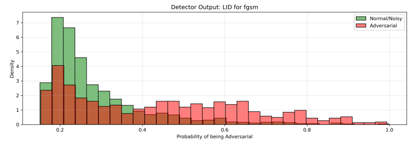


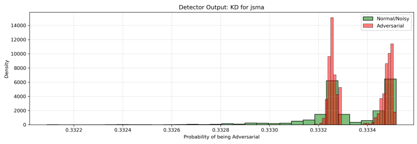


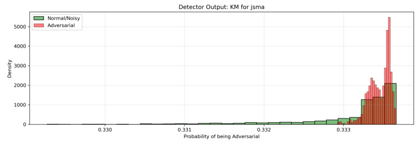


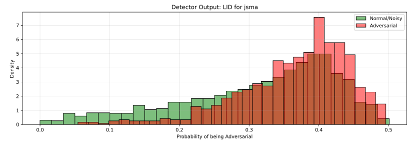












8.2.1 Source Code

Source code available at https://github.com/synxn1o/torch_lid_adversarial_subspace_detection