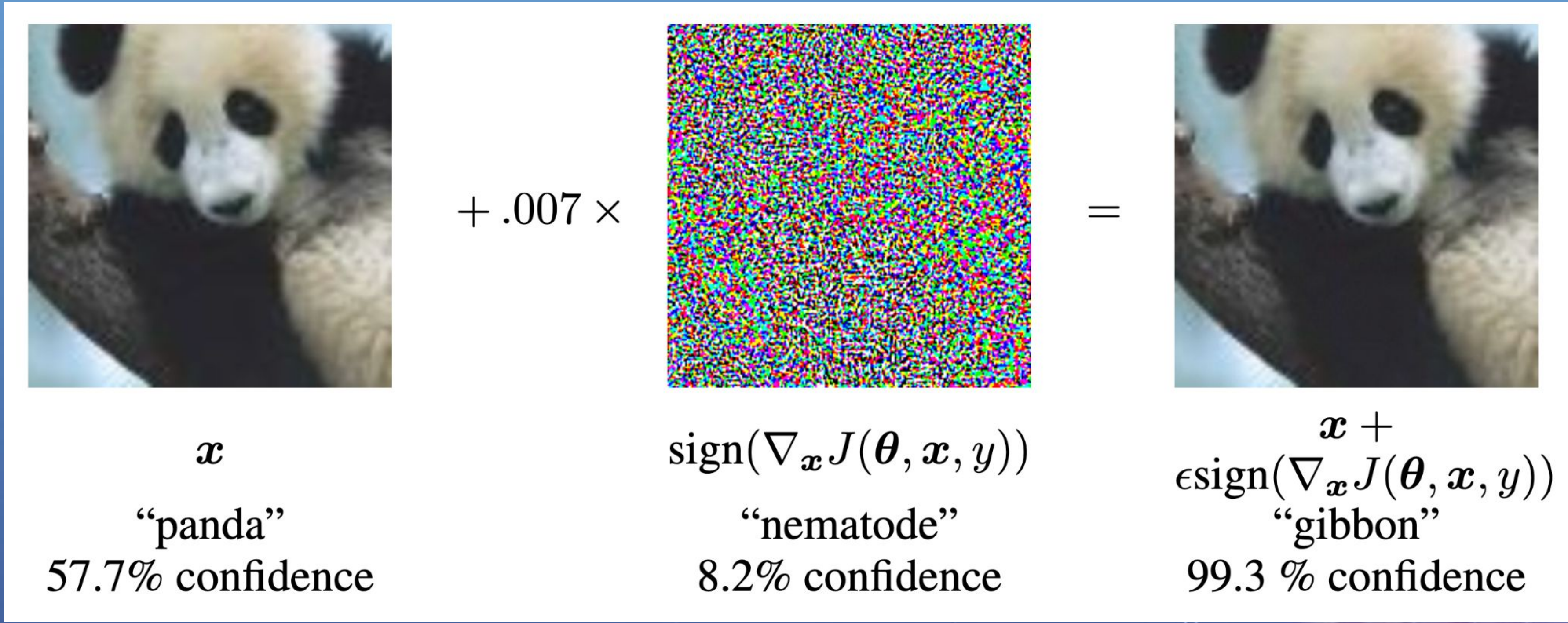
Adversarial Robustness and the Evolution of Latent Geometries in Neural Networks

Brandon Ismalej, Department of Computer Science - CSU Northridge[1]
Cal-Bridge, Department of Mathematics[2] & Halicioglu Data Science Institute[3] - UC San Diego, Dr. Alex Cloninger[2,3] & Dr. Gal Mishne[3]

Cal-Bridge Research Symposium | October 4, 2025

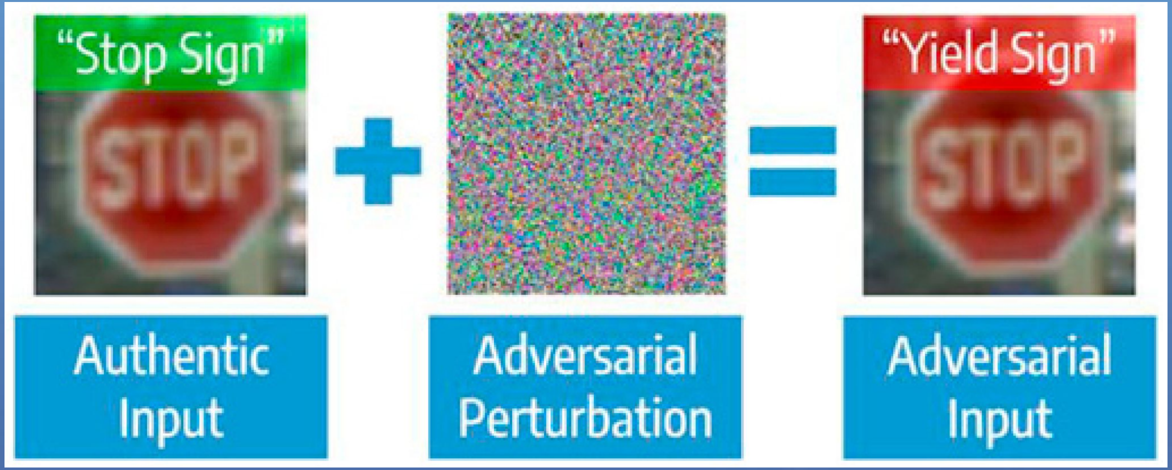
Introduction

Introduction



Goodfellow et al., 2015

Introduction



Zhai et al., 2020

Eykholt et al., 2018

CAL-BRIDGE

Motivation

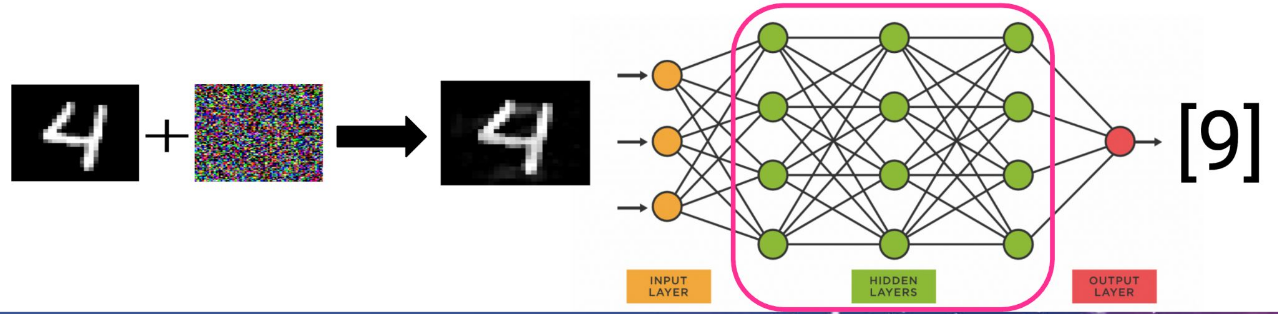
CAL-BRIDGE

Why Accuracy is not Enough

- Small, invisible changes can trick AI into confidently making wrong decisions.
- Even when we train AI to defend itself, high accuracy can hide deeper weaknesses.

The Central Question

Is accuracy a reliable measure of adversarial robustness or do the hidden geometries of the network contradict accuracy?



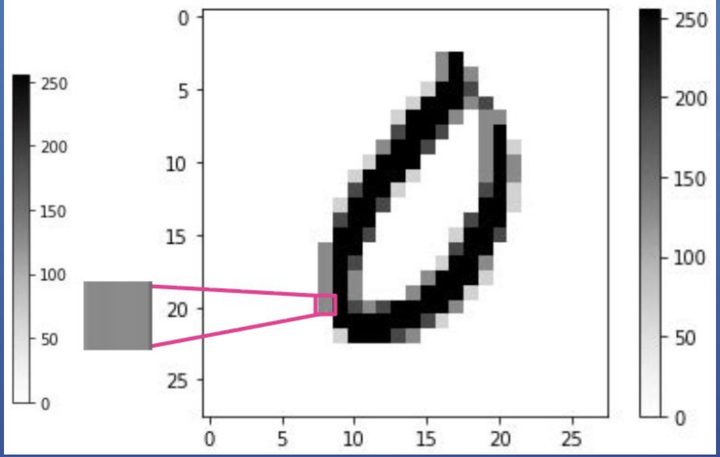
CAL-BRIDGE

Background

CAL-BRIDGE

Adversarial Attacks: Small Changes, Big Mistakes

- Adversarial attacks are small, carefully crafted changed to inputs.
- We can alter each pixel by a small, calculated amount.
- These attacks reveal how brittle and blind AI systems can be



Fast Gradient Sign Method (FGSM)

$$x_{\text{adv}} = x + \epsilon \cdot \text{sign}\left(\nabla_x J(\theta, x, y)\right)$$

L_2-Bounded

$$x_{\text{adv}} = x + \epsilon \cdot \frac{\nabla_x J(\theta,x,y)}{\|\nabla_x J(\theta,x,y)\|_2}$$

L_∞-Bounded

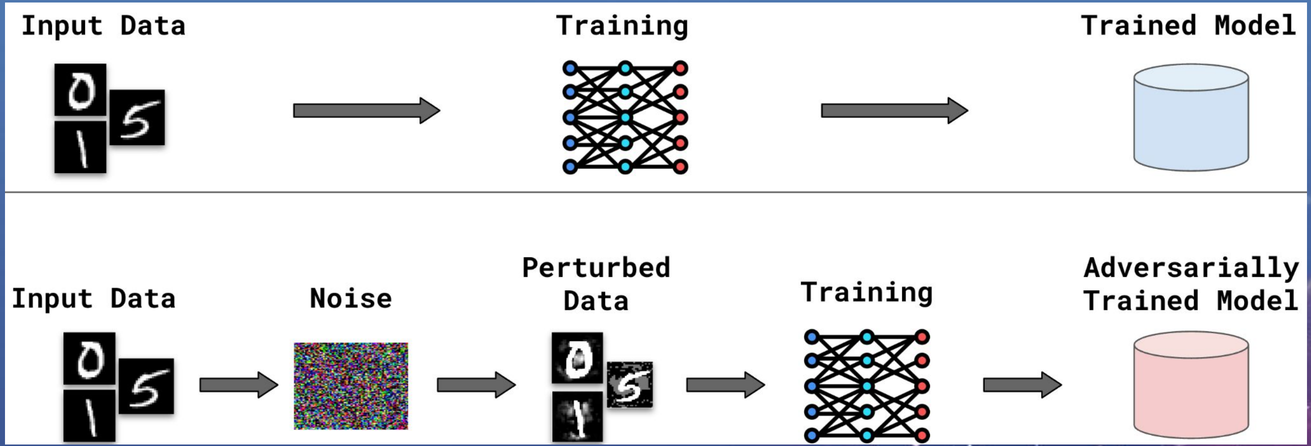
$$x_{\text{adv}} = x + \epsilon \cdot \frac{\nabla_x J(\theta,x,y)}{\|\nabla_x J(\theta,x,y)\|_\infty}$$

Key Insight

ϵ controls attack strength:
Larger ϵ = stronger but more visible attacks
Smaller ϵ = weaker but stealthier attacks

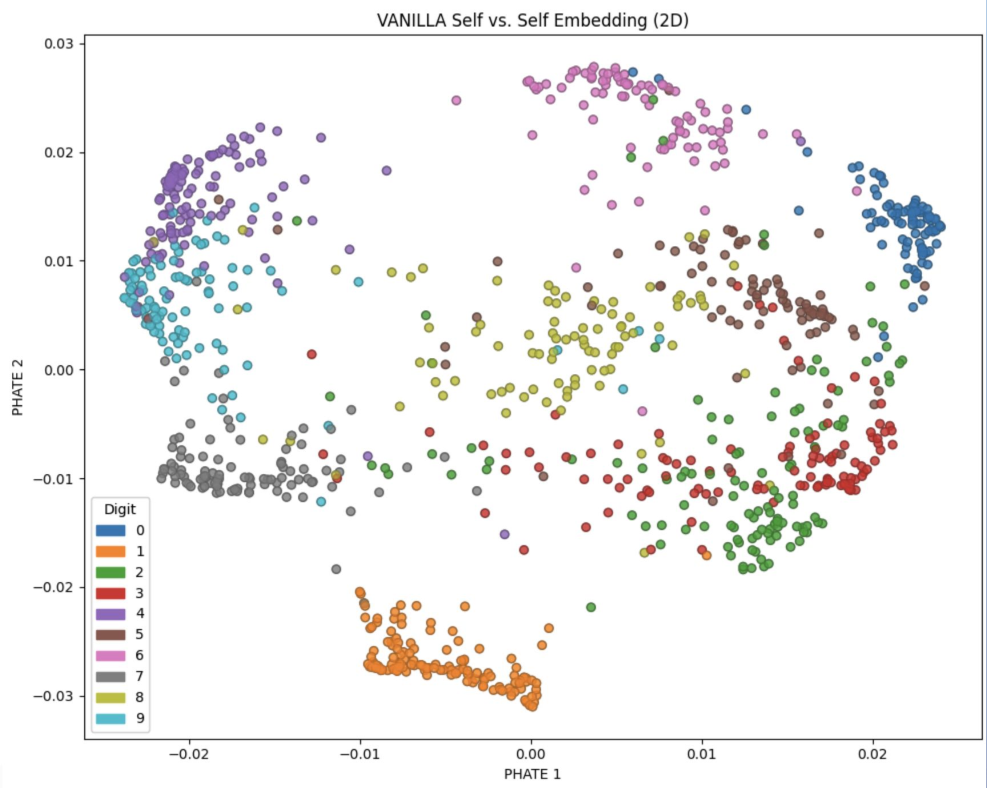
Adversarial Training and Robustness

- Adversarial training teaches models to defend exposing them to attacks during training.
- This can improve accuracy on attacked data, but doesn't guarantee real robustness
- Accuracy may stay high even when model is fragile inside



CAL-BRIDGE

M-PHATE & Why Geometry Matters



- Neural networks transform inputs into internal representations - their "geometry".
- We use a tool called M-PHATE to visualize these hidden geometries (Gigante et al., 2019)
- Helps us see how much network's understanding shifts under attack or defense

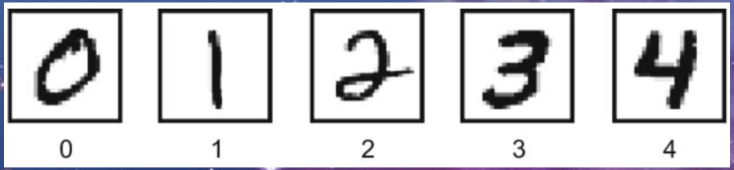
CAL-BRIDGE

Methods

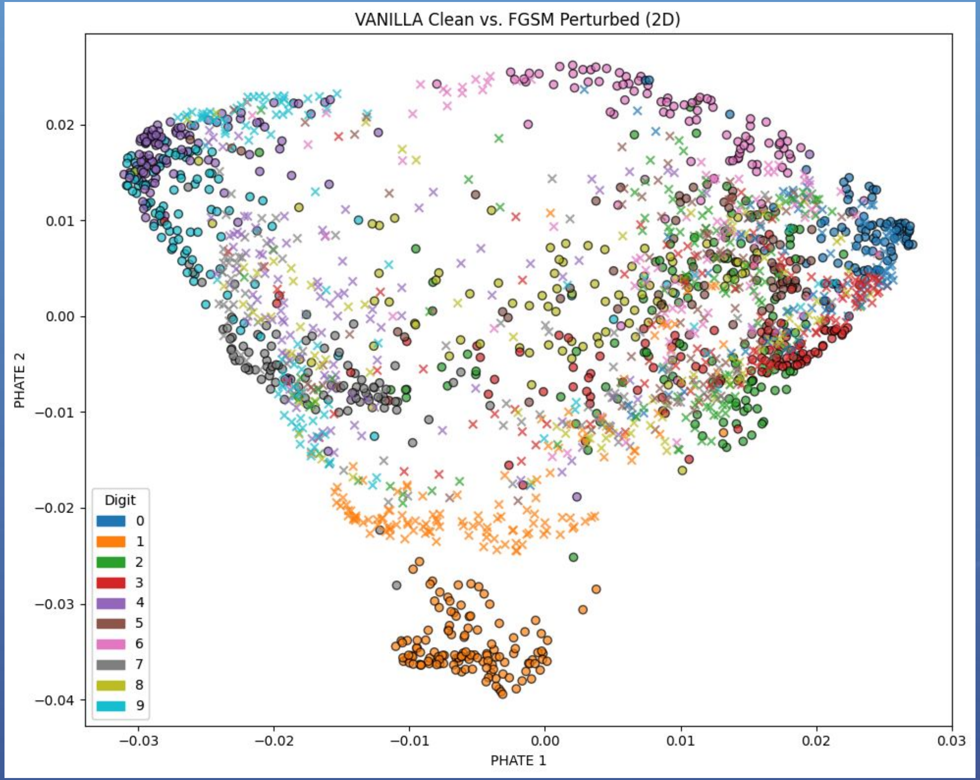
Training the Networks

- 5 sets of neural networks trained on MNIST Handwritten Digits:
- 4 networks/set
  - Standard (Baseline, no adv. attack)
  - FGSM attack-trained
  - L2 attack-trained
  - L∞ attack-trained
- Each set trained on varying levels of attack strength.
- We record their accuracy and clean and perturbed in puts, and their Δ accuracy

Extreme Low	FGSM: ε = 2/255 L_2: ε = 0.125 L_∞ ε = 2/255
Low	FGSM: ε = 4/255 L_2: ε = 0.25 L_∞ ε = 4/255
Medium	FGSM: ε = 8/255 L_2: ε = 0.5 L_∞ ε = 8/255
High	FGSM: ε = 16/255 L_2: ε = 1.0 L_∞ ε = 16/255
Extreme High	FGSM: ε = 32/255 L_2: ε = 2.0 L_∞ ε = 32/255



Looking inside the Network: M-PHATE

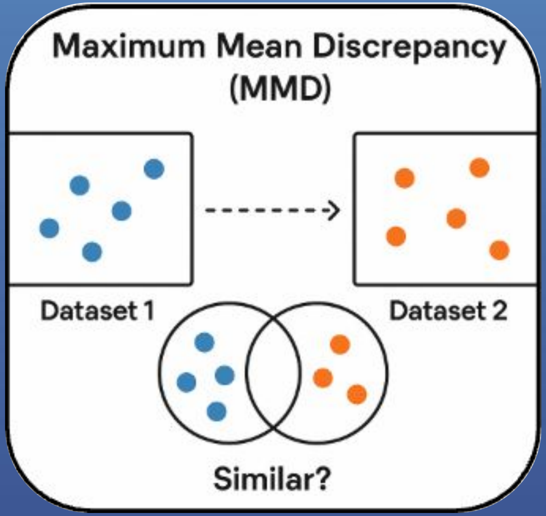


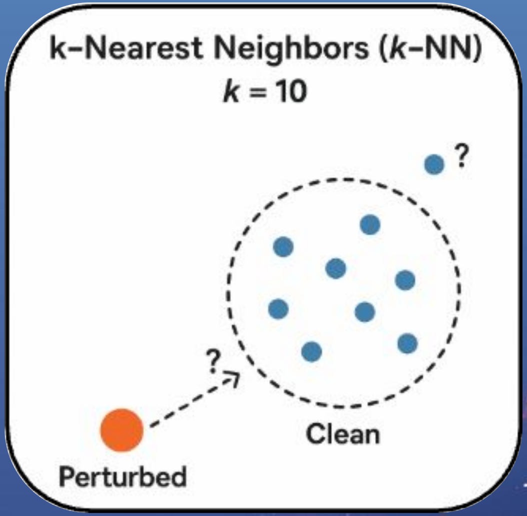
- We extracted each network's internal representation, its final-layer geometry.
- We visualized how clean and perturbed inputs are positioned in this space.
- **If the points are far apart, the network sees them as very different; if close, they are similar**

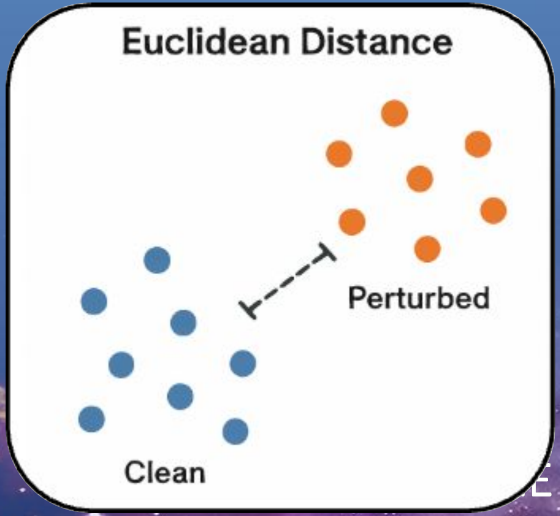
CAL-BRIDGE

Looking Inside: Quantifying Changes

- MMD + Hypothesis Test: Global difference in clean vs perturbed
- KNN Overlap: Proportion of perturbed point within clean k-NN
- Euclidean Distance: Per-point clean-perturbed movement



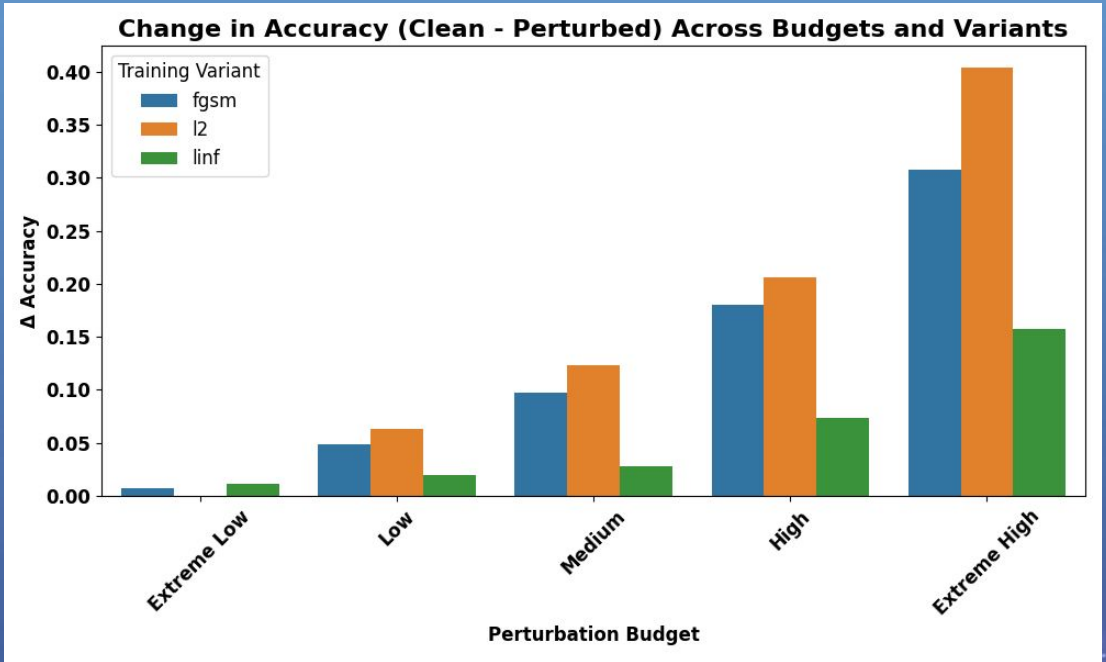




Results

Results: Accuracy and Perturbation

- L∞ appears to be the most robust - low accuracy drop.
- L2 appears to be the least robust - high accuracy drop.
- **Smaller drops in accuracy suggest more robustness, but is that the full story?**



CAL-BRIDGE

Results: Geometry Contradicts Accuracy

Metric	What it Measured	Key Finding
Δ Accuracy	Change from clean $\rightarrow$ perturbed performance	L_∞ often showed smallest drop
MMD p-value	Global geometry similarity in PHATE space	Sometimes suggested similarity despite large geometric shifts
kNN Recovery	Proportion of perturbed points near their clean neighbor	"Robust" models could still have poor neighbor recovery
Euclidean Distance	Median clean–perturbed separation (normalized)	L_∞ sometimes had largest separation despite high accuracy

- Accuracy alone can misrepresent robustness; small drops can hide large geometric changes.
- Geometry-based metrics expose vulnerabilities not seen through accuracy.
- Combining accuracy and geometry can offer a fuller view of robustness.

CAL-BRIDGE

Conclusions & Future Work

Conclusions & Future Work

Conclusions:

- Accuracy alone is not a reliable measure of adversarial robustness.
- Geometry-based metrics (MMD, kNN recovery, euclidean distance) reveal hidden vulnerabilities.
- Some models (e.g. $L\infty$) appear robust by accuracy but show large geometric shifts.

Future Work:

- Extend geometry analysis to other architectures and datasets.
- Further examination of why this contradiction occurs.
- Examine if we can see "when" adversarial robustness occurs, or geometries shift drastically.

CAL-BRIDGE

Thank you!

Questions?

brandon.ismalej.671@my.csun.edu

Acknowledgements:

- I would like to thank my PI's: Alex Cloninger and Gal Mishne
- My peers for their encouragement and camaraderie

References

- Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T., and Song, D. (2018). Robust physical-world attacks on deep learning models. (arXiv:1707.08945).
- Gigante, S., Charles, A. S., Krishnaswamy, S., and Mishne, G. (2019). Visualizing the phate of neural networks. In Advances in Neural Information Processing Systems, volume 32. Curran Associates, Inc.
- Goodfellow, I., Shlens, J., and Szegedy, C. (2015). Explaining and harnessing adversarial examples. In International Conference on Learning Representations.
- Zhai, J., Shen, W., Singh, I., Wanyama, T., and Gao, Z. (2020). A review of the evolution of deep learning architectures and comparison of their performances for histopathologic cancer detection. Procedia Manufacturing, 46:683–689.

CAL-BRIDGE

Appendix

Network Architecture & Training Configurations

Experimental Setup

Network Architecture	Training Configurations
• **Type:** Multilayer Perceptron (MLP) • **Input:** 28×28 (flattened image) • **Hidden Layers:** $128 \rightarrow 64$ neurons • **Output:** 10 classes • **Loss:** Cross Entropy • **Epochs:** 50	• **Standard:** Baseline (no adversarial attack) • **FGSM:** Attack-trained • **L2:** Attack-trained • **L∞:** Attack-trained

- This training setup is used for all experiments, for all 5 sets of varying perturbation budgets.
- Cross comparison is done only within each set.

Dataset Details

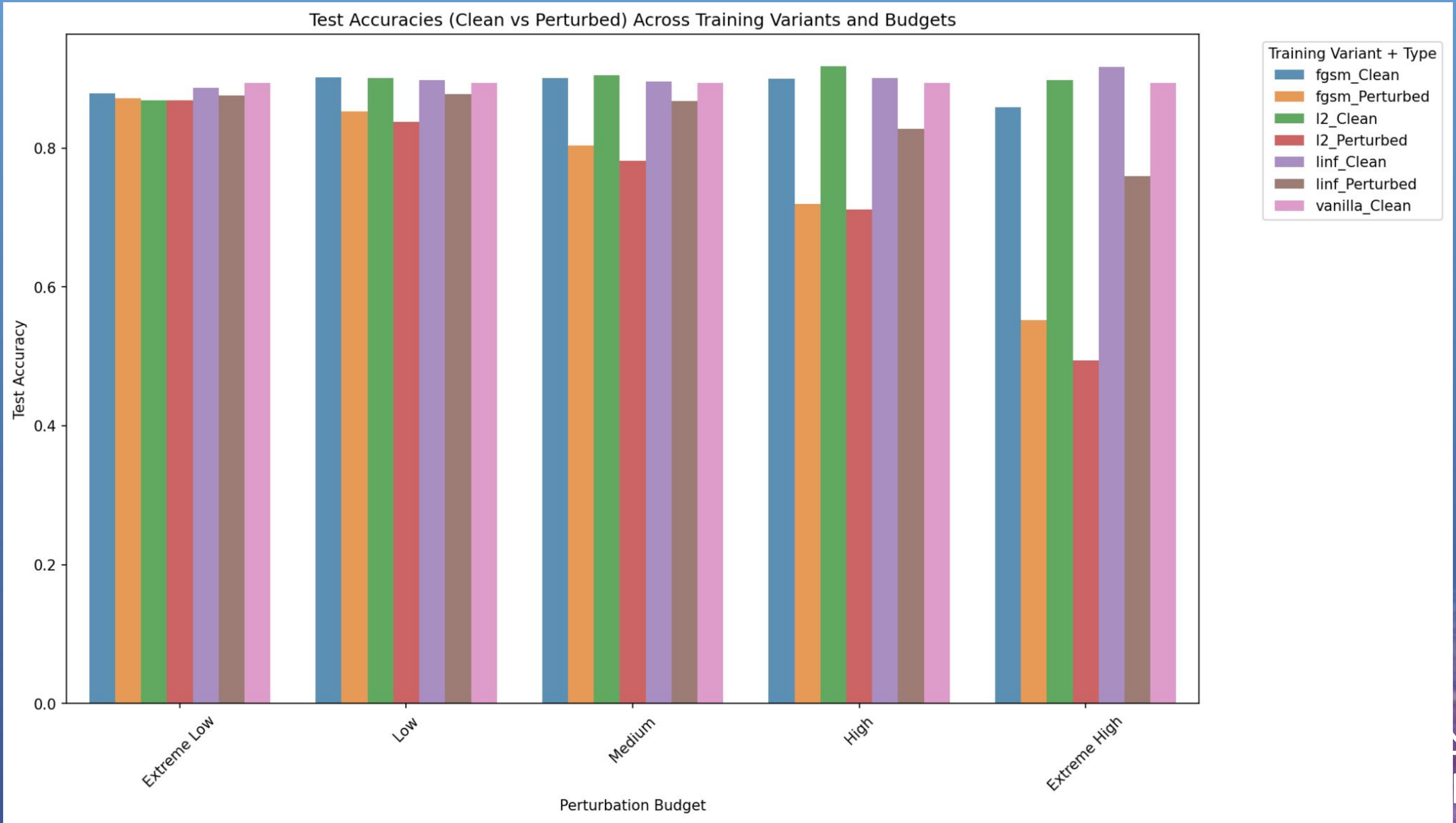
Property	**Value / Description**
Dataset Name	MNIST Handwritten Digits
Training Set Size	10,000 samples
Test Set Size	20,000 samples
Input Shape	28×28 grayscale images
Number of Classes	10
Preprocessing	Flattening (1D vector) of size $28 \times 28 = 784$

CAL-BRIDGE

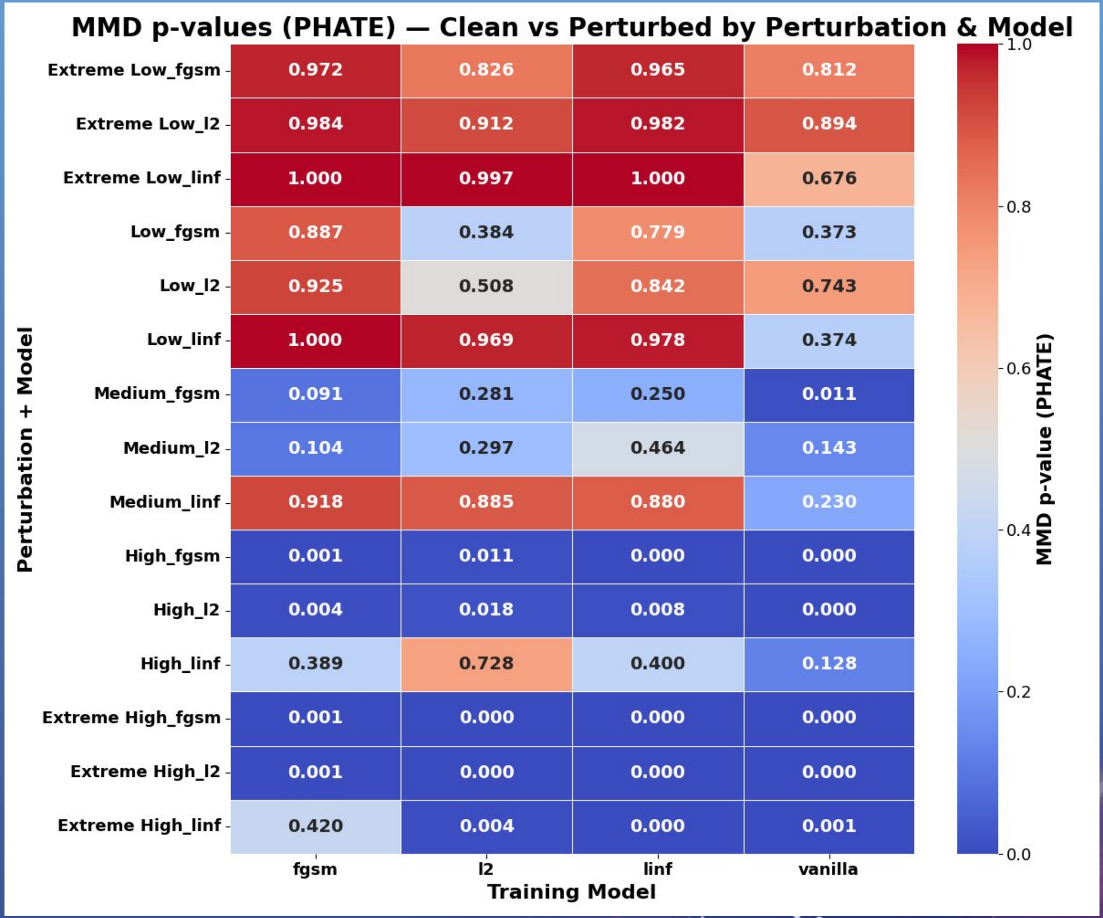
Training and Testing Accuracies

Budget / Variant	Train Acc	Test Acc (Clean)	Test Acc (Perturbed)	Train Loss	Test Loss (Clean)	Test Loss (Perturbed)
Vanilla	1.000	0.897	–	0.0029	0.4461	–
Extreme Low FGSM	1.000	0.879	0.872	0.0429	0.3614	0.4587
Extreme Low L2	1.000	0.869	0.869	0.0474	0.3500	0.4350
Extreme Low LINF	1.000	0.887	0.876	0.0297	0.3596	0.3974
Low FGSM	1.000	0.902	0.853	0.0925	0.4245	0.6742
Low L2	1.000	0.901	0.838	0.0070	0.4143	0.6459
Low LINF	1.000	0.898	0.878	0.0043	0.4271	0.5222
Medium FGSM	1.000	0.901	0.804	0.0347	0.03887	0.8339
Medium L2	1.000	0.905	0.782	0.0215	0.3836	0.8544
Medium LINF	1.000	0.896	0.868	0.0053	0.3988	0.5747
High FGSM	0.924	0.900	0.720	0.2495	0.3274	0.9582
High L2	0.981	0.918	0.712	0.1117	0.2919	1.0485
High LINF	1.000	0.901	0.828	0.0114	0.3900	0.7499
Extreme High FGSM	0.652	0.859	0.552	0.9419	0.4767	1.2358
Extreme High L2	0.678	0.898	0.494	0.8118	0.4032	1.3571
Extreme High LINF	0.996	0.917	0.760	0.0576	0.2981	0.8681

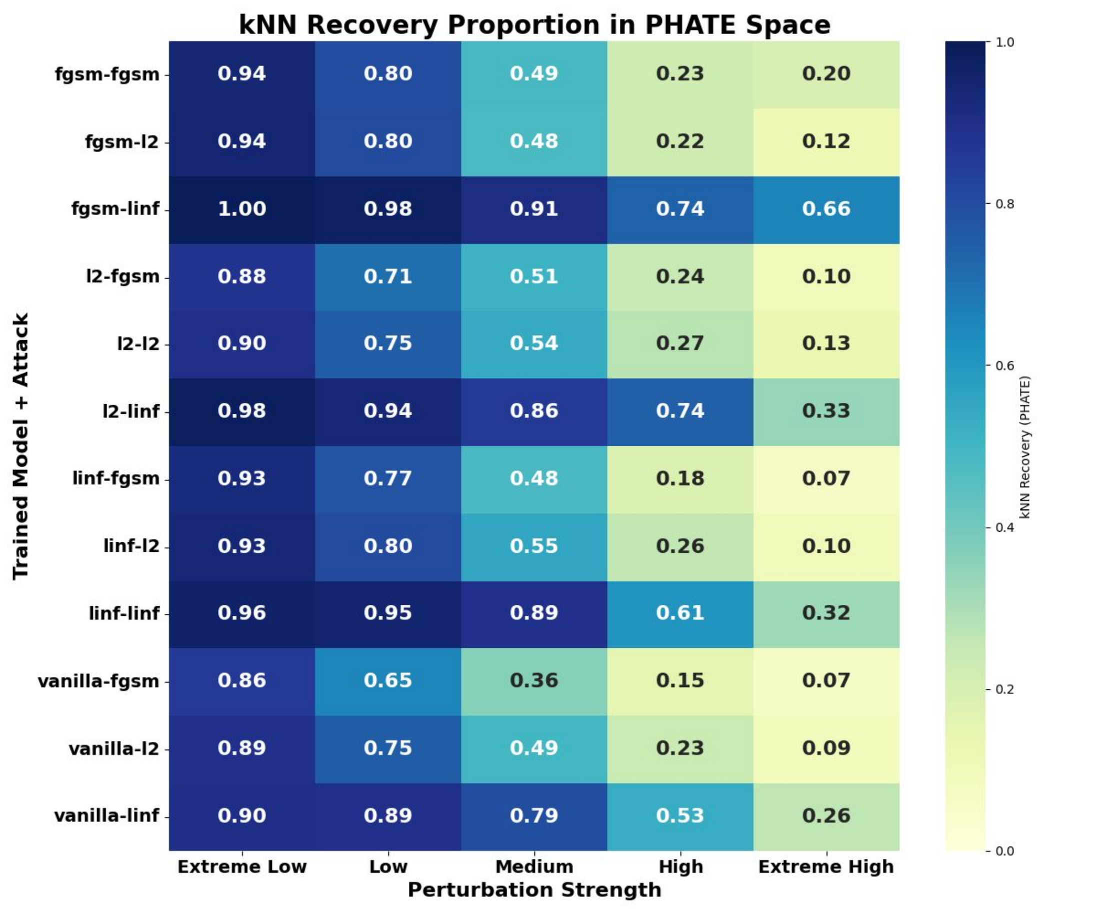
All Accuracies



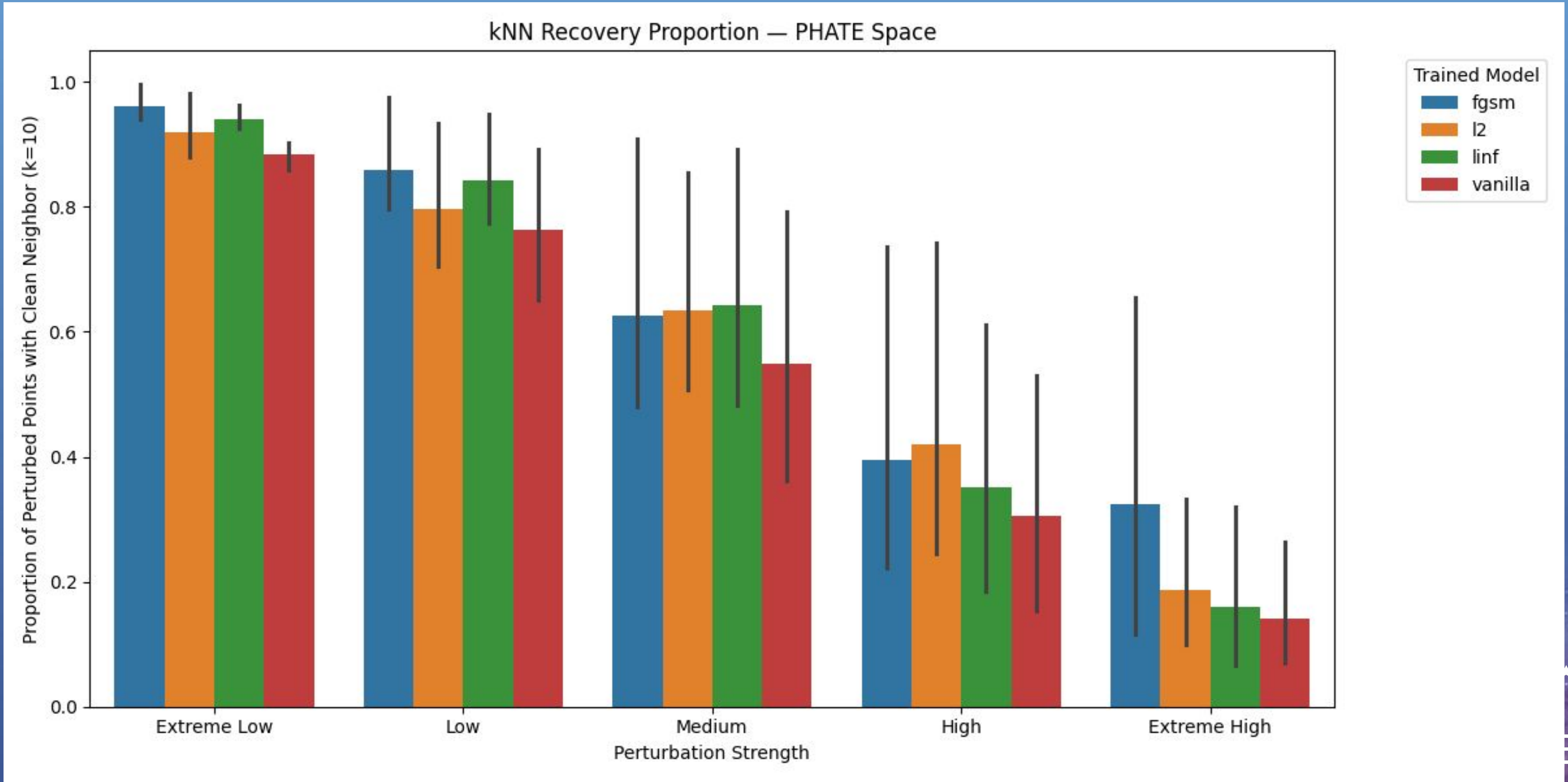
MMD P-values (PHATE)



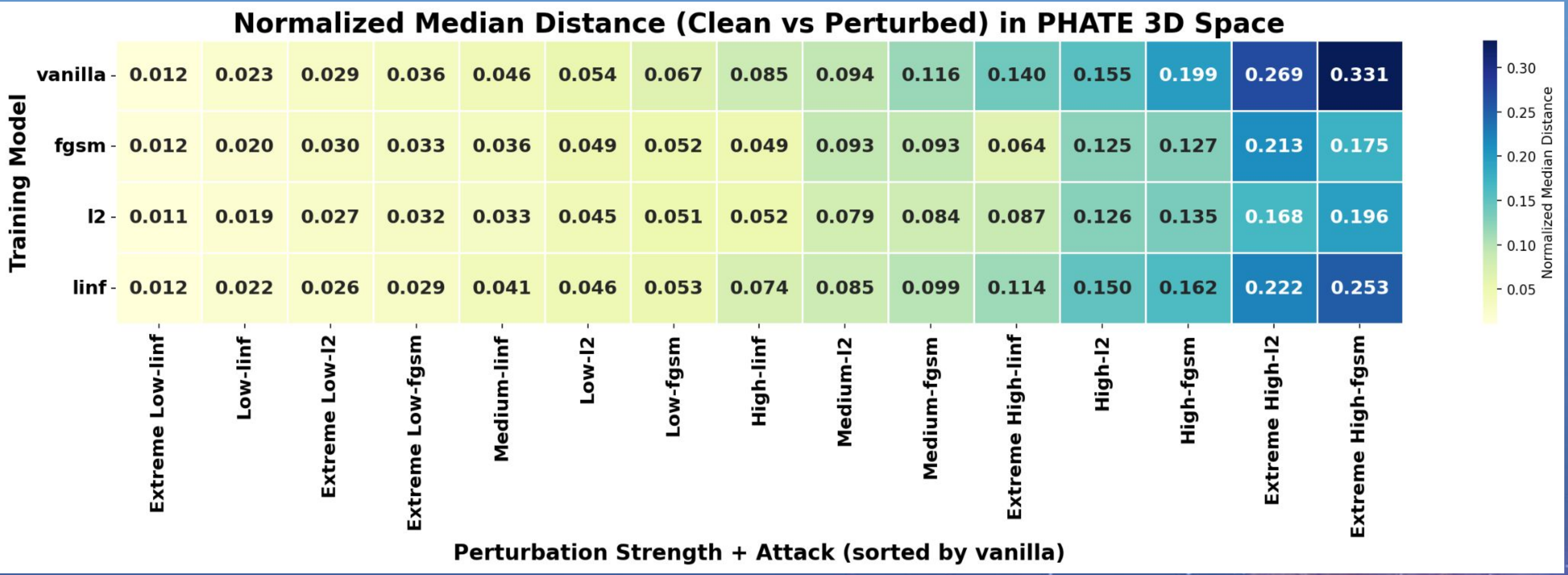
kNN Recovery Proportions (Heatmap)



kNN Recovery Proportions (Barplot)



Euclidean Distance between Clean & Perturbed Samples



Euclidean Distance Between Clean and Perturbed Samples