**DS603: Robust Machine Learning** **Autumn 2025**

Lecture 17 — 29/11/25

*Lecturer: Arjun Bhagoji* *Scribe: Anasmit*

## 17.1 Empirical Adversarial Training

Empirical adversarial training formulates robust learning as a saddle-point optimization problem. For a hypothesis $h \in \mathscr{H}$, loss function $\ell$, and an adversarial perturbation set $\mathscr{N}(x) = \{x' : \|x' - x\| \leqslant \varepsilon\}$, the objective is

$$h_{\text{robust}} \;=\; \operatorname*{arg\,min}_{h\in\mathscr{H}} \; {}_{(x,y)\sim P}\left[\max_{x'\in\mathscr{N}(x)} L_{\text{proxy}}(x', y; h)\right]. \tag{1}$$

The inner maximization adversarially perturbs the input to maximize loss, and the outer minimization updates parameters to reduce this worst-case loss. This corresponds to a first-order approximation of robust optimization.

### 17.1.1 Projected Gradient Descent (PGD) approximation

The inner maximization in **??** is commonly approximated by iterative first-order ascent (Projected Gradient Descent, PGD). Below we display the standard updates for the two most common perturbation norms.

**$\ell_\infty$ (elementwise) perturbations.** For an $\ell_\infty$ perturbation set $\mathscr{N}_\infty(x) = \{x' : \|x' - x\|_\infty \leqslant \varepsilon\}$, a standard iterative update (iterative-FGSM / PGD-$\ell_\infty$) is

$$x^{(k+1)} \;=\; \Pi_{\mathscr{N}_\infty(x)}\Big(x^{(k)} + \alpha_{\text{in}}\,\text{sign}\big(\nabla_x \ell(h_\theta(x^{(k)}), y)\big)\Big),$$

where $\text{sign}(\cdot)$ is applied elementwise, $\alpha_{\text{in}} > 0$ is the inner step size, and $\Pi_{\mathscr{N}_\infty(x)}$ denotes projection onto the $\ell_\infty$ ball (componentwise clipping).

**$\ell_2$ perturbations.** For an $\ell_2$ perturbation set $\mathscr{N}_2(x) = \{x' : \|x' - x\|_2 \leqslant \varepsilon\}$, one typically uses a normalized-gradient ascent step:

$$x^{(k+1)} \;=\; \Pi_{\mathscr{N}_2(x)}\left(x^{(k)} + \alpha_{\text{in}}\frac{\nabla_x \ell(h_\theta(x^{(k)}), y)}{\|\nabla_x \ell(h_\theta(x^{(k)}), y)\|_2}\right),$$

where the projection $\Pi_{\mathscr{N}_2(x)}$ rescales the perturbation to satisfy the $\ell_2$ constraint when required.

**Initialization and practical considerations.** A common initialization is $x^{(0)} = x + \xi$ where $\xi$ is sampled uniformly from the perturbation set (e.g., $\xi \sim \text{Uniform}([-\varepsilon, \varepsilon]^d)$ for $\ell_\infty$), and multiple random restarts may be used to improve the chance of finding high-loss perturbations. Typical heuristics for $\alpha_{\text{in}}$ include $\alpha_{\text{in}} = \varepsilon/K$ or tuning $\alpha_{\text{in}}$ together with the number of steps $K$. Using too large a step size can overshoot steep regions, while too small a step size may require many iterations.

**Remarks.**

- The ‘sign‘ update is specific to the $\ell_\infty$ geometry; it does not correctly approximate $\ell_2$ ascent.
- The projection operator $\Pi_{\mathcal{N}(x)}$ should be implemented to match the chosen norm: for $\ell_\infty$ it is componentwise clipping to $[x-\varepsilon, x+\varepsilon]$; for $\ell_2$ it is projection onto the Euclidean ball centered at $x$.
- In practice the inner maximization is not solved exactly; PGD provides a first-order approximation whose quality depends on $K$, $\alpha_{\text{in}}$, initialization, and the nonconvexity of the loss surface.

### 17.1.2 Parameter update

Given the adversarial sample $x_{\text{adv}}$, parameters $\theta$ are updated via stochastic gradient descent:

$$\theta_t \;=\; \theta_{t-1} - \alpha \nabla_\theta L_{\text{proxy}}\big(h_{\theta_{t-1}}(x_{\text{adv}}), y\big), \tag{2}$$

with learning rate $\alpha > 0$. This procedure alternates between adversarial sample generation and gradient-based parameter updates.

### 17.1.3 Properties

**Advantages.**

- Conceptually simple and compatible with standard stochastic optimization.
- Demonstrates strong empirical robustness against the specific attack class used for inner maximization.
- Effective when the inner maximization is solved to sufficient accuracy.

**Limitations.**

- Computationally expensive due to the iterative inner maximization.
- Robustness may not generalize across different attack models or perturbation sets.
- Does not provide certified guarantees of robustness.

### 17.1.4 Implementation considerations

Standard implementations frequently incorporate non-zero random initialization for the PGD process and choose $\eta_{\text{in}}$ comparable to the perturbation scale $\varepsilon$. These design choices help avoid local maxima in the inner optimization and improve coverage of the perturbation set.

## 17.2 Gradient Masking in Adversarially Trained Models

Gradient masking refers to a failure mode in which gradients of the loss with respect to the input become uninformative for generating adversarial perturbations, despite the classifier not being truly robust. This phenomenon is often associated with highly non-linear or irregular loss surfaces that obstruct effective first-order adversarial optimization.

### 17.2.1 Loss landscape geometry

Let $L(x)$ denote the loss of a trained classifier evaluated at input $x$. For an infinitesimal perturbation $\delta$, a local approximation of the loss is

$$L(x+\delta) \;\approx\; L(x)+\nabla_x L(x)^\top \delta+\tfrac{1}{2}\delta^\top \nabla_x^2 L(x)\,\delta. \tag{3}$$

Well-behaved gradients correspond to a loss surface in which the first-order term $\nabla_x L(x)$ reliably identifies directions that increase the loss. However, adversarially trained models frequently exhibit loss surfaces with the following characteristics:

- regions of extreme curvature, where higher-order terms dominate; and
- flat regions in which $\|\nabla_x L(x)\|$ is very small despite nearby points having significantly larger loss.

In such settings, first-order attacks that rely solely on the gradient may fail to find valid adversarial examples even though they exist, giving the appearance of robustness.

### 17.2.2 Two-dimensional slice visualization

A common diagnostic is to examine a two-dimensional slice of the loss surface around an input $x$, parametrized by orthonormal directions $v_1, v_2 \in \mathbb{R}^d$. Define

$$f(\varepsilon_1,\varepsilon_2)=L\big(x+\varepsilon_1 v_1+\varepsilon_2 v_2\big),$$

for $(\varepsilon_1,\varepsilon_2)$ in a small neighborhood of the origin. For robust behavior, $f$ should increase smoothly in the adversarial direction. In contrast, gradient masking manifests as irregular, sharply varying surfaces in adversarial directions and comparatively flat regions in benign directions.

Figure **??** illustrates such surfaces obtained from adversarially trained networks.

### 17.2.3 Implications

Gradient masking does not constitute robustness. Although it impedes gradient-based attacks, stronger attacks—such as randomized restarts, black-box optimization, or second-order methods—can typically circumvent it. Thus, robust evaluation requires verifying that gradients remain informative and that the model's adversarial accuracy persists under adaptive attacks.

## 17.3 TRADES: A Principled Accuracy–Robustness Trade-off

Adversarial examples typically arise near the classifier's decision boundary, where small perturbations can change the predicted label. The TRADES framework of Zhang et al. **zhang2019trades** gives a theoretical decomposition of adversarial risk that separates *natural (benign) classification error* from *boundary error*—instability of predictions in an adversarial neighborhood. This viewpoint leads to a surrogate training objective that explicitly balances accuracy and robustness.

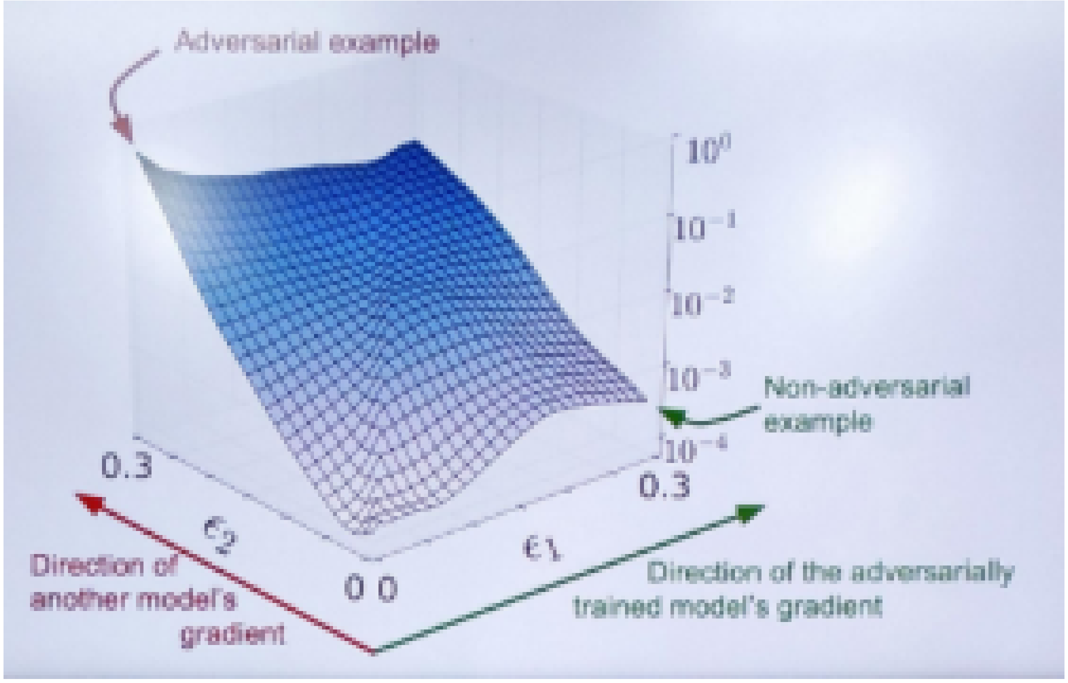


**Figure 17.1.** Two-dimensional slices of the loss landscape for an adversarially trained classifier. The axes $\varepsilon_1, \varepsilon_2$ correspond to orthonormal directions in input space. The surface exhibits highly curved regions near adversarial examples and comparatively flat regions near non-adversarial examples. Such geometry can cause first-order adversarial attacks to fail, despite the absence of true robustness.

### 17.3.1 Benign vs. adversarial risk

Given an adversarial perturbation set $\mathcal{N}(x)$, the adversarial risk of a classifier $h$ is

$$L_{\mathcal{N}}(h) =_{(x,y)\sim P}\left[\max_{x'\in\mathcal{N}(x)} \ell(h(x'),y)\right].$$

TRADES shows that this risk can be decomposed into two components:

- the benign risk $L(h)$, and
- a boundary-error term that measures how unstable the classifier is within the adversarial neighborhood around $x$.

This decomposition implies that reducing adversarial risk requires not only good standard accuracy but also controlling how rapidly the classifier's predictions change around inputs near the decision boundary.

### 17.3.2 The TRADES objective

Using the above decomposition, Zhang et al. derive a surrogate objective that jointly optimizes benign accuracy and local prediction stability. For a robustness weight $\beta > 0$, the TRADES loss is

$$\mathscr{L}_{\text{TRADES}} = \ell_{\text{CE}}\big(h(x),y\big) + \beta\cdot D_{\text{KL}}\big(h(x)\,\|\,h(x')\big), \tag{4}$$

where the inner point $x'$ is chosen via

$$x' \in \arg\max_{z \in \mathscr{N}(x)} D_{\mathrm{KL}}\big(h(x) \,\|\, h(z)\big).$$

**Interpretation.**

- The first term promotes standard (natural) accuracy.
- The second term encourages prediction *smoothness*: large KL divergence indicates that $h$ changes sharply within the adversarial neighborhood of $x$.
- The weight $\beta$ handles the accuracy–robustness trade-off: larger $\beta$ emphasizes robustness.

### 17.3.3 Inner maximization via PGD

The maximizer $x'$ of the KL divergence is approximated by PGD: starting from $x^{(0)} = x + \xi$, for $k = 0, \ldots, K-1$,

$$x^{(k+1)} = \Pi_{\mathscr{N}(x)}\Big(x^{(k)} + \eta_{\mathrm{in}}\,\mathrm{sign}\left(\nabla_x D_{\mathrm{KL}}(h(x), h(x^{(k)}))\right)\Big).$$

(This is the standard update for $\ell_\infty$ perturbations; for other norms, the gradient step is normalized accordingly.)

### 17.3.4 Parameter update

Given adversarial examples $x_{\mathrm{adv}}$, parameters are updated via

$$\theta \leftarrow \theta - \eta\left(\nabla_\theta \ell_{\mathrm{CE}}(h(x), y) + \beta \nabla_\theta D_{\mathrm{KL}}\big(h(x), h(x_{\mathrm{adv}})\big)\right).$$

### 17.3.5 Empirical performance

Table **??** summarizes representative adversarial robustness results from the TRADES paper under PGD attacks on CIFAR-10. These results illustrate that TRADES achieves substantially higher robust accuracy compared to previous defense strategies based on gradient masking or regularization.

## 17.4 Certified Robustness

Certified robustness methods aim to provide formal guarantees that a classifier's prediction remains unchanged for all perturbations within a specified adversarial region. Unlike empirical adversarial training, which approximates worst-case perturbations using gradient-based attacks, certified methods compute provable upper and lower bounds on the network's output under all perturbations belonging to the set $\mathscr{N}(x)$.

| **Defense** | **Type** | **Attack** | **Dataset** | $A_{\text{nat}}$ | $A_{\text{rob}}$ |
|---|---|---|---|---|---|
| BRRG18 | gradient mask | ACW18 | CIFAR10 | – | 0% |
| MLW18 | gradient mask | ACW18 | CIFAR10 | – | 0% |
| DAL18 | gradient mask | ACW18 | CIFAR10 | – | 0% |
| SKN18 | gradient mask | ACW18 | CIFAR10 | – | 9% |
| NKM17 | gradient mask | ACW18 | CIFAR10 | – | 15% |
| WSMK18 | robust opt. | $\text{FGSM}^{\infty}$(PGD) | CIFAR10 | 27.0% | 23.5% |
| MMS18 | robust opt. | $\text{FGSM}^{\infty}$(PGD) | CIFAR10 | 87.0% | 46.0% |
| ZSLG16 | regularization | $\text{FGSM}^{\infty}$(PGD) | CIFAR10 | 94.64% | 0.19% |
| KGB17 | regularization | $\text{FGSM}^{\infty}$(PGD) | CIFAR10 | 95.25% | 45.89% |
| RDV17 | regularization | $\text{FGSM}^{\infty}$(PGD) | CIFAR10 | 88.46% | 49.09% |
| TRADES ($1/\lambda = 1$) | regularization | $\text{FGSM}^{\infty}$(PGD) | CIFAR10 | 89.48% | 56.43% |
| TRADES ($1/\lambda = 2$) | regularization | $\text{FGSM}^{\infty}$(PGD) | CIFAR10 | 88.64% | 55.20% |
| TRADES ($1/\lambda = 6$) | regularization | $\text{FGSM}^{\infty}$(PGD) | CIFAR10 | 84.92% | 56.61% |

**Table 17.1.** Performance comparison of TRADES and prior defenses under white-box PGD attacks on CIFAR-10.

### 17.4.1 Certified prediction stability

A classifier $h$ is said to be certifiably robust at input $x$ with radius $\varepsilon$ if

$$h(x') = h(x) \quad \text{for all } x' \in \mathscr{N}(x) = \{x' : \|x' - x\| \leqslant \varepsilon\}.$$

To establish this property, certified defenses propagate interval or polytope bounds through each layer of the network to obtain tight output bounds. If the lower bound on the predicted logit for the true class exceeds the upper bounds of all other classes, the classification is guaranteed to remain invariant to all admissible perturbations.

### 17.4.2 Bounding methods

Several classes of techniques have been developed to compute certified output bounds:

- **Lipschitz-based certificates:** These methods enforce or estimate global or local Lipschitz constants of the classifier. If $h$ is $L$-Lipschitz, then $\|h(x) - h(x')\| \leqslant L\|x - x'\|$, enabling robustness certificates by bounding the separation between class logits.
- **Convex outer polytope approximations:** These approaches construct convex relaxations of the adversarial polytope $\{h(x') : x' \in \mathscr{N}(x)\}$. By solving linear or semidefinite programs over these relaxations, one obtains guaranteed bounds on the logits of perturbed inputs.
- **Interval Bound Propagation (IBP):** IBP propagates lower and upper bounds of activations through the network via interval arithmetic. For a layer $z^{(\ell+1)} = \phi(Wz^{(\ell)} + b)$, one computes explicit bounds $[z_{\text{min}}^{(\ell+1)}, z_{\text{max}}^{(\ell+1)}]$ using the known monotonicity of $\phi$ and the affine structure of $Wz + b$. The resulting bounds provide conservative but efficient certificates.

Let $[f_{\text{min}}(x'), f_{\text{max}}(x')]$ denote certified lower and upper bounds on the logits at any $x' \in \mathscr{N}(x)$. A prediction for class $y$ is certified if

$$f_{\text{min}}(x')_y > \max_{j \neq y} f_{\text{max}}(x')_j.$$

### 17.4.3 Properties

Certified robustness techniques possess several desirable characteristics:

- **Provable safety:** No adversarial examples exist within the certified perturbation set.
- **Resistance to adaptive attacks:** Since certificates do not rely on gradient approximations, they are not vulnerable to gradient masking.
- **No gradient masking:** Certified methods evaluate worst-case perturbations analytically, not by optimization.

### 17.4.4 Limitations

Despite their rigorous guarantees, certified defenses face practical limitations:

- **High computational cost:** Computing bounds during training substantially increases the per-iteration complexity.
- **Small certifiable radii:** Current methods typically certify only small perturbation norms, especially for high-dimensional datasets.

Overall, certified robustness provides mathematically rigorous guarantees of adversarial stability, while incurring increased computational demands and offering limited certified radii in practice.

## 17.5 Visualizing Adversarially Trained Models

A useful tool for understanding the internal representations learned by adversarially trained models is layer-wise similarity analysis. Centered Kernel Alignment (CKA) provides a measure of similarity between feature representations across layers or across different networks. Given two feature matrices $Z_1, Z_2 \in \mathbb{R}^{n\times d}$, the CKA similarity is defined as

$$\mathrm{CKA}(Z_1, Z_2) = \frac{\mathrm{HSIC}(Z_1, Z_2)}{\sqrt{\mathrm{HSIC}(Z_1, Z_1)\,\mathrm{HSIC}(Z_2, Z_2)}},$$

where HSIC denotes the Hilbert–Schmidt Independence Criterion.

### 17.5.1 Representation similarity under adversarial training

Layer-wise CKA heatmaps reveal structural differences between standard and adversarially trained networks. Several empirical patterns consistently emerge:

- **Higher self-similarity in deeper layers of robust models:** Robust networks exhibit more uniform representations across late layers, indicating increased invariance and compressed feature geometry.
- **Reduced cross-model similarity:** Representations of adversarially trained models differ substantially from those of standard models, demonstrating that adversarial training alters the feature hierarchy rather than merely modifying the classifier boundary.

- **Dataset dependence:** The degree of representational shift varies across datasets (e.g., CIFAR-10, ImageNet variants), but the trend of deeper-layer reorganization remains consistent.

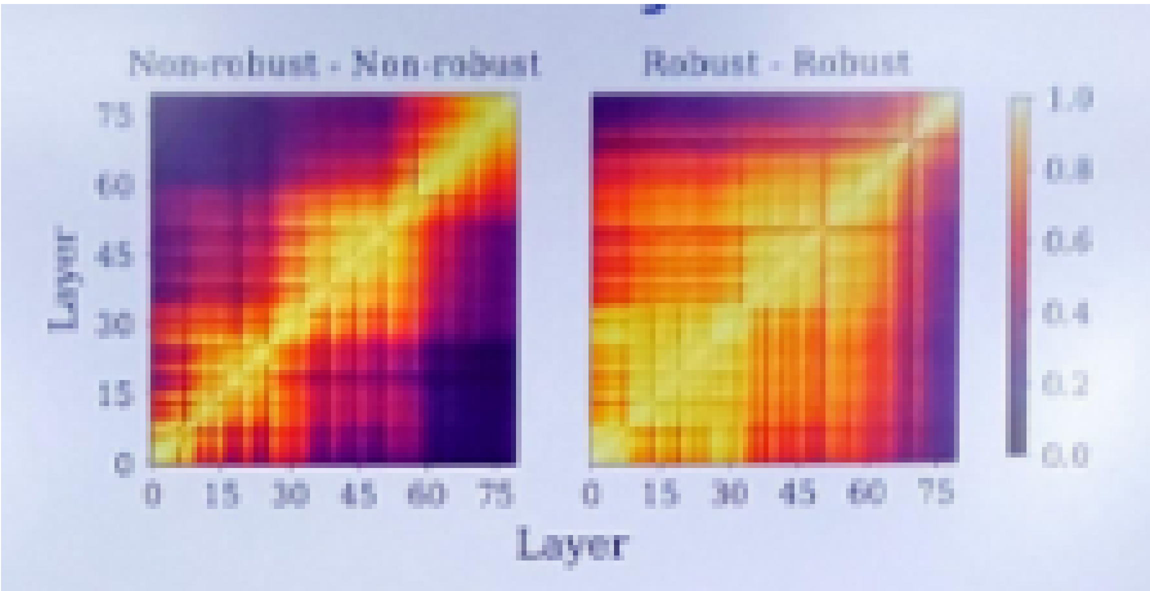


**Figure 17.2.** Layer-wise CKA similarity visualizations for standard and adversarially trained models. Top: self-similarity of non-robust and robust ResNet-50 networks on CIFAR-10. Bottom: cross-model similarity comparisons across datasets, illustrating the representational differences induced by adversarial training.

These observations indicate that adversarial robustness fundamentally modifies the organization of intermediate features, producing more stable and structured internal representations compared to standard training.

## 17.6 Optimal Adversarial Risk and Distribution-Induced Limitations

Adversarial learning differs from the benign setting because the learner must correctly classify not only each training point, but all points within an adversarial neighborhood surrounding it. Consequently, even the optimal classifier may incur nonzero adversarial risk due to unavoidable overlaps in these neighborhoods. This section formalizes these concepts and establishes distribution-dependent limits on robust classification.

### 17.6.1 Adversarial risk

Let $\mathcal{N}(x)$ be the adversarial neighborhood of $x$, typically

$$\mathcal{N}(x) = \{x' : \|x' - x\| \leqslant \varepsilon\}.$$

For a classifier $h$, the adversarial risk is

$$L_{\mathcal{N}}(h) =_{(x,y)\sim P} \left[ \sup_{x' \in \mathcal{N}(x)} \ell(h(x'), y) \right].$$

**Optimal robust classifier.**

$$h^*_{\mathscr{N}} \in \arg\min_{h \in \mathscr{H}} L_{\mathscr{N}}(h).$$

**Adversarial empirical risk minimizer.** Given samples $\{(x_i, y_i)\}_{i=1}^N$,

$$\widehat{h}_{\mathscr{N},N} \in \arg\min_{h \in \mathscr{H}} \frac{1}{N} \sum_{i=1}^N \sup_{x' \in \mathscr{N}(x_i)} \ell(h(x'), y_i).$$

**Three key quantities.** The adversarial setting mirrors classical learning theory, giving rise to:

1. *Adversarial optimization error*
$$L_{\mathscr{N}}(\widehat{h}_{\mathscr{N},N}) - \inf_{h \in \mathscr{H}} L_{\mathscr{N}}(h).$$
2. *Adversarial generalization gap*
$$L_{\mathscr{N}}(\widehat{h}_{\mathscr{N},N}) - \frac{1}{N} \sum_{i=1}^N \sup_{x' \in \mathscr{N}(x_i)} \ell(\widehat{h}_{\mathscr{N},N}(x'), y_i).$$
3. *Optimal adversarial risk*
$$L_{\mathscr{N}}(h^*_{\mathscr{N}}).$$

The third quantity is intrinsic to the data distribution and cannot be improved by algorithmic means.

### 17.6.2 Unavoidable adversarial error

Since $\mathscr{N}(x)$ always contains $x$, benign risk is a lower bound:

$$L_{\mathscr{N}}(h^*_{\mathscr{N}}) \geqslant L(h^*).$$

This inequality is strict whenever adversarial neighborhoods overlap regions assigned to different classes.

### 17.6.3 Univariate Gaussian Model: Adversarial Bayes Classification

We illustrate the effect of adversarial perturbations on the Bayes-optimal classifier in a simple one-dimensional setting. Consider a binary problem with equal class priors:

$$X \mid Y = 0 \sim \mathscr{N}(\mu_0, \sigma^2), \qquad X \mid Y = 1 \sim \mathscr{N}(\mu_1, \sigma^2), \quad \mu_1 > \mu_0.$$

For the benign (non-adversarial) problem, the Bayes classifier is

$$h^*(x) = \begin{cases} 1, & x > t_{\text{ben}}, \\ 0, & x \leqslant t_{\text{ben}}, \end{cases} \qquad t_{\text{ben}} = \frac{\mu_0 + \mu_1}{2}.$$

**Adversarial perturbations.** Let the adversary act in the $\ell_\infty$ model on $\mathbb{R}$, that is, for each input $x$ the perturbation set is

$$\mathcal{N}(x) = [x-\varepsilon, x+\varepsilon].$$

For a candidate classifier $h$, the adversarial risk is

$$L_{\mathcal{N}}(h) = \mathbb{E}_{(x,y)}\Big[\sup_{x'\in[x-\varepsilon,x+\varepsilon]} \mathbb{I}\big(h(x')\neq y\big)\Big].$$

A point $x$ is robustly classified as class 1 if *all* perturbed points $x' \in [x-\varepsilon, x+\varepsilon]$ lie on the side of the threshold assigned to class 1. Consequently, a threshold classifier is robust for label 1 at $x$ if and only if

$$x-\varepsilon > t,$$

where $t$ is the decision threshold of $h$. Analogously, it is robust for label 0 if and only if

$$x+\varepsilon \leqslant t.$$

**Bayes-optimal adversarial threshold.** Let $h_t(x) = \mathbb{I}[x>t]$ denote a threshold classifier. Under the above robustness constraints, classification of label 1 suffers adversarial error on the event $\{X \leqslant t+\varepsilon\}$, whereas classification of label 0 suffers adversarial error on the event $\{X \geqslant t-\varepsilon\}$. Thus the adversarial risk of $h_t$ is

$$L_{\mathcal{N}}(h_t) = \frac{1}{2}\Big(\mathbb{P}(X \leqslant t+\varepsilon \mid Y=1) + \mathbb{P}(X \geqslant t-\varepsilon \mid Y=0)\Big).$$

Using the Gaussian CDF $\Phi$,

$$L_{\mathcal{N}}(h_t) = \frac{1}{2}\Big(\Phi\Big(\frac{t+\varepsilon-\mu_1}{\sigma}\Big) + 1 - \Phi\Big(\frac{t-\varepsilon-\mu_0}{\sigma}\Big)\Big).$$

**Minimizing the adversarial risk.** Differentiating with respect to $t$ and setting the derivative to zero yields the optimal adversarial threshold:

$$t_{\text{adv}} = \frac{\mu_0+\mu_1}{2}.$$

Thus, in this symmetric Gaussian model, adversarial perturbations do *not* shift the optimal decision boundary; instead, they increase the overlap of the effective class-conditional regions by expanding the misclassification events.

**Resulting adversarial Bayes risk.** Substituting $t_{\text{adv}}$ into the adversarial risk expression gives

$$L_{\mathcal{N}}(h^*_{\mathcal{N}}) = \Phi\Big(\frac{\varepsilon-(\mu_1-\mu_0)/2}{\sigma}\Big),$$

which exceeds the benign Bayes error

$$L(h^*) = \Phi\Big(-\frac{\mu_1-\mu_0}{2\sigma}\Big).$$

**Interpretation.** Adversarial perturbations reduce the effective separation between the two Gaussian classes from $(\mu_1 - \mu_0)$ to $(\mu_1 - \mu_0 - 2\varepsilon)$. This contraction increases the Bayes-optimal error even though the optimal decision boundary remains unchanged. The phenomenon illustrates a general principle: the minimal achievable adversarial risk is determined by intrinsic overlap between adversarial neighborhoods of the class distributions.

### 17.6.4 Neighborhood-induced overlap: toy example

Let $x$ and $x'$ be samples from different classes. If their adversarial neighborhoods satisfy

$$\mathscr{N}(x) \cap \mathscr{N}(x') \neq \varnothing,$$

then no classifier can assign robust labels to both points. Thus, overlap of neighborhoods implies unavoidable adversarial misclassification.

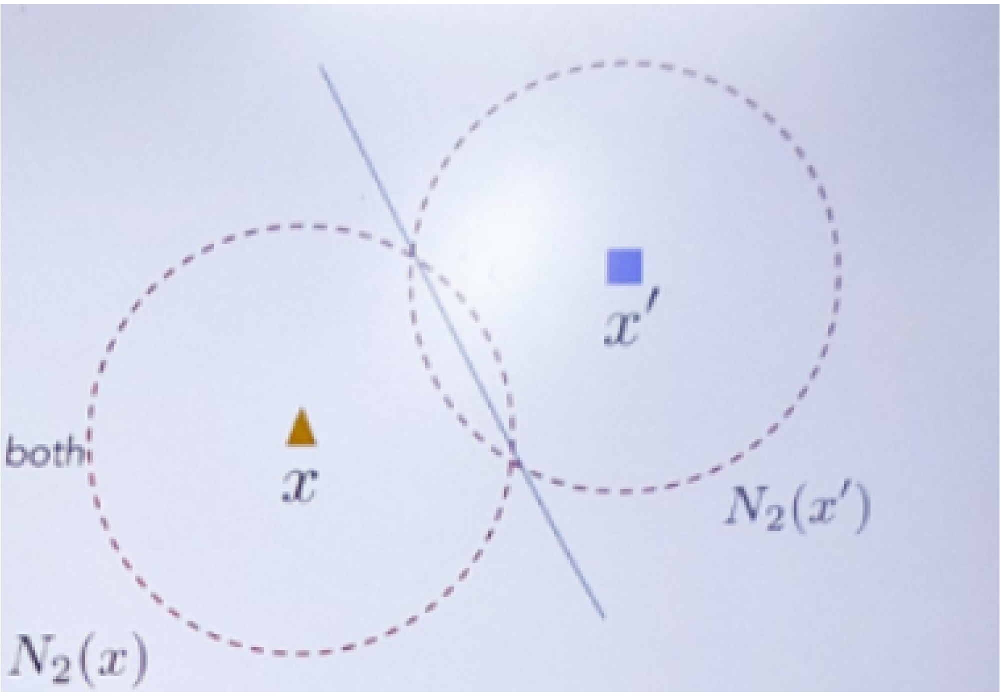


**Figure 17.3.** Left: disjoint neighborhoods allow robust separation. Right: intersecting neighborhoods force misclassification for at least one class, imposing a positive minimal adversarial risk.

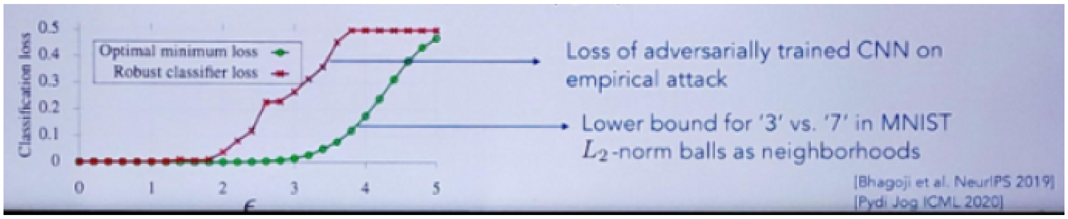
### 17.6.5 Optimal adversarial loss: two-class distributions

For class-conditional distributions $P_0, P_1$ and adversary $\mathscr{N}$, the minimal possible adversarial risk over all measurable classifiers can be expressed as

$$\inf_h L_{\mathscr{N}}(h) = \frac{1}{2}\big(1 - D_{\mathscr{N}}(P_0, P_1)\big),$$

where $D_{\mathscr{N}}$ is an adversary-dependent divergence capturing the degree to which adversarial neighborhoods make the two distributions indistinguishable.

This provides a fundamental lower bound: even optimal classifiers cannot achieve a risk below this value.



Loss of adversarially trained CNN on empirical attack

Lower bound for '3' vs. '7' in MNIST $L_2$-norm balls as neighborhoods

[Bhagoji et al. NeurIPS 2019]
[Pydi Jog ICML 2020]

**Figure 17.4.** Comparison of robust classifier loss and the theoretical minimum robust risk determined solely by class distributions and adversarial neighborhoods.

### 17.6.6 Optimal Transport View of Distribution-Induced Robustness Limits

Optimal adversarial risk reflects intrinsic geometric interactions between the class-conditional distributions under the perturbation model. A convenient way to formalize this interaction is through an optimal-transport (OT) viewpoint, which measures how difficult it is for an adversary to make the two distributions indistinguishable by applying perturbations within $\mathscr{N}(x)$.

**Setup.** Let $P_0$ and $P_1$ denote the class-conditional distributions of $X$ given $Y=0$ and $Y=1$, respectively, and assume equal class priors. For a measurable classifier $h:\mathscr{X}\to\{0,1\}$, the adversarial risk is

$$L_{\mathscr{N}}(h)=\frac{1}{2}\mathbb{E}_{X\sim P_0}\left[\sup_{x'\in\mathscr{N}(x)}\mathbb{I}(h(x')\neq 0)\right]+\frac{1}{2}\mathbb{E}_{X\sim P_1}\left[\sup_{x'\in\mathscr{N}(x)}\mathbb{I}(h(x')\neq 1)\right].$$

The minimal achievable adversarial risk is

$$L^*_{\mathscr{N}}=\inf_h L_{\mathscr{N}}(h),$$

which depends only on $(P_0,P_1)$ and the perturbation structure $\mathscr{N}$.

**Adversarial indistinguishability.** For two points $x,x'\in\mathscr{X}$, define

$$\Gamma_{\mathscr{N}}(x,x')=\begin{cases}1, & \mathscr{N}(x)\cap\mathscr{N}(x')=\varnothing,\\ 0, & \text{otherwise.}\end{cases}$$

This quantity records whether the adversary can map $x$ and $x'$ into a common region; if their neighborhoods intersect, then an adversary can make them locally indistinguishable to any classifier.

**Adversarial OT cost.** We define an OT-type cost function

$$c_{\mathscr{N}}(x,x')=\Gamma_{\mathscr{N}}(x,x'),$$

which penalizes pairs whose adversarial neighborhoods do *not* overlap. The associated optimal-transport divergence between $P_0$ and $P_1$ is

$$D_{\mathscr{N}}(P_0,P_1)=\inf_{\pi\in\Pi(P_0,P_1)}\int c_{\mathscr{N}}(x,x')\,d\pi(x,x'),$$

where $\Pi(P_0,P_1)$ is the set of joint couplings with marginals $P_0$ and $P_1$.

**Interpretation.** The quantity $D_{\mathscr{N}}(P_0,P_1)$ measures the minimal mass of pairs $(x,x')$ that *cannot* be made locally indistinguishable under adversarial perturbations. A small value means that most of the mass of $P_0$ can be transported to $P_1$ through pairs whose neighborhoods intersect. In such cases, no classifier can reliably distinguish the classes under adversarial perturbations.

**Connection to optimal adversarial risk.** For any measurable classifier $h$, one has the lower bound

$$L_{\mathscr{N}}(h) \geqslant \frac{1}{2}\Big(1 - D_{\mathscr{N}}(P_0,P_1)\Big).$$

Thus,

$$L^*_{\mathscr{N}} \geqslant \frac{1}{2}\Big(1 - D_{\mathscr{N}}(P_0,P_1)\Big),$$

which formalizes the intuition that adversarial robustness is limited whenever the perturbation model causes the class distributions to become locally intertwined. The divergence $D_{\mathscr{N}}$ thereby captures an intrinsic, distribution-induced barrier to robust classification. In particular, if $D_{\mathscr{N}}(P_0,P_1) = 0$, then the adversary can fully align the supports of $P_0$ and $P_1$ under perturbations, making the classes robustly indistinguishable.

**Geometric special case.** When $\mathscr{N}(x)$ is an $\varepsilon$-ball under a metric $d$, one has

$$\mathscr{N}(x) \cap \mathscr{N}(x') = \varnothing \quad \iff \quad d(x,x') > 2\varepsilon.$$

In this case,

$$c_{\mathscr{N}}(x,x') = \mathbb{I}[d(x,x') > 2\varepsilon],$$

and $D_{\mathscr{N}}$ quantifies the minimal transport mass that must cross a $2\varepsilon$-margin barrier separating the two distributions.

**Summary.** The OT formulation provides a principled method of quantifying the inherent adversarial overlap between class-conditional distributions. Robust classification is achievable only when the induced neighborhoods of the distributions remain sufficiently separated relative to the adversary's allowable perturbations.