WHICH FEATURES REMAIN VULNERABLE IN ROBUST MODELS?

Anonymous authors
Paper under double-blind review

ABSTRACT

Scaling adversarial training to larger models and hundreds of millions of synthetic images has improved robustness, yet these classifiers remain vulnerable to small input perturbations. What properties of their learned features make these attacks possible? We study 71 of the most robust CIFAR10 models publicly available, spanning a wide range of architectures and training conditions. Our analysis reveals that adversarial attacks manipulate the same features that are responsible for separating the classes on clean data. Moreover, this relationship is concentrated in a surprisingly small subset of features: for the most robust model, more than 70% of successful attacks can still succeed when restricted to just 10% of the features per class pair. Furthermore, we find that the samples that remain vulnerable can be identified as those which are poorly separated along these specific feature dimensions, and that successful attacks follow a consistent attack direction. In addition, at the model level, we demonstrate a gradient-free measure of margin spread that is strongly correlated with higher adversarial robustness. Together, these results provide new insight into the mechanisms that underlie the remaining vulnerability of highly robust classifiers.

1 INTRODUCTION

Adversarial examples remain one of the most intriguing yet perplexing phenomena observed in deep neural networks (DNNs). Despite over a decade of research (Szegedy et al., 2013; Goodfellow et al., 2014), effective defenses remain elusive. While methods such as adversarial training (AT) (Madry et al., 2018) and TRADES (Zhang et al., 2019) substantially improve adversarial robustness, they still leave models vulnerable to powerful attacks (Croce et al., 2021). Recent advances have increasingly relied on scaling the amount of training data (Gowal et al., 2021; Wang et al., 2023); however, this strategy yields diminishing returns while incurring substantial computational cost. For example, even under a training budget comparable to the estimated cost of GPT-4, the scaling laws of Bartoldson et al. (2024) predict only about 80% robust accuracy on CIFAR10.

The difficulty of defending against adversarial attacks has inspired a considerable amount of research investigating *why* adversarial examples exist in the first place. These include descriptions of concentration of measure (Shafahi et al., 2019), misalignment with human perception (Salvatore et al., 2026), interference due to superposition (Stevinson et al., 2026), and inherently fragile features (Ilyas et al., 2019), among many others. However, most of these explanations concern standard models, i.e. models not explicitly trained to resist adversarial attacks.

Of particular interest is the seminal paper by Ilyas et al. (2019), who posit that adversarial examples are due to *useful non-robust features*: features learned by the model that, while susceptible to tiny input perturbations, are highly predictive and useful to the task. It is further motivated that techniques such as adversarial training prevents models from learning such features. However, adversarial examples can still be found for the most robust of real-world models. In this work we also take a feature-centric view of robustness, but shift our focus to highly robust models. Our contributions are:

- **A gradient-free predictor of robustness.** We bound the fraction of samples that cannot be guaranteed safe against an ϵ-attack in terms of the mean and spread of clean output margins. Under two simplifying assumptions, the bound depends only on a normalized

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

output margin separation, which ranks 71 RobustBench models by robust accuracy ($\tau_b = 0.54$) using only clean training representations and the output layer (Section 3).

- **Attacks exploit class-separating features.** In a contribution space that weights each feature by its role in the output margin, minimal adversarial displacements align closely with the directions separating the class means (Section 4.1).
- **Vulnerability is concentrated in few features.** For the most robust model, about 70% of successful attacks still succeed when only the 10% most useful features per class pair may change, while restricting these features reduces attack success by more than 80% (Section 4.2).
- **Useful features track sample vulnerability.** Restricting the output margin to the top 5% of useful features retains most of its rank correlation with the input-space margin, linking margin consistency (Ngnawé et al., 2024) to a core feature subset (Section 4.3).

2 PRELIMINARIES

2.1 NOTATION AND DEFINITIONS

Let $f_\theta : \mathbb{R}^d \to \mathbb{R}^K$ denote a deep neural network (DNN) parameterized by θ, and let $\phi_\theta : \mathbb{R}^d \to \mathbb{R}^p$ denote its penultimate-layer feature extractor. For an input sample $\boldsymbol{x}^{(i)} \in \mathbb{R}^d$ with label $y^{(i)} \in \{1, \dots, K\}$, its feature representation is $\boldsymbol{h}^{(i)} = \phi_\theta(\boldsymbol{x}^{(i)}) \in \mathbb{R}^p$. The logit associated with class k is given by

$$[f_\theta(\boldsymbol{x}^{(i)})]_k = o_k\Big(\phi_\theta(\boldsymbol{x}^{(i)})\Big) = \boldsymbol{w}_k^\top \boldsymbol{h}^{(i)} + a_k, \tag{1}$$

where $\boldsymbol{w}_k \in \mathbb{R}^p$ is the weight vector linking the penultimate layer with the class k logit and a_k is the bias. The predicted class is then given by $\hat{y}^{(i)} = \arg\max_k o_k(\boldsymbol{h}^{(i)})$.

Adversarial example For a sample $\boldsymbol{x}^{(i)}$ with $y^{(i)} = b$, define its *output margin* as the difference between logits for a given class pair b, t,

$$m_{b,t}(\boldsymbol{x}) := [f_\theta(\boldsymbol{x})]_b - [f_\theta(\boldsymbol{x})]_t = (\boldsymbol{w}_b - \boldsymbol{w}_t)^\top \phi_\theta(\boldsymbol{x}) + (a_b - a_t). \tag{2}$$

Then, an adversarial example $\boldsymbol{x}^{(i)}_{\text{adv}} = \boldsymbol{x}^{(i)} + \boldsymbol{\delta}$ is a perturbed version of $\boldsymbol{x}^{(i)}$ such that the output margin is negative,

$$m_{b,t}(\boldsymbol{x}^{(i)} + \boldsymbol{\delta}) < 0 \quad \text{for some } t \neq b, \qquad \|\boldsymbol{\delta}\|_q \leq \epsilon, \tag{3}$$

where $\boldsymbol{\delta}$ is a perturbation, in input space, subject to a chosen perturbation bound ϵ under an appropriate ℓ_q norm. We assume here that the original sample is correctly classified, such that $m_{b,t}(\boldsymbol{x}^{(i)}) > 0, \forall\, t \neq b$. Besides bounded perturbations, we are also interested in the minimum perturbation required to cross the decision boundary from class b toward t,

$$\boldsymbol{\delta}^{(i)}_{b,t} := \arg\min_{\boldsymbol{\delta}} \left\{ \|\boldsymbol{\delta}\|_q : m_{b,t}\left(\boldsymbol{x}^{(i)} + \boldsymbol{\delta}\right) \leq 0 \right\}. \tag{4}$$

We refer to $m^{\text{IS}}_{b,t}(\boldsymbol{x}^{(i)}) := \|\boldsymbol{\delta}^{(i)}_{b,t}\|_q$ as the *input space margin*. Finally, let $m^{\text{IS}}_{b,\star}(\boldsymbol{x}^{(i)}) := \min_{t \neq b} m^{\text{IS}}_{b,t}(\boldsymbol{x}^{(i)})$ denote the smallest input space margin across all possible target classes.

Decision boundary normal The level set $\{\boldsymbol{h} \in \mathbb{R}^p : o_b(\boldsymbol{h}) - o_t(\boldsymbol{h}) = 0\}$ indicates the linear decision boundary between classes b and t in the feature space. Furthermore, the normal vector pointing toward class t is given by $\boldsymbol{n}_{b\to t} = \boldsymbol{w}_t - \boldsymbol{w}_b$. Therefore, a successful adversarial perturbation in the input space has to induce enough movement along this normal feature space vector such that $(\Delta\boldsymbol{h}^{(i)})^\top \boldsymbol{n}_{b\to t} > m_{b,t}(\boldsymbol{x}^{(i)})$ where $\Delta\boldsymbol{h}^{(i)} := \boldsymbol{h}^{(i)}_{\text{adv}} - \boldsymbol{h}^{(i)}$ and $\boldsymbol{h}^{(i)}_{\text{adv}} = \phi_\theta(\boldsymbol{x}^{(i)}_{\text{adv}}) \in \mathbb{R}^p$.

Class-mean margin Let $\mathcal{I}_k$ denote the index set of samples of class k, with $N_k := |\mathcal{I}_k|$ and class mean $\boldsymbol{\mu}_k := \frac{1}{N_k}\sum_{i\in\mathcal{I}_k} \boldsymbol{h}^{(i)}$. For base class b and a target class $t \neq b$, define the margin at the class-k mean, for $k \in \{b, t\}$, as

$$m_k \;:=\; (\boldsymbol{w}_b - \boldsymbol{w}_t)^\top \boldsymbol{\mu}_k + (a_b - a_t) \;=\; \frac{1}{N_k}\sum_{i\in\mathcal{I}_k} m_{b,t}(\boldsymbol{x}^{(i)}), \tag{5}$$

where the second equality follows from the linearity of the output margin in $\boldsymbol{h}$; m_k is therefore also the mean output margin of class k.

Margin spread Define the deviation of each sample's output margin from the margin at the class mean as

$$\eta^{(i)} \;:=\; m_{b,t}(\boldsymbol{x}^{(i)}) - m_b \;=\; (\boldsymbol{w}_b - \boldsymbol{w}_t)^\top (\boldsymbol{h}^{(i)} - \boldsymbol{\mu}_b), \tag{6}$$

so that a sample's margin is the margin at the class mean plus its deviation; and the spread of the output margins of class b as

$$s^2_{m,b} \;:=\; \frac{1}{N_b} \sum_{i \in \mathcal{I}_b} \left(\eta^{(i)}\right)^2 \;=\; (\boldsymbol{w}_b - \boldsymbol{w}_t)^\top \Sigma_W^{(b)} (\boldsymbol{w}_b - \boldsymbol{w}_t), \tag{7}$$

where $\Sigma_W^{(b)} := \frac{1}{N_b} \sum_{i \in \mathcal{I}_b} (\boldsymbol{h}^{(i)} - \boldsymbol{\mu}_b)(\boldsymbol{h}^{(i)} - \boldsymbol{\mu}_b)^\top$ is the within-class covariance of class b. All analysis is conducted for a fixed class pair (b, t) and attack budget ϵ; where it simplifies notation, the dependence of quantities on t and ϵ is not shown explicitly.

2.2 Models

We consider a set of pretrained models available in the RobustBench (Croce et al., 2021) model zoo for analysis. Specifically, we use the models available for the CIFAR10 dataset trained to resist $||\boldsymbol{\delta}||_\infty \leq \frac{8}{255}$ bounded adversarial attacks, the most common setting for studying adversarial examples. This provides us with a large number of models of varying degrees of robustness, architectures, sizes, and diverse training conditions. Of the 76 available models, we exclude three ensemble models and one model with unreliable gradients, as these are not suitable for feature-level analysis. We also exclude the 'Standard' baseline, which has 0% robust accuracy. This leaves 71 models for analysis in total.

We generally report findings as averages across all these models. However, we also report key results on a representative set of 5 models, individually. We select the single most and least robust models, and then 3 models approximately equally spaced in terms of robust accuracy between these two extremes. The set consists of the following RobustBench model IDs, with robust accuracies in parentheses: Bartoldson2024Adversarial_WRN-94-16 (73.71%), Huang2022Revisiting_WRN-A4 (65.79%), Debenedetti2022Light_XCiT-L12 (57.58%), Engstrom2019Robustness (49.25%), Ding2020MMA (41.44%). In addition to varying levels of robustness, these models also span 5 unique architectures, namely WRN-94-16, WRN-A4, XCiT-L12, ResNet-50, and WRN-28-4, respectively. We pay particular attention to Bartoldson2024Adversarial_WRN-94-16, given that it is the most robust model currently available on the RobustBench leaderboard.

3 Model analysis

We first consider the robustness of models from a purely geometric perspective. We ask what effect the concentration of feature representations has on a model's vulnerability to an ϵ-attack. We make a number of simplifying assumptions in order to define a gradient-free quantity based on the output margin that provides an upper bound on the fraction of samples that cannot be guaranteed safe from an ϵ-attack. Derivations are included in Appendix A.1.

Assumption 1: Feature space local Lipschitz continuity Since our focus is on feature geometry, we make a simplifying assumption with regard to Lipschitz continuity in feature space. Fix $q \in [1, \infty]$. Assume there is an $L > 0$ such that, for every $i \in \mathcal{I}_b$,

$$\|\phi_\theta(\boldsymbol{x}^{(i)} + \boldsymbol{\delta}) - \phi_\theta(\boldsymbol{x}^{(i)})\|_2 \quad \leq \quad L \, \|\boldsymbol{\delta}\|_q \qquad \text{for all } \|\boldsymbol{\delta}\|_q \leq \epsilon. \tag{8}$$

Note that Eq. 8 is a local condition, required only within the ϵ-ball around each $\boldsymbol{x}^{(i)}$, $i \in \mathcal{I}_b$, and is therefore weaker than global Lipschitz continuity. We investigate the Lipschitz smoothness of the models analysed in Appendix A.2.1.

Lemma 1: Output margin local Lipschitz continuity Let L_m be the smallest constant such that, for every $i \in \mathcal{I}_b$,

$$|m_{b,t}(\boldsymbol{x}^{(i)} + \boldsymbol{\delta}) - m_{b,t}(\boldsymbol{x}^{(i)})| \quad \leq \quad L_m \, \|\boldsymbol{\delta}\|_q \qquad \text{for all } \|\boldsymbol{\delta}\|_q \leq \epsilon. \tag{9}$$

If Eq. 8 holds, then L_m exists and $L_m \leq L \|\boldsymbol{w}_b - \boldsymbol{w}_t\|_2$.

Lemma 2: Sample-specific guarantee For $i \in \mathcal{I}_b$, if $m_{b,t}(\boldsymbol{x}^{(i)}) > L_m\,\epsilon$, then $m_{b,t}(\boldsymbol{x}^{(i)} + \boldsymbol{\delta}) > 0$ for every $\|\boldsymbol{\delta}\|_q \leq \epsilon$ (cf. Hein & Andriushchenko, 2017, Thm. 2.1; Tsuzuku et al., 2018, Prop. 2).

Proposition 1: Model-specific upper bound Let $\lambda_{m,b} := m_b - L_m\epsilon$, the margin at the class mean reduced by $L_m\epsilon$, assumed $\lambda_{m,b} > 0$. Then

$$\frac{1}{N_b}\left|\left\{i \in \mathcal{I}_b : m_{b,t}(\boldsymbol{x}^{(i)}) \leq L_m\epsilon\right\}\right| \quad \leq \quad \frac{s_{m,b}^2}{s_{m,b}^2 + \lambda_{m,b}^2}. \tag{10}$$

Assuming $\lambda_{m,b} > 0$ requires the class mean to lie on the b side of the b–t decision boundary, with an output margin exceeding $L_m\epsilon$; the bound applies to all samples in $\mathcal{I}_b$, with samples for which $m_{b,t}(\boldsymbol{x}^{(i)}) \leq 0$ counted as not guaranteed safe. Proposition 1 holds for any finite set of class-b samples (e.g., training or test), provided m_b, $s_{m,b}^2$ and L_m are computed on that same set. In addition to the attack budget ϵ and Lipschitz constant L_m, the upper bound on the fraction of samples that cannot be guaranteed safe depends on two properties of the distribution of class-b output margins: its mean m_b and its spread $s_{m,b}^2$.

Assumption 2: Simplification We introduce two further simplifying assumptions in the next proposition. Specifically, we assume that (i) $\gamma := L_m\epsilon / s_{m,b}$ is approximately constant across the models and class pairs compared, and (ii) the b–t decision boundary passes through the midpoint of the class means, $(\boldsymbol{\mu}_b + \boldsymbol{\mu}_t)/2$; equivalently, $m_b + m_t = 0$. The first assumption is specifically made in order to remove reliance on estimates that require gradient information.

Proposition 2: Geometric-only approximation of upper bound For $s_{m,b} > 0$, define the normalized class-mean separation as

$$\rho_{b,t} \quad := \quad \frac{m_b - m_t}{2\,s_{m,b}}, \tag{11}$$

where $m_b - m_t = (\boldsymbol{w}_b - \boldsymbol{w}_t)^\top(\boldsymbol{\mu}_b - \boldsymbol{\mu}_t)$. If $\rho_{b,t} > \gamma$, then

$$\frac{1}{N_b}\left|\left\{i \in \mathcal{I}_b : m_{b,t}(\boldsymbol{x}^{(i)}) \leq L_m\epsilon\right\}\right| \quad \leq \quad \frac{1}{1 + (\rho_{b,t} - \gamma)^2}. \tag{12}$$

We therefore expect the value $\rho_{b,t}$ to be indicative of model robustness, which we refer to as the 'normalized margin separation' henceforth.

Empirical analysis We calculate the normalized margin separation for each of the models specified in Section 2.2, and then determine whether it accurately ranks the various models according to their robust accuracy. This allows us to determine whether feature space geometry alone is sufficient to explain a substantial part of a model's robustness. Specifically, using the training set only, we compute the expected value of $\rho_{b,t}$ uniformly over all ordered class pairs (b, t), and compare the resulting model-level quantity with robust accuracy using Pearson correlation and Kendall's τ across the 71 models. This is shown in Figure 1.

We observe substantial correlations between normalized margin separation and robust accuracy, with Kendall's $\tau = 0.542$ and Pearson's $r = 0.730$. These results suggest that the separation and spread of clean output margins capture substantial variation in robustness across our model set. In particular, this geometric quantity allows models to be ranked using only their clean training representations and output layer parameters, *without calculating a single gradient*. This motivates our feature-level analysis that follows. Furthermore, we also compare our normalized margin separation to other robustness prediction baselines in Appendix A.2.2. Finally, some of the least robust models, particularly Ding2020MMA, deviate from the overall trend. The unusually high feature sensitivity observed for Ding2020MMA in Appendix A.2.1 may help explain this deviation.

4 Feature analysis

We have shown that the robustness of the models we consider is closely tied to their output margin behavior. In this section we take a more granulated view of this behavior and ask how different

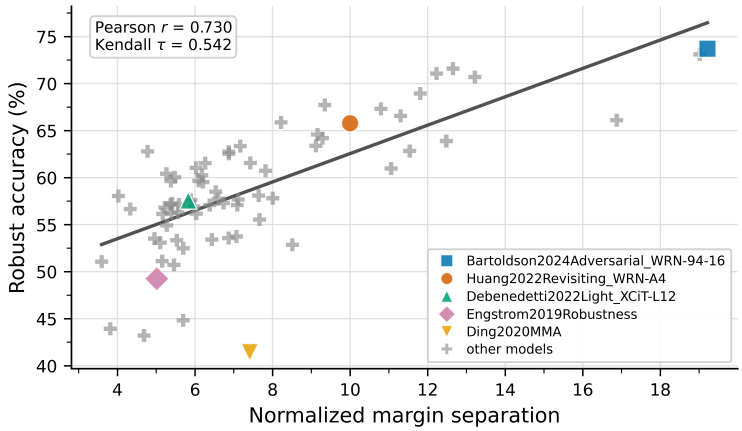

Figure 1: **Normalized margin separation and model robustness.** Robust accuracy versus normalized margin separation averaged uniformly over all ordered class pairs per model. Correlations are calculated across all 71 models.

features *contribute* to the output margin and adversarial examples in general. We start by defining a few relevant quantities for this analysis.

For a feature space vector $\boldsymbol{h}$, we can map each feature into the 'contribution space', which defines the extent to which each feature influences the output margin of the class pair. We define the contribution mapping as

$$\mathcal{C}_{b,t}(\boldsymbol{h}) := \boldsymbol{h} \odot (\boldsymbol{w}_b - \boldsymbol{w}_t), \tag{13}$$

where $\odot$ denotes the Hadamard product. Its jth component, $[\mathcal{C}_{b,t}(\boldsymbol{h})]_j = h_j(w_{b,j} - w_{t,j})$, captures the contribution of the corresponding feature to the output margin between classes b and t. Furthermore, for an adversarial example, we can then directly define the contribution of each feature to the *decrease* in output margin, given by

$$\Delta^{\text{Adv}}\mathcal{C}_{b,t}^{(i)} := \mathcal{C}_{b,t}(\boldsymbol{h}^{(i)}) - \mathcal{C}_{b,t}(\boldsymbol{h}_{\text{adv}}^{(i)}) = \mathcal{C}_{b,t}(-\Delta\boldsymbol{h}^{(i)}) = (-\Delta\boldsymbol{h}^{(i)}) \odot (\boldsymbol{w}_b - \boldsymbol{w}_t). \tag{14}$$

Thus, the contribution mapping decomposes the scalar change in output margin into its feature-wise components. The contribution space is useful as it can suppress large feature changes that do not directly influence the output margin, and similarly accounts for changes in scale between feature dimensions. We can also characterize the typical role of each feature in separating classes b and t. The difference between the average sample contributions of the two classes is given by

$$\begin{aligned}\Delta^{\text{CM}}\mathcal{C}_{b,t} &:= \mathbb{E}\Big[\mathcal{C}_{b,t}(\boldsymbol{h}^{(i)}) \mid y^{(i)} = b\Big] - \mathbb{E}\Big[\mathcal{C}_{b,t}(\boldsymbol{h}^{(i)}) \mid y^{(i)} = t\Big] \\ &= (\boldsymbol{\mu}_b - \boldsymbol{\mu}_t) \odot (\boldsymbol{w}_b - \boldsymbol{w}_t).\end{aligned} \tag{15}$$

Each component $[\Delta^{\text{CM}}\mathcal{C}_{b,t}]_j$ quantifies the contribution of feature j to the average separation between classes b and t induced by the output layer. Equation 15 can therefore be viewed as mapping the class mean separation vector $\boldsymbol{\mu}_b - \boldsymbol{\mu}_t$ into contribution space. Similarly, we define the contribution space counterpart of the reverse boundary normal direction as

$$\Delta^{\perp}\mathcal{C}_{b,t} := \mathcal{C}_{b,t}(-\boldsymbol{n}_{b\to t}) = (\boldsymbol{w}_b - \boldsymbol{w}_t) \odot (\boldsymbol{w}_b - \boldsymbol{w}_t). \tag{16}$$

Note that neither of the latter two quantities is sample-specific; both only depend on the class pair, so we additionally define

$$\Delta^{\text{TM}}\mathcal{C}_{b,t}^{(i)} := (\boldsymbol{h}^{(i)} - \boldsymbol{\mu}_t) \odot (\boldsymbol{w}_b - \boldsymbol{w}_t), \tag{17}$$

i.e. the contribution space counterpart of the (reverse) sample-to-target-mean direction.

4.1 Which features do adversarial examples manipulate?

For a linear decision boundary in feature space, the shortest path to the boundary follows its normal, $\boldsymbol{w}_t - \boldsymbol{w}_b$. An adversarial perturbation in input space, however, passes through the nonlinear feature

Table 1: Mean cosine similarity between the FAB displacement and reference directions in contribution and feature space. The final row considers only samples with input margins $\leq 8/255$.

	Contribution space			Feature space		
Model	$\Delta^{\mathrm{TM}}\mathcal{C}_{b,t}^{(i)}$	$\Delta^{\perp}\mathcal{C}_{b,t}$	$\Delta^{\mathrm{CM}}\mathcal{C}_{b,t}$	$\Delta^{\mathrm{TM},(i)}$	$\Delta_{b,t}^{\perp}$	$\Delta_{b,t}^{\mathrm{CM}}$
Bartoldson2024Adversarial_WRN-94-16	0.747	0.723	**0.816**	0.574	0.634	**0.690**
Huang2022Revisiting_WRN-A4	0.931	0.948	**0.958**	0.790	0.864	**0.878**
Debenedetti2022Light_XCiT-L12	0.604	0.462	**0.750**	0.528	0.321	**0.701**
Engstrom2019Robustness	0.679	0.706	**0.748**	0.600	0.607	**0.676**
Ding2020MMA	0.726	0.686	**0.774**	0.628	0.552	**0.655**
Selected-model mean ($n_{\mathrm{models}} = 5$)	0.738	0.705	**0.809**	0.624	0.595	**0.720**
All-model mean ($n_{\mathrm{models}} = 71$)	0.763	0.700	**0.811**	0.610	0.605	**0.710**
All-model mean, $m^{\mathrm{IS}} \leq 8/255$	0.686	0.697	**0.806**	0.493	0.612	**0.715**

extractor; thus, the smallest perturbation of the input will generally not be in that direction. In addition, various researchers have found that adversarial examples are moved towards the mean representation of the target class (Su et al., 2024; Mao et al., 2019). We instead ask whether the features changed by an adversarial example resemble those that separate the two classes on average, and compare against these two alternatives. We examine this both in raw feature space and in contribution space.

For this analysis we seek perturbations that move a sample just far enough to cross the decision boundary in the input space. This avoids any additional or unnecessary movement in the feature space past the point of misclassification. Therefore, we approximate $m_{b,\star}^{\mathrm{IS}}(\boldsymbol{x}^{(i)})$ with the Fast Adaptive Boundary attack (FAB) (Croce & Hein, 2020). For a sample from class b, we approximate the minimum ℓ_∞ perturbation for each other target class t, and then select the target class with the smallest perturbation. The resulting sample we denote as $\boldsymbol{x}_{\mathrm{FAB}}^{(i)}$, and its corresponding feature representation $\boldsymbol{h}_{\mathrm{FAB}}^{(i)}$. The resulting displacement is then given by $\boldsymbol{\Delta h}^{(i)} = \boldsymbol{h}_{\mathrm{FAB}}^{(i)} - \boldsymbol{h}^{(i)}$. We do this for the first $1\,024$ test set samples for each model, and only consider the correctly classified samples per model. We then calculate $\Delta^{\mathrm{Adv}}\mathcal{C}_{b,t}^{(i)}$, and compare this metric against $\Delta^{\mathrm{CM}}\mathcal{C}_{b,t}$, $\Delta^{\perp}\mathcal{C}_{b,t}$, and $\Delta^{\mathrm{TM}}\mathcal{C}_{b,t}^{(i)}$ via cosine similarity. Furthermore, to motivate our use of the contribution space, we also compare the displacement $-\boldsymbol{\Delta h}^{(i)}$ to the feature space equivalents of the above quantities. Also note that the *train set* samples are used to calculate the mean representations in all cases.

Results Considering the contribution space, we find very high cosine similarity between $\Delta^{\mathrm{Adv}}\mathcal{C}_{b,t}^{(i)}$ and $\Delta^{\mathrm{CM}}\mathcal{C}_{b,t}$ both per model as well as the average across models, and this exceeds both baseline comparisons (Table 1). This indicates that *the features which contribute to adversarial misclassification are also the features that contribute to separating the classes on average*. This is especially interesting given that $\Delta^{\mathrm{CM}}\mathcal{C}_{b,t}$ is fixed for a given class pair and not specific to any sample. The feature space shows similarly favorable results for $\Delta_{b,t}^{\mathrm{CM}}$, although it is clear that the cosine similarities are lower in the raw feature space. Furthermore, we find that these trends also hold on adversarially vulnerable samples, i.e. samples with an input space margin at most the epsilon bound (final row, Table 1). This is a noteworthy finding, as it implies that the adversarial contributions of features are very similar between samples which are vulnerable and those which are not. We show extended results in Appendix A.2.3.

4.2 How many features are vulnerable?

We now aim to determine how many features are necessary for a successful attack, and whether some features are disproportionately important to adversarial misclassification. As such, we aim to restrict the feature space geometry such that only some features can be altered during an adversarial attack. Accordingly, we define a binary feature mask $\boldsymbol{A}_{b,t}^{s}(\rho) \in \{0,1\}^p$ for the class pair (b,t), where $[\boldsymbol{A}_{b,t}^{s}(\rho)]_\ell = 0$ indicates that feature ℓ is fixed to its clean value, while $[\boldsymbol{A}_{b,t}^{s}(\rho)]_\ell = 1$ indicates that

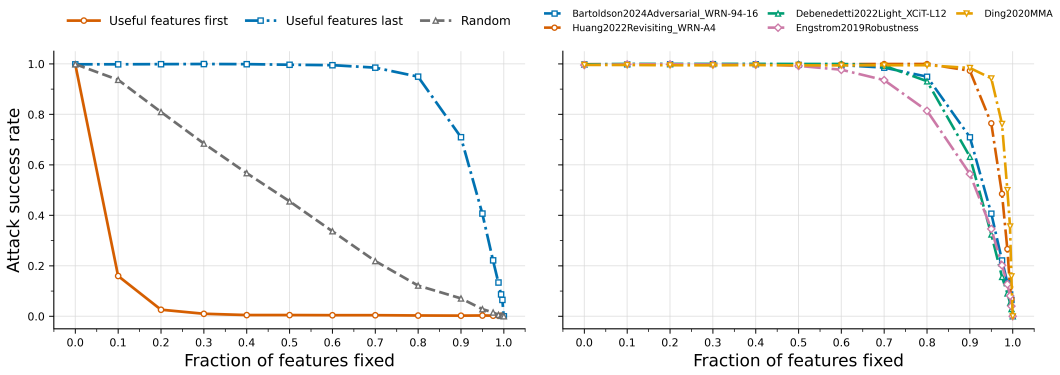

Figure 2: **Attack success under increasing feature fixing.** Attack success rate as the fraction of features fixed to their corresponding clean values, per sample, increases. *Left:* Results for Bartoldson2024Adversarial_WRN-94-16 under useful first, useful last, and random fix orderings. *Right:* Useful last results across the five selected models. The attack success rate is scaled relative to no features fixed, which is approximately 18.5% (1737 out of 9368 correctly classified samples), for the model on the left.

it is allowed to vary. Here ρ is the fraction of features fixed and s denotes their selection order. Then, for a sample $\boldsymbol{x}^{(i)}$ with $y^{(i)} = b$, let $\boldsymbol{x}^{(i)}_{\text{adv},r} = \boldsymbol{x}^{(i)} + \boldsymbol{\delta}^{(i)}_r$ denote the adversarial candidate at attack iteration $r \in \{1, \ldots, R\}$. At each iteration, we intervene on its feature representation such that

$$[\tilde{\boldsymbol{h}}^{(i)}_{\text{adv},r}]_\ell = \begin{cases} [\phi_\theta(\boldsymbol{x}^{(i)})]_\ell, & [\boldsymbol{A}^s_{b,t}(\rho)]_\ell = 0, \\ [\phi_\theta(\boldsymbol{x}^{(i)}_{\text{adv},r})]_\ell, & [\boldsymbol{A}^s_{b,t}(\rho)]_\ell = 1. \end{cases} \tag{18}$$

$\tilde{\boldsymbol{h}}^{(i)}_{\text{adv},r}$ rather than $\phi_\theta(\boldsymbol{x}^{(i)}_{\text{adv},r})$ is then passed to the output layer. The attack is then deemed to be successful if $\tilde{\boldsymbol{h}}^{(i)}_{\text{adv},R}$ is classified as class t.

Motivated by Section 4.1, we rank the features in decreasing order of their class-mean contribution $\Delta^{\text{CM}}\mathcal{C}_{b,t}$, i.e. in the order of how *useful* each feature is to adversarial examples, for each class pair (b, t). The mask $\boldsymbol{A}^s_{b,t}(\rho)$ is constructed by setting a fraction ρ of its entries to zero according to one of three orderings s: *useful first* fixes features from the top of the ranking, *useful last* fixes features from the bottom, and *random* follows a random permutation. We then increase ρ from 0 to 1 for each ordering individually and record the attack success rate for each mask. Practically, we perform the experiment in two stages. Firstly, we attack all the test samples with untargeted 100-step PGD (Madry et al., 2018) and $\epsilon = \frac{8}{255}$. For each successfully attacked sample, we take its resulting predicted class as the target class t and retain the corresponding clean sample. Secondly, we attack only this subset of clean samples using targeted PGD with the three masking methods outlined above, where each sample is matched to its corresponding class pair $\boldsymbol{A}^s_{b,t}(\rho)$ mask.

Results Remarkably, we achieve a (relative) attack success rate of $\approx 70\%$ even when the 90%least useful features are fixed to their clean values (blue curve in Figure 2, left) on the most robust model available. Conversely, if only 10% of the most useful features are fixed, the success rate quickly drops to $< 20\%$ (orange curve), while the random ordering shows a linear decrease in success rate (grey curve). This indicates that only a tiny fraction of the features are necessary per class pair for successful adversarial attacks, and these correspond to those responsible for separating the two classes on average. We repeat the 'useful last' ordering experiment on the other 4 selected models and find similar behavior (Figure 2, right). Finally, we also consider two other methods of restricting the usable features in Appendix A.2.5.

Are useful features unique? For Bartoldson2024Adversarial_WRN-94-16, different class pair contribution vectors have a mean cosine similarity of 0.35, while their top-20% most useful features have a mean Jaccard index of 0.21. This suggests that useful features are mostly unique across class pairs, although there is greater overlap for pairs sharing a class or involving related classes (see

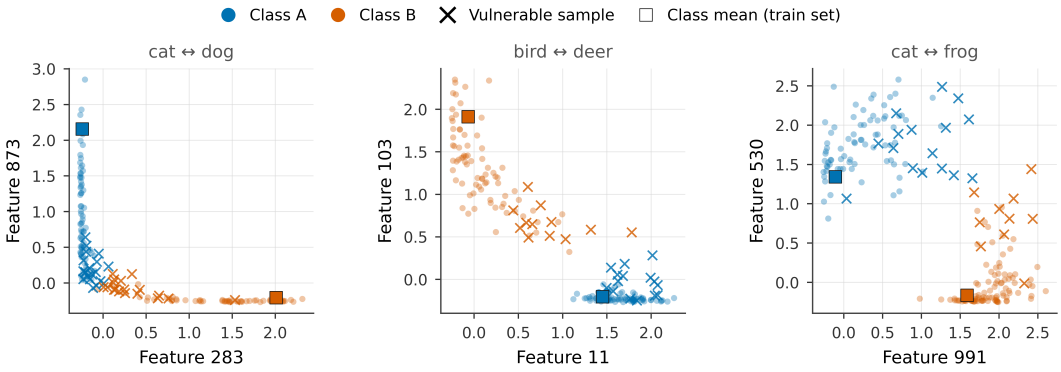

Figure 3: **Sample representations along the two most useful features for three class pairs in Bartoldson2024Adversarial_WRN-94-16.** Colors distinguish the two classes, squares mark their (train set) class means, and crosses indicate samples with an input space margin $\leq 8/255$ toward the other class.

Appendix A.2.6). This implies that useful features have a tendency to specialize to specific class pairs.

4.3 Which features explain sample vulnerability?

In Section 3, we motivated the relationship between output margins and adversarial vulnerability of individual samples. We now take a feature-level view of this relationship and ask: how much of the information about sample vulnerability is captured by the small subset of features that separates the classes?

As an initial visualization, we show the distribution of samples from a class pair by simply plotting the coordinates of the two most useful features according to $\Delta^{\mathrm{CM}}\mathcal{C}_{b,t}$ on a plane, as shown in Figure 3. Despite these scatter plots only showing 2 of 1 024 feature dimensions, we find that one can often visually identify which samples are vulnerable to attack within a class pair. Specifically, vulnerable samples tend to lie farther from their own class mean and closer to the opposing class mean, reflecting weaker separation along these two highly useful features.

We aim to quantify how well separation along a small subset of useful features captures sample vulnerability. Following Ngnawé et al. (2024), we evaluate this as a ranking problem, using the input margin as the reference measure of vulnerability. Their finding that the full output margin correlates strongly with the input margin in robust models provides our benchmark: we ask how much of this correlation is retained when only the most useful features are considered. Accordingly, we define the 'restricted output margin' as

$$m^{\mathrm{R}}_{b,t}\big(\boldsymbol{x}^{(i)};\rho,s\big) := \big[(\boldsymbol{w}_b-\boldsymbol{w}_t)\odot \boldsymbol{A}^{s}_{b,t}(\rho)\big]^{\top}\boldsymbol{h}^{(i)} + (a_b - a_t), \tag{19}$$

where $\boldsymbol{A}^{s}_{b,t}(\rho)$ is again the binary feature mask indicating which features are used ($=1$) and which are ignored ($=0$). Specifically, for our analysis, we calculate $m^{\mathrm{R}}_{b,t}$ with s set to the 'useful last' ordering and $\rho \in \{0.80, 0.90, 0.95, 0.99\}$, or equivalently the output margin using only the top 20%, 10%, 5%, and 1% most useful features. For comparison we also consider the 'random' ranking using the same feature fraction. In each case, we calculate the Kendall's τ_b between $m^{\mathrm{IS}}_{b,\star}$ (calculated in Section 4.1) and $m^{\mathrm{R}}_{b,\star}$ across samples per model, and then average this correlation across models. Furthermore, both the input margin and the restricted output margin independently select the nearest target class, which can potentially differ. Therefore, this is an indication of whether vulnerable samples can be identified *without calculating a single gradient* - the 'true' target class is not determined a priori for the output margin.

Results We show the mean correlation per feature selection in Table 2. For the 'Most useful' selection, we observe that using only the top 5% of features already recovers most of the correlation obtained using the full output margin on average (0.641 versus 0.692), and this substantially

Table 2: Mean Kendall's τ_b across all 71 models between the input space margin and the restricted output margin. Random results are means over 10 feature orderings. See extended results in Appendix A.2.4.

	Fraction of features used				
Selection	1%	5%	10%	20%	All
Most useful	**0.478**	**0.641**	**0.668**	**0.683**	0.692
Random	-0.026	0.199	0.343	0.488	

exceeds the random selection. The correlation increases further to 0.668 and 0.683 when retaining the top 10% and 20% of features, respectively. These results indicate that *the samples which remain vulnerable are poorly separated in terms of the most useful features*, confirming our intuitions from Figure 3.

5 Related work

Estimating model robustness. Song et al. (2025) propose the Robustness Difference Index (RDI), which is close in spirit to our normalized margin separation: it estimates model robustness through within-class compactness and between-class separation in the model's feature space. However, we find that our metric correlates substantially better with robust accuracy, see Appendix A.2.2. Our bound in Section 3 builds on Lipschitz-margin certification (Hein & Andriushchenko, 2017; Tsuzuku et al., 2018), which we use only to motivate a gradient-free predictor, not to certify models.

Adversarial movement and class representations. Prior work relates adversarial feature movement to the target class representation, including through the study of neural collapse under adversarial perturbations (Su et al., 2024; Mao et al., 2019). We distinguish the sample-to-target-mean direction from the difference between the two class means, and find that the latter better describes adversarial movement on average when measured in both feature and contribution space. Our analysis therefore identifies a relationship between the features manipulated by attacks and those responsible for average class separation, without assuming a neural-collapse geometry.

Useful features. Ilyas et al. (2019) characterize feature usefulness through correlation with the label, whereas our class mean contribution additionally accounts for how the learned classification head uses each feature to separate a particular class pair. As shown in Appendix A.2.7, features can exhibit considerable label correlation while contributing little to average class pair separation. This is orthogonal to the view that robust training yields markedly different, more perceptually aligned representations (Tsipras et al., 2019), and that adversarial samples manipulate both robust and non-robust features (Crawford et al., 2026).

Identifying vulnerable samples. Ngnawé et al. (2024) establish margin consistency as a basis for identifying vulnerable predictions using the output margin in robust classifiers. We provide a feature-level account of this relationship by showing that restricting the output margin to a small subset of features selected by their class mean contributions retains much of its correlation with the input space margin. This indicates that the relationship is concentrated in the features most responsible for average class-pair separation.

6 Conclusion

The remaining vulnerability of highly robust models is concentrated in the few features that separate each class pair on clean data: attacks move samples along these features, a small subset of them suffices for most successful attacks, and separation along them identifies which samples remain vulnerable. At the model level, class-mean separation relative to margin spread tracks robust accuracy without computing gradients. Our analysis is limited to penultimate-layer features of CIFAR-10 models under ℓ_∞ attacks with $\epsilon = 8/255$. Within this scope, the results indicate where future defenses might intervene - in how robust models separate classes along their most useful features.

AI USE STATEMENT

Required: In this work, we used generative AI tools to provide feedback on research methodology and experiments, and review hypotheses. We did not use generative AI tools to conceptualize the core ideas, develop the conceptual framework, propose hypotheses, design research methodology or experiments, or to write the initial draft. Mathematical claims were presented to and reviewed with AI tools. One ingredient (the use of Cantelli's inequality in Proposition 1) was AI-proposed. The rest of the claims and proofs were developed by the authors. The remainder of the required disclosure tasks (generating synthetic data sets, translation assistance, cleaning or reformatting datasets) are not applicable to this work.

Recommended: Additionally, we used generative AI tools to co-author Python software code, typeset sections of the paper, generate matplotlib images from our results, and determine whether we have overlooked closely related literature.

Verification: We have reviewed all AI-assisted work. Specifically, all LLM-assisted code was verified and tested for correctness by two of the authors. The use of Cantelli's inequality was reviewed from first principles. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We believe this work to be fully reproducible: we rely on publicly available models (the RobustBench model zoo) and dataset (CIFAR10), and we fully specify the experimental details. Code will be made available after the anonymized review process, while mathematical derivations can be found in Appendix A.1.

REFERENCES

Brian R. Bartoldson, James Diffenderfer, Konstantinos Parasyris, and Bhavya Kailkhura. Adversarial robustness limits via scaling-law and human-alignment studies. In *International Conference on Machine Learning*, 2024.

Jennifer Crawford, Amol Khanna, Fred Lu, Amy R Wagoner, Stella Biderman, Andre T Nguyen, and Edward Raff. Adversarial samples are not created equal. *arXiv preprint arXiv:2601.00577*, 2026.

Francesco Croce and Matthias Hein. Minimally distorted adversarial examples with a fast adaptive boundary attack. In *International Conference on Machine Learning*, 2020.

Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. RobustBench: a standardized adversarial robustness benchmark. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2021.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Sven Gowal, Sylvestre-Alvise Rebuffi, Olivia Wiles, Florian Stimberg, Dan Andrei Calian, and Timothy A Mann. Improving robustness using generated data. In *Advances in Neural Information Processing Systems*, 2021.

Matthias Hein and Maksym Andriushchenko. Formal guarantees on the robustness of a classifier against adversarial manipulation. In *Advances in Neural Information Processing Systems*, 2017.

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems*, 2019.

Yiding Jiang, Dilip Krishnan, Hossein Mobahi, and Samy Bengio. Predicting the generalization gap in deep networks with margin distributions. In *International Conference on Learning Representations*, 2019.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

Chengzhi Mao, Ziyuan Zhong, Junfeng Yang, Carl Vondrick, and Baishakhi Ray. Metric learning for adversarial robustness. In *Advances in Neural Information Processing Systems*, 2019.

Jonas Ngnawé, Sabyasachi Sahoo, Yann Pequignot, Frédéric Precioso, and Christian Gagné. Detecting brittle decisions for free: Leveraging margin consistency in deep robust classifiers. *Advances in Neural Information Processing Systems*, 2024.

Alessandro Salvatore, Stanislav Fort, and Surya Ganguli. Solving adversarial examples requires solving exponential misalignment. *arXiv preprint arXiv:2603.03507*, 2026.

Ali Shafahi, W. Ronny Huang, Christoph Studer, Soheil Feizi, and Tom Goldstein. Are adversarial examples inevitable? In *International Conference on Learning Representations*, 2019.

Jialei Song, Xingquan Zuo, Feiyang Wang, Hai Huang, and Tianle Zhang. RDI: An adversarial robustness evaluation metric for deep neural networks based on model statistical features. In *The 41st Conference on Uncertainty in Artificial Intelligence*, 2025.

Edward Stevinson, Lucas Prieto, Melih Barsbey, and Tolga Birdal. Adversarial vulnerability from interference between features in superposition. In *International Conference on Machine Learning*, 2026.

Jingtong Su, Ya Shi Zhang, Nikolaos Tsilivis, and Julia Kempe. On the robustness of neural collapse and the neural collapse of robustness. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In *International Conference on Learning Representations*, 2019.

Yusuke Tsuzuku, Issei Sato, and Masashi Sugiyama. Lipschitz-margin training: Scalable certification of perturbation invariance for deep neural networks. In *Advances in Neural Information Processing Systems*, 2018.

Zekai Wang, Tianyu Pang, Chao Du, Min Lin, Weiwei Liu, and Shuicheng Yan. Better diffusion models further improve adversarial training. In *International Conference on Machine Learning*, 2023.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P. Xing, Laurent El Ghaoui, and Michael I. Jordan. Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine Learning*, 2019.

A Appendix

A.1 Derivations

Setup For base class b and a target class $t \neq b$, let m_b, $\eta^{(i)}$, $s^2_{m,b}$ and $\Sigma_W^{(b)}$ be as defined in Eqs. 5–7, and define the deviation of each sample from the class mean in feature space as

$$\boldsymbol{z}_b^{(i)} \;:=\; \boldsymbol{h}^{(i)} - \boldsymbol{\mu}_b. \tag{20}$$

From Eq. 2 and the definition of $\boldsymbol{\mu}_b$,

$$\begin{aligned} \frac{1}{N_b}\sum_{i\in\mathcal{I}_b} m_{b,t}(\boldsymbol{x}^{(i)}) &= (\boldsymbol{w}_b - \boldsymbol{w}_t)^\top \frac{1}{N_b}\sum_{i\in\mathcal{I}_b}\boldsymbol{h}^{(i)} + (a_b - a_t) \\ &= (\boldsymbol{w}_b - \boldsymbol{w}_t)^\top \boldsymbol{\mu}_b + (a_b - a_t) \;=\; m_b, \end{aligned}$$

which gives the second equality in Eq. 5. Consequently, $\eta^{(i)} = (\boldsymbol{w}_b - \boldsymbol{w}_t)^\top \boldsymbol{z}_b^{(i)}$, and the deviations sum to zero:

$$\sum_{i \in \mathcal{I}_b} \eta^{(i)} \;=\; (\boldsymbol{w}_b - \boldsymbol{w}_t)^\top \sum_{i \in \mathcal{I}_b} \boldsymbol{z}_b^{(i)} \;=\; 0. \tag{21}$$

Furthermore,

$$\begin{aligned} s_{m,b}^2 &= \frac{1}{N_b} \sum_{i \in \mathcal{I}_b} (\boldsymbol{w}_b - \boldsymbol{w}_t)^\top \boldsymbol{z}_b^{(i)} \boldsymbol{z}_b^{(i)\top} (\boldsymbol{w}_b - \boldsymbol{w}_t) \\ &= (\boldsymbol{w}_b - \boldsymbol{w}_t)^\top \Sigma_W^{(b)} (\boldsymbol{w}_b - \boldsymbol{w}_t), \end{aligned}$$

which gives the second equality in Eq. 7.

Lemma 1: Output margin local Lipschitz continuity Let L_m be the smallest constant such that, for every $i \in \mathcal{I}_b$,

$$|m_{b,t}(\boldsymbol{x}^{(i)} + \boldsymbol{\delta}) - m_{b,t}(\boldsymbol{x}^{(i)})| \;\le\; L_m \, \|\boldsymbol{\delta}\|_q \qquad \text{for all } \|\boldsymbol{\delta}\|_q \le \epsilon.$$

If Eq. 8 holds, then L_m exists and $L_m \le L\|\boldsymbol{w}_b - \boldsymbol{w}_t\|_2$.

Proof. Let $\Delta \boldsymbol{h} := \phi_\theta(\boldsymbol{x}^{(i)} + \boldsymbol{\delta}) - \phi_\theta(\boldsymbol{x}^{(i)})$ with $\|\boldsymbol{\delta}\|_q \le \epsilon$. Using Eq. 2, Cauchy–Schwarz and Eq. 8, it follows that:

$$\begin{aligned} |m_{b,t}(\boldsymbol{x}^{(i)} + \boldsymbol{\delta}) - m_{b,t}(\boldsymbol{x}^{(i)})| &= |(\boldsymbol{w}_b - \boldsymbol{w}_t)^\top \Delta \boldsymbol{h}| \\ &\le \|\boldsymbol{w}_b - \boldsymbol{w}_t\|_2 \, \|\Delta \boldsymbol{h}\|_2 \;\le\; L\|\boldsymbol{w}_b - \boldsymbol{w}_t\|_2 \, \|\boldsymbol{\delta}\|_q. \end{aligned}$$

Eq. 9 therefore holds with the constant $L\|\boldsymbol{w}_b - \boldsymbol{w}_t\|_2$, so the smallest such constant exists (the set of valid constants is closed, so the infimum is attained) and is at most $L\|\boldsymbol{w}_b - \boldsymbol{w}_t\|_2$.

Lemma 2: Sample-specific guarantee For $i \in \mathcal{I}_b$, if

$$m_{b,t}(\boldsymbol{x}^{(i)}) \;>\; L_m \, \epsilon, \tag{22}$$

then $m_{b,t}(\boldsymbol{x}^{(i)} + \boldsymbol{\delta}) > 0$ for every $\|\boldsymbol{\delta}\|_q \le \epsilon$. This is a single-pair, fixed-radius form of the local Lipschitz-margin guarantee of Hein & Andriushchenko (2017, Thm. 2.1). Combined with Lemma 1, it is a local, ℓ_q version of the certificate in Tsuzuku et al. (2018, Prop. 2).

Proof. Using Eq. 9 and $\|\boldsymbol{\delta}\|_q \le \epsilon$, it follows that:

$$m_{b,t}(\boldsymbol{x}^{(i)} + \boldsymbol{\delta}) - m_{b,t}(\boldsymbol{x}^{(i)}) \;\ge\; -L_m \, \|\boldsymbol{\delta}\|_q \;\ge\; -L_m \epsilon.$$

Finally, for the condition specified in Eq. 22 it will hold that

$$m_{b,t}(\boldsymbol{x}^{(i)} + \boldsymbol{\delta}) \;\ge\; m_{b,t}(\boldsymbol{x}^{(i)}) - L_m \epsilon \;>\; 0.$$

The condition is sufficient, not necessary: a sample violating Eq. 22 cannot be guaranteed safe, but is not necessarily attackable. Since $L_m \le L\|\boldsymbol{w}_b - \boldsymbol{w}_t\|_2$, Eq. 22 is never harder to satisfy than the feature space condition $m_{b,t}(\boldsymbol{x}^{(i)}) > L\epsilon \, \|\boldsymbol{w}_b - \boldsymbol{w}_t\|_2$.

Proposition 1: Model-specific upper bound Let $\lambda_{m,b} := m_b - L_m \epsilon$, assumed $\lambda_{m,b} > 0$. Then

$$\frac{1}{N_b} \left| \left\{ i \in \mathcal{I}_b : m_{b,t}(\boldsymbol{x}^{(i)}) \le L_m \epsilon \right\} \right| \;\le\; \frac{s_{m,b}^2}{s_{m,b}^2 + \lambda_{m,b}^2}.$$

Proof. By the definition of $\eta^{(i)}$, $m_{b,t}(\boldsymbol{x}^{(i)}) = m_b + \eta^{(i)}$. Then

$$m_{b,t}(\boldsymbol{x}^{(i)}) \le L_m \epsilon \;\iff\; \eta^{(i)} \;\le\; L_m \epsilon - m_b \;=\; -\lambda_{m,b}.$$

Define $U_m := \{ i \in \mathcal{I}_b : \eta^{(i)} \le -\lambda_{m,b} \}$. Then the left-hand side of Eq. 10 is exactly $|U_m|/N_b$. Let I be an index drawn uniformly from $\mathcal{I}_b$ and $u_m := \eta^{(I)}$. Then

$$\Pr(u_m \le -\lambda_{m,b}) \;=\; \Pr(I \in U_m) \;=\; \frac{|U_m|}{N_b}.$$

By Eq. 21,

$$\mathbb{E}[u_m] \;=\; \frac{1}{N_b}\sum_{i\in\mathcal{I}_b}\eta^{(i)} \;=\; 0,$$

and, since $\mathbb{E}[u_m]=0$,

$$\mathrm{Var}(u_m) \;=\; \mathbb{E}[u_m^2] \;=\; \frac{1}{N_b}\sum_{i\in\mathcal{I}_b}\left(\eta^{(i)}\right)^2 \;=\; s_{m,b}^2.$$

Thus, u_m has mean 0 and variance $s_{m,b}^2$, and $\lambda_{m,b}>0$ by assumption. Cantelli's (one-sided Chebyshev) inequality states that, for a random variable u with mean 0, variance σ^2 and any $\lambda>0$, $\Pr(u\le-\lambda)\le\sigma^2/(\sigma^2+\lambda^2)$. Applied to u_m with $\lambda=\lambda_{m,b}$, it gives

$$\Pr(u_m\le-\lambda_{m,b}) \;\le\; \frac{s_{m,b}^2}{s_{m,b}^2+\lambda_{m,b}^2}$$

$$\Longrightarrow\; \frac{1}{N_b}\left|\left\{i\in\mathcal{I}_b : m_{b,t}(\boldsymbol{x}^{(i)})\le L_m\epsilon\right\}\right| \;=\; \Pr(u_m\le-\lambda_{m,b}) \;\le\; \frac{s_{m,b}^2}{s_{m,b}^2+\lambda_{m,b}^2}.$$

The proof does not use the minimality of L_m: Lemma 2 and Proposition 1 hold with L_m replaced by any constant satisfying Eq. 9, such as $L\|\boldsymbol{w}_b-\boldsymbol{w}_t\|_2$ (Lemma 1). Since the bound decreases in $\lambda_{m,b}$ and $m_b-L_m\epsilon\ge m_b-L\|\boldsymbol{w}_b-\boldsymbol{w}_t\|_2\,\epsilon$, using L_m gives a bound that is never looser than using $L\|\boldsymbol{w}_b-\boldsymbol{w}_t\|_2$, and strictly tighter whenever $L_m<L\|\boldsymbol{w}_b-\boldsymbol{w}_t\|_2$, $\epsilon>0$ and $s_{m,b}^2>0$.

Writing the dependence on t explicitly, let $\lambda_{m,b,t}:=m_b-L_m\epsilon$, $s_{m,b,t}^2$ and L_m be computed for the pair (b,t), and assume

$$\lambda_{m,b,t} \;>\; 0 \qquad \text{for all } t\neq b, \tag{23}$$

i.e., the class-b mean lies on the b side of every b–t decision boundary, with an output margin exceeding $L_m\epsilon$. Proposition 1 then applies to each target t, and the union bound gives

$$\frac{1}{N_b}\left|\left\{i\in\mathcal{I}_b : \exists\, t\neq b,\; m_{b,t}(\boldsymbol{x}^{(i)})\le L_m\epsilon\right\}\right| \;\le\; \sum_{t\neq b}\frac{s_{m,b,t}^2}{s_{m,b,t}^2+\lambda_{m,b,t}^2}.$$

Assumption 2: Simplification Here m_t is as in Eq. 5, and conditions (i) and (ii) are as defined in Section 3. As for class b, Eq. 2 and the definition of $\boldsymbol{\mu}_t$ give $m_t=(\boldsymbol{w}_b-\boldsymbol{w}_t)^\top\boldsymbol{\mu}_t+(a_b-a_t)=\frac{1}{N_t}\sum_{i\in\mathcal{I}_t}m_{b,t}(\boldsymbol{x}^{(i)})$. With $\bar{\boldsymbol{h}}:=(\boldsymbol{\mu}_b+\boldsymbol{\mu}_t)/2$,

$$(\boldsymbol{w}_b-\boldsymbol{w}_t)^\top\bar{\boldsymbol{h}}+(a_b-a_t) \;=\; \tfrac{1}{2}\left(m_b+m_t\right),$$

so the b–t decision boundary passes through $\bar{\boldsymbol{h}}$ if and only if $m_b+m_t=0$, which establishes the equivalence in (ii).

Proposition 2: Geometric-only approximation of upper bound Under Assumption 2, if $\rho_{b,t}>\gamma$, then Eq. 12 holds.

Proof. By (ii) and Eq. 11,

$$m_b \;=\; \tfrac{1}{2}(m_b-m_t) \;=\; \tfrac{1}{2}(\boldsymbol{w}_b-\boldsymbol{w}_t)^\top(\boldsymbol{\mu}_b-\boldsymbol{\mu}_t) \;=\; \rho_{b,t}\,s_{m,b}.$$

With (i), $\lambda_{m,b}=m_b-L_m\epsilon=(\rho_{b,t}-\gamma)\,s_{m,b}>0$, so Proposition 1 applies and gives

$$\frac{s_{m,b}^2}{s_{m,b}^2+\lambda_{m,b}^2} \;=\; \frac{s_{m,b}^2}{s_{m,b}^2+(\rho_{b,t}-\gamma)^2 s_{m,b}^2} \;=\; \frac{1}{1+(\rho_{b,t}-\gamma)^2}.$$

A.2 Additional results

A.2.1 Lipschitz smoothness of model set

To determine how smoothly the feature space varies with perturbations in the input space, we measure a normalized variant of the Lipschitz constant.

Setup We apply 100-step PGD to the test set and retain the successful adversarial examples for each model. For each $\boldsymbol{x}_{\text{adv}}^{(i)}$, we calculate

$$\bar{L}(\boldsymbol{x}^{(i)}) = \frac{\|\boldsymbol{h}_{\text{adv}}^{(i)} - \boldsymbol{h}^{(i)}\|_\infty}{\sqrt{\nu_\theta}\,\|\boldsymbol{x}_{\text{adv}}^{(i)} - \boldsymbol{x}^{(i)}\|_\infty}. \tag{24}$$

where ν_θ is the feature space normalization proposed by Jiang et al. (2019), defined as

$$\nu_\theta = \frac{1}{N}\sum_{i=1}^{N} \|\boldsymbol{h}^{(i)} - \boldsymbol{\mu}\|_2^2, \tag{25}$$

where $\boldsymbol{\mu}$ is the mean feature representation of all samples. Equivalently, ν_θ is the sum of the individual feature variances. To compare models normalization is required, as each model's feature space can use completely different scales - so we normalize by the natural spread expected across different samples as in prior work (Jiang et al., 2019). Note that we use the train set samples to calculate ν_θ.

Finally, we take the mean of $\bar{L}$ across all adversarial examples, which we define as the 'Mean normalized Lipschitz estimate'. We plot this against the robust accuracy of each model we consider in Figure 4, and we also include the 'Standard' model as reference.

Results We find that most models behave rather similarly, with the exception of Standard (as expected) and Ding2020MMA. This indicates that the feature extractor of the latter model is much more sensitive than the other models we consider, and is closer in resemblance to a non-robust model. We therefore interpret the results for Ding2020MMA with caution, given its atypical nature, but still include it in all our analyses.

Excluded models Not shown in Figure 4 are the four models we exclude from all our analysis, namely the three ensemble models: Bai2024MixedNUTS, Bai2023Improving_edm, Chen2020Adversarial, and one with unreliable gradients: Kang2021Stable.

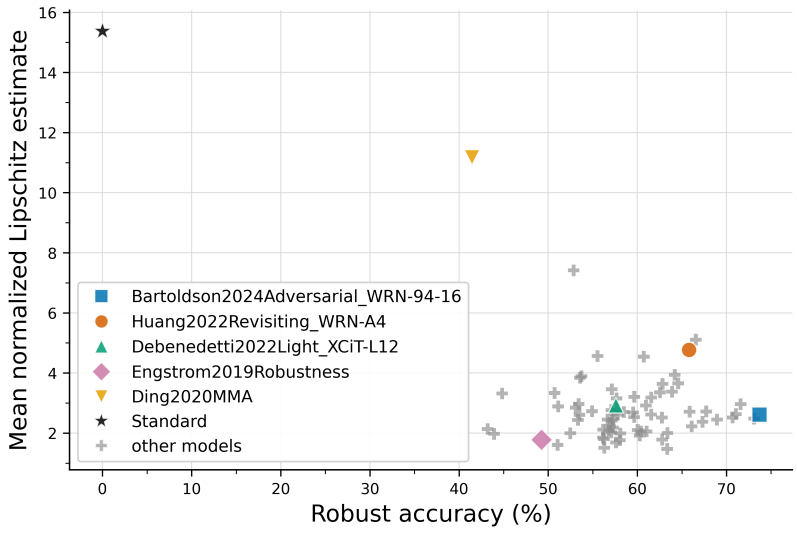

Figure 4: **Normalized feature sensitivity across models.** Mean feature space displacement per unit input perturbation, normalized by each model's feature space variation, plotted against robust accuracy.

A.2.2 Robustness-prediction baselines

This analysis is related to Section 3. We compare normalized margin separation with the following model-level baselines and ablations.

Normalized base margin $m_b/s_{m,b}$. This retains the base class margin and its spread normalization, but omits the target class mean.

Unnormalized margin separation $m_b - m_t$. This retains the separation between class means, but omits normalization by the base-class margin spread.

Raw mean margin $m_{b,\star}(\boldsymbol{x}^{(i)})$ the mean signed margin between the ground truth logit and the largest competing logit. Misclassified samples therefore contribute negative margins.

Robustness Difference Index (RDI) Song et al. (2025) propose the Robustness Difference Index (RDI), which is close in spirit to our metric: an estimate of robustness based on within-class compactness and between-class separation in feature space. Let $\widehat{\mathcal{I}}_k$ contain the indices of samples predicted as class k, $\widehat{N}_k$ their number, $\widehat{\boldsymbol{\mu}}_k$ their mean representation, and $\widehat{\boldsymbol{\mu}}_0$ the mean of these class means. Then

$$\text{IntraD} := \frac{1}{K}\sum_{k=1}^{K}\frac{1}{\widehat{N}_k}\sum_{i\in\widehat{\mathcal{I}}_k}\left\|\boldsymbol{h}^{(i)} - \widehat{\boldsymbol{\mu}}_k\right\|_2, \tag{26}$$

$$\text{InterD} := \frac{1}{K}\sum_{k=1}^{K}\left\|\widehat{\boldsymbol{\mu}}_k - \widehat{\boldsymbol{\mu}}_0\right\|_2, \tag{27}$$

$$\text{RDI} := \frac{\text{InterD} - \text{IntraD}}{\max(\text{InterD}, \text{IntraD})}. \tag{28}$$

Higher RDI indicates greater between-class separation relative to within-class spread.

Results As shown in Table 3, normalized margin separation achieves the highest Kendall correlation with robust accuracy, substantially exceeding RDI. The normalized base margin performs virtually identically (in fact, equal Pearson correlation), as expected under our midpoint assumption: when $m_t = -m_b$, $(m_b - m_t)/(2s_{m,b}) = m_b/s_{m,b}$. Both unnormalized margin metrics perform poorly, highlighting the importance of accounting for within-class margin spread.

Table 3: Correlation between each metric and robust accuracy across all 71 models.

Metric	Kendall's τ	Pearson's r
Normalized margin separation	**0.5423**	**0.7302**
RDI	0.2693	0.3536
Normalized base margin	0.5382	**0.7302**
Unnormalized margin separation	-0.1510	-0.2428
Raw mean margin	0.1824	0.0933

A.2.3 ALL CONTRIBUTION SIMILARITIES

This analysis complements Section 4.1. In Figure 5 we show the mean cosine similarity between $\Delta^{\text{Adv}}\mathcal{C}_{b,t}^{(i)}$ and $\Delta^{\text{CM}}\mathcal{C}_{b,t}$ for all 71 models we consider on the x-axis. We then contrast this with the same measurement for $\Delta^{\perp}\mathcal{C}_{b,t}$ and $\Delta^{\text{TM}}\mathcal{C}_{b,t}^{(i)}$ on the y-axis, respectively. For all models to the right of the identity line, $\Delta^{\text{CM}}\mathcal{C}_{b,t}$ is a better predictor of $\Delta^{\text{Adv}}\mathcal{C}_{b,t}^{(i)}$ than the alternatives.

In Table 4 we complement Table 1 and show the full contribution/feature space similarity results for samples with an input margin less than $\frac{8}{255}$.

A.2.4 ALL MARGIN CORRELATIONS

This analysis is an extension of the results presented in Section 4.3. Table 5 shows the full correlation results between the restricted output margin and input margin, coroborating those presented in the main paper. We note that the Ding2020MMA model shows contrary behavior - we suspect this is due to its atypical feature sensitivity (see Appendix A.2.1).

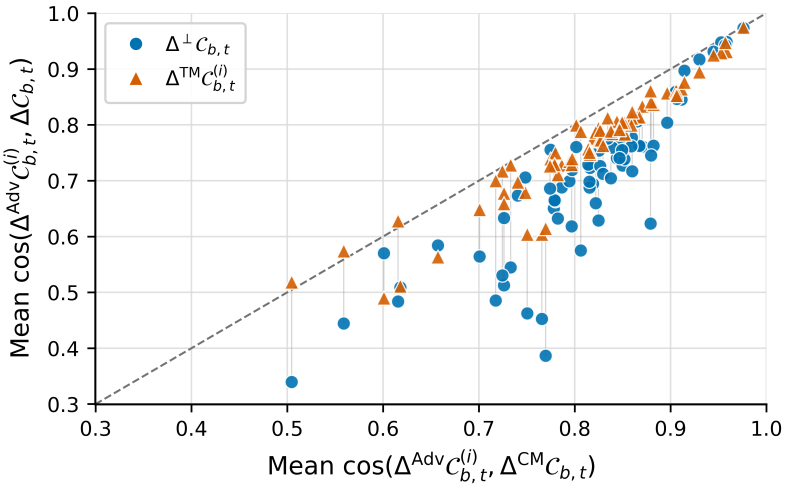

Figure 5: **Mean cosine similarity between adversarial contribution and reference contribution vectors across 71 models.** Each model is represented by two symbols, comparing class mean similarity on the x-axis with the boundary normal (blue dots) and sample-to-target similarity (orange triangle) on the y-axis. Points below the dashed identity line indicate stronger alignment with the class mean contribution vector.

Table 4: The same as Table 1, but restricted to samples with $m^{\mathrm{IS}}_{b,\star}(\boldsymbol{x}^{(i)}) \leq 8/255$.

	Contribution space			Feature space		
Model	$\Delta^{\mathrm{TM}}\mathcal{C}^{(i)}_{b,t}$	$\Delta^{\perp}\mathcal{C}_{b,t}$	$\Delta^{\mathrm{CM}}\mathcal{C}_{b,t}$	$\Delta^{\mathrm{TM},(i)}$	$\Delta^{\perp}_{b,t}$	$\Delta^{\mathrm{CM}}_{b,t}$
Bartoldson2024Adversarial_WRN-94-16	0.626	0.760	**0.809**	0.394	0.664	**0.682**
Huang2022Revisiting_WRN-A4	0.899	0.953	**0.963**	0.757	0.890	**0.907**
Debenedetti2022Light_XCiT-L12	0.513	0.455	**0.755**	0.434	0.314	**0.711**
Engstrom2019Robustness	0.556	0.676	**0.717**	0.444	0.569	**0.647**
Ding2020MMA	0.710	0.739	**0.805**	0.558	0.618	**0.674**
Selected-model mean ($n_{\mathrm{models}} = 5$)	0.661	0.717	**0.810**	0.517	0.611	**0.724**
All-model mean ($n_{\mathrm{models}} = 71$)	0.686	0.697	**0.806**	0.493	0.612	**0.715**

A.2.5 Limiting attacks via gradient masking and flipping

In Section 4.2 we fix some features to their corresponding clean values and then determine whether an adversarial attack can still be successful. Here, we consider two additional methods of restricting the features.

Gradient masking We retain the same mask method, although in this case each feature can move freely, and only the gradients of the masked features are zeroed during the PGD gradient calculation. This implies that the directional information that the feature provides to the attack gradients is removed. In doing so, we can determine whether the attack must explicitly exploit these features, or whether it can induce adversarial change in them indirectly through the remaining features.

Flipping In this case, we find each adversarial example beforehand. We then reset all the features to their clean values (the masked state), before gradually changing each feature to that of the adversarial value (unmasked state). This implies that we do not interfere with the adversarial attack process, but rather determine which features are sufficient for adversarial misclassification in a 'natural' adversarial example.

Results We show the masking and flipping results in Figure 6. As with the feature fixing experiment, we find that masking the gradients of the most useful features quickly reduces the attack

Table 5: Kendall's τ_b between the input space margin and the restricted output margin. Random results are means over 10 feature orderings.

		Fraction of features used				
Model	Selection	1%	5%	10%	20%	All
Bartoldson2024Adversarial_WRN-94-16	Most useful	**0.467**	**0.664**	**0.686**	**0.685**	0.681
	Random	0.154	0.422	0.503	0.604	
Huang2022Revisiting_WRN-A4	Most useful	**0.507**	**0.635**	**0.660**	**0.668**	0.668
	Random	-0.059	0.148	0.305	0.445	
Debenedetti2022Light_XCiT-L12	Most useful	**0.455**	**0.615**	**0.644**	**0.673**	0.647
	Random	0.083	0.280	0.351	0.467	
Engstrom2019Robustness	Most useful	**0.581**	**0.643**	**0.671**	**0.690**	0.703
	Random	0.036	0.452	0.583	0.649	
Ding2020MMA	Most useful	0.088	0.120	0.143	0.146	0.163
	Random	**0.105**	**0.138**	**0.172**	**0.155**	
All-model mean (71 models)	Most useful	**0.478**	**0.641**	**0.668**	**0.683**	0.692
	Random	-0.026	0.199	0.343	0.488	

success rate, which implies the directional information these features provide is essential. Remarkably, when considering the 'useful last' curve(s), we find that 99.70% of the features can be masked (second-to-lastmarker) and the attack is still successful in almost all cases. On the other hand, the 'random' ordering is slightly less successful, likely because some very useful features are masked earlier.

The flipping experiment yields the complementary result: flipping the most useful features first produces misclassification using a small fraction of features, whereas introducing these features last substantially delays the success of the attack. The agreement between all these experiments suggests that the observed ordering is not merely an artifact of constraining PGD.

A.2.6 Similarity between class pairs

As mentioned in Section 4.2, we calculate the cosine similarity and the top-20% feature Jaccard index between the $\Delta^{\text{CM}}\mathcal{C}_{b,t}$ vectors for each unique class pair. We show a heatmap of the similarity in the lower triangle of Figure 7, while the upper triangle shows the Jaccard index between the top 20% most useful features per pair.

The cosine similarities are consistently positive, indicating that the contribution vectors for different class pairs share a common direction, to an extent. However, the overlap between their most useful features is slight in most cases, confirming that the features most relevant to class separation vary across pairs. That said, both measures tend to be higher for pairs that share a class. There are particularly high overlaps when the remaining classes are also related, e.g. (cat,ship) and (dog,ship). To illustrate this, we show the top 5 most similar pairs in Table 6, which all contain both the 'dog' and 'cat' class, except the fifth which contains the related 'automobile' and 'truck' (paired with 'frog').

Table 6: Cosine similarity and Jaccard index of the five most similar pairs of class-mean contribution vectors.

Class pair 1	Class pair 2	Cosine similarity	Jaccard index
(cat,ship)	(dog,ship)	0.738	0.419
(automobile,cat)	(automobile,dog)	0.698	0.449
(cat,truck)	(dog,truck)	0.684	0.424
(airplane,cat)	(airplane,dog)	0.682	0.371
(frog,automobile)	(frog,truck)	0.679	0.327

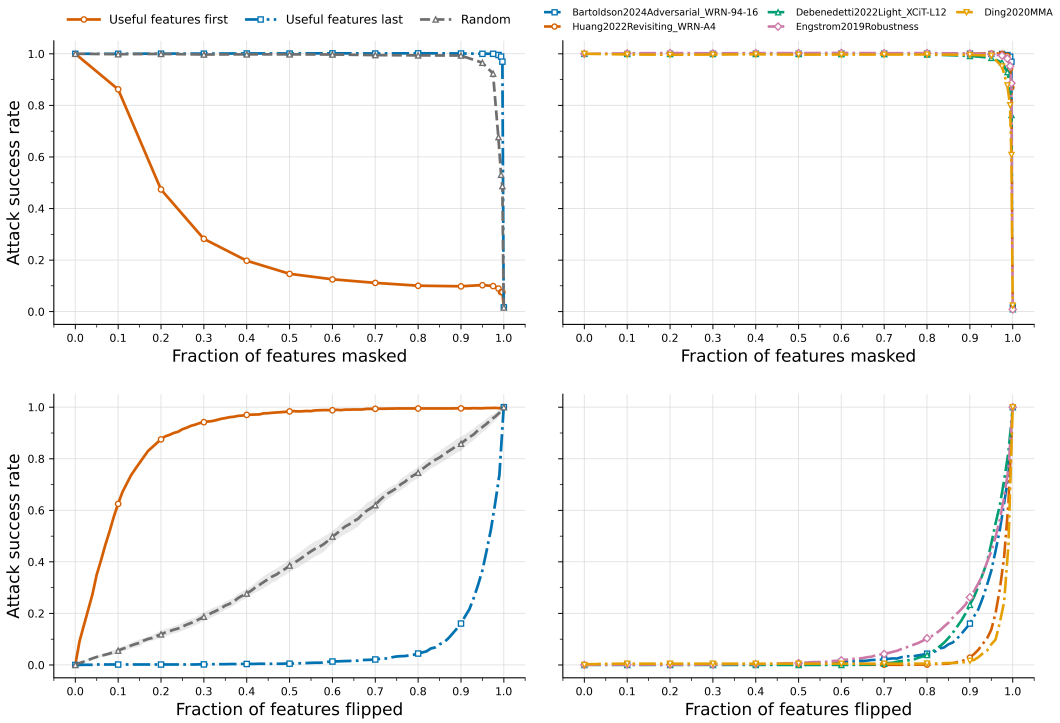

Figure 6: Restricting adversarial features through gradient masking (top) and feature flipping (bottom). Left: results for Bartoldson2024Adversarial_WRN-94-16 using all three orderings. Right: Useful last results for the five selected models.

A.2.7 Correlation versus class mean contribution

Throughout, we have relied on our mean contribution vector $\Delta^{\text{CM}}\mathcal{C}_{b,t}$ to determine which features are *useful* for a given class pair. However, Ilyas et al. (2019) define 'κ-useful features' as features which are at least κ-correlated with the label under a given data distribution. Formally, for binary labels $y \in \{-1, +1\}$, a feature is κ-useful if

$$\mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[y \cdot [\phi(\boldsymbol{x})]_j\right] \geq \kappa, \tag{29}$$

where $[\phi(\boldsymbol{x})]_j$ denotes the value of the j-th feature for input $\boldsymbol{x}$, $\mathcal{D}$ is the data distribution, and we assume that each feature is normalized to have zero mean and unit variance under $\mathcal{D}$. This is equivalent to the Pearson correlation between the feature and the label when the two classes are equally likely. In this section we compare these two definitions of 'usefulness'.

Setup To calculate the correlation, we restrict the dataset to classes b and t, assign labels $+1$ and -1, respectively, and normalize each feature to have zero mean and unit variance over this dataset. We then calculate each feature's Pearson correlation with these binary labels and compare it with the corresponding class mean contribution $[\Delta^{\text{CM}}\mathcal{C}_{b,t}]_j$. To match $\Delta^{\text{CM}}\mathcal{C}_{b,t}$, we calculate these correlations using the train set.

Results We show a scatter plot of the absolute value of the correlation and the class mean contribution per feature for four class pairs, using the Bartoldson2024Adversarial_WRN-94-16 model in Figure 8. More specifically, we show the four most common class pairs between which adversarial examples can be found. We find that find that features which have a very high correlation with the class label also tend to have a very high class mean contribution. On the other hand, there are many features with considerable correlation and near 0 contribution. This is interesting as it implies that these features contain discriminative information, but they are not utilized by the classification head. We suspect that this might be due to lower correlation features being more vulnerable to adversarial attack, although we leave further investigation to future work. Taken together, we believe that the

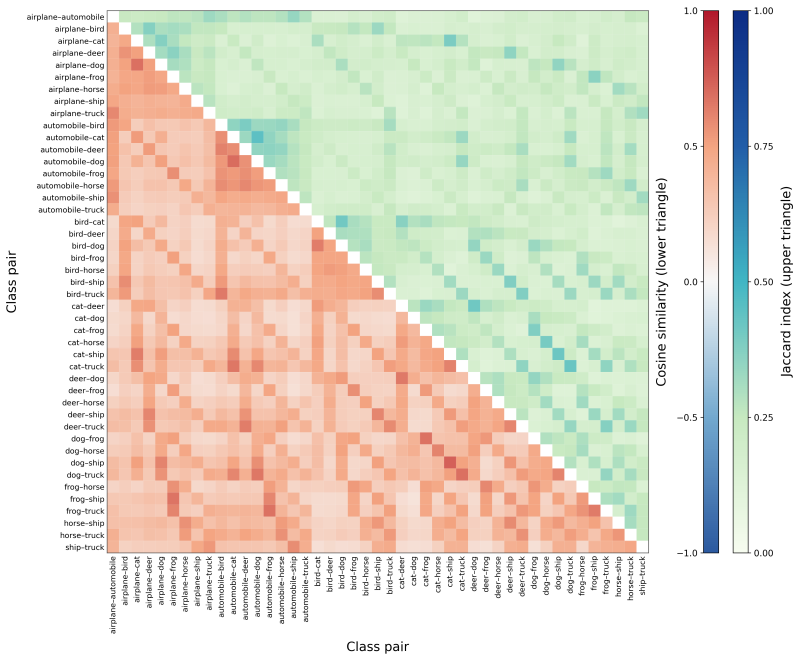

Figure 7: Similarity of class pair contribution vectors for Bartoldson2024Adversarial_WRN-94-16 for unique pairs. The lower triangle shows the cosine similarity between $\Delta^{\mathrm{CM}}\mathcal{C}_{b,t}$ vectors, while the upper triangle shows the Jaccard index between their top 20% most useful features.

class mean contribution metric is a better indication than correlation of which features are actually used to separate two classes.

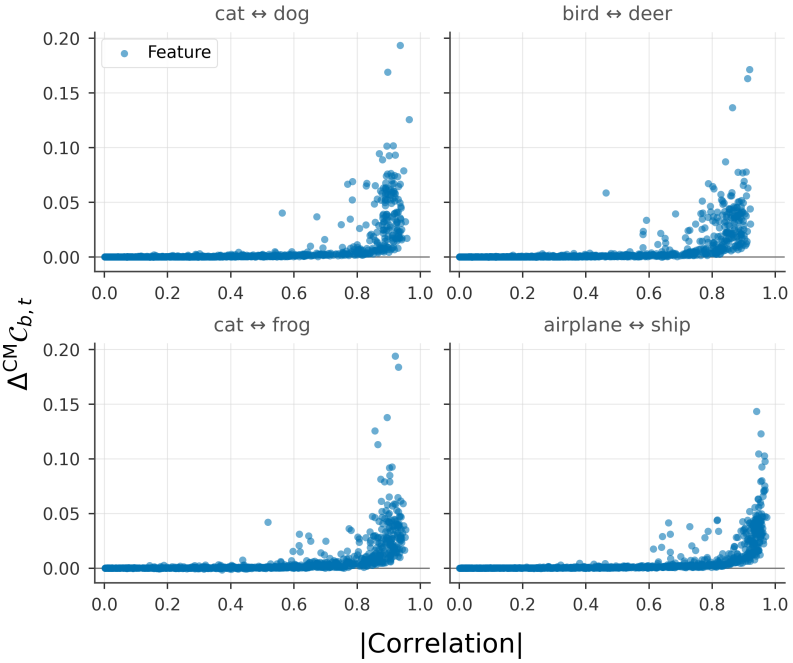

Figure 8: Absolute Pearson correlation with the binary class label versus class mean contribution $[\Delta^{\mathrm{CM}}\mathcal{C}_{b,t}]_j$ for the four most common adversarial class pairs in Bartoldson2024Adversarial_WRN-94-16. Each point represents one feature, with both quantities calculated on the training set.