A GEOMETRIC OBSTRUCTION TO ROBUST AND ACCURATE DIFFERENTIABLE CLASSIFIERS

Anonymous authors
Paper under double-blind review

ABSTRACT

We ask when a robust and accurate classifier can be represented as a differentiable function. Reading the gradient of a hypothesis as a *sensitivity field*, we observe that a differentiable classifier's field must be conservative — curl-free and path-independent — while robustness under the manifold hypothesis imposes a separate *strong tangency* condition: invariance to perturbations off a data manifold $\mathcal{X}$, represented as the zero-set of ϕ. Given an accurate hypothesis ψ, we construct the tangent field $F = P_\phi \nabla \psi$, where P_ϕ orthogonally projects onto $\ker(J_\phi)$, and show it reproduces ψ exactly on $\mathcal{X}$ via path integration. We then characterize when F remains conservative: testing the antisymmetric part of J_F with arbitrary vectors, it decomposes into bilinear forms whose leading term is governed by the Hessian of ϕ, i.e., the curvature of $\mathcal{X}$, and vanishes only in degenerate cases (flat manifolds, or $l = 1$ normal-normal directions). For the unit circle, this yields a rigidity result under a specific strong tangency condition — any hypothesis continuous at the origin and strongly tangent to S^1 must be constant on it — showing the obstruction might be absolute rather than generic. To test our thesis, we relax conservativeness and represent classifiers as path-dependent functionals evaluated by line integrals of a learned field, using an oracle-path scheme for adversarial training. On MNIST, FashionMNIST, and CIFAR-10, adversarially trained path-dependent hypotheses achieve substantially higher robust accuracy than gradient-field baselines at comparable clean accuracy, align more closely with a surrogate data-manifold tangent space, and exhibit statistically significant non-zero circulation — direct evidence that the learned fields are not gradients. Together, the theory and experiments indicate that the accuracy–robustness trade-off is, at least in part, a structural consequence of manifold curvature, motivating path-dependent (multi-valued) architectures as a principled alternative to single-valued differentiable classifiers.

1 INTRODUCTION

The phenomenon of adversarial examples has challenged our understanding of modern machine learning systems. Despite achieving superhuman performance on many tasks, state-of-the-art models remain highly sensitive to small, often imperceptible perturbations (Bala et al., 2026). Explaining this vulnerability has proven difficult, and there is still no consensus on its root causes (Han et al., 2023). Existing perspectives range from data distribution mismatch (Grosse et al., 2017) to model linearity (Goodfellow et al., 2015), yet none fully reconcile the coexistence of high accuracy and extreme brittleness.

A natural interpretation is that adversarial examples arise because models fail to capture the "true" underlying patterns in the data. However, efforts to improve robustness often lead to a degradation in accuracy on benign samples (Tsipras et al., 2019), suggesting a more intricate relationship between predictive features and adversarial directions. This apparent trade-off raises a fundamental question: is the tension between accuracy and robustness a property of the data, or of the hypothesis class used to model it?

For a differentiable hypothesis, we argue that the tension is partly a conflict between two constraints acting on the same object. The key observation is that adversarial robustness is, at its core, a constraint on the gradient of a differentiable hypothesis. This is not merely a conceptual point: standard

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.

robustness methods act directly on the gradient. It has been shown that gradient regularization improves robustness (Jukić & Šnajder, 2025), and robustness certificates are derived from gradient information (Singla & Feizi, 2020). In both cases, what is being controlled is the way a prediction responds to small changes in its input.

It is therefore natural to read the gradient of a hypothesis as a vector field — a *sensitivity field* — that encodes, at each input, how the prediction changes with the input. Because this field is the gradient of a scalar function, it is a priori constrained to be conservative: its curl vanishes and its line integrals are path-independent. Robustness, by contrast, imposes a separate set of geometric requirements on the same field, asking it to remain small in adversarial directions while staying informative in the directions that carry signal. Our main thesis is that the accuracy–robustness trade-off observed in applications is a manifestation of the conflict between these two kinds of constraints: the conservativeness that every scalar-function sensitivity field must satisfy, and the directional constraints that robustness and accuracy impose on it.

Having identified this conflict, we develop a principled construction of a sensitivity field that respects the robustness and accuracy constraints, and then study when the resulting field can still be realized as the gradient of a scalar function. We start from an accurate hypothesis ψ and take its gradient $\nabla\psi$ as the base sensitivity field. To filter out the sensitivity to adversarial directions, we apply a projection operator P, yielding the cleaned field $F = P\nabla\psi$. The central question we study is under what conditions this cleaned field F remains conservative. In other words, whether there exists a well-defined scalar potential whose gradient is F, and hence whether a single-valued classifier can realize the robust and accurate behavior we have constructed.

This question only becomes tractable once a concrete form for P is fixed, so we commit to one, guided by the manifold hypothesis. The data is assumed to lie on a low-dimensional manifold, which we encode as the zero-set of a differentiable function ϕ. We take P to be the orthogonal projection onto the null space of the Jacobian J_ϕ. On the data manifold, this choice cleanly separates the on-manifold directions, in which the classifier should remain sensitive, from the off-manifold directions, in which it should be invariant. With this P, $F = P\nabla\psi$ is precisely the component of $\nabla\psi$ tangent to the data manifold which enforces, to first order, invariance to off-manifold perturbations. For an off-manifold point, F would remain tangent to level-sets of ϕ which cannot intersect with the data manifold. Under this regiment, the question of conservativeness of F becomes a well-posed geometric one.

Our contributions are as follows: (i) we reinterpret adversarial robustness as a *strong tangency* constraint on the gradient of a differentiable hypothesis, and identify the accuracy–robustness trade-off as a conflict between this constraint and the conservativeness every gradient field must satisfy; (ii) fixing P_ϕ as the orthogonal projection onto $\ker(J_\phi)$ under the manifold hypothesis, we construct the projected sensitivity field $F = P_\phi\nabla\psi$, prove it reproduces an accurate hypothesis exactly on the data manifold (Theorem 3.1), and give a complete decomposition of the antisymmetric part of its Jacobian (Theorem 3.2), showing the obstruction to conservativeness reduces, in the generic case, to a curvature term governed by the Hessian of ϕ — and, for the unit circle and under mild conditions, forces a rigidity result in which every continuous strongly-tangent hypothesis is constant (Theorem 3.4); and (iii) we introduce path-dependent, multi-valued hypotheses as a representation that sidesteps this obstruction, and show empirically on MNIST, FashionMNIST, and CIFAR-10 that adversarially trained path-dependent sensitivity fields achieve better robust accuracy than gradient-field baselines, align more closely with the tangent space of the data manifold, and exhibit statistically significant non-zero circulation, confirming that the learned fields are not, in general, gradients of any single-valued function.

2 METHOD

2.1 NOTATION

Given some differentiable function $\gamma : \mathbb{R} \to \mathbb{R}^d$ of time t, we will denote its derivative with respect to t by $\dot{\gamma} = \frac{d\gamma}{dt}$. To represent partial derivatives of a differentiable function $f : \mathbb{R}^d \to \mathbb{R}$, we will use the notation $\partial_i f = \frac{\partial f}{\partial x_i}$. More generally, $\partial_v f$ represents the directional derivative of f in direction of $v \in \mathbb{R}^d$. Moreover and assuming appropriate smoothness conditions on f, we will use ∇f and H_f

to represent its gradient vector and Hessian matrix, respectively. Lastly, given some differentiable vector-valued function $F : \mathbb{R}^d \to \mathbb{R}^l$, its Jacobian is shown by $J_F : \mathbb{R}^d \to \mathbb{R}^{l \times d}$.

We will use the notation $x \mapsto y$ to represent an absolutely continuous path from $x \in \mathbb{R}^d$ to $y \in \mathbb{R}^d$. Similarly, $x \mapsto y \mapsto z$ represents a path that is the result of attaching a path from x to y to a path from y to z. We will use the following shorthand notation for path integrals of a vector field $F : \mathbb{R}^d \to \mathbb{R}^d$ over a differentiable path $\gamma : [a, b] \to \mathbb{R}^d$:

$$\int_\gamma F \cdot \mathrm{d}\gamma = \int_a^b \langle F(\gamma(t)), \dot{\gamma}(t) \rangle \, \mathrm{d}t, \tag{1}$$

where $\langle \cdot, \cdot \rangle$ is the usual Euclidean inner product.

2.2 Tangency condition

Let $\mathcal{H}$ be a nonuniform learnable (Shalev-Shwartz & Ben-David, 2014) hypothesis space consisting of functions $h : \mathbb{X} \to \mathbb{Y}$ where $\mathbb{X} \subset \mathbb{R}^d$ is some ambient space and $\mathbb{Y} \subset \mathbb{R}^k$ is some label space, e.g., $\mathbb{X} = [0, 1]^d$, $\mathbb{Y} = \{0, 1\}$, and $\mathcal{H}$ is a space of artificial neural networks with a specific architecture. Suppose we have random variables X, Y supported on $\mathbb{X}, \mathbb{Y}$ respectively with a joint distribution $\mathcal{D}$. Our aim in a standard learning problem is to find some $h \in \mathcal{H}$ which minimizes the expected value $\mathbb{E}_{\mathcal{D}}\left[L\left(h(x), y\right)\right]$ of some loss function $L : \mathbb{Y} \times \mathbb{Y} \to \mathbb{R}_{\geq 0}$ with respect to $\mathcal{D}$, given a finite sample generated from $\mathcal{D}$.

We are, however, concerned with a robust learning problem where we want to minimize the supremum of the loss in an ℓ_p-ball $B_r^p(x)$ with radius r around each sample x. This setting is chosen because we can assume the label of a training sample does not change in a small enough ball around x in most real-world scenarios. More specifically, we want to analyze a robust classification problem from the perspective of geometry of the embedding of samples inside $\mathbb{X}$. To this end, it is a useful simplification to assume X is only supported on some low-dimensional manifold $\mathcal{X} \subset \mathbb{X}$ for which $B_r^p(x) \subset \mathbb{X}$ for all $x \in \mathcal{X}$. The research question for us then is to determine how geometry of $\mathcal{X}$ affects learning of a robust hypothesis.

Intuitively, brittleness of a hypothesis could be interpreted as its sensitivity to changes in the test sample that do not correspond to a meaningful change in the content of the sample. This suggests a construction of a robust hypothesis by first finding an appropriate sensitivity field $F : \mathbb{X} \to \mathbb{R}^d$, and then constructing a hypothesis h by solving a partial differential equation $\nabla h(x) = F(x)$.

However, this equation implies that F is curl-free:

$$J_F(x) - J_F(x)^T = 0. \tag{2}$$

The other constraint on a gradient field is path-independence of path integrals: a path integral over an absolutely continuous path $\gamma : x_0 \mapsto x$ should only depend on the end-points of the path and not the trajectory taken by the path:

$$h(x) - h(x_0) = \int_\gamma \nabla h \cdot \mathrm{d}\gamma = \int_\gamma F \cdot \mathrm{d}\gamma, \tag{3}$$

in which we have used the fundamental theorem of calculus to represent h as a path integral of ∇h. Otherwise, it would not be possible to assign a unique value to $h(x)$ and h cannot be a function. A vector field that satisfies both conditions of being curl-free and path-independent is called a *conservative* field.

So, a sensitivity field that is the gradient of a differentiable function is quite constrained already. This observation naturally leads us to a question regarding the realizability of a robust and accurate hypothesis by artificial neural networks (ANNs): is the desired sensitivity field a gradient to begin with? To answer this question, we first have to determine what we consider to be a robust sensitivity field. That is where we introduce the geometry of $\mathcal{X}$ into our exposition of the question. We will consider sensitivity fields that are in some sense *parallel* to $\mathcal{X}$.

First, assume $\mathcal{X}$ is a hyperplane represented by a few axis coordinates of $\mathbb{X}$ and let $\mathbb{X} = \mathbb{R}^d$. Hence, we can decompose the coordinates into two partitions $x = \begin{bmatrix} c & s \end{bmatrix}^T$ in which $s \in \mathbb{R}^l, l < d$ represents the *style* variable and $c \in \mathbb{R}^{d-l}$ is the *content* variable of a sample $x \in \mathbb{R}^d$. The content variable c

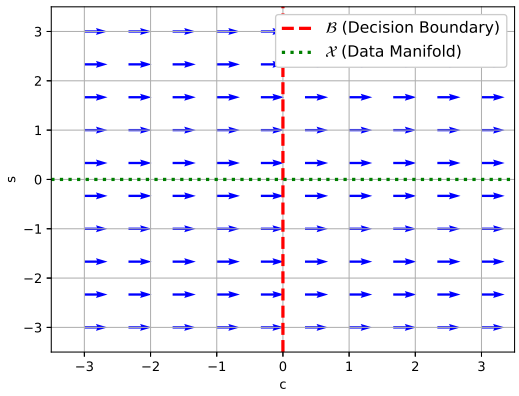


Figure 1: Assuming $x = (c, s) \in \mathbb{R}^2$, the gradient field of a robust hypothesis should only point in a direction parallel to the c axis. (Visualization created using Matplotlib 3.10.9 (Hunter, 2007))

carries the information about the position of the orthogonal projection of x into $\mathcal{X}$, and s describes how x is positioned outside $\mathcal{X}$. It is arguable then that a robust hypothesis should base its predictions only on information in c and not s. Hence, we expect the gradient field of a robust hypothesis to be parallel to this hyperplane everywhere in $\mathbb{X}$ as shown in Figure 1 for $d = 2$. This setup is inspired by the causal picture of robustness presented in (Tang et al., 2021) and the boundary tilting perspective of (Tanay & Griffin, 2016).

For a hypothesis $h : \mathbb{R}^d \to \mathbb{R}$ that is robust in this sense, we have $\nabla h \in T_x(\mathcal{X})$ in which $T_x(\mathcal{X})$ is the space tangent to $\mathcal{X}$ at $x \in \mathcal{X}$. Consequently, we say ∇h is *tangent* to $\mathcal{X}$. Similarly, for all $x \in \mathbb{X}$, we see $\nabla h(x)$ is in some sense parallel to $\mathcal{X}$. In other words, $\partial_v h(x) = 0$ for all $v = (0, s)$, and we say ∇h is *strongly tangent* to $\mathcal{X}$.

It is rarely the case that $\mathcal{X}$ is a hyperplane in practice, and more often than not it is embedded in $\mathbb{X}$ in a quite complicated manner. However, the same principles are applicable if we define a notion of parallel for a curved manifold. We approach this issue by assuming $\mathcal{X}$ is represented as the zero-set of a function $\phi : \mathbb{R}^d \to \mathbb{R}^l$:

$$\mathcal{X} = \{x \in \mathbb{X} \mid \phi(x) = 0\}. \tag{4}$$

Then, we can define a strongly tangent sensitivity field F as a vector field that is tangent to level-sets of ϕ:

Definition 2.1 (ϕ-strong tangency)**.** Let $F : \mathbb{R}^d \to \mathbb{R}^d$ be a vector field and assume $\mathcal{X}$ is a manifold represented by the zero-set of a differentiable function $\phi : \mathbb{R}^d \to \mathbb{R}^l$. Then, F is strongly tangent to $\mathcal{X}$ when

$$J_\phi(x)F(x) = 0 \quad \text{for all} \quad x \in \mathbb{X}. \tag{5}$$

Choosing $\phi(x) = \begin{bmatrix} 0 & 0 \\ 0 & I_l \end{bmatrix} x$, for example, represents the simple case we mentioned earlier.

3 Obstruction to Robustness

Suppose we have an accurate hypothesis $\psi : \mathbb{X} \to \mathbb{R}$ which is not robust and we have access to some $\phi : \mathbb{X} \to \mathbb{R}^l$ which represents $\mathcal{X}$ as its zero-set. We can construct a ϕ-strongly tangent sensitivity field by projecting $\nabla\psi$ into the tangent space of level-sets of ϕ:

$$F(x) = P_\phi(x)\nabla\psi(x), \tag{6}$$

where $P_\phi : \mathbb{X} \to \mathbb{R}^{d\times d}$ is the orthogonal projection defined by $\ker J_\phi(x)$:

$$P_\phi = I_d - J_\phi^T \left(J_\phi J_\phi^T\right)^{-1} J_\phi, \tag{7}$$

where we have omitted the dependence on x for brevity. A sensitivity field constructed in this manner trivially satisfies $J_\phi(x)F(x) = 0$ as long as $J_\phi(x)$ has full row-rank, and F would show a

discontinuity where $|J_\phi(x)J_\phi(x)^T| = 0$. The discontinuity can be removed if we construct the sensitivity field instead as $F(x) = |J_\phi(x)J_\phi(x)^T|P_\phi(x)\nabla\psi(x)$, but we will stick to the discontinuous construction for simplicity. As we will show in the following theorem, constructing a sensitivity field in this manner gives rise to a hypothesis that is both single-valued and accurate on $\mathcal{X}$ under mild conditions:

Theorem 3.1. *Let $\psi : \mathbb{R}^d \to \mathbb{R}$ and $\phi : \mathbb{R}^d \to \mathbb{R}^l$ be differentiable and let $\mathcal{X} = \{x \in \mathbb{X} \,|\, \phi(x) = 0\}$ be a connected manifold for which $J_\phi(x)$ has full row-rank for all $x \in \mathcal{X}$. Consider $x, x_0 \in \mathcal{X}$ and let $\gamma \subset \mathcal{X} : x_0 \mapsto x$ be a differentiable path. Then, assuming $F(x) = P_\phi(x)\nabla\psi(x)$,*

$$\psi(x) = \psi(x_0) + \int_\gamma F \cdot \mathrm{d}\gamma. \tag{8}$$

Hence, it is enough to find some solution $\nabla h(x) = F(x)$ to construct a robust and accurate hypothesis. However, the next theorem shows that such a function would not always exist:

Theorem 3.2. *Assume twice-differentiable functions $\psi : \mathbb{R}^d \to \mathbb{R}$, $\phi : \mathbb{R}^d \to \mathbb{R}^l$, $x \in \mathcal{X} \subset \mathbb{R}^d$ and further assume $J_\phi(x)$ has full row-rank. Let $u, v \in \mathbb{R}^d$ and $F(x) = P_\phi(x)\nabla\psi(x)$. Define the bilinear form:*

$$\begin{aligned}\mathcal{B}(u,v) = \big\langle u_\parallel, H_\psi(x)v_\perp \big\rangle \\ + \sum_{j=1}^{l} \widetilde{v}_{\perp,j} \, \big\langle H_{\phi_j}(x)u_\parallel, \nabla\psi(x)_\parallel \big\rangle \\ - \sum_{j=1}^{l} \big\langle H_{\phi_j}(x)u_\parallel, v_\perp \big\rangle \big\langle c_j(x), \nabla\psi(x)_\perp \big\rangle,\end{aligned} \tag{9}$$

and the antisymmetric form:

$$\mathcal{A}(u,v) = \left\langle \sum_{j=1}^{l} \tilde{v}_{\perp,j} H_{\phi_j}(x)u_\perp - \tilde{u}_{\perp,j} H_{\phi_j}(x)v_\perp, \nabla\psi(x)_\parallel \right\rangle, \tag{10}$$

where $a_\parallel \in T_x(\mathcal{X})$ and $a_\perp \in T_x(\mathcal{X})^\perp$ are the tangent and normal components of $a \in \mathbb{R}^d$ to $\mathcal{X}$, respectively. Then,

$$\big\langle u, \big(J_F(x) - J_F(x)^T\big)v \big\rangle \;=\; \mathcal{A}(u,v) \;+\; \mathcal{B}(u,v) \;-\; \mathcal{B}(v,u). \tag{11}$$

The full definition of $\tilde{a}_\perp \in \mathbb{R}^l, a \in \mathbb{R}^d$ and $c : \mathbb{R}^d \to \mathbb{R}^{l\times d}$ in Theorem 3.2 are available in Appendix A.2.

Theorem 3.2 shows that, depending on our choice of ϕ and ψ, the curl of F might not vanish. In such a case, it would not be possible to represent this sensitivity field as a gradient. Assuming $l = 1$, the following corollary simplifies the obstruction so that it is easier to read:

Corollary 3.3. *Let $l = 1$, $x \in \mathcal{X}$ and $u, v \in \mathbb{R}^d$. Then, the antisymmetric form $\mathcal{A}(u,v)$ vanishes and the bilinear form reduces to:*

$$\mathcal{B}(u,v) = \big\langle u_\parallel, H_\psi(x)v_\perp \big\rangle + \tilde{v}_\perp \big\langle H_\phi(x)u_\parallel, \nabla\psi(x)_\parallel - \nabla\psi(x)_\perp \big\rangle. \tag{12}$$

Then,

$$\big\langle u, \big(J_F(x) - J_F(x)^T\big)v \big\rangle = \; \mathcal{B}(u,v) \;-\; \mathcal{B}(v,u). \tag{13}$$

Corollary 3.3 shows that the main components in the obstruction are H_ψ and H_ϕ. Considering the definition of ϕ-strong tangency, we argue that the relevant obstruction to conservativeness of F is H_ϕ. Given the fact that ϕ has $\mathcal{X}$ as its zero-set, $H_\phi(x)$ encodes the curvature information of $\mathcal{X}$ at $x \in \mathcal{X}$. In other words, constructing conservative sensitivity fields that are ϕ-strongly tangent to $\mathcal{X}$ is obstructed when $\mathcal{X}$ is curved. This obstruction is independent of our choice of ϕ since any such choice is constrained to represent $\mathcal{X}$ as its zero-set.

In this light, we face another question: can we find some $\mathcal{X}$ for which no accurate differentiable classifier could be ϕ-strongly tangent to? The next theorem shows that for some functions ϕ at the very least the answer is affirmative:

Theorem 3.4. *Let $\phi(x) = (\|x\|^2 - 1)/2, x \in \mathbb{R}^2$. Then, $h : \mathbb{R}^2 \to \mathbb{R}$ is both continuous and ϕ-strongly tangent to the unit circle $\mathcal{X} = S^1$ iff h is constant.*

To summarize, depending on our choice of the definition of a robust sensitivity field, no differentiable function might be able to realize the robust and accurate hypothesis, irrespective of the size and the architecture of the model. The obstruction to some extent is caused by the shape of the support of the random variable X, and it is not a side-effect of overlapping categories. This suggests that topological defects (i.e. holes and twists) in the support of X might obstruct learning of a robust and accurate classifier by differentiable functions, as these topological complexities survive any continuous transformation.

We want to be forthcoming with what our analysis is not showing. The analysis is not ruling out the possibility of learning robust and accurate classifiers by differentiable functions, rather, it shows that a simple reading of robustness definitions available in the literature suggests that modeling a robust and accurate classifier might need novel architectures that go beyond functions, or might need models that represent a sensitivity field directly.

4 MULTI-VALUED HYPOTHESES

To test our thesis in real-world applications, we present a model for building multi-valued hypotheses. The idea is to learn a sensitivity field $F : \mathbb{R}^d \to \mathbb{R}^d$ instead of a function, and then define a path-dependent function as the hypothesis:

$$h(x \mid \gamma, b) = b + \int_{\gamma} F \cdot \mathrm{d}\gamma, \tag{14}$$

where $\gamma : x_0 \mapsto x$ is an absolutely continuous path and b is a value of h at x_0. Under this model, γ chooses which value for $h(x)$ is returned. In other words, by fixing a family of paths and assigning a unique path to each $x \in \mathbb{X}$, we are choosing a single-valued *branch* of the multi-valued hypotheses. Nevertheless, if sensitivity fields of robust and accurate classifiers for a particular application are not gradients, none of these branches can be robust, accurate and differentiable simultaneously.

This model is a good choice in the sense that it has all differentiable functions as a subset: if we set F equal to the gradient of some function f and choose the inference paths to be straight paths $\gamma_x : x_0 \mapsto x$ for some $x_0 \in \mathbb{X}$ and let $b = f(x_0)$, then we have $h(x \mid \gamma_x, b) = f(x)$. However, the extra representation power comes at the cost of added complexity of inferring the optimal inference path for some test sample. While this issue is of utmost importance in practical use of this model, it is not a concern of this paper, and we employ oracle-paths to test whether it is possible to find sensitivity fields that have a branch that is both accurate and robust without any loss of accuracy for real-world applications.

We choose the oracle-path, inspired by the observation that accurate differentiable classifiers exist. So, a sensitivity field exists for which the branch corresponding to straight paths from a unique starting point $x_0 \in \mathbb{X}$ is accurate for benign samples. For adversarial examples, however, we choose the path $x_0 \mapsto x \mapsto \tilde{x}$ which results from attaching two straight paths $x_0 \mapsto x$ and $x \mapsto \tilde{x}$:

$$h(\tilde{x} \mid x_0 \mapsto x \mapsto \tilde{x}, b) = h(x \mid x_0 \mapsto x, b) + h(\tilde{x} \mid x \mapsto \tilde{x}, 0). \tag{15}$$

This choice of an oracle-path allows us to adversarially train sensitivity fields in the neighborhood of training samples and empirically test various characteristics of the trained sensitivity fields to see if they align with predictions of our thesis.

The difference $h(\tilde{x} \mid x_0 \mapsto x \mapsto \tilde{x}, b) - h(x_0 \mid x_0 \mapsto \tilde{x}, b)$ gives us a local measure of non-conservativeness of F. This quantity computes the difference between two branches of h, and, in effect, computes h for a closed loop:

Definition 4.1 (circulation)**.** Given a vector field $F : \mathbb{R}^d \to \mathbb{R}^d$ and an absolutely continuous loop $\pi : S^1 \to \mathbb{X}$, the circulation C of F for π is:

$$\mathrm{C}(F \mid \pi) = \oint_{\pi} F \cdot \mathrm{d}\pi. \tag{16}$$

This definition is basically an application of the generalized Stokes theorem (Tu, 2011). While $\mathrm{C}(F \mid \pi) = 0$ does not mean F is conservative inside π, a non-zero value signals a non-conservative vector field.

Table 1: Benign and adversarial accuracy of the branch of oracle-paths up to first-order approximation for batches of 256 samples. *Omitted Square Attack on CIFAR10 due to extreme computational burden.

Dataset	Clean	Adv.
MNIST	0.975 ± 0.016	0.974 ± 0.017
FMNIST	0.861 ± 0.028	0.860 ± 0.029
CIFAR10*	0.714 ± 0.028	0.713 ± 0.028

5 EXPERIMENTS

Our experiments will focus on three aspects: robustness of a path-dependent hypothesis, tangency to $\mathcal{X}$, and the conservativeness of trained vector fields. We will empirically study these issues on MNIST (LeCun & Cortes, 2010), FMNIST (Xiao et al., 2017), and CIFAR10 (Krizhevsky, 2009) datasets. In all experiments, we have $k = 10$ sensitivity fields that are represented by k parallel ANNs $F : \mathbb{R}^d \to \mathbb{R}^{k \times d}$ implemented in PyTorch 2.11 (Paszke et al., 2019). The path integrals are computed through numerical ODE integration of these fields using TorchDiffEq 0.2.5 (Chen et al., 2018), with $b = 0$ and $x_0 = 0$ unless otherwise specified. Adversarial examples for adversarial training are generated for the branch of straight paths using FGSM (Goodfellow et al., 2015) and then used to adversarially train the branch of oracle-paths following the TRADES framework (Zhang et al., 2019). In assessing the generalization of the robustness of a branch, we use the AutoAttack (Croce & Hein, 2020) suite consisting of AutoPGD (Croce & Hein, 2020), DeepFool (Moosavi-Dezfooli et al., 2016), and Square Attack (Andriushchenko et al., 2020). All attacks are performed under ℓ_∞ constraints, and the radius of the attack is set to 0.3 for the MNIST family and $8/255$ for CIFAR10. The adversarial attack implementations are provided by Foolbox 3.3.4 (Rauber et al., 2020) and ART 1.20.1 (Nicolae et al., 2018) libraries. Statistical hypothesis tests were implemented using Scipy 1.17.1 (Virtanen et al., 2020), and PCA was performed using the implementation provided by ScikitLearn 1.8.0 (Pedregosa et al., 2011). All experiments were run on an Apple Mac Studio M4 Max machine. The details of results and implementations are available in Appendix B.4.

5.1 ROBUSTNESS TO ATTACKS

In our first experiment, we consider the robustness of adversarially trained sensitivity fields up to first-order perturbations. We achieve this by choosing a forward one-step Euler method to approximate $h(\tilde{x} \mid x \mapsto \tilde{x}, 0) \approx \langle F(x), \tilde{x} - x \rangle$ and a backward step for $h(x \mid x_0 \mapsto x, b) \approx \langle F(x), x - x_0 \rangle$. The results are reported in Table 1. We can see that the first-order response of the sensitivity fields is almost maximally robust. Together with the acceptable accuracy on benign samples, the data suggests that trained sensitivity fields are quite tangent to $\mathcal{X}$.

First-order response, however, is not informative about strong tangency. We compute the path integrals accurately in the next experiment to this end using the DoPri5 (Dormand & Prince, 1980) adaptive ODE solver in Table 2. The computational burden for this method is, however, too high to perform the experiment on the whole test sets. Consequently, we performed the experiment for 10 batches of 32 samples.

Furthermore, we compare the performance of sensitivity fields with gradient fields trained with the same adversarial training methodology to show that relaxing the conservativeness constraint significantly improves the performance. However, the most reliable method of constructing conservative fields is through gradients, and the prediction of the conservative fields is actually computed through standard single-valued ANNs without involving the ODE integrator. It is arguably impossible to equate the two ANNs from a capacity and architecture perspectives due to the mismatch between the output layers. Nevertheless, given the fact that gradient fields are a strict subset of sensitivity fields, we can be sure that the performance of the best sensitivity field is always better than or equal to the performance of the best gradient field.

For completeness, we also compare the performance of the choice of $x_0 = 0$ with choosing the empirical mean $\bar{X}$ of the training sets as x_0 in training. The results further show that the choice of a branch is quite consequential for the performance of a multi-valued hypothesis. We intuitively

Table 2: Comparison of clean and adversarial accuracies of single-valued and multi-valued models based on oracle paths. Multi-valued models show great potential in representing robust hypotheses. The centroid $\bar{X}$ showed inferior performance compared to 0 as a starting point on MNIST and FMNIST, so we omitted the test on CIFAR10 to save on computation time.

Dataset	Single-valued		Multi-valued			
			Trained with $x_0 = 0$		Trained with $x_0 = \bar{X}$	
	Clean	Adv.	Clean	Adv.	Clean	Adv.
MNIST	0.96 ± 0.03	0.00 ± 0.00	0.98 ± 0.02	0.56 ± 0.11	0.97 ± 0.03	0.22 ± 0.06
FMNIST	0.88 ± 0.06	0.01 ± 0.01	0.86 ± 0.04	0.48 ± 0.07	0.89 ± 0.05	0.12 ± 0.08
CIFAR10	0.74 ± 0.07	0.28 ± 0.07	0.74 ± 0.10	0.70 ± 0.09	-	-

Table 3: Non-negligible p-values of Wilcoxon tests for circulation separated by label. All entries not listed here are ≈ 0. Assuming $\alpha = 0.01$, the only case where the null hypothesis is not rejected is for digit 1 of MNIST. (1: 1/Trouser/Automobile, 4: 4/Coat/Deer)

Dataset	Label	p-value
MNIST	1	0.02
CIFAR10	4	2e-8

interpret this observation under our model as suggesting that straight paths from the centroid of $\mathcal{X}$ are, in some sense, *strongly normal* to $\mathcal{X}$ and that making this branch accurate is arguably in conflict with strong tangency of F to $\mathcal{X}$. In the same spirit, we also argue that even though $x_0 = 0$ is an improvement over the centroid, straight paths are probably far from the best choice of inference paths for training strongly tangent and accurate sensitivity fields.

5.2 Circulation

Next, we statistically test whether the trained sensitivity fields are conservative. To this end, we compute the circulation C of F in a loop $x_0 \mapsto x \mapsto \tilde{x} \mapsto x_0$. We utilized a two-sided Wilcoxon test with 3200 samples in each test. The p-values are reported in Table 3. We can see that the p-values equal zero up to machine precision for most hypotheses except the one corresponding to digit 1 in MNIST. So we cannot reject the null hypothesis for digit 1 with $\alpha = 0.01$. To control for the possibility of rejection due to numerical errors, we performed the Wilcoxon test on the gradient field of the single-valued MNIST classifier, for which the null hypothesis was not rejected.

5.3 Tangency

We perform two experiments with regards to tangency. In the first experiment, we approximate $T_x(\mathcal{X})$ for some benign test sample x and then compare the alignment of the adversarially trained gradient fields and sensitivity fields. Since we are working with probability distributions rather than an explicit manifold, we construct a surrogate tangent space per test sample via PCA on same-class near neighbors, and compare how strongly sensitivity fields align with this surrogate tangent space relative to gradient fields, using one-sided t-tests. Informally, for a benign test sample x we find its 50 nearest neighbors that share its class, call them z, remove the test sample (z-x), and run PCA keeping 95% of the variance. The resulting PCA subspace serves as a surrogate for $T_x(\mathcal{X})$. We then compared the norm of the projection of the vector fields at x into this surrogate space against each other, to test whether the learned vector field is more aligned with it.

The results are reported in Table 4. Comparing Table 4 with Table 2, we see that higher adversarial accuracy correlates with improved tangency — i.e. the sensitivity fields that show higher robustness also show better alignment with the tangent space of $\mathcal{X}$.

We also perform a second experiment that applies the projected field construction $F(x) = P_\phi(x)\nabla\psi(x)$. Here we use a known invariance of image classifiers: robustness to the normalization of pixel intensities. This invariance can be encoded as ϕ-strong tangency of the sensitivity

Table 4: Categories for which the alignment of sensitivity fields with the surrogate tangent space of $\mathcal{X}$ was statistically worse than that of gradient fields. All categories for CIFAR10 rejected the null-hypothesis. (1: 1/Trouser/Automobile, 3: 3/Dress/Cat, 5: 5/Sandal/Dog, 7: 7/Sneaker/Horse, 8: 8/Bag/Ship, 9: 9/Ankle Boot/Truck)

Dataset	H_0 not Rejected
MNIST	7,8
FMNIST	1,3,5,9

field with $\phi(x) = \|x\|^2 - 1$. Concretely, we project the gradient field of an accurate hypothesis ψ to construct a ϕ-strongly tangent sensitivity field F. We then compute the curl of F and take its Frobenius norm as a test for conservativeness. We then perform one-sample t-tests to statistically test for conservativeness of F for each category over 10 batches of 32 samples of the MNIST and FMNIST datasets. Every class in both datasets rejects H_0 at $p < 0.002$ and a large majority at $p < 0.001$, so the deviation from zero is not just numerically visible but statistically robust given the batch-level variability we observe.

6 Conclusion and future research

In this paper, we tried to find a robust and accurate hypothesis by first constructing a desirable sensitivity field. However, the empirical and theoretical exposition presented strongly points to non-conservativeness of sensitivity fields of robust and accurate hypotheses in real-world applications. These observations suggest that a move to a hypothesis space of either multi-valued or non-differentiable functions might be inevitable for building robust and accurate models. Given that modern machine learning heavily relies on gradient information for tasks related to training, explanation and robustness, we argue that the only real alternative to differentiable hypotheses is multi-valued ones. This returns us to the question posed at the outset — whether the accuracy–robustness trade-off reflects a property of the data or of the hypothesis class: our analysis suggests it is fundamentally the later, since the obstruction would vanish by changing the hypothesis class.

While we believe our method for revealing the obstruction in this paper is easy to understand, it is probably unnecessarily long and dependent on the choice of coordinate space. It is imaginable that a coordinate-free method or an argument based on cohomology can provide a cleaner exposition of the issue. It also remains open whether a combination of a decision boundary and a data manifold exists for which the robust and accurate classifier is not realizable by differentiable functions. Nevertheless, we believe the discussion in this paper takes a step in the right direction, and we conjecture that the answer to this question is affirmative, and a prerequisite for such a manifold is existence of non-trivial cohomology groups for $\mathcal{X}$.

We also presented a simple method for exploiting sensitivity fields and constructing multi-valued hypotheses. This paper was mainly focused on the feasibility of these models in real-world problems. Our experiments showed the potential of multi-valued hypotheses in modeling robust and accurate classifiers. However, our construction is far from being applicable to practical problems. The main blockers in this regard are the lack of a method for determining the optimal branch of the multi-valued hypotheses and the computational costs of using these models. The best clue the paper gives for the issue of the choice of a branch is Theorem 3.1 which places emphasis on streamlines of sensitivity fields. Considering the computational burden, it is conceivable that mixing single-valued and multi-valued hypotheses can cut the size of the multi-valued models down enough to enable fast inference.

AI use statement

In this work, we used generative AI tools to help with writing of proofs. We have not used generative AI tools to help develop theoretical models or conceptual frameworks, formulate mathematical claims, interpret results, design or provide feedback on research methodology or experiments, implement methods, or generate synthetic datasets. Assisting with translation, cleaning and reformatting datasets, and supporting qualitative or thematic data analysis are not applicable to this work. Addi-

tionally, we used generative AI tools for the creation of software code, drafting parts of the research paper, brainstorming, and editing the paper to improve readability. We have reviewed all AI-assisted work. LLM-generated code and proofs were verified and tested for correctness by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square attack: a query-efficient black-box adversarial attack via random search. 2020.

Anju Bala, Ritu, Utkarsh Mishra, Nagendar Yamsani, Chandrapal Singh Arya, and Tushar Gupta. Deep learning in cybersecurity: A review of attacks, defenses, and interpretability approaches. In *2026 1st International Conference on Advancing Sustainable Solutions through Technologies (ICASST)*, pp. 1–5, 2026. doi: 10.1109/ICASST68917.2026.11484115.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations. *Advances in Neural Information Processing Systems*, 2018.

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020.

J.R. Dormand and P.J. Prince. A family of embedded runge-kutta formulae. *Journal of Computational and Applied Mathematics*, 6(1):19–26, 1980. ISSN 0377-0427. doi: https://doi.org/10.1016/0771-050X(80)90013-3. URL https://www.sciencedirect.com/science/article/pii/0771050X80900133.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2015. URL https://arxiv.org/abs/1412.6572.

Kathrin Grosse, Praveen Manoharan, Nicolas Papernot, Michael Backes, and Patrick McDaniel. On the (statistical) detection of adversarial examples. *arXiv preprint arXiv:1702.06280*, 2017.

Sicong Han, Chenhao Lin, Chao Shen, Qian Wang, and Xiaohong Guan. Interpreting adversarial examples in deep learning: A review. *ACM Comput. Surv.*, 55(14s), July 2023. ISSN 0360-0300. doi: 10.1145/3594869. URL https://doi.org/10.1145/3594869.

J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3): 90–95, 2007. doi: 10.1109/MCSE.2007.55.

Josip Jukić and Jan Šnajder. From robustness to improved generalization and calibration in pretrained language models. *Transactions of the Association for Computational Linguistics*, 13: 264–280, 2025.

Alex Krizhevsky. Learning multiple layers of features from tiny images. pp. 32–33, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: A simple and accurate method to fool deep neural networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2574–2582, 2016. doi: 10.1109/CVPR.2016.282.

Maria-Irina Nicolae, Mathieu Sinn, Minh Ngoc Tran, Beat Buesser, Ambrish Rawat, Martin Wistuba, Valentina Zantedeschi, Nathalie Baracaldo, Bryant Chen, Heiko Ludwig, Ian Molloy, and Ben Edwards. Adversarial robustness toolbox v1.2.0. *CoRR*, 1807.01069, 2018. URL https://arxiv.org/pdf/1807.01069.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance

deep learning library. In *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019.

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

Jonas Rauber, Roland Zimmermann, Matthias Bethge, and Wieland Brendel. Foolbox native: Fast adversarial attacks to benchmark the robustness of machine learning models in pytorch, tensorflow, and jax. *Journal of Open Source Software*, 5(53):2607, 2020. doi: 10.21105/joss.02607. URL https://doi.org/10.21105/joss.02607.

S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press, 2014. ISBN 9781107057135.

Sahil Singla and Soheil Feizi. Second-order provable defenses against adversarial attacks. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 8981–8991. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/singla20a.html.

Thomas Tanay and Lewis Griffin. A boundary tilting persepective on the phenomenon of adversarial examples, 2016. URL https://arxiv.org/abs/1608.07690.

Kaihua Tang, Mingyuan Tao, and Hanwang Zhang. Adversarial visual robustness by causal intervention, 2021. URL https://arxiv.org/abs/2106.09534.

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. URL https://openreview.net/forum?id=SyxAb30cY7.

Loring W Tu. *An Introduction to Manifolds*. Universitext. Springer Science & Business Media, New York, NY, 2 edition, 2011. ISBN 978-1-4419-7399-3. doi: 10.1007/978-1-4419-7400-6.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, 2017. cite arxiv:1708.07747Comment: Dataset is freely available at https://github.com/zalandoresearch/fashion-mnist Benchmark is available at http://fashion-mnist.s3-website.eu-central-1.amazonaws.com/.

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 Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, Proceedings of Machine Learning Research, pp. 7472–7482. PMLR, 2019. URL http://proceedings.mlr.press/v97/zhang19p.html.

A PROOFS

A.1 THEOREM 3.1

Proof. Since $\gamma(\tau) \in \mathcal{X}$, we have

$$\phi(\gamma(\tau)) = 0 \qquad \forall \tau \in [0, 1]. \tag{17}$$

Differentiating with respect to τ,

$$J_\phi(\gamma(\tau))\dot{\gamma}(\tau) = 0. \tag{18}$$

Hence

$$\dot{\gamma}(\tau) \in \ker J_\phi(\gamma(\tau)). \tag{19}$$

For a regular level-set manifold, the tangent space satisfies

$$T_{\gamma(\tau)}(\mathcal{X}) = \ker J_\phi(\gamma(\tau)). \tag{20}$$

Therefore every tangent vector $\dot{\gamma}(\tau)$ is orthogonal to the row space of $J_\phi(\gamma(\tau))$.

Now decompose the gradient $\nabla\psi$ into orthogonal components:

$$\nabla\psi(x) = \nabla\psi_\parallel(x) + \nabla\psi_\perp(x), \tag{21}$$

where $\nabla\psi_\perp(x) \in \text{row}\,(J_\phi(x))$. By definition,

$$F(x) = \nabla\psi_\parallel(x). \tag{22}$$

Thus

$$\nabla\psi(\gamma(\tau)) = F(\gamma(\tau)) + \nabla\psi_\perp(\gamma(\tau)). \tag{23}$$

Since $\dot{\gamma}(\tau) \in \ker J_\phi(\gamma(\tau))$, we get orthogonality:

$$\langle \nabla\psi_\perp(\gamma(\tau)), \dot{\gamma}(\tau)\rangle = 0. \tag{24}$$

Hence

$$\langle \nabla\psi(\gamma(\tau)), \dot{\gamma}(\tau)\rangle = \langle F(\gamma(\tau)), \dot{\gamma}(\tau)\rangle. \tag{25}$$

By the chain rule,

$$\frac{d}{d\tau}\psi(\gamma(\tau)) = \langle \nabla\psi(\gamma(\tau)), \dot{\gamma}(\tau)\rangle. \tag{26}$$

Therefore,

$$\frac{d}{d\tau}\psi(\gamma(\tau)) = \langle F(\gamma(\tau)), \dot{\gamma}(\tau)\rangle. \tag{27}$$

Integrating over $[0, 1]$,

$$\psi(\gamma(1)) - \psi(\gamma(0)) = \int_0^1 \langle F(\gamma(\tau)), \dot{\gamma}(\tau)\rangle \, d\tau. \tag{28}$$

Using $\gamma(0) = x_0$ and $\gamma(1) = x$ we have:

$$\psi(x) = \psi(x_0) + \int_0^1 \left\langle F(\gamma(\tau)), \dot{\gamma}(\tau)\right\rangle d\tau. \tag{29}$$

□

A.2 THEOREM 3.2 AND COROLLARY 3.3

F has to be conservative for it to be the gradient field of some function. For a conservative vector field we have

$$J_F(x) - J_F^T(x) = 0. \tag{30}$$

A direct calculation of the anti-symmetric part of J_F reveals:

$$\partial_j F_i(x) - \partial_i F_j(x) = \sum_{s=1}^{d} \left(\partial_j P_{is}(x) - \partial_i P_{js}(x)\right) \partial_s\psi(x) + \sum_{s=1}^{d} \left(P_{is}(x)\partial_j\partial_s\psi(x) - P_{js}(x)\partial_i\partial_s\psi(x)\right). \tag{31}$$

Defining $\Phi : \mathbb{R}^d \to \mathbb{R}^{d\times d}$ as

$$\Phi_{ij}(x) = \sum_{s=1}^{d} \left(\partial_j P_{is}(x) - \partial_i P_{js}(x)\right) \partial_s \psi(x), \tag{32}$$

we can write $J_F(x) - J_F^T(x)$ more compactly as

$$J_F(x) - J_F^T(x) = \Phi(x) + P(x)H_\psi(x) - H_\psi(x)^T P(x)^T, \tag{33}$$
$$= \Phi(x) + \left[P(x), H_\psi(x)\right], \tag{34}$$

in which $[A, B]$ is the commutator of A and B and we have used the fact that both H_ψ and P are symmetric matrices.

This expression identifies two distinct sources of non-conservativeness: (i) the commutator of P and H_ψ, and (ii) the spatial variation of the tangent projector $P(x)$ itself. To see what could obstruct conservativeness, we will choose vectors $u, v \in T_x(\mathbb{X})$ and compute

$$\left\langle u, \left(J_F(x) - J_F^T(x)\right) v \right\rangle = \langle u, \Phi(x)v\rangle + \langle u, \left[P(x), H_\psi(x)\right] v\rangle . \tag{35}$$

If J_F has no anti-symmetric part, then equation 35 should vanish for all u, v. We begin the analysis with the following lemma:

Lemma A.1. *For $u, v \in \mathbb{R}^d$ we have*

$$\langle u, \Phi(x)v\rangle = \langle (\partial_v P(x))u - (\partial_u P(x))v, \nabla\psi(x)\rangle . \tag{36}$$

Proof. Assume $g \in \mathbb{R}^d$ and let $\tilde{F} : \mathbb{R}^d \to \mathbb{R}^d$ be the following:

$$\tilde{F}(x) = \sum_{i=1}^{d} g_i P_{:,i}(x), \tag{37}$$

We can recover Φ by differentiating $\tilde{F}$ to get

$$J_{\tilde{F}}(x) = \sum_{i=1}^{d} g_i J_{P_{:i}}(x), \tag{38}$$

and applying to u we have

$$J_{\tilde{F}}(x)u = \sum_{i=1}^{d} g_i J_{P_{:i}}(x)u = (\partial_u P(x))g. \tag{39}$$

Consequently, substituting $g = \nabla\psi(x)$ we get

$$\langle u, \Phi(x)v\rangle = \langle u, J_{\tilde{F}}(x)v\rangle - \langle J_{\tilde{F}}(x)u, v\rangle , \tag{40}$$
$$= \langle u, (\partial_v P(x))\nabla\psi(x)\rangle - \langle (\partial_u P(x))\nabla\psi(x), v\rangle , \tag{41}$$
$$= \langle (\partial_v P(x))u - (\partial_u P(x))v, \nabla\psi(x)\rangle . \tag{42}$$

□

Decomposing all vectors into tangent and normal components, we can categorize relevant choices of u, v for some $x \in \mathcal{X}$ into three cases:

1. both reside in $T_x(\mathcal{X})$,
2. both reside in $T_x(\mathcal{X})^\perp$,
3. one resides in $T_x(\mathcal{X})$ and the other is a member of $T_x(\mathcal{X})^\perp$.

We first analyse the case where $u, v \in T_x(\mathcal{X})$. Following lemma helps to show that the commutator part vanishes in this case.

Lemma A.2. *Let $x \in \mathcal{X}$ and $u, v \in T_x(\mathcal{X})$. Then*

$$\langle u, P(x)H_\psi(x)v\rangle = \langle v, P(x)H_\psi(x)u\rangle . \tag{43}$$

Proof. We know both $P(x)$ and $H_\psi(x)$ are symmetric matrices. Moreover, we know for all $u \in T_x(\mathcal{X})$ we have $P(x)u = u$ since $P(x)$ is the orthogonal projection inside $T_x(\mathcal{X})$. Hence,

$$\begin{aligned}\langle u, P(x)H_\psi(x)v\rangle &= \langle P(x)u, H_\psi(x)v\rangle, &(44)\\ &= \langle u, H_\psi(x)v\rangle, &(45)\\ &= \langle v, H_\psi(x)u\rangle, &(46)\\ &= \langle P(x)v, H_\psi(x)u\rangle, &(47)\\ &= \langle v, P(x)H_\psi(x)u\rangle. &(48)\end{aligned}$$

□

Lemma A.2 shows that $P(x)H_\psi(x)$ acts as a symmetric matrix for certain vectors and thus for $x \in \mathcal{X}$ and $u, v \in T_x(\mathcal{X})$ we have

$$\begin{aligned}\langle u, [P(x), H_\psi(x)]\, v\rangle &= \langle u, P(x)H_\psi(x)v\rangle - \langle u, H_\psi(x)P(x)v\rangle, &(49)\\ &= \langle v, P(x)H_\psi(x)u\rangle - \langle u, H_\psi(x)P(x)v\rangle, &(50)\\ &= 0. &(51)\end{aligned}$$

Hence,

$$\left\langle u, \left(J_F(x) - J_F^T(x)\right) v\right\rangle = \langle u, \Phi(x)v\rangle. \tag{52}$$

Next lemma helps to show that $\Phi(x)$ obstruction also vanishes in this case:

Lemma A.3. *Assume $x \in \mathcal{X}$, $u \in T_x(\mathcal{X})$ and $w \in \mathbb{R}^d$. Letting c_j be the j-th column of $J_\phi(x)^T \left(J_\phi(x)J_\phi(x)^T\right)^{-1}$ we have*

$$(\partial_w P)u = -\sum_{j=1}^{l} \left(u^T H_{\phi_j} w\right) c_j. \tag{53}$$

Proof. Let $M = J_\phi J_\phi^T$ and $Q = J_\phi^T M^{-1} J_\phi$ and then we have that $P = I - Q$ where we have removed the dependence on x for brevity.

We have

$$\partial_w Q = (\partial_w J_\phi)^T M^{-1} J_\phi + J_\phi^T (\partial_w M^{-1}) J_\phi + J_\phi^T M^{-1} (\partial_w J_\phi). \tag{54}$$

Since $J_\phi u = 0$, we get

$$(\partial_w Q)u = J_\phi^T M^{-1} (\partial_w J_\phi) u. \tag{55}$$

Moreover,

$$(\partial_w J_\phi)u = \begin{bmatrix} u^T H_{\phi_1} w \\ \vdots \\ u^T H_{\phi_l} w \end{bmatrix}, \tag{56}$$

hence,

$$(\partial_w Q)u = \sum_{j=1}^{l} \left(u^T H_{\phi_j} w\right) c_j. \tag{57}$$

Consequently,

$$\begin{aligned}(\partial_w P)u &= -(\partial_w Q)u, &(58)\\ &= -\sum_{j=1}^{l} \left(u^T H_{\phi_j} w\right) c_j. &(59)\end{aligned}$$

□

Corollary A.4. *For every $x \in \mathcal{X}$ and $u, v \in T_x(\mathcal{X})$ we have*

$$\left(\partial_u P(x)\right) v = \left(\partial_v P(x)\right) u. \tag{60}$$

Proof. Continuing from Lemma A.3, it is enough to show $(\partial_u Q)v = (\partial_v Q)u$. We have

$$(\partial_u Q)v = \sum_{j=1}^{l} \left(u^T H_{\phi_j} v\right) c_j. \tag{61}$$

However, $u^T H_{\phi_j} v = v^T H_{\phi_j} u$ since H_{ϕ_j} is symmetric and thus

$$(\partial_u Q)v = \sum_{j=1}^{l} \left(u^T H_{\phi_j} v\right) c_j, \tag{62}$$

$$= \sum_{j=1}^{l} \left(v^T H_{\phi_j} u\right) c_j, \tag{63}$$

$$= (\partial_v Q)u. \tag{64}$$

□

Consequently, we reach the following:

Proposition A.5. *For every $x \in \mathcal{X}$ and $u, v \in T_x(\mathcal{X})$ we have*

$$\left\langle u, \left(J_F(x) - J_F^T(x)\right) v \right\rangle = 0. \tag{65}$$

Proof. Combining Lemmas A.1, A.2 and Corollary A.4:

$$\left\langle u, \left(J_F(x) - J_F^T(x)\right) v \right\rangle = \langle u, \Phi(x)v \rangle, \tag{66}$$

$$= \langle (\partial_v P(x))u - (\partial_u P(x))v, \nabla\psi(x) \rangle, \tag{67}$$

$$= 0. \tag{68}$$

□

Proposition A.5 shows that the obstruction vanishes as long as test vectors u, v are tangent to $\mathcal{X}$.

Next, we will test F with vectors $u, v \in T_x(\mathcal{X})^\perp$. In this case, the commutator part trivially vanishes as $P(x)u = P(x)v = 0$. So, we will start again from

$$\left\langle u, \left(J_F(x) - J_F^T(x)\right) v \right\rangle = \langle u, \Phi(x)v \rangle, \tag{69}$$

but this time $u, v \in T_x(\mathcal{X})^\perp$.

Lemma A.6. *Assume $x \in \mathcal{X}$, $u \in T_x(\mathcal{X})^\perp$ and $w \in \mathbb{R}^d$. Let $u = J_\phi(x)^T \tilde{u}$ and $\tilde{u} = (J_\phi(x)J_\phi(x)^T)^{-1} J_\phi(x)u \in \mathbb{R}^l$. Then,*

$$(\partial_w P(x))u = -P(x) \sum_{j=1}^{l} \tilde{u}_j H_{\phi_j}(x) w. \tag{70}$$

Proof. Let $M = J_\phi J_\phi^T$ and $Q = J_\phi^T M^{-1} J_\phi$ and then we have that $P = I - Q$ where we have removed the dependence on x for brevity. Keep in mind that M is invertible since we have assumed that J_ϕ has full row-rank. Thus, $\partial_w P = -\partial_w Q$ and

$$\partial_w Q = (\partial_w J_\phi)^T M^{-1} J_\phi + J_\phi^T (\partial_w M^{-1}) J_\phi + J_\phi^T M^{-1} (\partial_w J_\phi). \tag{71}$$

Applying to u and using

$$\partial_w M^{-1} = -M^{-1} \left((\partial_w J_\phi) J_\phi^T + J_\phi^T (\partial_w J_\phi)\right) M^{-1} \tag{72}$$

we get

$$(\partial_w Q)u = (\partial_w J_\phi)^T \tilde{u} - J_\phi^T M^{-1} (\partial_w J_\phi) u - Q(\partial_w J_\phi)^T \tilde{u} + J_\phi^T M^{-1} (\partial_w J_\phi) u, \tag{73}$$

$$= (I - Q)(\partial_w J_\phi)^T \tilde{u}, \tag{74}$$

$$= P(\partial_w J_\phi)^T \tilde{u}. \tag{75}$$

□

We reach the following proposition:

Proposition A.7. *For every $x \in \mathcal{X}$ and $u, v \in T_x(\mathcal{X})^\perp$, and letting $\nabla\psi(x)_\parallel = F(x)$ be the tangent component of $\nabla\psi(x)$, we have*

$$\left\langle u, \left(J_F(x) - J_F^T(x)\right) v \right\rangle = \left\langle \sum_{j=1}^{l} \tilde{v}_j H_{\phi_j}(x)u - \tilde{u}_j H_{\phi_j}(x)v, \nabla\psi(x)_\parallel \right\rangle. \tag{76}$$

Proof. Using Lemmas A.1 and A.6 we have

$$\begin{aligned}
\left\langle u, \left(J_F(x) - J_F^T(x)\right) v \right\rangle &= \langle u, \Phi(x)v \rangle, && (77)\\
&= \langle (\partial_v P(x))u - (\partial_u P(x))v, \nabla\psi(x) \rangle, && (78)\\
&= \left\langle P(x) \sum_{j=1}^{l} \tilde{v}_j H_{\phi_j}(x)u - \tilde{u}_j H_{\phi_j}(x)v, \nabla\psi(x) \right\rangle, && (79)\\
&= \left\langle \sum_{j=1}^{l} \tilde{v}_j H_{\phi_j}(x)u - \tilde{u}_j H_{\phi_j}(x)v, \nabla\psi(x)_\parallel \right\rangle. && (80)
\end{aligned}$$

□

Unlike the case where both test vectors were tangent, choosing u, v to be normal reveals that the Hessian of ϕ can obstruct conservativeness of F as long as ψ is not constant on $\mathcal{X}$. The Hessian of $\phi(x)$ determines the curvature of $\mathcal{X}$ at some $x \in \mathcal{X}$; thus, the curvature of $\mathcal{X}$ is the main obstruction in this case. Nevertheless, this obstruction vanishes as well when $l = 1$:

Corollary A.8. *Let $l = 1$, $x \in \mathcal{X}$ and $u, v \in T_x(\mathcal{X})^\perp$. Then,*

$$\left\langle u, \left(J_F(x) - J_F^T(x)\right) v \right\rangle = 0. \tag{81}$$

Proof. In this case $\tilde{u}, \tilde{v}$ are scalars and u and v have to be parallel:

$$\tilde{v}H_\phi(x)u - \tilde{u}H_\phi(x)v = \tilde{v}H_\phi(x)(\tilde{u}\nabla\phi(x)) - \tilde{u}H_\phi(x)(\tilde{v}\nabla\phi(x)) = 0. \tag{82}$$

□

So, it could be argued that both cases are not fundamental geometric obstructions to conservativeness. Finally, we consider the mixed case where we assume $u \in T_x(\mathcal{X}), v \in T_x(\mathcal{X})^\perp$ without loss of generality.

Lemma A.9. *For $x \in \mathcal{X}$ and $u \in T_x(\mathcal{X}), v \in T_x(\mathcal{X})^\perp$ we have*

$$\langle u, [P(x), H_\psi(x)]\, v \rangle = \langle u, H_\psi(x)v \rangle. \tag{83}$$

Proof. Since $P(x)u = u$ and $P(x)v = 0$ we have

$$\begin{aligned}
\langle u, [P(x), H_\psi(x)]\, v \rangle &= \langle P(x)u, H_\psi(x)v \rangle - \langle H_\psi(x)u, P(x)v \rangle, && (84)\\
&= \langle u, H_\psi(x)v \rangle. && (85)
\end{aligned}$$

□

Proposition A.10. *Consider $x \in \mathcal{X}$, $u \in T_x(\mathcal{X})$ and $v \in T_x(\mathcal{X})^\perp$. Let $z = z_\parallel + z_\perp$ be the decomposition of $z \in T_x(\mathbb{X})$ into tangent $z_\parallel \in T_x(\mathcal{X})$ and normal $z_\perp \in T_x(\mathcal{X})^\perp$ components. Further, assume c_j and $\tilde{u}, \tilde{v}$ are defined as in Lemmas A.3 and A.6. Then we have:*

$$\begin{aligned}
\left\langle u, \left(J_F(x) - J_F^T(x)\right) v \right\rangle = \langle u, H_\psi(x)v \rangle &\\
+ \sum_{j=1}^{l} \tilde{v}_j \left\langle H_{\phi_j}(x)u, \nabla\psi(x)_\parallel \right\rangle &\\
- \sum_{j=1}^{l} \left\langle u, H_{\phi_j}(x)v \right\rangle \langle c_j(x), \nabla\psi(x)_\perp \rangle. && (86)
\end{aligned}$$

Proof. We will omit dependence on x for brevity. From Lemma A.3 we have

$$\langle (\partial_v P)u, \nabla\psi\rangle = -\sum_{j=1}^{l} \left\langle u, H_{\phi_j} v\right\rangle \langle c_j, \nabla\psi\rangle . \tag{87}$$

Similarly, from Lemma A.6 we have

$$\langle (\partial_u P)v, \nabla\psi\rangle = -\sum_{j=1}^{l} \tilde{v}_j \left\langle H_{\phi_j} u, \nabla\psi_{\|}\right\rangle . \tag{88}$$

Since $c_j \in T_x(\mathcal{X})^{\perp}$ we have $\langle c_j, \nabla\psi\rangle = \langle c_j, \nabla\psi_{\perp}\rangle$. Combining with Lemma A.1 we have

$$\langle u, \Phi v\rangle = \langle (\partial_v P)u - (\partial_u P)v, \nabla\psi\rangle , \tag{89}$$

$$= -\sum_{j=1}^{l} \left\langle u, H_{\phi_j} v\right\rangle \langle c_j, \nabla\psi_{\perp}\rangle + \sum_{j=1}^{l} \tilde{v}_j \left\langle H_{\phi_j} u, \nabla\psi_{\|}\right\rangle . \tag{90}$$

Adding the commutator effect from Lemma A.9 to $\langle u, \Phi v\rangle$ we conclude the proposition. □

Corollary A.11. *Let $l = 1$. Then, for every $x \in \mathcal{X}$, $u \in T_x(\mathcal{X})$ and $v \in T_x(\mathcal{X})^{\perp}$ we have*

$$\left\langle u, \left(J_F(x) - J_F^T(x)\right) v\right\rangle = \\ \langle u, H_{\psi}(x)v\rangle + \tilde{v}\left\langle H_{\phi}(x)u, \nabla\psi(x)_{\|} - \nabla\psi(x)_{\perp}\right\rangle . \tag{91}$$

Proof. For $l = 1$ we have $c = \nabla\phi/\|\nabla\phi\|^2$ and the formula for $\langle u, \Phi v\rangle$ collapses to

$$\langle u, \Phi v\rangle = \tilde{v}\left\langle H_{\phi} u, \nabla\psi_{\|}\right\rangle - \langle u, H_{\phi} v\rangle \langle c, \nabla\psi_{\perp}\rangle , \tag{92}$$

$$= \tilde{v}\left\langle H_{\phi} u, \nabla\psi_{\|}\right\rangle - \tilde{v}\langle H_{\phi} u, \nabla\phi\rangle \frac{\langle \nabla\phi, \nabla\psi_{\perp}\rangle}{\|\nabla\phi\|^2}, \tag{93}$$

$$= \tilde{v}\left\langle H_{\phi} u, \nabla\psi_{\|} - \nabla\psi_{\perp}\right\rangle . \tag{94}$$

□

We can see that the curvature of $\mathcal{X}$ reappears in Corollary A.11. This time, however, the obstruction from the curvature does not vanish without making assumptions about ψ. As a result, we argue that the most fundamental obstruction to conservativeness of F is the curvature of $\mathcal{X}$.

We now have the required machinery to prove Theorem 3.2:

Proof. By bilinearity of $\langle u, (J_F - J_F^T)v\rangle$ in (u, v), we expand over the four blocks obtained from the decompositions $u = u_{\|} + u_{\perp}$, $v = v_{\|} + v_{\perp}$:

$$\left\langle u, (J_F - J_F^T)v\right\rangle = \left\langle u_{\|} + u_{\perp}, (J_F - J_F^T)(v_{\|} + v_{\perp})\right\rangle . \tag{95}$$

The remainder of the proof applies the proved propositions to the resulting four expressions:

- $\left\langle u_{\|}, (J_F - J_F^T)v_{\|}\right\rangle$: According to Proposition A.5 we have

$$\left\langle u_{\|}, (J_F - J_F^T)v_{\|}\right\rangle = 0. \tag{96}$$

- $\left\langle u_{\perp}, (J_F - J_F^T)v_{\perp}\right\rangle$: According to Proposition A.7 we have

$$\left\langle u_{\perp}, (J_F - J_F^T)v_{\perp}\right\rangle = \mathcal{A}(u, v). \tag{97}$$

- $\left\langle u_{\|}, (J_F - J_F^T)v_{\perp}\right\rangle$: According to Proposition A.10 we have

$$\left\langle u_{\|}, (J_F - J_F^T)v_{\perp}\right\rangle = \mathcal{B}(u, v). \tag{98}$$

- $\langle u_\perp, (J_F - J_F^T)v_\parallel \rangle$: Since $J_F - J_F^T$ is an antisymmetric matrix and according to Proposition A.10 we have

$$\langle u_\perp, (J_F - J_F^T)v_\parallel \rangle = -\langle v_\parallel, (J_F - J_F^T)u_\perp \rangle, \tag{99}$$
$$= -\mathcal{B}(v, u). \tag{100}$$

□

Corollary 3.3 also can now be proved as well:

Proof. This is a direct result of combining Theorem 3.2 with Corollaries A.8 and A.11. □

A.3 THEOREM 3.4

Proof. Assuming h to be constant, it is trivial to show that it is both ϕ-strongly tangent and continuous. For the reverse, we separate the two roles a point $x \in \mathbb{R}^2 \setminus \{0\}$ plays: its distance from $\mathcal{X}$, captured by the style variable $s^2 := \|x\|^2 = x_1^2 + x_2^2$, and its position along $\mathcal{X}$, captured by the content variable c, the angular coordinate with $x = s\,[\cos c \quad \sin c]^T$. For h to be ϕ-strongly tangent to $\mathcal{X}$, it has to be a solution to the PDE $\langle \nabla\phi(x), \nabla h(x) \rangle = 0$.

It is easy to see that $\nabla\phi(x) = x$ and consequently:

$$\langle x, \nabla h(x) \rangle = 0. \tag{101}$$

We can calculate the following:

$$s\nabla s = x, \quad \|\nabla s\|^2 = 1 \quad \text{and} \quad \langle \nabla s, \nabla c \rangle = 0. \tag{102}$$

So, we can attempt to solve the PDE by separation of variables:

$$h(x) = f(c)g(s) \quad \text{and} \quad \nabla h(x) = f(c)g'(s)\nabla s + f'(c)g(s)\nabla c. \tag{103}$$

Hence,

$$\langle x, \nabla h(x) \rangle = \langle s\nabla s, f(c)g'(s)\nabla s + f'(c)g(s)\nabla c \rangle, \tag{104}$$
$$= sf(c)g'(s) = 0. \tag{105}$$

But, $s \neq 0$ and $f(c) = 0$ implies $h(x) = 0$, which is constant. Thus, the only other possibility is:

$$g'(s) = 0 \quad \Rightarrow \quad g(s) = \text{const}. \tag{106}$$

So, it must be that h is only a function of c: $h(x) = f(c)$.

Continuity of h at the origin means that $\lim_{x\to 0} h(x)$ exists and is the same along every path approaching 0. In particular, approaching along each ray of fixed angle c (i.e. letting $s \to 0^+$ with c held fixed) must give the same limit L:

$$\lim_{s\to 0^+} f(c) = L \qquad \text{for every fixed } c. \tag{107}$$

Fix any two angles c_1, c_2. Subtracting the two corresponding limit equations, we have

$$f(c_1) - f(c_2) = \lim_{s\to 0^+} f(c_1) - \lim_{s\to 0^+} f(c_2) = L - L = 0. \tag{108}$$

Since c_1, c_2 were arbitrary, f is constant. As a result, h has to be constant as well. □

B EXPERIMENT DETAILS

B.1 DETAILED RESULTS

The results in Table 5 show that the straight path branch is indeed vulnerable to AutoAttack even though the oracle-path branch is robust. Together with the results presented in Table 3, they show that non-conservativeness is an integral part of the observed robustness.

The details of the alignment experiment of Table 4 can be found in Tables 6,7 and 8.

The details of the experiment regarding the projected vector fields for $\phi(x) = \|x\|^2 - 1$ are reported in Tables 9 and 10.

Table 5: A comparison of robustness of straight and oracle path branches. We can see that the straight paths branch does not enjoy the same robustness as the oracle-paths branch.

Dataset	$x_0 \mapsto x$	$x_0 \mapsto \tilde{x}$	$x_0 \mapsto x \mapsto \tilde{x}$
MNIST	0.9812 ± 0.0219	0.1094 ± 0.0980	0.9812 ± 0.0219
FMNIST	0.8594 ± 0.0423	0.1969 ± 0.0860	0.8562 ± 0.0367
CIFAR10	0.7094 ± 0.0967	0.2375 ± 0.0535	0.7156 ± 0.0877

Table 6: Statistical test results for tangency of sensitivity fields of MNIST classes.

Class (MNIST)	Mean	Std	t	p	Reject H_0
0	0.049708	0.013892	11.3153	0.000001	True
1	0.065469	0.015497	13.3597	0.000000	True
2	0.053670	0.026845	6.3222	0.000069	True
3	0.070206	0.023881	9.2966	0.000003	True
4	0.037869	0.019774	6.0562	0.000095	True
5	0.017238	0.018347	2.9711	0.007838	True
6	0.097039	0.017669	17.3670	0.000000	True
7	0.000740	0.018832	0.1242	0.451943	False
8	-0.017825	0.020326	-2.7732	0.989179	False
9	0.121233	0.019758	19.4032	0.000000	True

B.2 Training hyperparameters and execution time

All training runs used a batch size of 32. Reported runtimes correspond to the approximate wall-clock time required for a complete experimental run.

B.3 Neural Network Architectures

This section summarizes the neural network architectures used in the experiments.

B.3.1 Vector Field Models

MNIST and FashionMNIST. For MNIST and FashionMNIST experiments, each vector field component was modeled using a multilayer perceptron (MLP) with three hidden layers and ReLU activations. The input image was flattened into a vector of dimension $d = 784$.

The complete vector field model consists of k such subnetworks operating in parallel using linear operators.

CIFAR10. For CIFAR10 experiments, each vector field component was modeled using a convolutional neural network with four convolutional blocks followed by fully connected layers.

As with the MNIST and FashionMNIST experiments, the full vector field model consists of k parallel subnetworks coupled through block-diagonal operators.

B.3.2 Single-Valued Classification Models

MNIST and FashionMNIST. For MNIST and FashionMNIST classification experiments, we used a standard multilayer perceptron with two hidden layers.

CIFAR10. For CIFAR10 classification experiments, we used a convolutional neural network with four convolutional layers followed by three fully connected hidden layers.

Table 7: Statistical test results for tangency of sensitivity fields FMNIST classes.

Class (FMNIST)	Mean	Std	t	p	Reject H_0
0	0.031649	0.017503	5.7180	0.000144	True
1	0.014090	0.031566	1.4115	0.095861	False
2	0.023002	0.009499	7.6576	0.000016	True
3	-0.014175	0.019385	-2.3125	0.976975	False
4	0.038417	0.015064	8.0645	0.000010	True
5	-0.018305	0.013679	-4.2317	0.998900	False
6	0.036350	0.015069	7.6283	0.000016	True
7	0.030101	0.025293	3.7634	0.002231	True
8	0.112302	0.021207	16.7459	0.000000	True
9	0.006247	0.023542	0.8392	0.211551	False

Table 8: Statistical test results for tangency of sensitivity fields CIFAR10 classes.

Class (CIFAR10)	Mean	Std	t	p	Reject H_0
0	0.199310	0.037169	16.9571	0.000000	True
1	0.260445	0.031372	26.2530	0.000000	True
2	0.044944	0.015238	9.3273	0.000003	True
3	0.063505	0.009836	20.4168	0.000000	True
4	0.312961	0.014637	67.6141	0.000000	True
5	0.167604	0.031557	16.7953	0.000000	True
6	0.245204	0.019867	39.0293	0.000000	True
7	0.140325	0.026404	16.8058	0.000000	True
8	0.305110	0.023278	41.4488	0.000000	True
9	0.266938	0.023690	35.6325	0.000000	True

B.4 Adversarial Attack Hyperparameters

Adversarial robustness evaluations were performed using the AutoAttack implementation provided by the Adversarial Robustness Toolbox (ART). Unless otherwise stated, all attack hyperparameters were kept at their default ART values. Total execution time was approximately 6 hours for MNIST, 2 hours for FMNIST, and 24 hours for CIFAR10.

For MNIST and FashionMNIST experiments, we used an ℓ_∞ perturbation budget of $\epsilon = 0.3$. For CIFAR10 experiments, we used $\epsilon = 8/255$.

The default ensemble of attacks used by `AutoAttack` in ART consists of:

- Auto-PGD with cross-entropy loss (APGD-CE),
- Auto-PGD with difference-of-logits-ratio loss (APGD-DLR),
- DeepFool,
- Square Attack.

The detailed hyperparameters for each attack are summarized in Table 17.

Table 9: Statistical test results for conservativeness of MNIST classes

Class (MNIST)	Mean	Std	t	p
Overall	0.3812	0.1904	6.3307	0.000136
0	0.5299	0.2481	6.7550	0.000083
1	0.3873	0.1529	8.0085	0.000022
2	0.4513	0.1973	7.2326	0.000049
3	0.2812	0.0888	10.0162	0.000004
4	0.4247	0.1589	8.4489	0.000014
5	0.3643	0.1740	6.6190	0.000097
6	0.4117	0.1465	8.8836	0.000009
7	0.3090	0.1308	7.4685	0.000038
8	0.2378	0.0884	8.5023	0.000014
9	0.4151	0.2502	5.2454	0.000531

Table 10: Statistical test results for conservativeness of FMNIST classes

Class (FMNIST)	Mean	Std	t	p
Overall	0.1880	0.1187	5.0101	0.000729
0	0.1675	0.1093	4.8457	0.000914
1	0.2381	0.1323	5.6899	0.000298
2	0.1800	0.1275	4.4632	0.001570
3	0.1531	0.1039	4.6601	0.001185
4	0.1766	0.1020	5.4760	0.000392
5	0.2437	0.1608	4.7910	0.000986
6	0.1866	0.1275	4.6278	0.001240
7	0.2100	0.0863	7.6948	0.000030
8	0.1879	0.1044	5.6884	0.000299
9	0.1370	0.0575	7.5361	0.000036

Table 11: Training hyperparameters and approximate execution times.

Dataset	Model Type	Epochs	Optimizer / Learning Rate	Approx. Runtime
MNIST	Multi-valued	3	Adam / 10^{-4}	$\sim$ 30 min
FashionMNIST	Multi-valued	3	Adam / 10^{-4}	$\sim$ 30 min
CIFAR10	Multi-valued	10	Adam / 10^{-4}	$\sim$ 16 h

Table 12: Architecture of an individual vector field network for MNIST and FashionMNIST.

Layer	Output Dimension	Activation
Flatten	784	–
Linear	256	ReLU
Linear	128	ReLU
Linear	256	ReLU
Linear	784	–
Unflatten	$(1, 28, 28)$	–

Table 13: Architecture of an individual vector field network for CIFAR10.

Layer	Kernel / Stride	Output Channels	Activation
Conv2D	3×3 / 1	64	ReLU
MaxPool2D	2×2 / 2	–	–
Conv2D	3×3 / 1	128	ReLU
MaxPool2D	2×2 / 2	–	–
Conv2D	3×3 / 1	256	ReLU
MaxPool2D	2×2 / 2	–	–
Conv2D	3×3 / 1	512	ReLU
MaxPool2D	2×2 / 2	–	–
Flatten	2048	–	
Linear	1024	ReLU	
Linear	2048	ReLU	
Linear	3072	–	
Unflatten	$(3, 32, 32)$	–	

Table 14: Single-valued classifier architecture for MNIST and FashionMNIST.

Layer	Output Dimension	Activation
Flatten	784	–
Linear	512	ReLU
Linear	256	ReLU
Linear	10	–

Table 15: Single-valued classifier architecture for CIFAR10.

Layer	Kernel / Stride	Output Channels	Activation
Conv2D	3×3 / 1	64	ReLU
MaxPool2D	2×2 / 2	–	–
Conv2D	3×3 / 1	128	ReLU
MaxPool2D	2×2 / 2	–	–
Conv2D	3×3 / 1	256	ReLU
MaxPool2D	2×2 / 2	–	–
Conv2D	3×3 / 1	512	ReLU
MaxPool2D	2×2 / 2	–	–
Flatten	2048	–	
Linear	1024	ReLU	
Linear	512	ReLU	
Linear	256	ReLU	
Linear	10	–	

Table 16: Global AutoAttack hyperparameters.

Hyperparameter	Value
Norm	ℓ_∞
ϵ (MNIST/FashionMNIST)	0.3
ϵ (CIFAR10)	$8/255$
Step size (ϵ_{step})	0.1
Batch size	32
Targeted attacks	False
Parallel pool size	0

Table 17: Component attacks and corresponding hyperparameters used in ART AutoAttack.

Attack	Hyperparameter	Value
APGD-CE	Max iterations	100
	Random restarts	5
	Loss	Cross-entropy
	Targeted	False
	Batch size	32
APGD-DLR	Max iterations	100
	Random restarts	5
	Loss	Difference-of-logits-ratio
	Targeted	False
	Batch size	32
DeepFool	Max iterations	100
	Overshoot parameter (ϵ)	10^{-3}
	Number of gradients	10
	Batch size	32
Square Attack	Max iterations	5000
	Initial square fraction (p_{init})	0.8
	Restarts	5