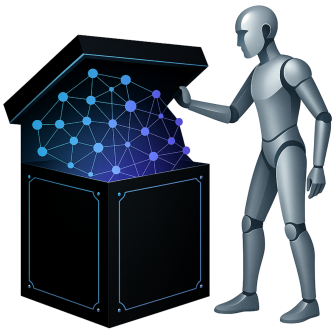
ZUSE INSTITUTE BERLIN

Explaining Deep Neural Networks Through Inverse Classification

PhD Defense

Shpresim Sadiku

October 13, 2025



ZUSE INSTITUTE BERLIN

Explaining DNNs Through Inverse Classification

Explainable Artificial Intelligence (XAI)

Goal. Enhance transparency/interpretability of ML models by providing intelligible justifications for decisions in high-stakes domains.

Interpretability. *Degree to which a human can consistently predict the model's output.*
Explainability. *Degree to which a human can understand the cause of a decision.*

Why it matters.

- DNNs are accurate yet opaque ("black-box"). Trust, accountability, and governance require explanations.
- Distinguish *prediction* (model output) from *prescription* (human action).
- Aim: alignment of model reasoning with domain knowledge and real-world expectations.

Inverse Classification and Adversarial Perturbations

Given a trained classifier f_{θ} and input $\boldsymbol{x} \in \mathbb{R}^d$ with $f_{\theta}(\boldsymbol{x}) = y$, **inverse classification** seeks a minimally modified $\tilde{\boldsymbol{x}} = \boldsymbol{x} + \boldsymbol{r}$ with desired label $\tilde{y} \neq y$:

$$\min_{\boldsymbol{r} \in \mathbb{R}^d} \ \mathcal{L}(f_{\theta}(\boldsymbol{x} + \boldsymbol{r}), \tilde{y}) \quad \text{s.t.} \quad \|\boldsymbol{r}\|_p \leq \epsilon. \tag{1}$$

Two major classes of such perturbations:

- **Counterfactual Explanations (CFEs)**: human-oriented; prioritize plausibility, feasibility, and recourse.
- **Adversarial Attacks**: robustness evaluation; imperceptible yet effective (esp. in images).

Shared maths: both solve constrained optimization that minimally alters the decision; they differ in downstream *constraints* (plausibility vs. worst-case failure).

Explanations: Mapping and Desiderata

What is an explanation? A mapping $E(\boldsymbol{x}, f_{\boldsymbol{\theta}})$ to a human-interpretable object (textual, visual, symbolic).

Causal query. Why output y for input $\boldsymbol{x}$? (e.g., loan rejection due to low credit score.)

Desirable properties

- *Comprehensibility*: understandable to non-experts.
- *Stability*: small input changes $\Rightarrow$ similar explanations.
- *Consistency*: same input $\Rightarrow$ similar explanations across runs/models.
- *Realism*: counterfactuals should be feasible/in-manifold.

Open question: what constitutes a "good" explanation remains unsettled.

Existence of Adversarial Perturbations (Theory)

For two-layer ReLU networks $f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \sum_{i=1}^{m} u_i \, \sigma(\boldsymbol{w}_i^\top \boldsymbol{x})$ with inputs near a subspace $P \subset \mathbb{R}^d$ ($\dim P^\perp = \ell$):

- After training, the input gradient has a **large $P^\perp$ component** with high probability:

$$\left\| \Pi_{P^\perp} (\partial f_{\boldsymbol{\theta}} / \partial \boldsymbol{x}) \right\| \;\geq\; \sqrt{\tfrac{k\ell}{2md}}, \tag{2}$$

  where $k = |\{\text{active neurons}\}|$.

- There exists a **universal $\boldsymbol{r} \in P^\perp$** such that $\operatorname{sign}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x} + \boldsymbol{r})\big) \neq \operatorname{sign}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x})\big)$ and

$$\|\boldsymbol{r}\| \;\leq\; \mathcal{O}\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}) \cdot \sqrt{\tfrac{m}{k_y}} \cdot \sqrt{\tfrac{d}{\ell}} \right), \tag{3}$$

  with k_y neurons aligned with label y.

Takeaway. Even small off-manifold perturbations can flip decisions; universal directions may exist.

Adversarial and Counterfactual Methods

FGSM (targeted/untargeted).

$$\tilde{\boldsymbol{x}} = \boldsymbol{x} - \epsilon \operatorname{sign}\big(\nabla_{\boldsymbol{x}} \mathcal{L}(f_{\boldsymbol{\theta}}(\boldsymbol{x}), \tilde{y})\big). \tag{4}$$

PGD (ℓ_∞-bounded).

$$\boldsymbol{x}_{t+1} = \Pi_{\mathcal{B}_\epsilon^\infty(\boldsymbol{x})}\Big(\boldsymbol{x}_t - \alpha \operatorname{sign}(\nabla_{\boldsymbol{x}} \mathcal{L}(f_{\boldsymbol{\theta}}(\boldsymbol{x}_t), \tilde{y}))\Big). \tag{5}$$

CFEs. Solve variants of (1) with *plausibility* constraints/regularisation:

- Distance-based (e.g., Manhattan/Mahalanobis), actionability constraints.
- Density-aware: hard constraints via GMM components; or soft regularisers (e.g., LOF-based).

Structured attacks (images). Group-sparse (e.g., ADMM-based), nuclear-group norms, homotopy sparse attacks.

Optimization Toolbox for Perturbations

First-order methods.

- GD/SGD: $\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \eta \nabla_{\boldsymbol{x}} s_\theta(\boldsymbol{x}_t)$; momentum and Nesterov variants improve stability/speed.

Proximal gradient (PG).

$$\boldsymbol{x}_{t+1} = \mathrm{prox}_{\lambda g}\Big(\boldsymbol{x}_t - \lambda \nabla h(\boldsymbol{x}_t)\Big), \quad \lambda \approx 1/L. \tag{6}$$

Acceleration (FISTA). Nesterov-type extrapolation yields $\mathcal{O}(1/t^2)$ in convex settings.

Thresholding operators.

- *Soft* (ℓ_1) and *hard* (ℓ_0) thresholding (closed forms).
- Nonconvex $\ell_{1/2}$: explicit proximal update using $\phi_{2\lambda}(x_i)$ and threshold $g(2\lambda)$.

Message. These tools enable principled trade-offs between imperceptibility, sparsity/structure, and target success.

ZUSE INSTITUTE BERLIN

Explaining DNNs Through Inverse Classification

Introduction and Motivation

- DNNs are vulnerable to adversarial perturbations across tasks: classification, captioning, retrieval, QA, autonomous driving, face recognition/detection, etc. (e.g., Carlini et al. 2017, Athalye et al. 2018, Zhang et al. 2020).
- Beyond ℓ_p with $p \geq 1$, the $p = 0$ (sparse) regime is compelling: few pixels changed *without* constraints on *where* and *by how much* $\Rightarrow$ perceptible artifacts (Su et al. 2019).
- **Need:** impose *structure* $\Rightarrow$ **group-wise sparse** perturbations targeted at the object of interest (Xu et al. 2018, Zhu et al. 2021, Imtiaz et al. 2022, Kazemi et al. 2023).
- Bridges human perception vs. machine features (Ilyas et al. 2019); perturbations become *explainable*.

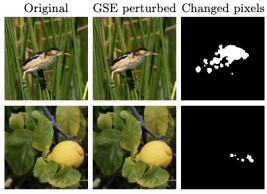


Figure: Adversarial attacks generated by GSE algorithm.

Contributions (GSE)

Two-phase algorithm for group-wise sparse, low-magnitude, explainable attacks.

1. **Phase I (Selection).** Non-convex regularisation with proximal splitting + a proximity-based update of per-pixel tradeoff parameters $\boldsymbol{\lambda}$ to select salient pixel *groups*.
2. **Phase II (Refinement).** Nesterov's accelerated gradient (projected onto selected coordinates) with ℓ_2-regularisation to minimise perturbation magnitude.
3. **Empirics.** CIFAR-10 and ImageNet: up to **50.9%** (CIFAR-10) and **38.4%** (ImageNet) higher group-wise sparsity (targeted, average case) at **100%** ASR.

Evaluation: ASR; sparsity (ACP), grouping (ANC, $d_{2,0}$), magnitude (ℓ_2), explainability (ASM-based IS), runtime.

Explainability. Quantitatively aligns perturbations with salient regions (ASM/CAM), outperforming SOTA sparse and group-wise sparse attacks.

Related Work (Sparse & Group-wise Sparse Attacks)

- **Sparse (p=0):** one-pixel (Su et al. 2019); local search (Narodytska et al. 2016); evolutionary methods (Croce et al. 2019); ℓ_1 relaxations, e.g., SparseFool (Modas et al. 2019). Often perceptible; location/magnitude unconstrained.
- **Group-wise sparse:**
  - StrAttack (Xu et al. 2018): ADMM with sliding masks.
  - SAPF (Fan et al. 2020): ℓ_p-Box ADMM with binary selections.
  - Homotopy-Attack (Zhu et al. 2021): nmAPG; SLIC-based $2,0-$'norm' regularisation.
  - FWnucl (Kazemi et al. 2023): nuclear group norm.
- Contrary to benchmarks, GSE method does not depend on **pixel partitionings**.
- **Links to explanations:** hitting-set duality on MNIST (Ignatiev et al. 2019); perturbations trace discriminative features (Xu et al. 2018).

Adversarial Attack Formulation

Feasible images: $\mathcal{X} = [I_{\min}, I_{\max}]^{M \times N \times C}$. Benign image $\boldsymbol{x} \in \mathcal{X}$ with label $y \in \mathbb{N}$, target $\tilde{y} \in \mathbb{N}$ ($\tilde{y} \neq y$). Classifier $f_{\boldsymbol{\theta}}$ and loss $\mathcal{L}$.

$$\min_{\boldsymbol{r} \in \mathbb{R}^{M \times N \times C}} \mathcal{L}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x} + \boldsymbol{r}), \tilde{y}\big) + \lambda \mathcal{D}(\boldsymbol{r}). \tag{7}$$

$$\max_{\boldsymbol{r} \in \mathbb{R}^{M \times N \times C}} \mathcal{L}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x} + \boldsymbol{r}), y\big) - \lambda \mathcal{D}(\boldsymbol{r}). \tag{8}$$

Sparse regularisation: $\mathcal{D}(\cdot) = \|\cdot\|_p^p$, $0 < p < 1$.

1/2-Quasinorm Regularisation and FBS

Quasinorm-regularised objective (sparse attacks):

$$\min_{\boldsymbol{r}} \; \mathcal{L}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}+\boldsymbol{r}), y\big) + \lambda \, \|\boldsymbol{r}\|_p^p, \quad 0 < p < 1. \tag{9}$$

For $p = \frac{1}{2}$, the proximal operator admits a **closed form** (component-wise).

Algorithm Forward–Backward Splitting Attack (sketch)

1: Initialise $\boldsymbol{r}_0 \leftarrow \mathbf{0}$
2: **for** $t = 0, \ldots, T-1$ **do**
3: $\quad \boldsymbol{r}_{t+1} \leftarrow \mathrm{prox}_{\alpha_t \lambda \|\cdot\|_p^p}\Big(\boldsymbol{r}_t - \alpha_t \, \nabla_{\boldsymbol{r}} \, \mathcal{L}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}+\boldsymbol{r}_t), y\big)\Big)$
4: **end for**
5: Return $\tilde{\boldsymbol{r}} = \boldsymbol{r}_T$

Limitation: yields very sparse but often **large-magnitude** and poorly localised perturbations (Fan et al. 2020).

GSE: Group-wise Sparse, Low-Magnitude Attack

Phase I (Select coordinates):

- Use a per-pixel vector $\boldsymbol{\lambda} \in \mathbb{R}_{\geq 0}^{M \times N \times C}$ in the $\frac{1}{2}$-quasinorm proximal step.
- Build $\boldsymbol{m} = \operatorname{sign}\left(\sum_{c=1}^{C} |\boldsymbol{r}_t|_{:,:,c}\right)$; blur with a Gaussian kernel $\boldsymbol{K}$ to obtain $\boldsymbol{M} = \boldsymbol{m} * * \boldsymbol{K}$.
- Form $\overline{\boldsymbol{M}}$ via $\overline{M}_{ij} = M_{ij} + 1$ if $M_{ij} \neq 0$, else $q \in (0, 1]$; update $\lambda_{t+1}^{i,j,:} = \frac{1}{\overline{M}_{ij}} \lambda_t^{i,j,:}$.
- After $\hat{t}$ iters, define selected subspace $V = \operatorname{span}\{e_{i,j,c} \mid \lambda_{\hat{t}}^{i,j,c} < \lambda_0^{i,j,c}\}$.

Phase II (Refine on V):

$$\min_{\boldsymbol{r} \in V} \mathcal{L}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x} + \boldsymbol{r}), y\big) + \mu \|\boldsymbol{r}\|_2, \quad \text{solve by projected NAG.} \tag{10}$$

Lemma (Sadiku, Wagner, and Pokutta 2025)

The projected NAG solving Eq. (10) converges as NAG solving an unconstrained problem.

ZUSE INSTITUTE BERLIN

Explaining DNNs Through Inverse Classification

Counterfactual Explanations (CFEs): Motivation

- ML systems operate in high-stakes domains (finance, healthcare, justice, hiring). Opacity ⇒ transparency, fairness, accountability concerns.
- CFEs answer *what-if*: minimal (feasible) changes to flip the decision to a target label (Wachter et al. 2017).
- Contrast with LRP/LIME: attribution of *present* features (Bach et al. 2015, Ribeiro et al. 2016) vs. CFEs identify *absent* features whose presence would change the outcome.

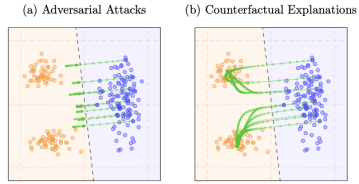


Figure: (a) Without the plausibility term, points cluster near the blue factual data but far from the orange distribution. (b) With the plausibility term, points lie in high-density regions. The dashed line shows the linear decision boundary.

Principles: Proximity, Validity, Actionability, Plausibility, Sparsity

Basic principles.

- *Proximity* (small ℓ_2 distance to factual) and *Validity* ($f_\theta(\tilde{\mathbf{x}}) = \tilde{y}$).
- *Actionability*: respect feature ranges; avoid impossible edits.
- *Plausibility*: move toward target class manifold (not merely across boundary).
- *Sparsity*: change as few features as possible (short explanations are preferred (Mothilal et al. 2020, Naumann et al. 2021)).

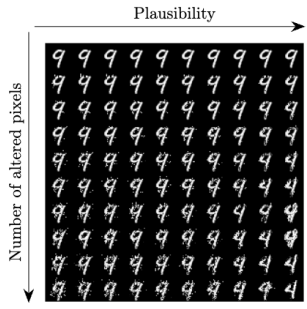


Figure: CFEs for changing $9 \rightarrow 4$: sparsity alone gives adversarial results, plausibility gives unrealistic ones, combining both yields sparse and realistic CFEs.

Canonical CFE Optimisation and Challenges

Canonical form for a factual $\boldsymbol{x}$ (conceptual):

$$\min_{\boldsymbol{x}' \in \text{actionable set}} \Big[\underbrace{\text{CFE loss}}_{\text{validity}} + \underbrace{\text{dist}(\boldsymbol{x}', \boldsymbol{x})}_{\text{proximity}} + \underbrace{\text{dist to manifold}}_{\text{plausibility}} + \underbrace{\#\text{changes}}_{\text{sparsity}} \Big]. \tag{11}$$

Difficulties.

- Nonconvex classifier losses; non-smooth sparsity terms (e.g., ℓ_0); complex manifold penalties; box constraints.
- Prior work tackles subsets: linear/trees with GMM constraints (Artelt et al. 2020); ReLU MIP with LOF (Tsiourvas et al. 2024); density-regularised relaxations (Zhang et al. 2023).

Related Work Landscape

- Early CFEs: weighted ℓ_1/Mahalanobis for sparsity and proximity (Wachter et al. 2017, Verma et al. 2024, Karimi et al. 2020).
- DNNs with VAEs for plausibility (CEM) (Dhurandhar et al. 2018); density-based plausibility with elastic-net (DCFE) (Zhang et al. 2023).
- Convex/GMM approach for simple classifiers (PCFE) (Artelt et al. 2020).
- MIP over ReLU polytopes with LOF constraint (limited to ReLU) (Tsiourvas et al. 2024).

S-CFE: A Simple APG Framework (FISTA-style)

Relaxed objective (penalty form):

$$\min_{x'} \; h(x', \tilde{y}) + g_p(x'), \quad h = \|x' - x\|_2^2 + \gamma\, \mathcal{L}(f_\theta(x'), \tilde{y}) - \tau\, \hat{q}(x', \tilde{y}), \tag{12}$$

$$g_p = I_{\mathcal{A}}(x') + \beta\, \|x' - x\|_p^p, \quad p \in \{\tfrac{1}{2}, \tfrac{2}{3}, 1\} \tag{13}$$

APG step (cf. FISTA):

$$x'_{t+1} = \mathrm{prox}_{\sigma_t g_p}\Big(z_t - \sigma_t \nabla h(z_t, \tilde{y})\Big), \quad z_{t+1} = x'_{t+1} + \alpha_t (x'_{t+1} - x'_t). \tag{14}$$

Plausibility choices: differentiable $\hat{q} \in \{\hat{q}_{\mathrm{KDE}}, \hat{q}_{\mathrm{GMM}}, \hat{q}_{k\mathrm{NN}}\}$.

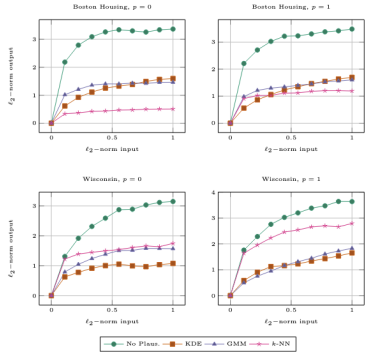
Sparsity control (constrained form):

$$g_0 = I_{\mathcal{A}} + \beta\, I_{\|x' - x\|_0 \le m} \;\Rightarrow\; \mathrm{prox} = \text{projection onto } \{\|x' - x\|_0 \le m\} \cap \mathcal{A}. \tag{15}$$

Empirical Highlights and Robustness

Setup. Boston Housing, Wine, MNIST; logistic/DNN/CNN classifiers; metrics: Validity (%), proximity (ℓ_2), sparsity (ℓ_0), plausibility (LOF), runtime.

- S-CFE variants (KDE/GMM/kNN) produce **sparse** (ℓ_0 controlled), **plausible** (low LOF) CFEs with strong validity—while keeping proximity and runtime competitive.
- Projection onto $\{\|\boldsymbol{x}' - \boldsymbol{x}\|_0 \leq m\} \cap \mathcal{A}$ offers **explicit** sparsity control; density terms steer toward target manifolds.



- Robustness: plausibility constraints improve stability to small input shifts; promotes individual fairness.

ZUSE INSTITUTE BERLIN

Explaining DNNs Through Inverse Classification

Learning from *plausible* counterfactuals (p-CFEs): Why?

- **Goal:** Flip a model's prediction via *minimal* input changes.
- **Two worlds:** Adversarial attacks vs. p-CFEs (plausible, manifold-aligned, interpretable).
- Recent theory: adversarial perturbations contain generalizable, class-specific features (Ilyas et al. 2019, Kumano et al. 2024).

Question: Do p-CFEs share this representational richness? And can they be *better* for learning—especially under spurious correlations?

- **Claim:** Training on p-CFEs attains competitive accuracy *and* mitigates spurious correlations (strong WGA gains).

Contributions

1. **Learning from p-CFEs:** Extend the *learning from perturbations* paradigm from adversarial examples to *plausible* counterfactuals.
2. **Accuracy:** Models trained on p-CFEs reach test accuracy comparable to models trained on adversarial examples (PGD ℓ_2, ℓ_∞) and CFE-ℓ_2.
3. **Spurious correlations:** p-CFE training substantially improves worst-group accuracy (WGA); on WaterBirds it **surpasses** standard training by $\approx 12\,\%$.

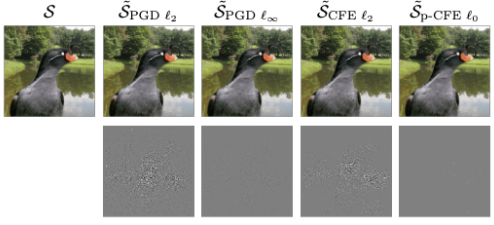


Figure: Random WaterBirds samples with perturbations ($\times 40$) targeting landbird labels from true waterbirds.

Learning from perturbations: setup & objectives

Definition (Learning from perturbations)

Given a dataset $\mathcal{S} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$, create a perturbed set $\tilde{\mathcal{S}} = \{(\tilde{\boldsymbol{x}}_i, \tilde{y}_i)\}_{i=1}^n$ by targeting labels $\tilde{y}_i \neq y_i$; then train a new model on $\tilde{\mathcal{S}}$ and evaluate on the clean test set.

PGD (targeted)

$$\min_{\tilde{\boldsymbol{x}}} \mathcal{L}(f_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}}), \tilde{y})$$
$$\text{s.t. } \|\tilde{\boldsymbol{x}} - \boldsymbol{x}\|_p \leq \epsilon.$$

p-CFE (targeted)

$$\tilde{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}' \in \mathcal{A}} \Big\{ \|\boldsymbol{x}' - \boldsymbol{x}\|_2^2 + \gamma\, \mathcal{L}(f_{\boldsymbol{\theta}}(\boldsymbol{x}'), \tilde{y}) - \tau\, \hat{q}(\boldsymbol{x}', \tilde{y}) + \beta\, \|\boldsymbol{x}' - \boldsymbol{x}\|_0 \Big\}.$$

- Key distinction: the **plausibility** term ($-\tau\, \hat{q}$) pulls counterfactuals toward the target-class manifold; ℓ_0 promotes **sparsity**.

Experimental setup: data, training, metrics

- **Datasets with spurious correlations:**
  - **WaterBirds** (Sagawa et al. 2019): label (land vs. water) spuriously correlates with *background*.
  - **SpuCoAnimals** (Joshi et al. 2023): big vs. small dogs spuriously correlate with *indoor/outdoor*.

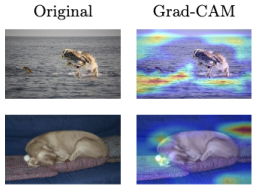


Figure: Grad-CAM visualizations show misclassifications: a landbird on water labeled as a waterbird and a big dog indoors as a small dog.

- **Training:** Fine-tune ResNet50 on perturbed sets (PGD-ℓ_2, PGD-ℓ_∞, CFE-ℓ_2, p-CFE-ℓ_0); target labels $\tilde{y}$ chosen uniformly at random.
- **Metrics:** Train/Test accuracy.
- **Worst-Group Accuracy (WGA)** to quantify spurious reliance.

Results: Accuracy and Worst-Group Accuracy (WGA)

Test accuracy (%)

	PGD-ℓ_2	PGD-ℓ_∞	CFE-ℓ_2	**p-CFE**	Orig.
WaterBirds	86.08	86.02	**88.58**	86.54	87.56
SpuCoAnimals	78.10	79.43	79.00	**81.78**	83.13

Worst-Group Acc. (%)

	PGD-ℓ_2	PGD-ℓ_∞	CFE-ℓ_2	**p-CFE**	Orig.
WaterBirds	56.58	61.72	63.04	**76.05**	64.97
SpuCoAnimals	56.06	57.53	56.60	**63.53**	65.60

- **Takeaways.** p-CFE training: (i) matches adversarial/CFE-ℓ_2 on accuracy; (ii) **strongly** mitigates spurious correlations—+11–12% WGA vs. standard training on WaterBirds.

Qualitative evidence (Grad-CAM) & conclusions

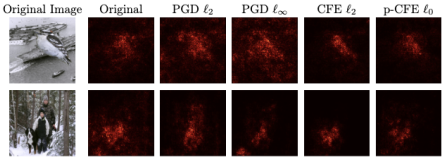


Figure: Saliency maps for a landbird and dog: original, standard, PGD (ℓ_2, ℓ_∞), CFE (ℓ_2), and p-CFE (ℓ_0) models.

Observed focus (Grad-CAM):

- Standard/PGD/CFE-ℓ_2 tend to over-weight *background*.
- **p-CFE** shifts attention to *semantic object* (bird/dog).
- Simple, model-agnostic recipe—no group labels needed.

Conclusions

- p-CFEs are effective training signals: accurate & robust to spurious cues.
- Manifold alignment (plausibility) steers learning toward semantic features.

ZUSE INSTITUTE BERLIN

Explaining DNNs Through Inverse Classification

Future Directions

Adversarial Training with GSE

- Use GSE examples in adversarial training.
- Report robustness–sparsity–explainability–time trade-offs.

S-CFE: Method

- From predictor acceptance → outcome improvement (causal constraints).
- Train on data-level targets, not only model loss.

Model Shifts / Black-Box

- Test KDE / density-gravity plausibility under model change.
- Black-box CFEs: finite-diff or surrogate; consider validity-free variant (accuracy trade-off).

Mixed & Categorical Features

- Design discrete prox/projection (beyond one-hot + APG).

High-Dimensional CFEs

- Swap KDE/GMM for differentiable VAEs/flows; stabilize gradients in $\hat{q}$.

Learning from p-CFEs @ Scale

- Extend to LLMs/VLMs; connect with theory of learning from perturbations.

Adversarial ⇔ p-CFE

- Conjecture: on robust models, targeted attacks ≈ manifold-aligned p-CFEs.
- Diagnostic: angle between attack and p-CFE directions.

ZUSE INSTITUTE BERLIN

Thank you for your attention!

From FISTA to NAG when $g = 0$

- **Set** $g = 0$**.** Then $\text{prox}_{\alpha g} = \text{Id}$, so the update rule of FISTA becomes a plain gradient step at the look-ahead point $\boldsymbol{y}_t$:

$$\boldsymbol{x}_{t+1} = \boldsymbol{y}_t - \alpha \nabla f(\boldsymbol{y}_t). \tag{1}$$

- The extrapolation coefficient is

$$\mu_{t+1} := \frac{\beta_t - 1}{\beta_{t+1}} \quad \Longrightarrow \quad \boldsymbol{y}_{t+1} = \boldsymbol{x}_{t+1} + \mu_{t+1}(\boldsymbol{x}_{t+1} - \boldsymbol{x}_t). \tag{2}$$

- Define instead the time-aligned coefficient

$$\mu_t := \frac{\beta_{t-1} - 1}{\beta_t}. \tag{3}$$

- From (2) with index shifted, this gives

$$\boldsymbol{y}_t = \boldsymbol{x}_t + \mu_t(\boldsymbol{x}_t - \boldsymbol{x}_{t-1}). \tag{4}$$

From FISTA to NAG when $g = 0$ (cont.)

- **Introduce the "velocity" $\boldsymbol{v}_t$:**

$$\boldsymbol{v}_t := \boldsymbol{x}_t - \boldsymbol{x}_{t-1}. \tag{5}$$

- Using (4), the look-ahead point is $\boldsymbol{y}_t = \boldsymbol{x}_t + \mu_t \boldsymbol{v}_t$. Plug this into the gradient step (1):

$$\boldsymbol{x}_{t+1} = \boldsymbol{x}_t + \mu_t \boldsymbol{v}_t - \alpha \nabla f(\boldsymbol{x}_t + \mu_t \boldsymbol{v}_t). \tag{6}$$

- Now rewrite (6) in velocity form by subtracting $\boldsymbol{x}_t$ from both sides:

$$\boldsymbol{v}_{t+1} = \boldsymbol{x}_{t+1} - \boldsymbol{x}_t = \mu_t \boldsymbol{v}_t - \alpha \nabla f(\boldsymbol{x}_t + \mu_t \boldsymbol{v}_t), \quad \boldsymbol{x}_{t+1} = \boldsymbol{x}_t + \boldsymbol{v}_{t+1}. \tag{7}$$

- **Conclusion:** Equations (7) are exactly the Nesterov Accelerated Gradient (NAG) updates, where $\mu_t = \frac{\beta_{t-1}-1}{\beta_t}$ provides the momentum parameter.

Projected NAG example

Figure: Second, third and fifth coordinates of $\boldsymbol{r}$ are set to 0, the other two are perturbed.

- Define the selection matrix

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \qquad A^\top = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{16}$$

- Perform a QR decomposition of $A^\top$: find orthogonal H and upper–triangular R such that

$$H^\top H = I, \qquad H A^\top = \begin{bmatrix} R \\ 0 \end{bmatrix}. \tag{17}$$

Projected NAG example (cont.)

- Since the columns of $A^\top$ are already orthonormal up to permutations/signs, one valid choice is obtained by permuting rows; H is not unique.
- Split $H = [\, Y \; Z \,]^\top$ so that the columns of Y span range(A) and the columns of Z span its orthogonal complement. A concrete valid choice is

$$Y = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad Z = \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \end{bmatrix}. \tag{18}$$

- Hence any $\boldsymbol{r} \in \mathbb{R}^5$ can be written as

$$\boldsymbol{r} = Y \boldsymbol{r}_y + Z \boldsymbol{r}_z, \qquad \boldsymbol{r}_y \in \mathbb{R}^3, \; \boldsymbol{r}_z \in \mathbb{R}^2. \tag{19}$$

Coordinates, permutation, and reduced problem

- If

$$\boldsymbol{r} = \begin{bmatrix} a \\ b \\ c \\ d \\ e \end{bmatrix}, \qquad \text{then} \quad \boldsymbol{r}_y = \begin{bmatrix} b \\ c \\ e \end{bmatrix}, \quad \boldsymbol{r}_z = \begin{bmatrix} a \\ d \end{bmatrix}. \tag{20}$$

- Indeed,

$$Y\,\boldsymbol{r}_y + Z\,\boldsymbol{r}_z = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} b \\ c \\ e \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} d \\ a \end{bmatrix} = \begin{bmatrix} a \\ b \\ c \\ d \\ e \end{bmatrix} = \boldsymbol{r}. \tag{21}$$

Reduced problem

- Stacking $(\boldsymbol{r}_y, \boldsymbol{r}_z)$ and applying $H^\top = [\, Y \; Z \,]$ gives a fixed permutation of the entries of $\boldsymbol{r}$:

$$H^\top \begin{bmatrix} \boldsymbol{r}_y \\ \boldsymbol{r}_z \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} b \\ c \\ e \\ a \\ d \end{bmatrix} = \begin{bmatrix} a \\ b \\ c \\ d \\ e \end{bmatrix}. \tag{22}$$

- **New (reduced) problem:** with Z as above,

$$\min_{\boldsymbol{z} \in \mathbb{R}^2} \; \mathcal{L}(f_\theta(\boldsymbol{x} + Z\boldsymbol{z}),\, t) \; + \; \mu \, \|Z\boldsymbol{z}\|_2. \tag{23}$$

Projection matrix P_V

- **Why** $ZZ^\top = P_V$, i.e., projection matrix onto ker A?

$$ZZ^\top = \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{24}$$

- The matrix product that reorders coordinates equals P_V, and applying it to a vector (e.g., a gradient) yields

$$P_V \nabla f(\boldsymbol{r}_t) \tag{25}$$

by the definition of P_V (cf. Eq. (2.14)), which zeros entries outside V.

GSE results on targeted adversarial attacks

Table: Targeted attacks performed on ResNet20 classifier for CIFAR-10, and ResNet50 and ViT_B_16 classifiers for ImageNet. Tested on 1k images from each dataset, 9 target labels for CIFAR-10 and 10 target labels for ImageNet.

		Best case					Average case					Worst case				
	Attack	**ASR**	**ACP**	**ANC**	ℓ_2	$d_{2,0}$	**ASR**	**ACP**	**ANC**	ℓ_2	$d_{2,0}$	**ASR**	**ACP**	**ANC**	ℓ_2	$d_{2,0}$
CIFAR-10 ResNet20	GSE (Ours)	**100%**	**29.6**	**1.06**	**0.68**	**137**	**100%**	**86.3**	**1.76**	**1.13**	**262**	**100%**	**162**	**3.31**	**1.57**	**399**
	StrAttack	**100%**	78.4	4.56	0.79	352	**100%**	231	10.1	1.86	534	**100%**	406	15.9	4.72	619
	FWnucl	**100%**	283	1.18	1.48	515	85.8%	373	2.52	2.54	564	40.5%	495	4.27	3.36	609
ImageNet ResNet50	GSE (Ours)	**100%**	**3516**	5.89	2.16	**5967**	**100%**	**12014**	14.6	**2.93**	**16724**	**100%**	**21675**	**22.8**	**3.51**	**29538**
	StrAttack	**100%**	6579	7.18	2.45	9620	**100%**	15071	18.0	3.97	20921	**100%**	26908	32.1	6.13	34768
	FWnucl	31.1%	9897	**3.81**	**2.02**	11295	7.34%	19356	**7.58**	3.17	26591	0.0%	N/A	N/A	N/A	N/A
ImageNet ViT_B_16	GSE (Ours)	**100%**	**916**	**3.35**	2.20	**1782**	**100%**	**2667**	**7.72**	**2.87**	**4571**	**100%**	**5920**	**14.3**	**3.60**	**9228**
	StrAttack	**100%**	3550	7.85	**2.14**	5964	**100%**	8729	17.2	3.50	13349	**100%**	16047	27.4	5.68	22447
	FWnucl	53.2%	5483	4.13	2.77	6718	11.2%	6002	9.73	3.51	7427	0.0%	N/A	N/A	N/A	N/A

Quantitative evaluation

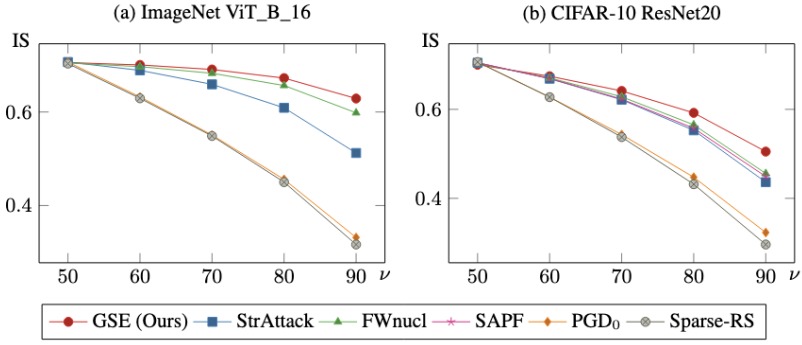


Figure: IS vs. percentile ν for targeted versions of GSE vs. five other attacks. Evaluated on an ImageNet ViT_B_16 classifier (a), and CIFAR-10 ResNet20 classifier (b). Tested on 1k images from each dataset, 9 target labels for CIFAR-10 and 10 target labels for ImageNet.

Constraining the Sparsity

- Regularize using the indicator function of the sparsity constraint

  $\hookrightarrow$ Improved control over sparsity

$$I_{\|\boldsymbol{x}'-\boldsymbol{x}\|_0 \leq m}(\boldsymbol{x}') := \begin{cases} 0, & \text{if } \|\boldsymbol{x}'-\boldsymbol{x}\|_0 \leq m \\ +\infty, & \text{otherwise.} \end{cases}$$

- New $g(\boldsymbol{x}') := I_{\mathcal{A}}(\boldsymbol{x}') + \beta I_{\|\boldsymbol{x}'-\boldsymbol{x}\|_0 \leq m}(\boldsymbol{x}')$ is an indicator function

  $\hookrightarrow$ Proximal operator coincides with the projection onto the intersection

$$\{\|\boldsymbol{x}'-\boldsymbol{x}\|_0 \leq m\} \cap \mathcal{A}.$$

Proximal operator of an indicator function

- For any indicator function $I_S(\boldsymbol{y}) = \begin{cases} 0, & \text{if } \boldsymbol{y} \in S, \\ +\infty, & \text{if } \boldsymbol{y} \notin S. \end{cases}$, its proximal operator is the projection onto the set S:

$$\text{prox}_{I_S}(\boldsymbol{x}) = \arg\min_{\boldsymbol{y}} \left\{ \tfrac{1}{2}\|\boldsymbol{y} - \boldsymbol{x}\|^2 + I_S(\boldsymbol{y}) \right\} = \arg\min_{\boldsymbol{y} \in S} \tfrac{1}{2}\|\boldsymbol{y} - \boldsymbol{x}\|^2 = P_S(\boldsymbol{x}).$$

- Therefore, when $g_p(\boldsymbol{y})$ is a sum of indicator functions, its proximal operator is the projection onto the intersection of the sets defining those indicators (provided that the intersection is nonempty).

Learning from adversarial perturbations

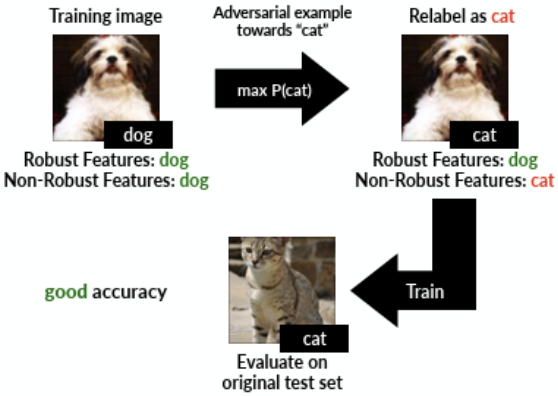


Figure: Training on a dataset which appears mislabeled to humans (via adversarial examples) results in good accuracy on the original test set (Ilyas et al. 2019).