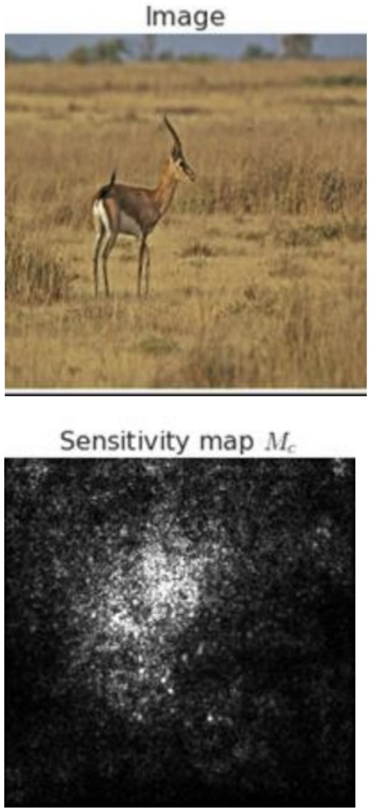
Harvard John A. Paulson
School of Engineering
and Applied Sciences

Do Input Gradients Highlight Discriminative Features?

Authors: Harshay Shah, Prateek Jain, Praneeth Netrapalli
Presenters: Karly Hou, Eshika Saxena, Kat Zhang

Introduction

- Instance-specific explanations of model predictions
- Input coordinates are ranked in decreasing order of input gradient magnitude
- **Assumption (A)**: Larger input gradient magnitude = higher contribution to prediction

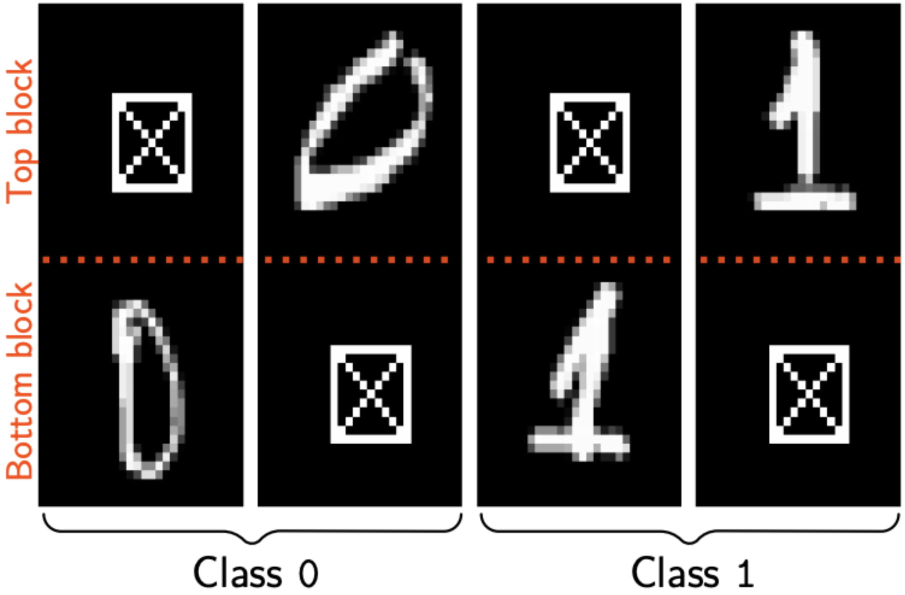


Key Contributions

- New evaluation framework **DiffROAR** for evaluating **Assumption (A)** on real datasets
  - compares the top-ranked and bottom-ranked features
- Evaluate input gradient attributions for MLPs and CNNs
- Found that standard models violate Assumption (A)
- Some adversarially trained models might satisfy it

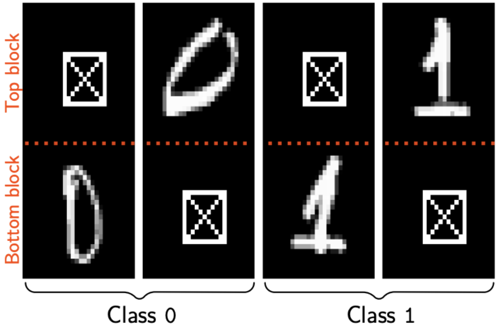
Key Contributions: BlockMNIST Dataset

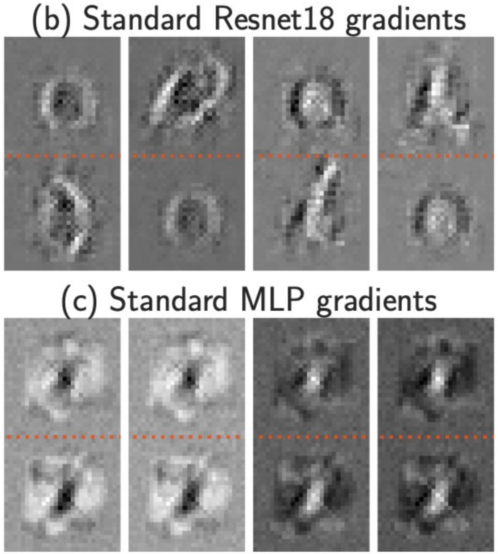
- Each datapoint contains two images, a signal (the actual digit), and a null block (a blank square)
- Want to see if the classification is actually occurring as a result of the block with the information



Key Contributions: BlockMNIST Dataset

- Feature leakage is a potential reason why Assumption (A) is violated
- Feature leakage is shown empirically and proved theoretically




(b) Standard Resnet18 gradients

(c) Standard MLP gradients

Related Work

- Sanity checks for explanations
  - Visual assessments are unreliable
  - Other well-known gradient-based attribution methods fare worse in these sanity checks

- Evaluating explanation fidelity
  - Evaluations traditionally performed without knowing the ground-truth explanations
  - Usually evaluated using masking or ROAR
  - Input gradients lack explanatory power and are as good as random attributions

Related Work

- Adversarial Robustness
  - Adversarial training can improve the visual quality of input gradients
  - Adversarial model gradients are more stable than standard ones
  - Are they also more faithful?

DiffROAR Evaluation Framework

Setting

- Standard classification setting with each independently drawn data point as a pair of (instance, labe $(x^{(i)}, y^{(i)})$
- $x^{(i)}_j$ denotes jth coordinate/feature of $x^{(i)}$
- **Feature attribution scheme** $\mathcal{A} : \mathbb{R}^d \to \{\sigma : \sigma \text{ is a permutation of } [d]\}$ maps a d-dimensional instance x to a permutation of its features
- e.g. **Input gradient attribution scheme** takes input instance x, predicted label y, outputs ordering [d] that ranks features in decreasing order of input gradient magni $\mathcal{A}(x) : [d] \to [d]$

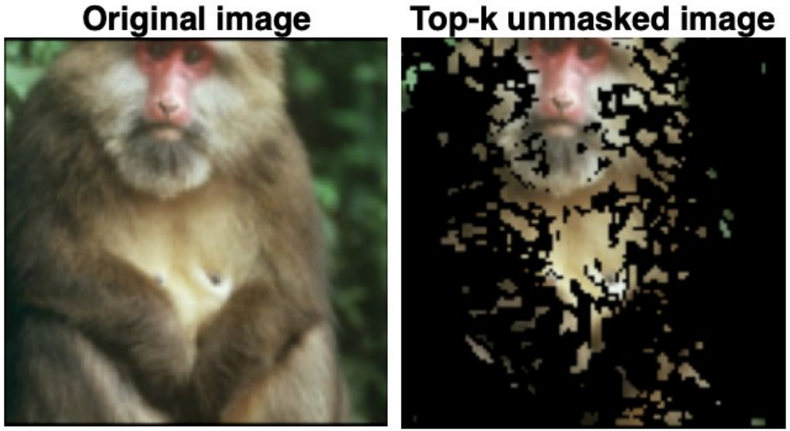
Unmasking Schemes

- **Unmasked instance** x^S zeroes out all coordinates not in subset S
- **Unmasking scheme** maps instance x to subset A(x) of coordinates to obtain unmasked instance $x^{A(x)}$

$$A : \mathbb{R}^d \to \{S : S \subseteq [d]\}$$

- Tc ng scheme

$$\mathcal{A}_k^{\text{top}}(x) := \{\mathcal{A}(x)_j : j \leq k\},$$
$$\mathcal{A}_k^{\text{bot}}(x) := \{\mathcal{A}(x)_j : d-k < j \leq d\}.$$



Predictive Power of Unmasking Schemes

- **Predictive power**: best classification accuracy that can be attained by training a model with architecture **M** on unmasked instances that are obtained via unmasking scheme **A**

$$\texttt{PredPower}_M(A) := \sup_{f \in M, f:\mathbb{R}^d \to \mathcal{Y}} \mathbb{E}_{\mathcal{D}}\left[\mathbb{1}\left[f(x^{A(x)}) = y\right]\right].$$

- Estimate predictive power in two steps:
  (1) Use unmasking scheme A to obtain unmasked train and test datasets that comprise data points of the form ($x^{A(x)}$, y)
  (2) Retrain a new model with the same architecture M on unmasked train data and evaluate its accuracy on unmasked test data

DiffROAR Metric

$$\mathtt{DiffROAR}_M(\mathcal{A}, k) = \mathtt{PredPower}_M(\mathcal{A}_k^{\text{top}}) - \mathtt{PredPower}_M(\mathcal{A}_k^{\text{bot}})$$

Interpretation of the metric:

- Sign of the metric indicates whether the assumption is satisfied (> 0) or violated (< 0)
- Magnitude of the metric quantifies extent of separation of most and least discriminative coordinates into two disjoint subsets

Experiment 1: Image Classification Benchmarks

Setup: Datasets and Models

- **Datasets**: SVHN, Fashion MNIST, CIFAR-10, ImageNet-10
- **Models**: Standard and adversarially trained two-hidden-layer MLPs and Resnets
  - Adversarial: l_2 and l_∞ epsilon-robust models with perturbation budget epsilon using PGD adversarial training

Procedure: Computing DiffROAR Metric

As a function of unmasking fraction k, compare input gradient attributions of models to baselines of model-agnostic and input-agnostic attributions:

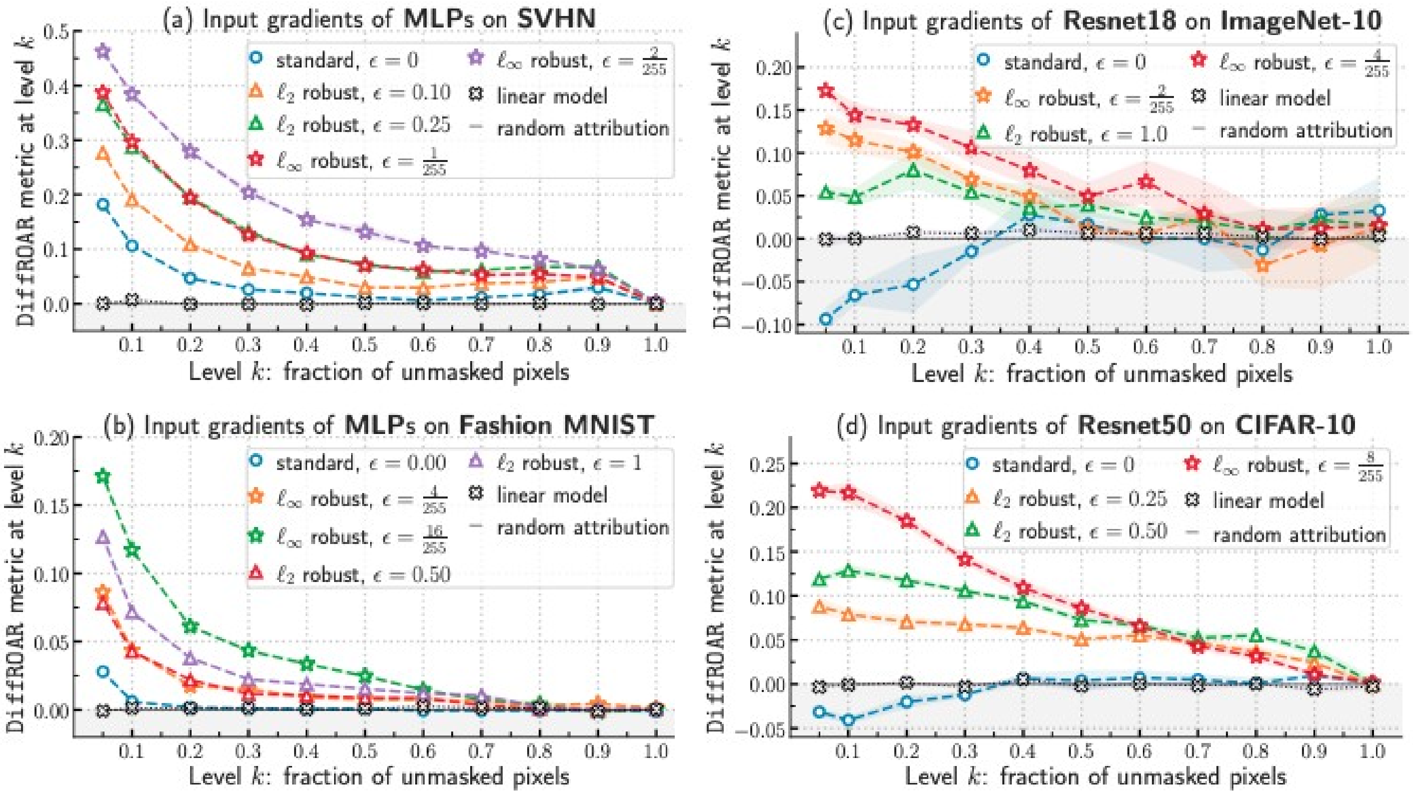
1. Train a standard or robust model with architecture M on the original dataset and obtain its input gradient attribution scheme A.

2. Use attribution scheme A and level k (i.e., fraction of pixels to be unmasked) to extract the top-k and bottom-k unmasking schemes: A top k and A bot k

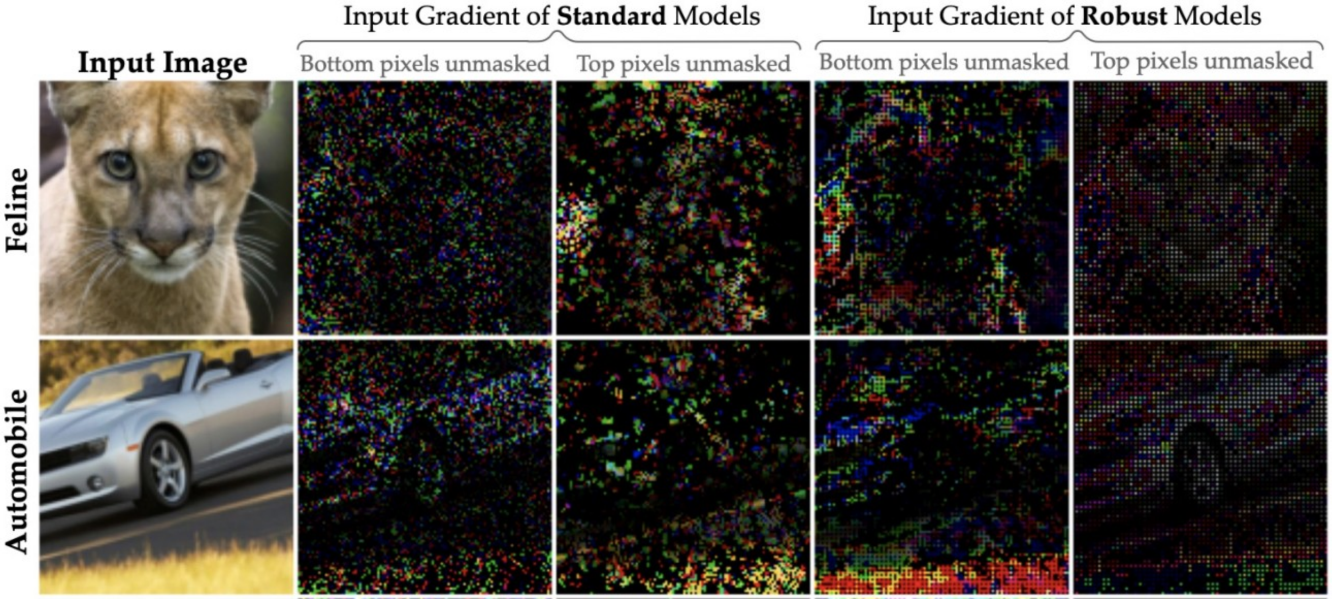
Procedure: Computing DiffROAR Metric

3. Apply A top k and A bot k on the original train & test datasets to obtain top-k and bottom-k unmasked datasets respectively (unmask individual image pixels without grouping them channel-wise)
4. Estimate top-k and bottom-k predictive power by retraining new models with architecture M on top-k and bottom-k unmasked datasets respectively and compute the DiffROAR metric.
5. Average the DiffROAR metric over five runs for each model and unmasking fraction or level k

Results


(a) Input gradients of MLPs on SVHN
DiffROAR metric at level k
Level k: fraction of unmasked pixels
standard, $\epsilon = 0$
ℓ_2 robust, $\epsilon = 0.10$
ℓ_2 robust, $\epsilon = 0.25$
ℓ_∞ robust, $\epsilon = \frac{1}{255}$
ℓ_∞ robust, $\epsilon = \frac{2}{255}$
linear model
random attribution
0.0 0.1 0.2 0.3 0.4 0.5
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
(b) Input gradients of MLPs on Fashion MNIST
DiffROAR metric at level k
Level k: fraction of unmasked pixels
standard, $\epsilon = 0.00$
ℓ_∞ robust, $\epsilon = \frac{4}{255}$
ℓ_∞ robust, $\epsilon = \frac{16}{255}$
ℓ_2 robust, $\epsilon = 0.50$
ℓ_2 robust, $\epsilon = 1$
linear model
random attribution
0.00 0.05 0.10 0.15 0.20
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
(c) Input gradients of Resnet18 on ImageNet-10
DiffROAR metric at level k
Level k: fraction of unmasked pixels
standard, $\epsilon = 0$
ℓ_∞ robust, $\epsilon = \frac{2}{255}$
ℓ_2 robust, $\epsilon = 1.0$
ℓ_∞ robust, $\epsilon = \frac{4}{255}$
linear model
random attribution
−0.10 −0.05 0.00 0.05 0.10 0.15 0.20
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
(d) Input gradients of Resnet50 on CIFAR-10
DiffROAR metric at level k
Level k: fraction of unmasked pixels
standard, $\epsilon = 0$
ℓ_2 robust, $\epsilon = 0.25$
ℓ_2 robust, $\epsilon = 0.50$
ℓ_∞ robust, $\epsilon = \frac{8}{255}$
linear model
random attribution
−0.05 0.00 0.05 0.10 0.15 0.20 0.25
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0

Results - Qualitative Analysis



Experiment 2: BlockMNIST Data

Analyzing input gradient attributions: BlockMNIST data

Dataset design based on intuitive properties of classification tasks:

- Object of interest may appear in **different parts** of an image
- Object of interest and the rest of the image often share low-level patterns like edges that are **not informative** of the label on their own

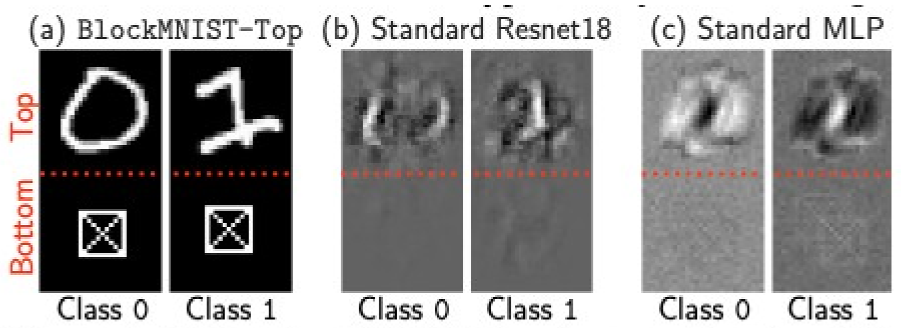
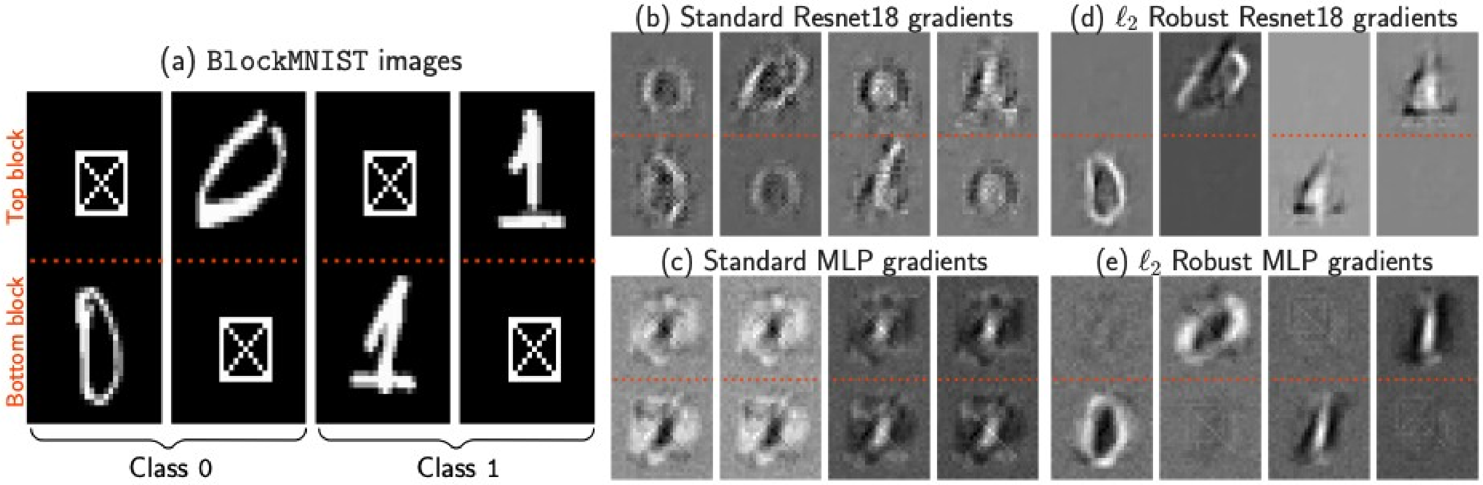


Figure 4: (a) In BlockMNIST-Top images, the signal & null blocks are fixed at the top & bottom respectively. In contrast to results on BlockMNIST in fig. 1, input gradients of standard (b) Resnet18 and (c) MLP trained on BlockMNIST-Top highlight discriminative features in the signal block, suppress the null block, and satisfy (A).

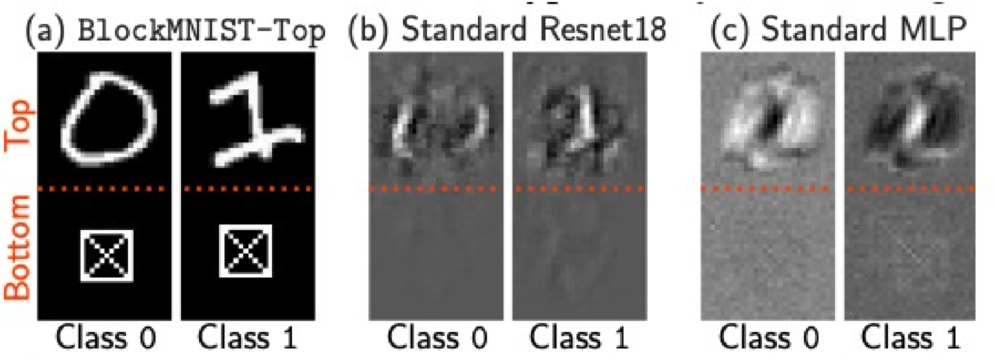
Standard + adversarially robust models trained on BlockMNIST data attain 99.99% test accuracy.

Do gradient attributions highlight signal block over null block? Not always!


(a) BlockMNIST images
Top block
Bottom block
Class 0
Class 1
(b) Standard Resnet18 gradients
(c) Standard MLP gradients
(d) ℓ_2 Robust Resnet18 gradients
(e) ℓ_2 Robust MLP gradients

Feature leakage hypothesis

- When discriminative features vary across instances (e.g. signal block at top vs bottom), input gradients of standard models may not only highlight instance-specific features but also **leak discriminative features** from other instances
- Altered datasets: BlockMNIST-Top → now gradients of standard Resnet18, MLP highlight discriminative features in signal block and suppress null block



Proof 3: Feature Leakage

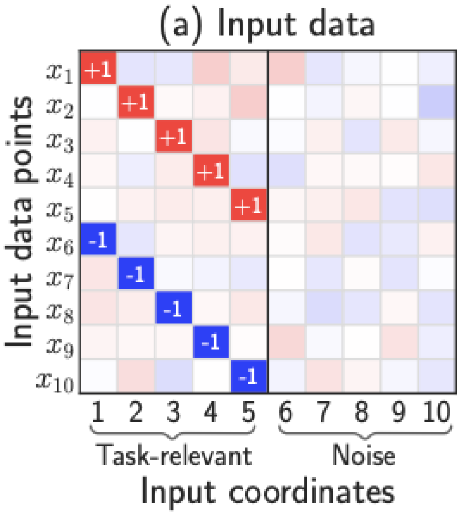
Dataset

- Simplified version of BlockMNIST
- Draw sample $(x, y) \in \mathbb{R}^{d \cdot d} \times \{\pm 1\}$
  - eta = noise parameter
  - g_i drawn uniformly at random from the unit ball
  - Let d even so that d/2 is an integer
- We can think of each x as a concatenation of blocks {x_1...x_d}
  - The first d/2 blocks are task-relevant: each example contains an instance-specific signal block that is informative of its label
  - The remaining blocks are noise blocks that don’t contain task-relevant signal
- At a high level, these correspond to the discriminative MNIST digit and null square patch in BlockMNIST

:

$y = \pm 1$ with probability 0.5 and

$x = [\eta g_1, \quad \eta g_2, \quad \ldots, \quad y u^* + \eta g_j, \quad \ldots, \quad \eta g_d]$ with j chosen at random from $[{}^{d}/_{2}]$



Model

- Consider one-hidden layer MLPs with ReLU nonlinearity
- Given an input instance, the output logit f and cross-entropy loss L are (for given layer width m)

$$f((w,R,b),x) := \langle w, \phi(Rx+b) \rangle, \quad \mathcal{L}((w,R,b),(x,y)) := \log\left(1+\exp\left(-y \cdot f((w,R,b),x)\right)\right)$$

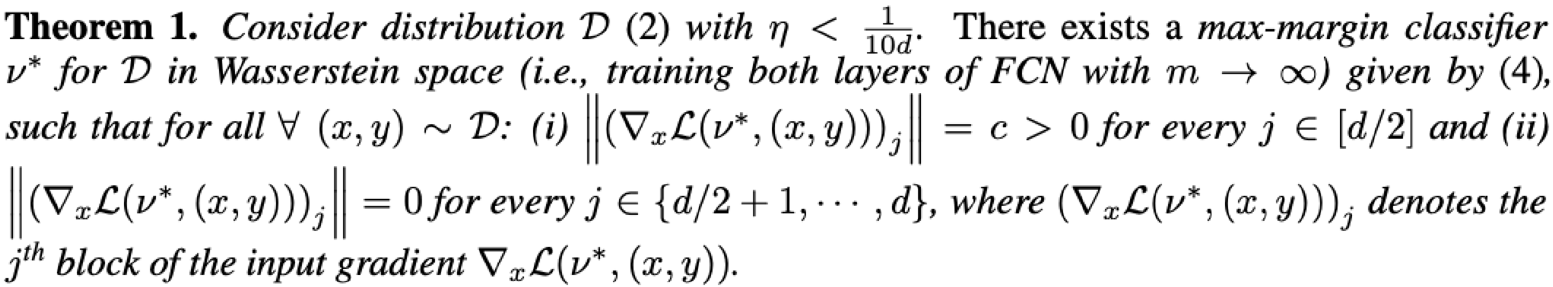
- As m → infinity, the training procedure equivalent to gradient descent on infinite-dimensional Wasserstein space
- Wasserstein space: network interpreted as probability distribution nu with output score, CE loss:

$$f(\nu,x) := \mathbb{E}_{(w,r,b)\sim\nu}\left[w\phi\left(\langle r,x\rangle + b\right)\right], \quad \mathcal{L}(\nu,(x,y)) := \log\left(1+\exp\left(-y \cdot f(\nu,x)\right)\right)$$

Theoretical analysis

- Prior research shows that if gradient descent in the Wasserstein space $\mathcal{W}^2\left(\mathbb{R} \times \mathbb{R}^{\tilde{d}d} \times \mathbb{R}\right)$ on $\mathbb{E}_{\mathcal{D}}\left[\mathcal{L}(\nu, (x, y))\right]$ converges, it does to a max-margin classifier: $$\nu^* := \underset{\nu \in \mathcal{P}\left(\mathbb{S}^{d\tilde{d}+1}\right)}{\arg\max} \; \min_{(x,y)\sim\mathcal{D}} y \cdot f(\nu, x),$$
- S = surface of Euclidean unit ball
- P(S) = space of probability distributions
- Intuitively, our results show that on any data point (x, y) ~ D, the input gradient magnitude of the max-margin classifier v* is equal over all task-relevant blocks and zero on noise blocks

Theorem



Theorem 1. *Consider distribution $\mathcal{D}$ (2) with $\eta < \frac{1}{10d}$.* There exists a *max-margin classifier ν^* for $\mathcal{D}$ in Wasserstein space (i.e., training both layers of FCN with $m \to \infty$) given by* (4), *such that for all $\forall$ $(x, y) \sim \mathcal{D}$: (i) $\left\|\left(\nabla_x \mathcal{L}(\nu^*, (x, y))\right)_j\right\| = c > 0$ for every $j \in [d/2]$ and (ii) $\left\|\left(\nabla_x \mathcal{L}(\nu^*, (x, y))\right)_j\right\| = 0$ for every $j \in \{d/2+1, \cdots, d\}$, where $\left(\nabla_x \mathcal{L}(\nu^*, (x, y))\right)_j$ denotes the j^{th} block of the input gradient $\nabla_x \mathcal{L}(\nu^*, (x, y))$.*

- Guarantees **existence of max-margin classifier** such that the input gradient magnitude for any given instance is a nonzero constant on the task-relevant blocks, and zero on noise blocks
- However, input gradients fail at highlighting the **unique instance-specific signal block** over the task-relevant blocks
- **Feature leakage:** input gradients highlight task-relevant features that are not specific to the given instance

Empirical results

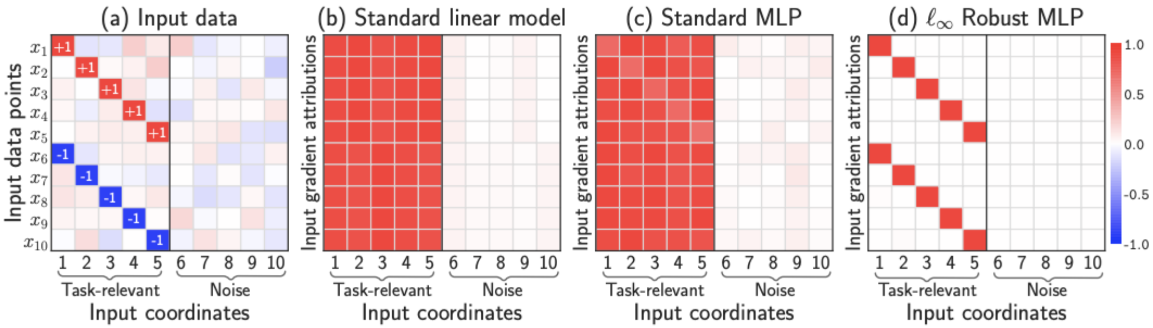


Figure 5: Input gradients of linear models and standard & robust MLPs trained on data from eq. (2) with $d = 10, \widetilde{d} = 1, \eta = 0$ and $u^* = 1$. (a) Each row in corresponds to an instance x, and the highlighted coordinate denotes the signal block $j^*(x)$ & label y. (b) Linear models suppress noise coordinates but lack the expressive power to highlight instance-specific signal $j^*(x)$, as their input gradients in subplot (b) are identical across all examples. (c) Despite the expressive power to highlight instance-specific signal coordinate $j^*(x)$, input gradients of standard MLPs exhibit feature leakage (see Theorem 1) and violate (A) as well. (d) In stark contrast, input gradients of adversarially trained MLPs suppress feature leakage and starkly highlight instance-specific signal coordinates $j^*(x)$.

- One-hidden-layer ReLU MLPs with width 10000
- All models obtain **100% test accuracy**
- Due to insufficient expressive power, linear models have input-agnostic gradients that suppress noise but **do not differentiate instance-specific signal coordinates**

Conclusion & Discussion

Discussion and Limitations

- Assuming that the model trained on the unmasked dataset learns the same features as the model trained on the original dataset
- When all features are equally informative, ROAR/DiffROAR can't be used
- This work only focuses on "vanilla" input gradients
- Why does adversarial training actually mitigate feature leakage?

Conclusion

Assumption (A): Larger input gradient magnitude = higher contribution to prediction

- **Assumption (A)** is not necessarily true for standard models
- Adversarially robust models satisfy **Assumption (A)** consistently
- Feature leakage is the reason why **Assumption (A)** does not hold

Discussion Questions

- After seeing that the fundamental assumption regarding the correspondence between input gradients and feature importance may not hold, **do we still find post-hoc explanations for individual or groups of data points convincing?**
- What are other major assumptions in explainability that we've discussed that you think could benefit from a sanity check with an **approach similar to this paper?**
- Do you have any additional **doubts** about any of the approaches taken in this paper?