Deep Learning: Advanced topics

Interpretability & Robustness

Speaker: Ma Jin

ma_jin@pku.edu.cn

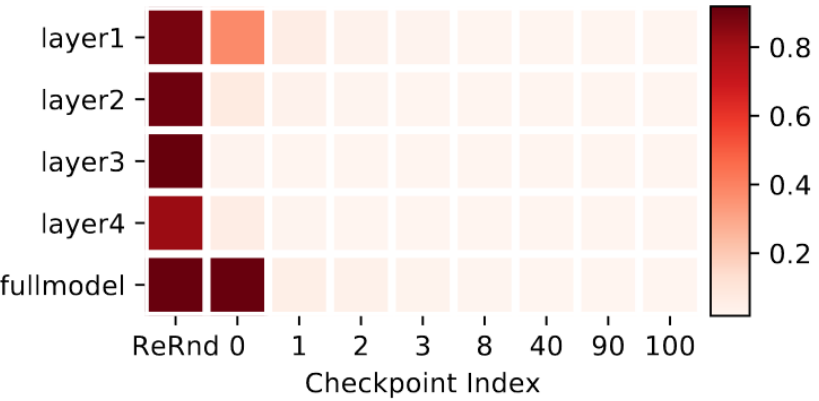
Outline

- The layers' robustness in the neural network
  - Layers are heterogeneous
- Robust interpretability
  - Connect neural networks to linear models
- How to explain adversarial examples
  - Establish bridge between interpretability and adversarial attacks
  - Adversarial example detection

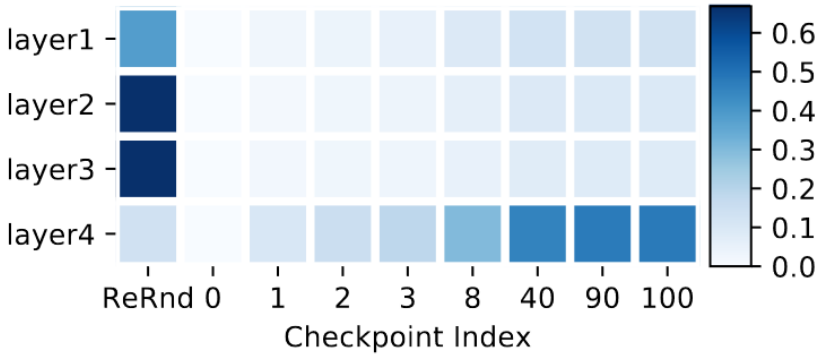
Are All Layers Created Equal?

- Tools
  - Re-initialization robustness
  - Re-randomization robustness
- Re-initialization robustness:
  - For layer d, checkpoint τ, re-initialize the parameters: $\theta_d^T \leftarrow \theta_d^\tau$
- Re-randomization robustness
  - For layer d, re-sample random values $\tilde{\theta}_d \sim \mathcal{P}_d$
- no re-training or fine-tuning – just evaluated directly with mixed post-trained and re-initialized/re-randomized weights

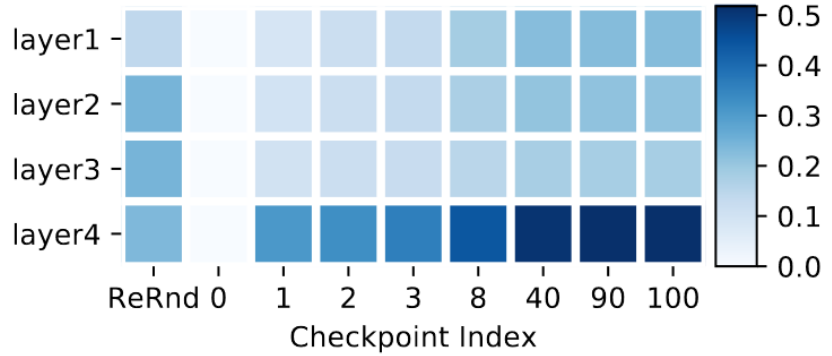
FCN 3×256



(a) Test error



(b) $\|\theta_d^\tau - \theta_d^0\|$



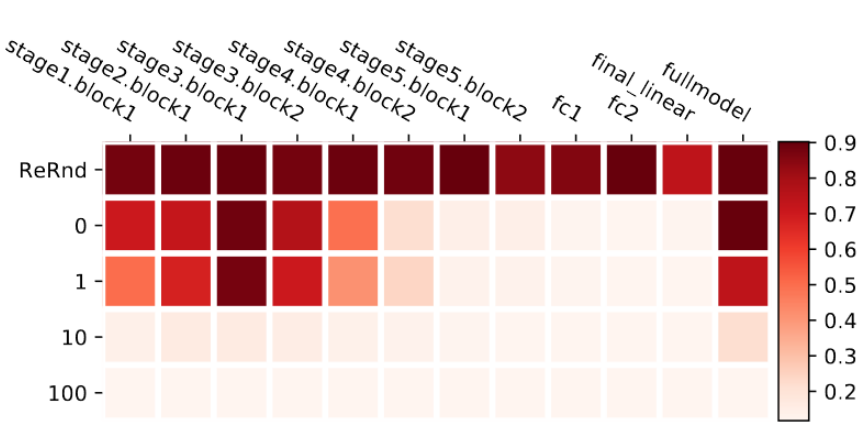
(c) $\|\theta_d^\tau - \theta_d^0\|_\infty$

- Re-randomizing any of the layers completely -> drops to the level of random guessing
- The robustness to re-initialization does not obviously correlate with either of the distances

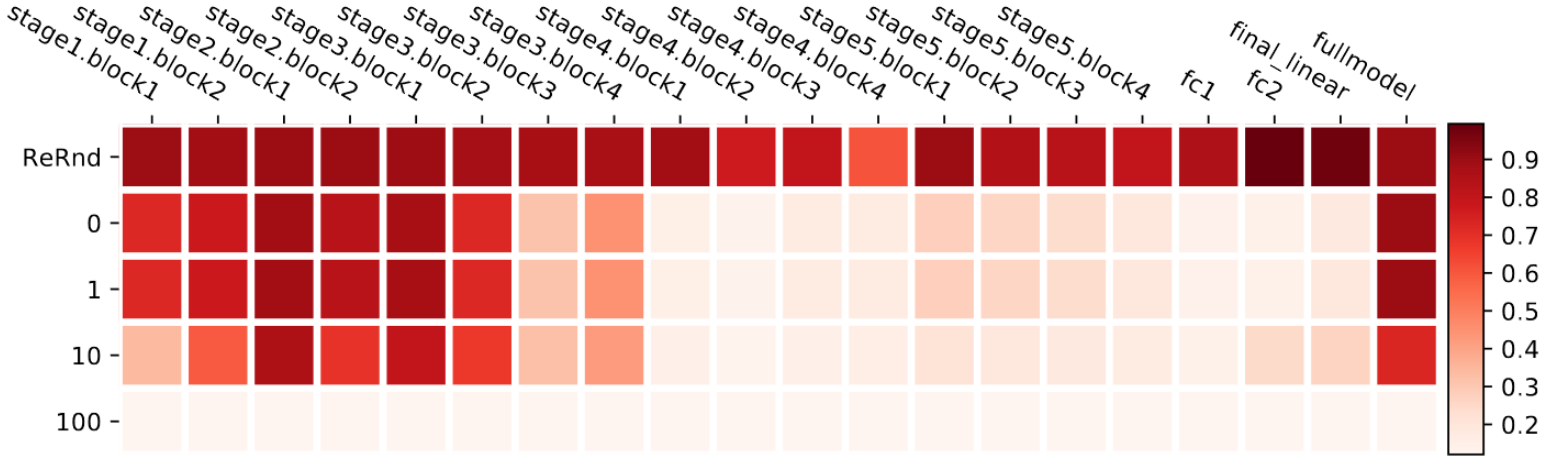
Over-capacitated deep networks trained with stochastic gradient have low-complexity due to self-restriction of the number of critical layers.

- In summary, the empirical results of this section provide some evidence that deep networks automatically adjust their de-facto capacity. When a big network is trained on an easy task, only a few layers seem to be playing critical roles

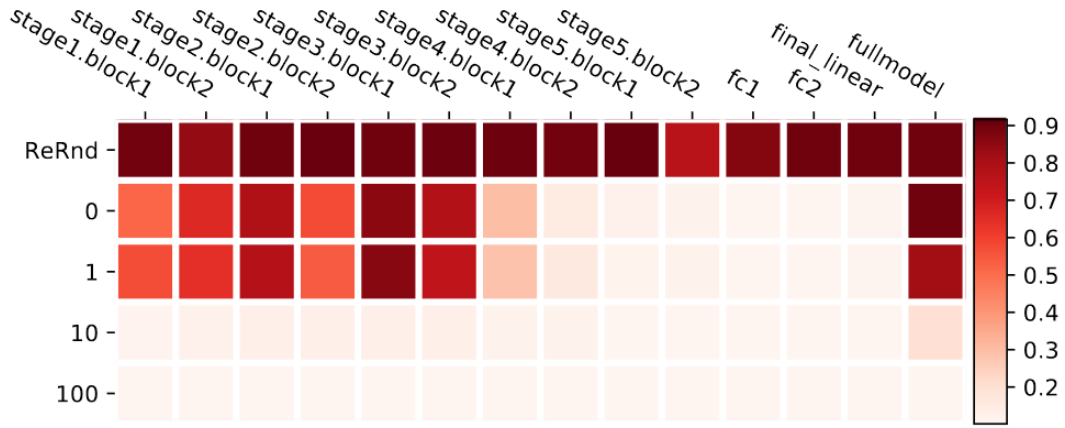
CNN: VGG



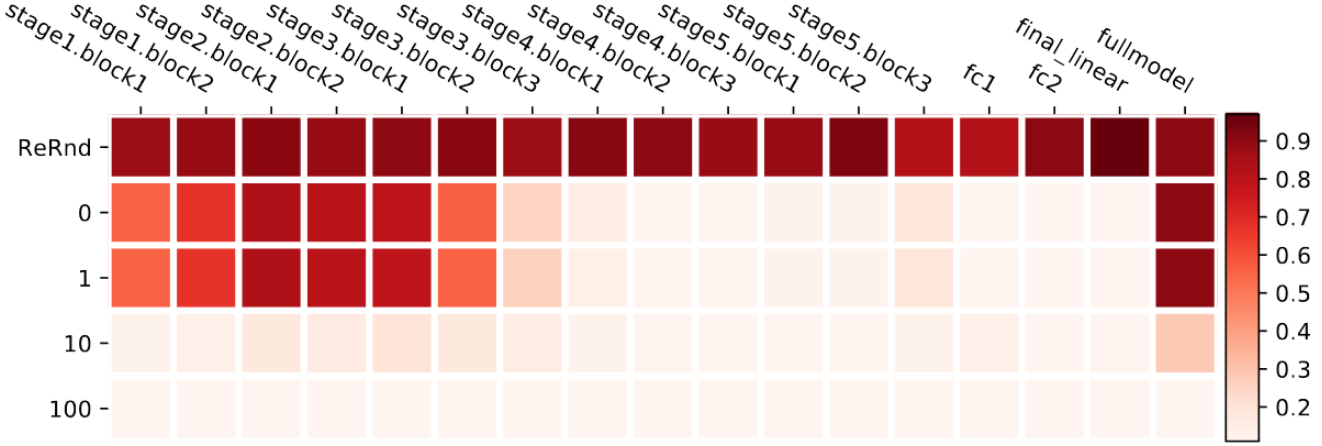
(a) VGG11



(b) VGG19



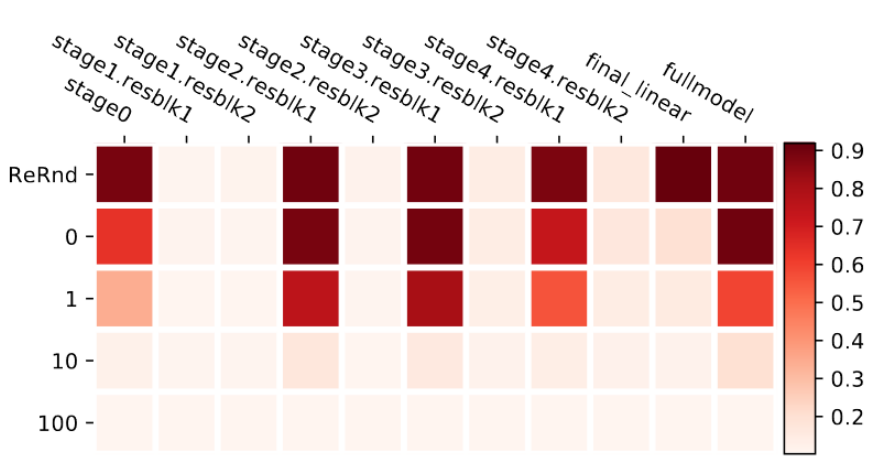
(c) VGG13



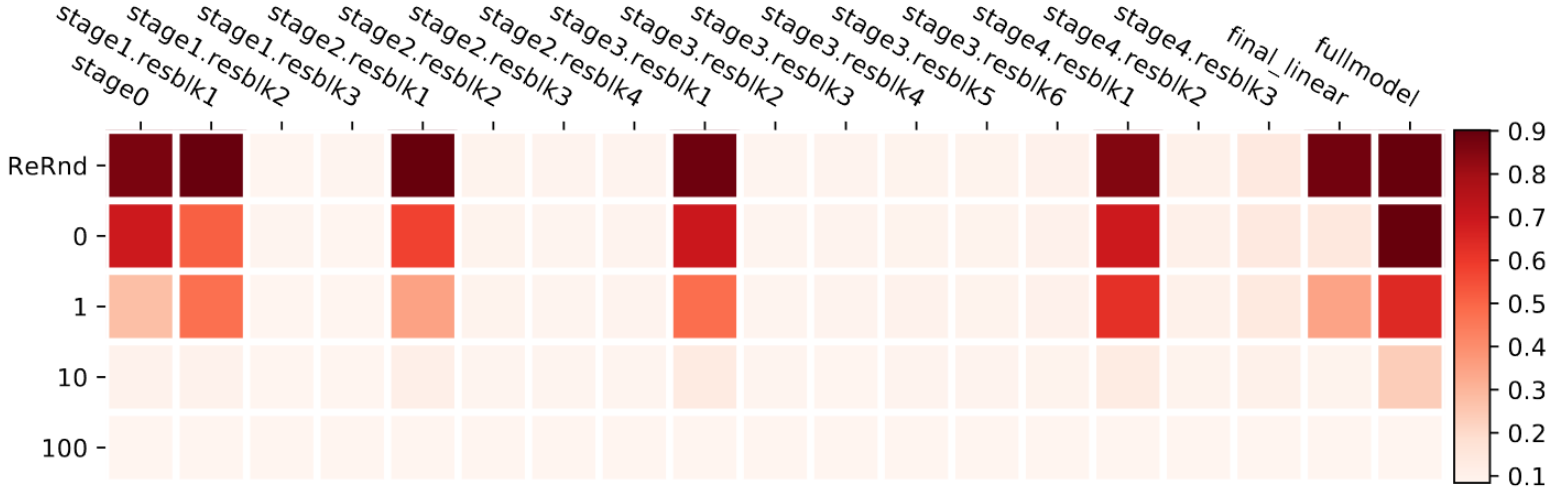
(d) VGG16

- More layers are sensitive to re-initialization,
- The bottom layers are also sensitive while the top layers are robust to re-initialization.

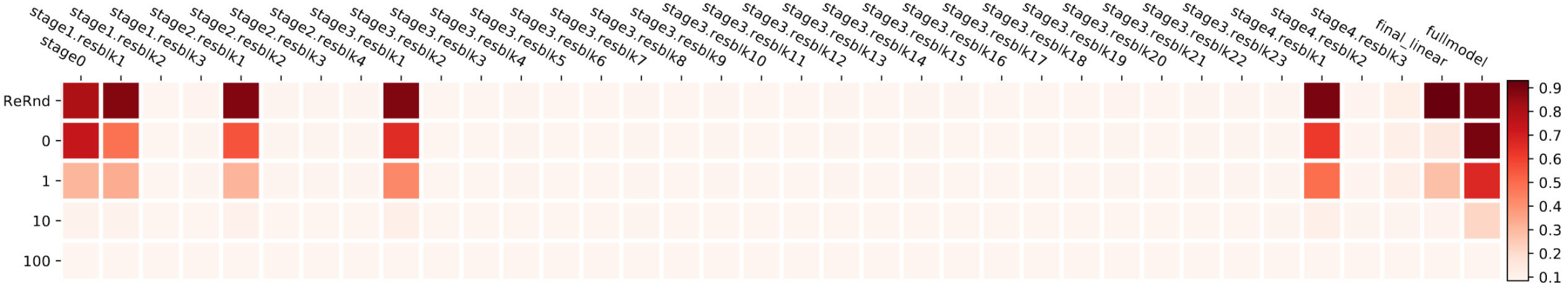
ResNets



(a) ResNet18



(b) ResNet50

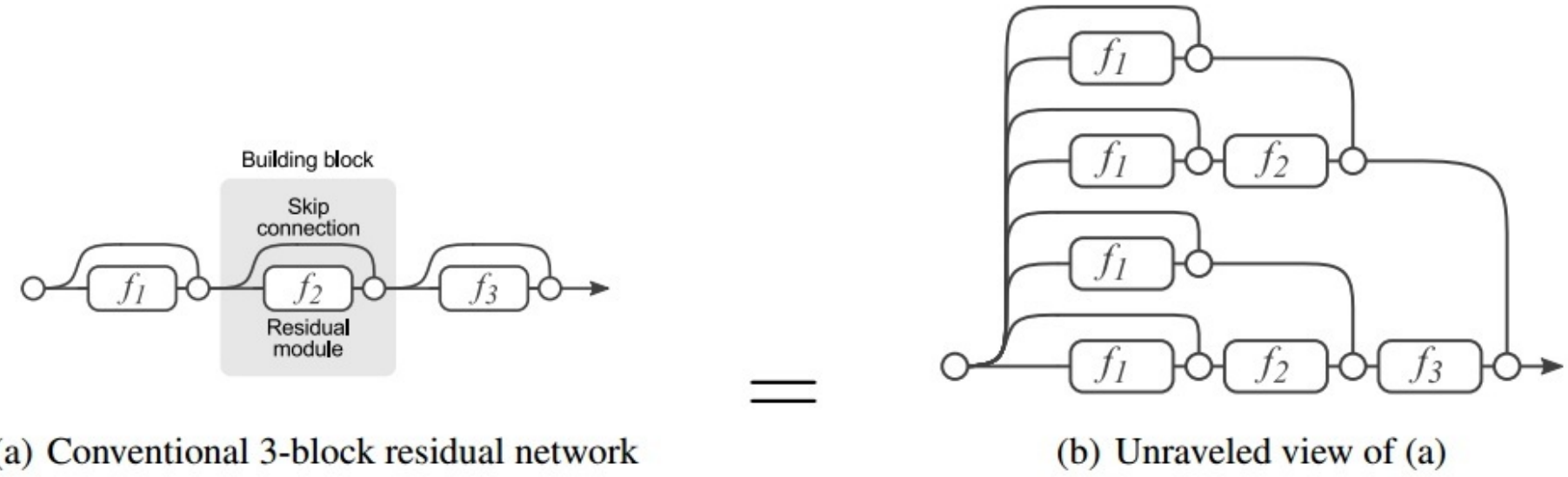


(c) ResNet101

More interesting

- ResNets re-distribute critical layers
  - Every stage’s first block: connected to the last block of the previous stage, has a non-identity skip connection due to different input-output shapes
- Residual blocks can be robust to re-randomization

Residual Networks Behave Like Ensembles of Relativelv Shallow Networks

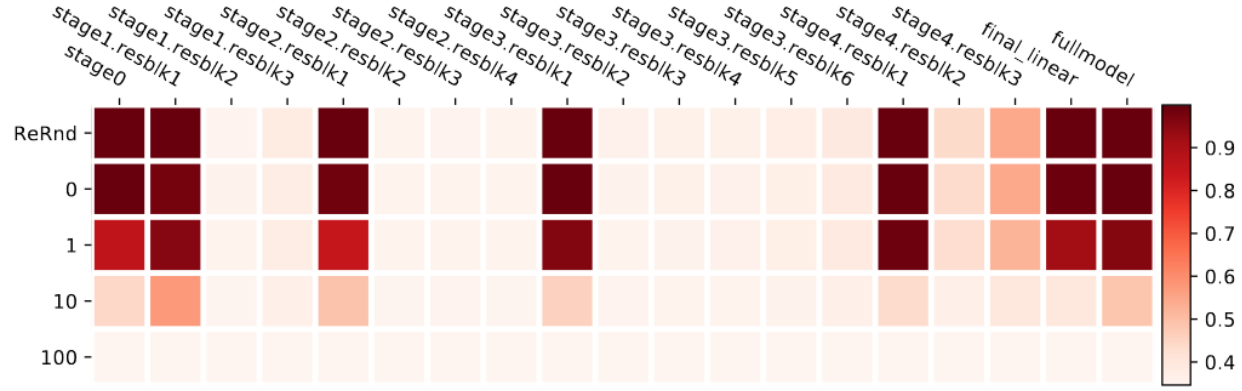


(a) Conventional 3-block residual network

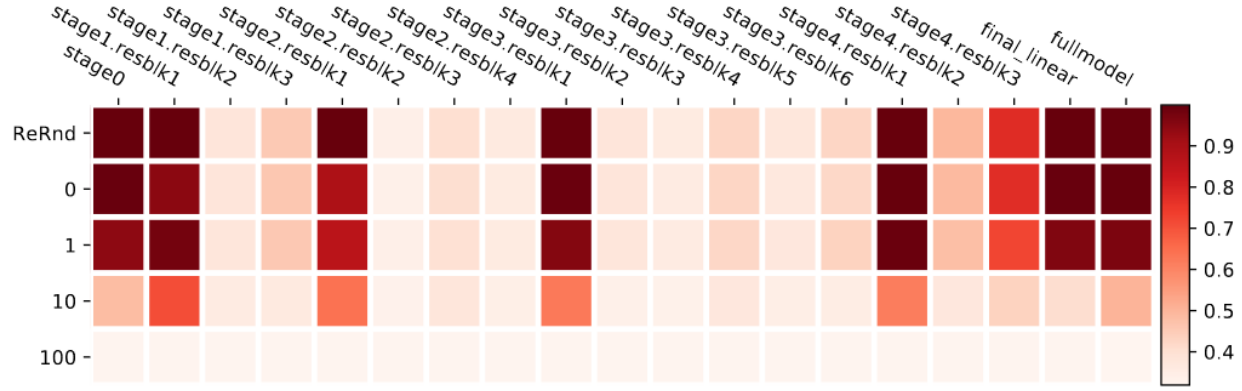
(b) Unraveled view of (a)

- Removing a single layer of ResNet does not significantly affect network performance (traditional networks -> random guess)

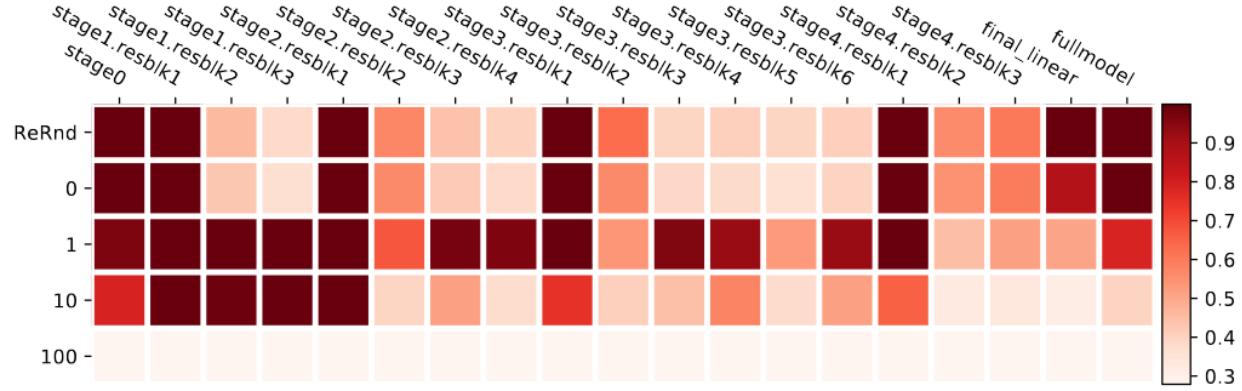
Weight decay & batch normalization



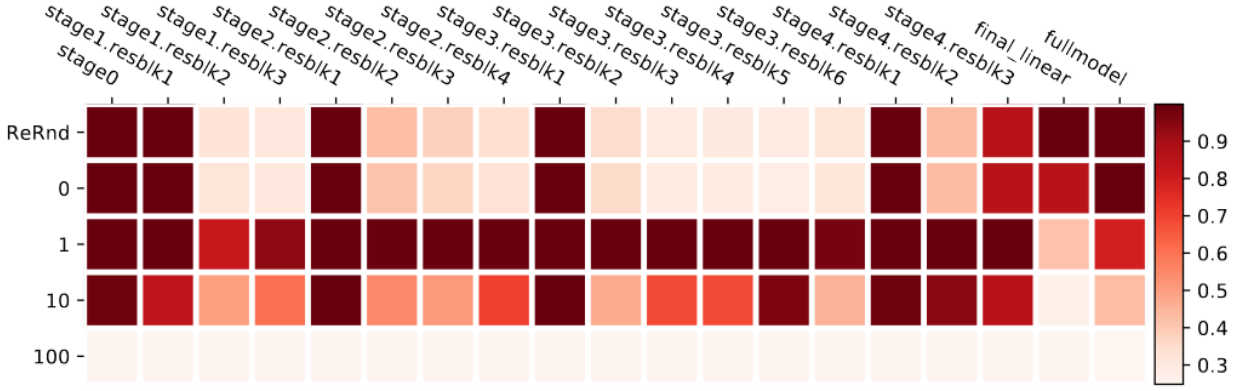
(a) ResNet50



(b) ResNet50 +wd



(c) ResNet50 +bn



(d) ResNet50 +wd +bn

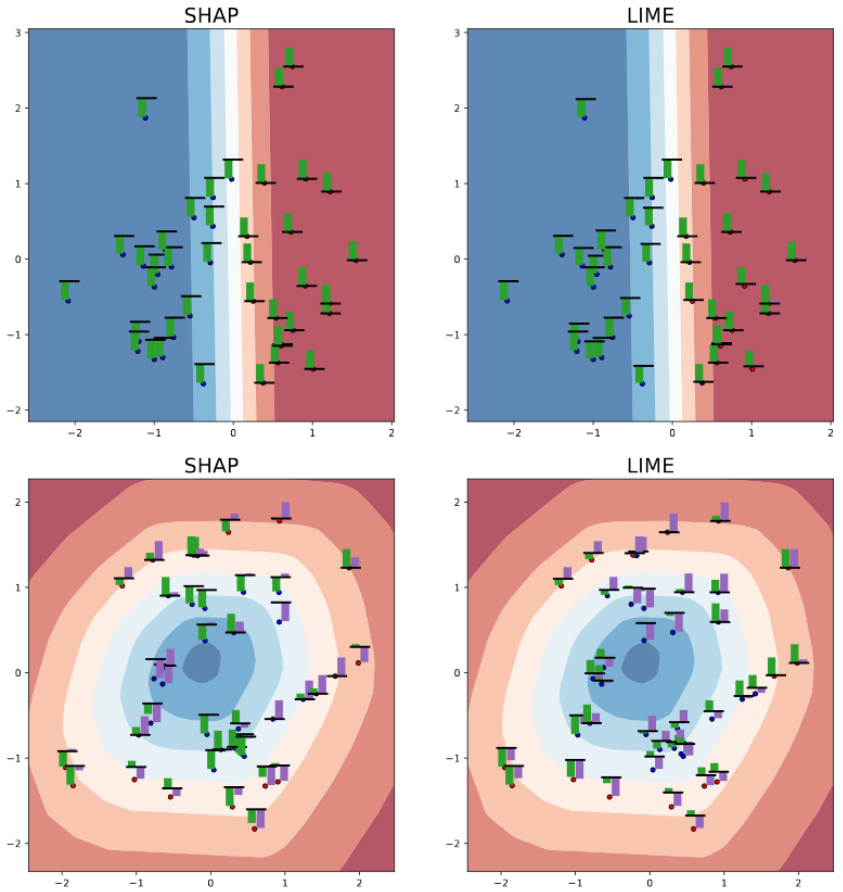
Conclusion

- heterogeneous characteristic of layers
  - Ambient: re-setting doesn’t hurt performance
  - Critical:
- Parameter counting or norm accounting is not enough !
- Implications on generalization
  - heavy overparameterization
  - generalization bounds are vacuous
- Key: partially random subnetworks within one large network may be strong.

On the Robustness of Interpretability Methods

- Robustness of explanation - similar inputs should give rise to similar explanations

- $f(x)$: attribution arrays



Robustness quantity

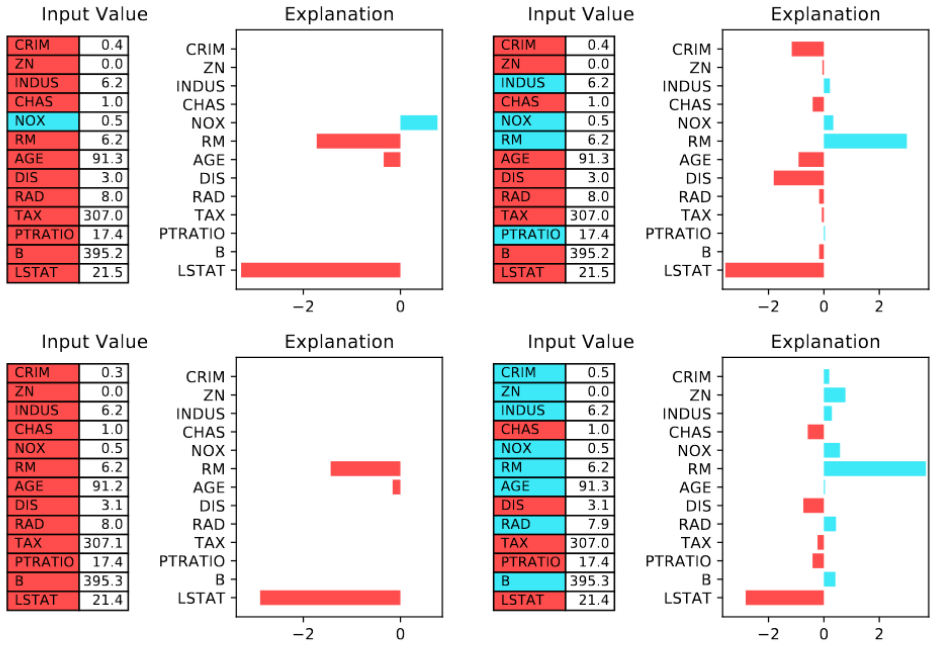
- Locally Lipschitz:

Definition 2.1. *$f : \mathcal{X} \subseteq \mathbb{R}^n \to \mathbb{R}^m$ is **locally Lipschitz** if for every x_0 there exist $\delta > 0$ and $L \in \mathbb{R}$ such that $\|x - x_0\| < \delta$ implies $\|f(x) - f(x_0)\| \leq L\|x - x_0\|$.*

- δ, L depend on x_0
- Optimization problem to seek L:

$$\hat{L}(x_i) = \underset{x_j \in B_\epsilon(x_i)}{\operatorname{argmax}} \frac{\|f(x_i) - f(x_j)\|_2}{\|x_i - x_j\|_2}$$

Regression datasets



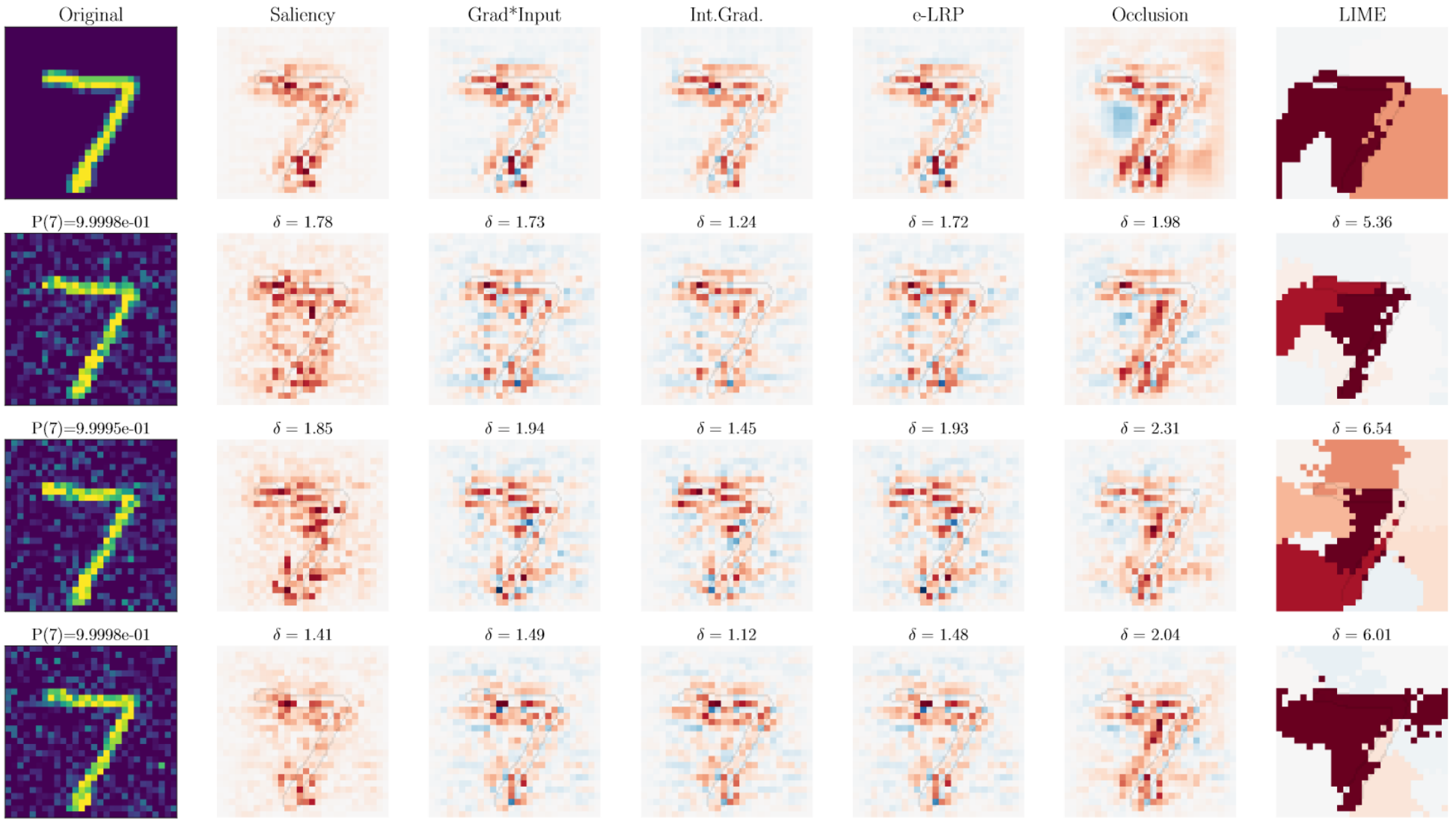
(a) SHAP ($L = 6.98$) (b) LIME ($L = 18.17$)

Figure 3: **Top**: example x_i from the BOSTON dataset and its *explanations* (attributions). **Bottom**: explanations for the maximizer of the Lipschitz estimate $L(x_i)$ as per (1).

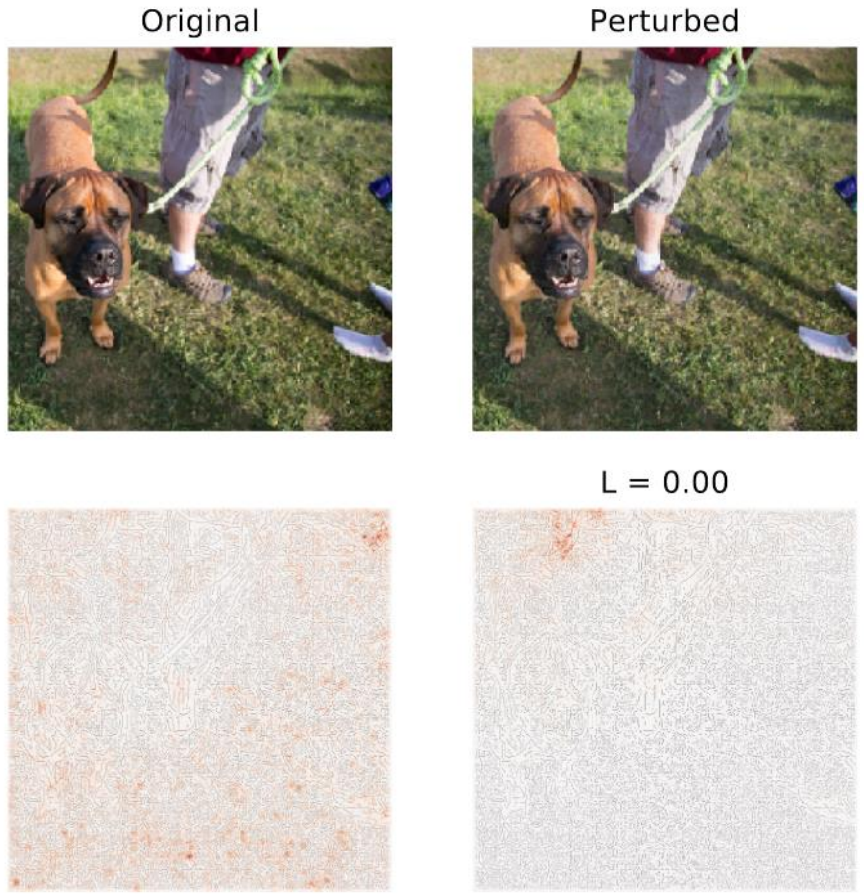
(a) SHAP ($L = 3.63$) (b) LIME ($L = 0.67$)

Figure 4: Robustness upon explaining a classifier on the COMPAS dataset. The two rows correspond to the pair maximizing $\tilde{L}_X$ (2) over the entire test fold, with $\epsilon = 0.1$.

MNIST


Original
Saliency
Grad*Input
Int.Grad.
e-LRP
Occlusion
LIME
P(7)=9.9998e-01
δ = 1.78
δ = 1.73
δ = 1.24
δ = 1.72
δ = 1.98
δ = 5.36
P(7)=9.9995e-01
δ = 1.85
δ = 1.94
δ = 1.45
δ = 1.93
δ = 2.31
δ = 6.54
P(7)=9.9998e-01
δ = 1.41
δ = 1.49
δ = 1.12
δ = 1.48
δ = 2.04
δ = 6.01

- Model predicts the same class in both cases with almost identical probabilities, yet the explanations are remarkably different



- Small perturbations that have minimal (or no) effect on the underlying model's predictions, yet have significant effects on the explanations.

- Q： Is it necessary to get a robust explanation system？
  - *Are the explanations immediate and understandable ?*
  - *Do relevance scores correspond to truly relevant features ?*
  - *How consistent are the explanations for similar/neighboring examples ?*

Towards Robust Interpretability with Self-Explaining Neural Networks

- Explanations are supposed to be:
  - explicitness, faithfulness and stability
- Idea
  - Explicitness：progressively generalizing linear classifiers to complex yet architecturally explicit models
  - Faithfulness and stability： enforced via regularization
  - Built from bottom to up

Linear and beyond

- $f(x) = \sum_i^n \theta_i x_i + \theta_0$
- $f(x) = \theta(x)^T x,\ \theta(x) \sim network(x)$
- Ensure robustness: $\nabla_x f(x) \approx \theta(x_0)$, $x \in Neighborhood(x_0)$
  - Locally linear: $\theta(x_0)$
- Feature mapping to higher level: $h(x) : \mathcal{X} \to \mathcal{Z} \subset \mathbb{R}^k$
- $f(x) = \theta(x)^T h(x) = \sum_{i=1}^{K} \theta(x)_i h(x)_i$
- Release the aggregation:
  - permutation invariant
  - isolate the effect of individual feature
  - relative magnitude of θ

Self-explaining models

- Locally Lipschitz condition on $\theta(x) \to h(x)$:

Definition 2.1. *$f : \mathcal{X} \subseteq \mathbb{R}^n \to \mathbb{R}^m$ is **locally Lipschitz** if for every x_0 there exist $\delta > 0$ and $L \in \mathbb{R}$ such that $\|x - x_0\| < \delta$ implies $\|f(x) - f(x_0)\| \leq L\|x - x_0\|$.*

- Self-explaining prediction model：

$$f(x) = g\big(\theta_1(x)h_1(x), \ldots, \theta_k(x)h_k(x)\big)$$

where:

P1) g is monotone and completely additively separable
P2) For every $z_i := \theta_i(x)h_i(x)$, g satisfies $\frac{\partial g}{\partial z_i} \geq 0$
P3) θ is locally difference bounded by h
P4) $h_i(x)$ is an interpretable representation of x
P5) k is small.

In that case, for a given input x, we define the explanation of $f(x)$ to be the set $\mathcal{E}_f(x) \equiv \{(h_i(x), \theta_i(x))\}_{i=1}^k$ of basis concepts and their influence scores.

Enforce the conditions

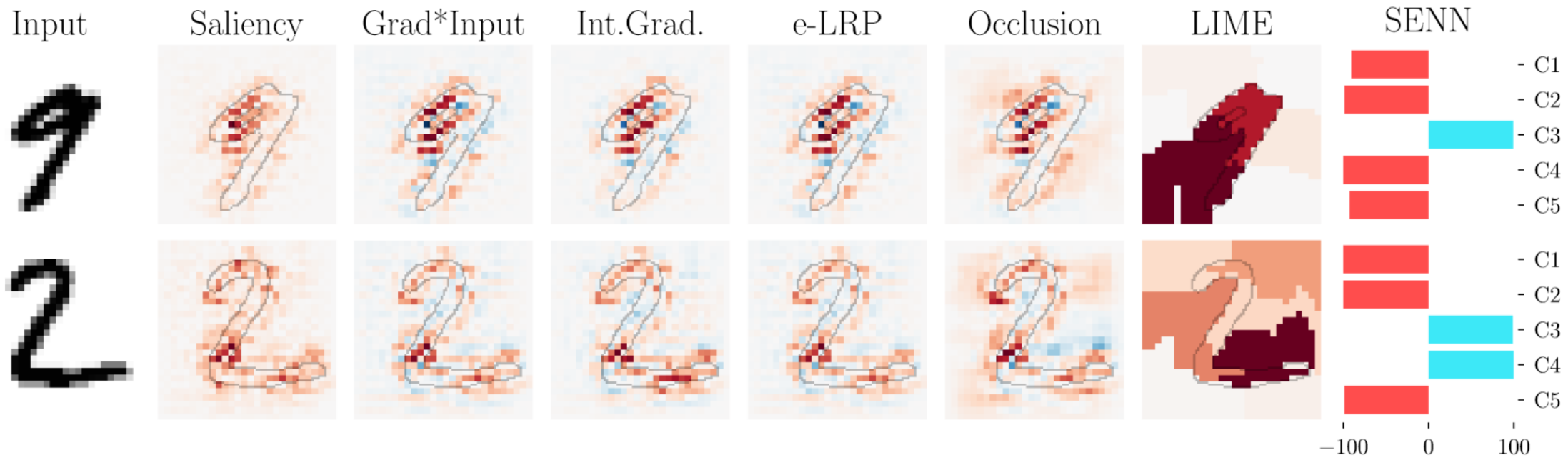
- P1, P2: depend on the choice of aggregating function g
  - Positive affine functions
- P4, P5: application-dependent
- P3
  - Chain rule: $\nabla_x f = \nabla_z f \cdot J_x h$
- Ansatz to enforce P3: $\mathcal{L}_\theta(f) := \|\nabla_x f(x) - \theta(x)^T J_x h(x)\| \approx 0$
- Gradient: $\mathcal{L}_y(f(x), y) + \lambda \mathcal{L}_\theta(f)$

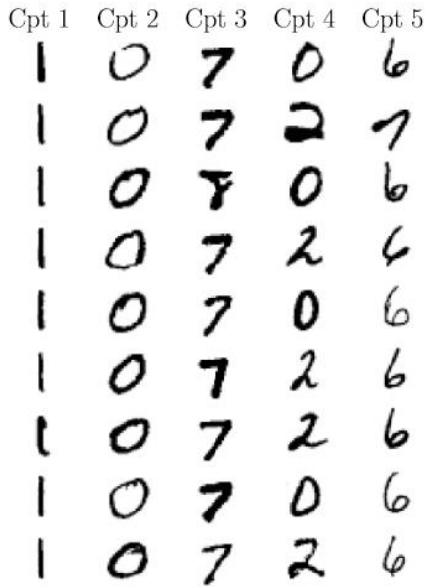
Learning interpretable basis concepts(h)

- For high-dimensional inputs, raw features lead to noisy
  - lack of robustness
- Informed by expert knowledge or learnt
  - Fidelity
  - Diversity
  - Grounding
- h: autoencoder

$$\mathcal{L}_y(f(x), y) + \lambda \mathcal{L}_\theta(f) + \xi \mathcal{L}_h(x, \hat{x})$$

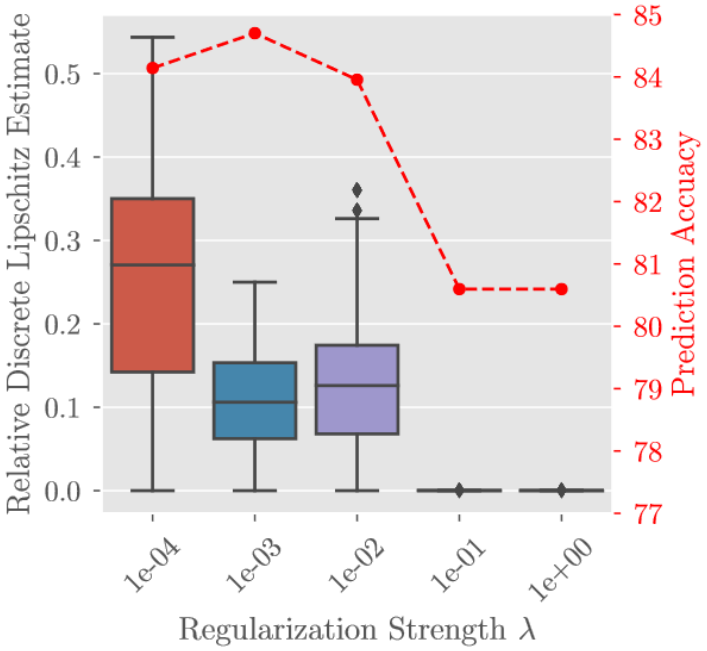
$$X^i = \underset{\hat{X} \subseteq X, |\hat{X}| = l}{\operatorname{argmax}} \sum_{x \in \hat{X}} h(x)_i$$


Input
Saliency
Grad*Input
Int.Grad.
e-LRP
Occlusion
LIME
SENN
- C1
- C2
- C3
- C4
- C5
- C1
- C2
- C3
- C4
- C5
−100
0
100

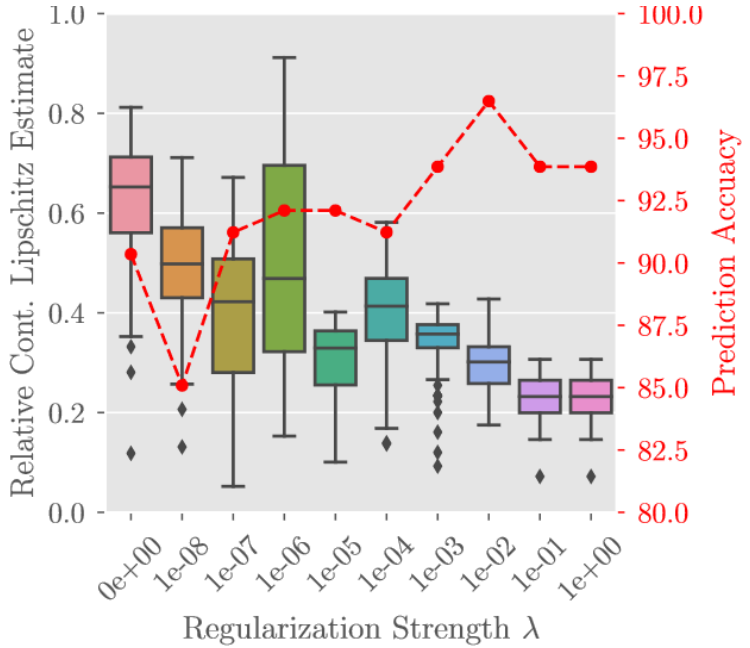

Cpt 1
Cpt 2
Cpt 3
Cpt 4
Cpt 5

Robustness

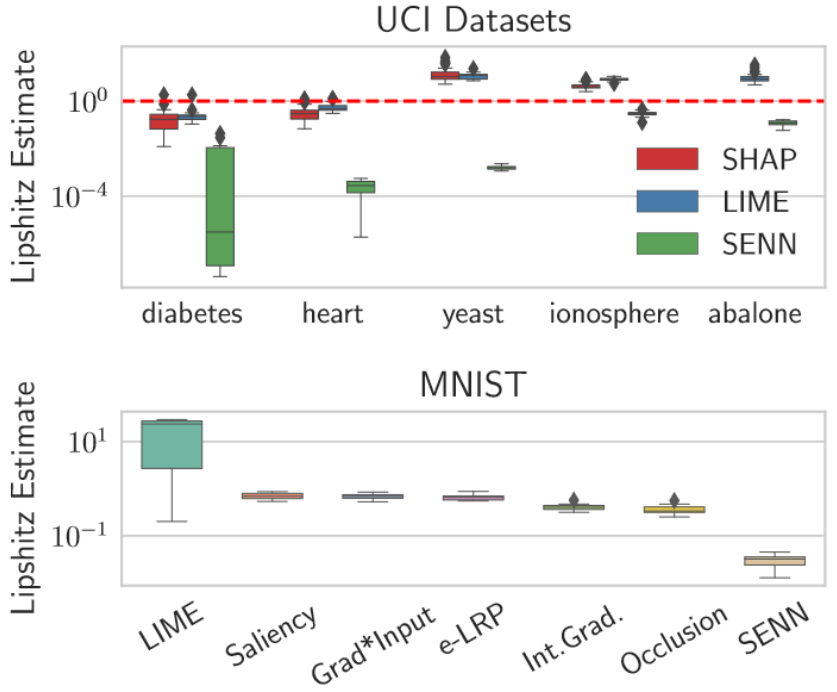
$$\hat{L}(x_i) = \underset{x_j \in B_\epsilon(x_i)}{\operatorname{argmax}} \frac{\|f_{\text{expl}}(x_i) - f_{\text{expl}}(x_j)\|_2}{\|h(x_i) - h(x_j)\|_2}$$



(A) Senn on Compas



(B) Senn on Breast-Cancer



(C) All methods on UCI/Mnist

Towards robust, locally linear deep networks

- Aim - gradient stability : robustness on the derivative of the mapping with respect to input
- Method - two steps:
  - Identifies a region around a point where linear approximation is provably stable
  - Optimization step to expand such regions
- ResNet & RNN

Notation

- $f_\theta: \mathbb{R}^D \to \mathbb{R}^L$Neural network θ with M hidden layers and N_i neurons in the i-th layer
- $z^i, a^i \in \mathbb{R}^{N_i}, W^i \in \mathbb{R}^{N_i \times N_{i-1}}, b^i \in \mathbb{R}^{N_i}$

$$\mathbf{a}^i = \mathrm{ReLU}(\mathbf{z}^i) := \max(\mathbf{0}, \mathbf{z}^i), \quad \mathbf{z}^i = \mathbf{W}^i \mathbf{a}^{i-1} + \mathbf{b}^i, \forall i \in [M], \quad \mathbf{a}^0 = \mathbf{x},$$

- Output layer:

$$f_\theta(\mathbf{x}) = \mathbf{W}^{M+1}\mathbf{a}^M + \mathbf{b}^{M+1} \text{ with } \mathbf{W}^{M+1} \in \mathbb{R}^{L \times N_M} \text{ and } \mathbf{b}^{M+1} \in \mathbb{R}^L.$$

$$\mathcal{B}_{\epsilon,p}(\mathbf{x}) := \{\bar{\mathbf{x}} \in \mathbb{R}^D : \|\bar{\mathbf{x}} - \mathbf{x}\|_p \leq \epsilon\}$$

Definition 1. *(Activation Pattern) An activation pattern is a set of indicators for neurons* $\mathcal{O} = \{\mathbf{o}^i \in \{-1, 1\}^{N_i} | i \in [M]\}$ *that specifies the following functional constraints:*

$$\mathbf{z}^i_j \geq 0, \quad \text{if } \mathbf{o}^i_j = 1; \quad \mathbf{z}^i_j \leq 0, \quad \text{if } \mathbf{o}^i_j = -1. \tag{2}$$

$$\partial \mathbf{a}^{i'}_{j'} / \partial \mathbf{z}^{i'}_{j'} := \max(\mathbf{o}^{i'}_{j'}, 0), \forall j' \in [N_{i'}], i' \in [i-1],$$

Step1-Inference for regions with stable derivative

- Derive an explicit characterization of the feasible set on the input space

Lemma 2. *Given an activation pattern $\mathcal{O}$ with any feasible point $\mathbf{x}$, each activation indicator $\mathbf{o}_j^i \in \mathcal{O}$ induces a feasible set $S_j^i(\mathbf{x}) = \{\bar{\mathbf{x}} \in \mathbb{R}^D : \mathbf{o}_j^i[(\nabla_{\mathbf{x}}\mathbf{z}_j^i)^\top \bar{\mathbf{x}} + (\mathbf{z}_j^i - (\nabla_{\mathbf{x}}\mathbf{z}_j^i)^\top \mathbf{x})] \geq 0\}$, and the feasible set of the activation pattern is equivalent to $S(\mathbf{x}) = \cap_{i=1}^{M} \cap_{j=1}^{N_i} S_j^i(\mathbf{x})$.*

- Definition the ℓ_p margin of $\mathbf{x}$ subject to its activation pattern:

$$\hat{\epsilon}_{\mathbf{x},p} := \max_{\epsilon \geq 0 : \mathbf{x}' \in S(\mathbf{x}), \forall \mathbf{x}' \in \mathcal{B}_{\epsilon,p}(\mathbf{x})} \epsilon = \max_{\epsilon \geq 0 : \mathcal{B}_{\epsilon,p}(\mathbf{x}) \subseteq S(\mathbf{x})} \epsilon$$

- $\mathrm{p} = 2$, can be certified analytically, minimal of the distances between x and hyperplane

$$\hat{\epsilon}_{\mathbf{x},2} = \min_{(i,j)\in\mathcal{I}} |\mathbf{z}_j^i| / \|\nabla_{\mathbf{x}}\mathbf{z}_j^i\|_2$$

Step2-Learning: maximizing the margins of stable derivatives

- Not trackable optimization problem:

$$\min_{\theta} \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}} \left[\mathcal{L}(f_\theta(\mathbf{x}),\mathbf{y}) - \lambda \min_{(i,j)\in\mathcal{I}} \frac{|\mathbf{z}_j^i|}{\|\nabla_{\mathbf{x}}\mathbf{z}_j^i\|_2} \right]$$

- Hinge-based relaxation inspired by SVM：

$$\min_{\mathbf{w},b} \sum_{\mathbf{x}\in\mathcal{D}_{\mathbf{x}}} \|\mathbf{w}\|_2^2 + C\max(0, 1 - |\mathbf{w}^T\mathbf{x} + b|)$$

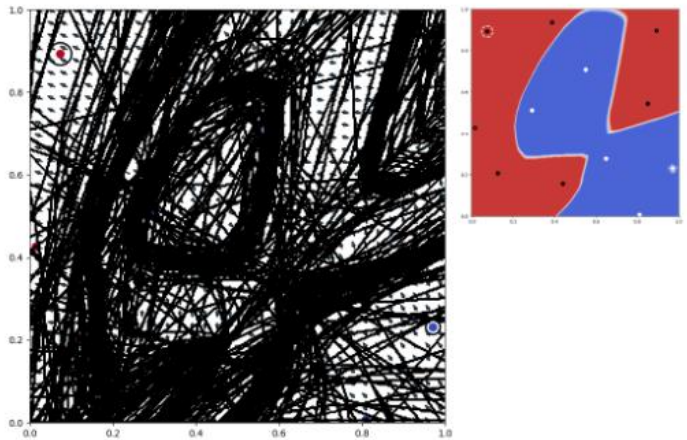
$$\min_{\theta} \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}} \mathcal{L}(f_\theta(\mathbf{x}),\mathbf{y}) - \lambda \min_{(i,j)\in\mathcal{I}} \frac{|\mathbf{z}_j^i|}{\|\nabla_{\mathbf{x}}\mathbf{z}_j^i\|_2}, \quad s.t. \min_{(i,j)\in\mathcal{I}} |\mathbf{z}_j^i| \geq 1, \forall(\mathbf{x},\mathbf{y}) \in \mathcal{D}$$

$$\min_{\theta} \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}} \mathcal{L}(f_\theta(\mathbf{x}),\mathbf{y}) - \lambda \min_{(i,j)\in\mathcal{I}} \frac{1}{\|\nabla_{\mathbf{x}}\mathbf{z}_j^i\|_2}, \quad s.t. \min_{(i,j)\in\mathcal{I}} |\mathbf{z}_j^i| \geq 1, \forall(\mathbf{x},\mathbf{y}) \in \mathcal{D}$$

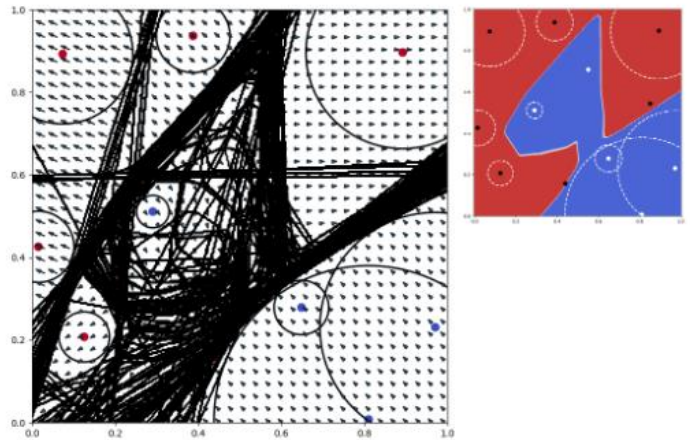
$$\min_{\theta} \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}} \mathcal{L}(f_\theta(\mathbf{x}),\mathbf{y}) + \lambda \max_{(i,j)\in\mathcal{I}} \left[\|\nabla_{\mathbf{x}}\mathbf{z}_j^i\|_2^2 + C\max(0, 1 - |\mathbf{z}_j^i|) \right]$$

$$\min_{\theta} \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}} \mathcal{L}(f_\theta(\mathbf{x}),\mathbf{y}) + \frac{\lambda}{\#\hat{\mathcal{I}}(\mathbf{x},\gamma)} \sum_{(i,j)\in\hat{\mathcal{I}}(\mathbf{x},\gamma)} \left[\|\nabla_{\mathbf{x}}\mathbf{z}_j^i\|_2^2 + C\max(0, 1 - |\mathbf{z}_j^i|) \right]$$

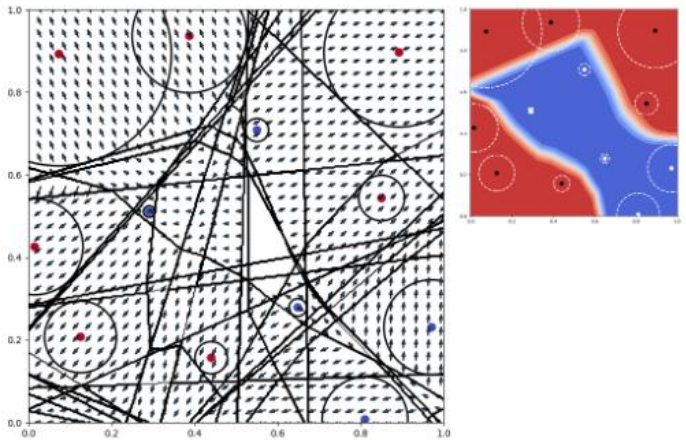
Visualize the effect of the proposed methods



(a) Vanilla loss



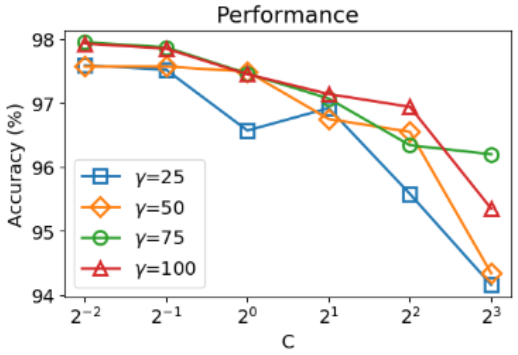
(b) Distance regularization



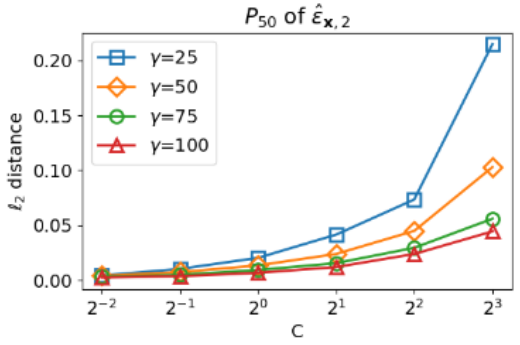
(c) Relaxed regularization

MNIST

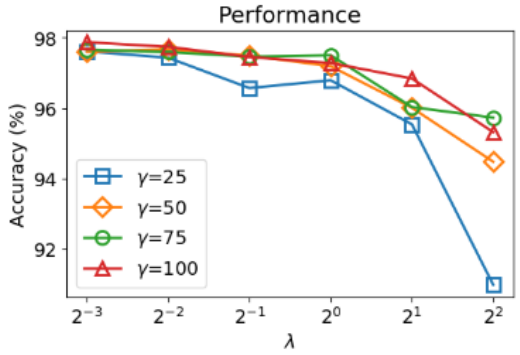
Loss	C	ACC	#CLR	$\hat{\epsilon}_{\mathbf{x},1}(\times 10^{-4})$				$\hat{\epsilon}_{\mathbf{x},2}(\times 10^{-4})$			
				P_{25}	P_{50}	P_{75}	P_{100}	P_{25}	P_{50}	P_{75}	P_{100}
Vanilla		98%	10000	22	53	106	866	3	6	13	91
ROLL	0.25	98%	9986	219	530	1056	6347	37	92	182	1070
ROLL	1.00	97%	8523	665	1593	3175	21825	125	297	604	4345



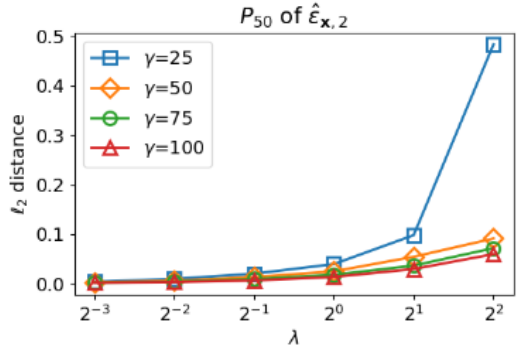
(a) $\lambda = 0.5$



(b) $\lambda = 0.5$



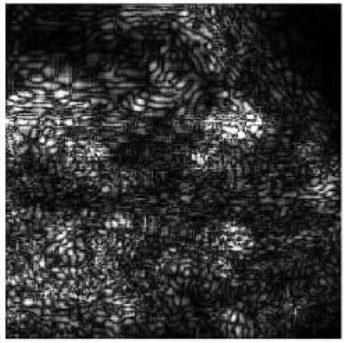
(c) $C = 1$



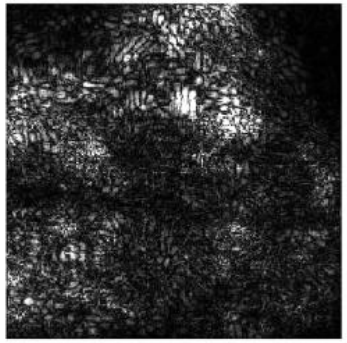
(d) $C = 1$

CALTECH-256

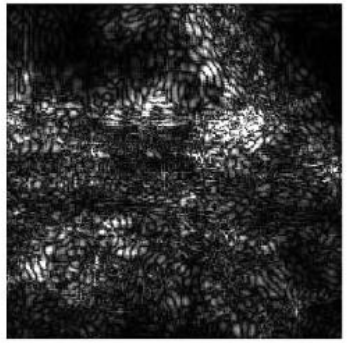
(a) Image (Laptop)



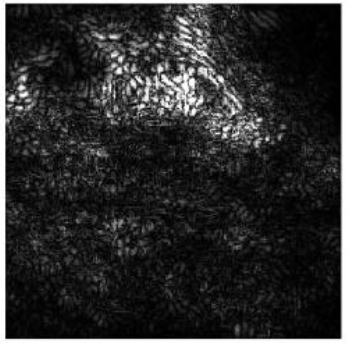
(b) Orig. gradient (ROLL)



(c) Adv. gradient (ROLL)

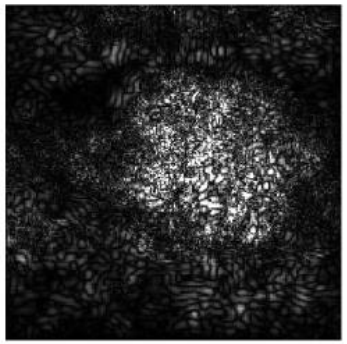


(d) Orig. gradient (Vanilla)

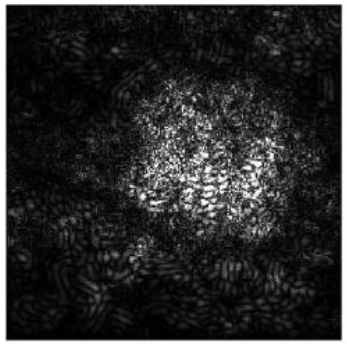


(e) Adv. gradient (Vanilla)

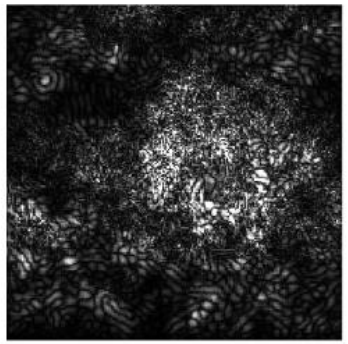
(f) Image (Bear)



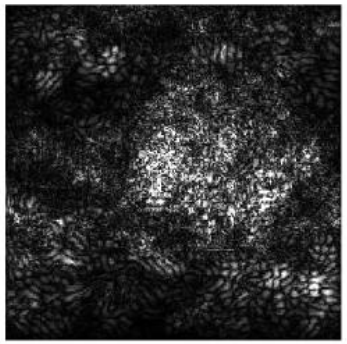
(g) Orig. gradient (ROLL)



(h) Adv. gradient (ROLL)



(i) Orig. gradient (Vanilla)



(j) Adv. gradient (Vanilla)

Interpreting Adversarially Trained Convolutional Neural Networks

- Key problem: how AT-CNNs classify objects?

- Hypothesis:
  - AT-CNNs are more biased towards global structures, such as shapes and edges
  - AT-CNNs are less sensitive to the texture distortion and focus more on shape information, while the normally trained CNNs the other way around

Verification

- Experiment 1: Visually reflect the different characteristics of the two networks by comparing the mapping of AT-CNNs and CNNs on the saliency map
- Experiment 2: construct new datasets containing only texture or shape to evaluate the performance of AT-CNNs and CNN

- Only normal training or adversarial training on the original training sets, and then evaluate their generalizability over the transformed data

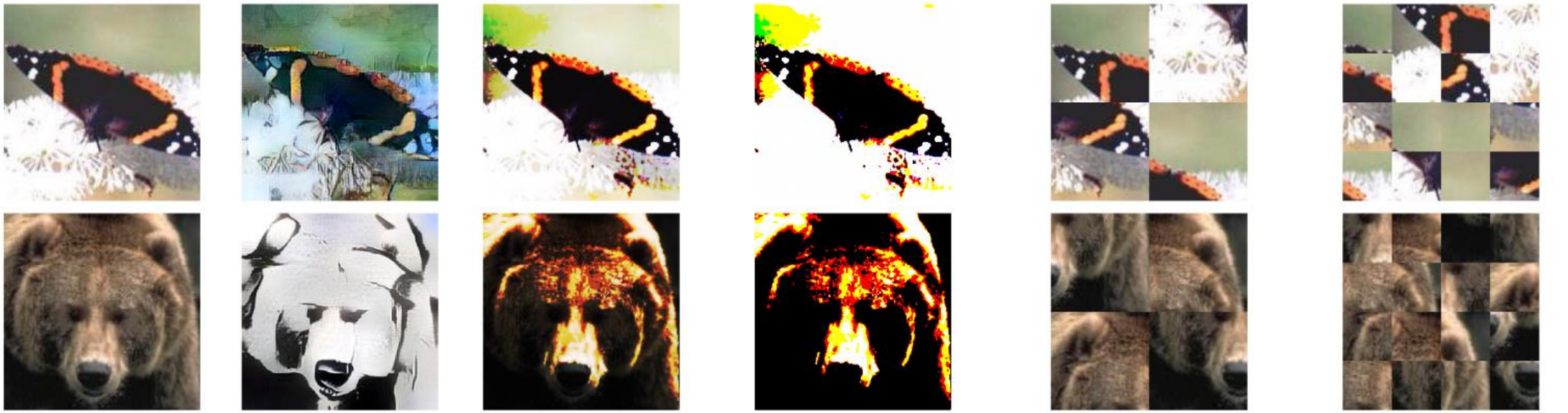
Saliency map

- Show the sensitivity of the output to each pixel of input image
- Gaussian smooth gradient-based method

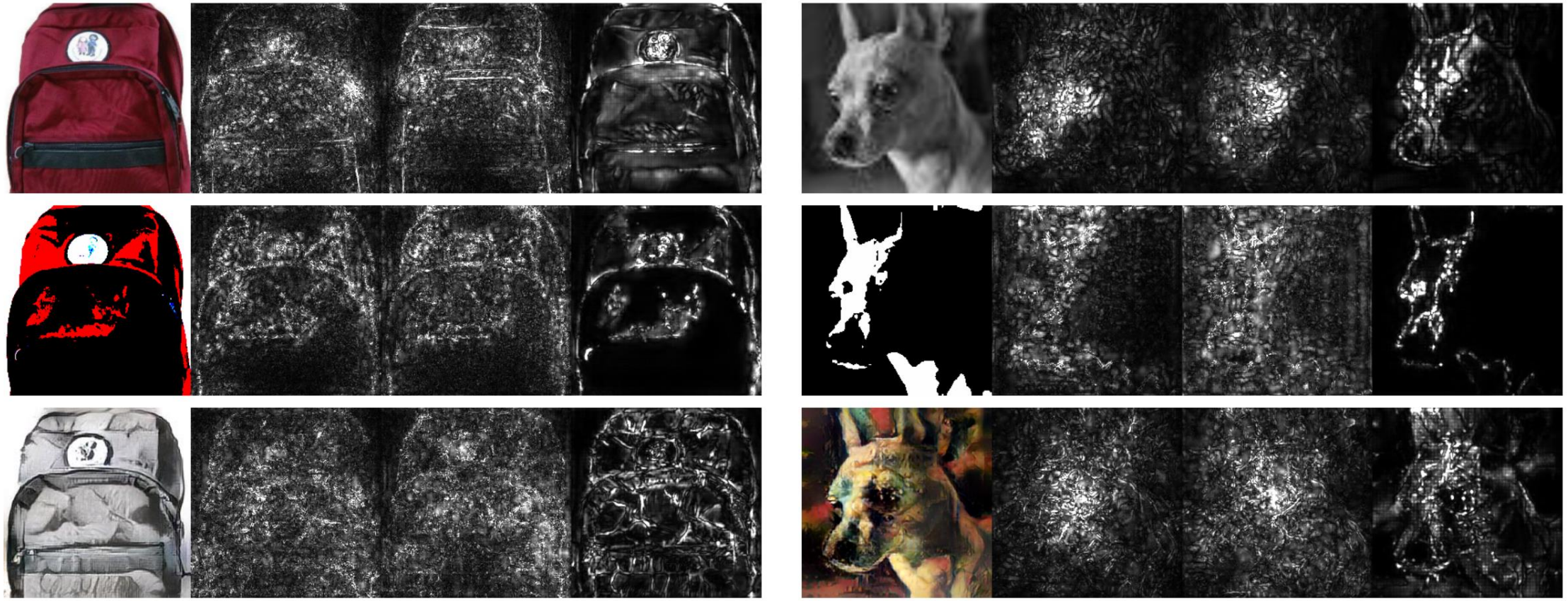
$$E = \frac{1}{n}\sum_{i=1}^{n}\frac{\partial S_c(x_i)}{\partial x_i}$$

- $x_i = x + g_i$, where $g_i \sim \mathcal{N}(0, \sigma^2)$

Transformation preserving texture or shape

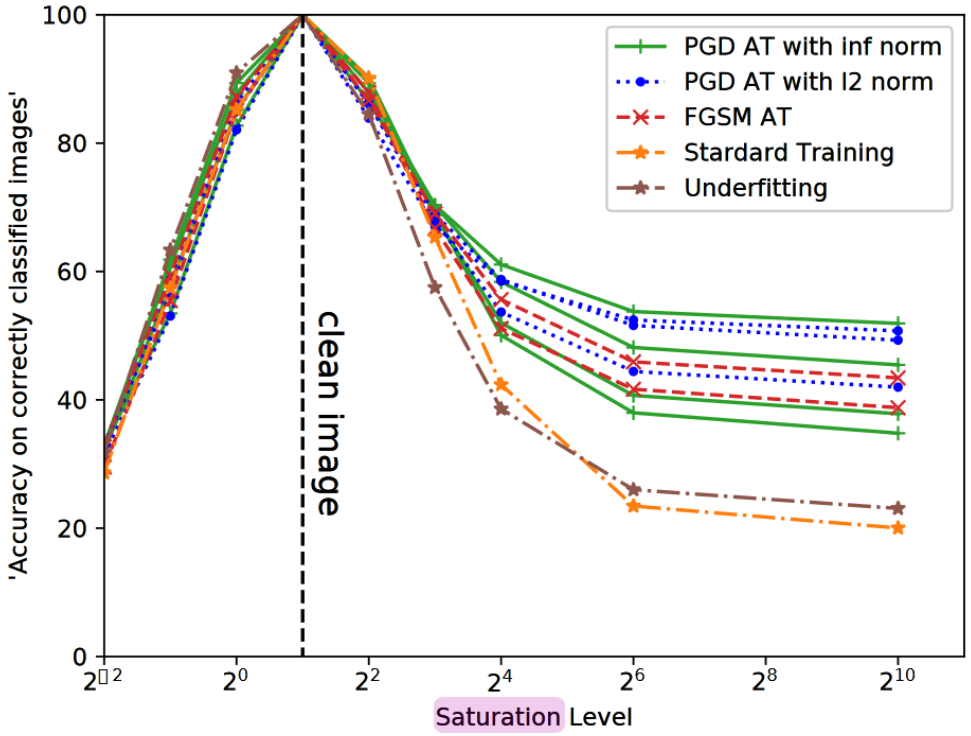


(a) Original (b) Stylized (c) Saturated 8 (d) Saturated 1024 (e) patch-shuffle 2 (f) patch-shuffle 4

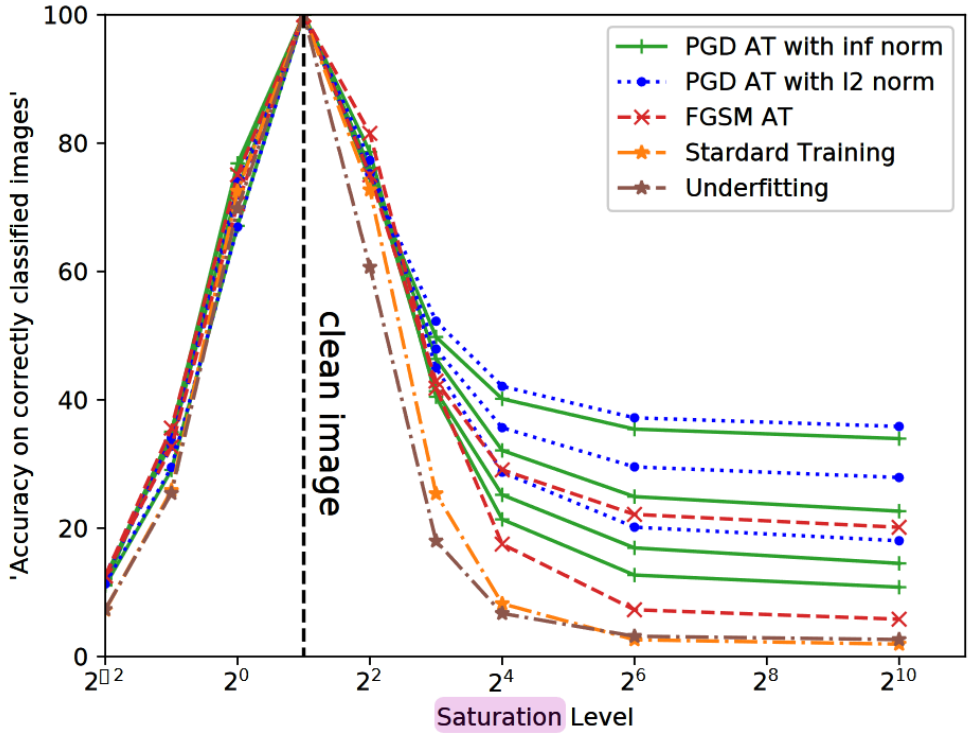


- CNN, underfitting CNN, AT-CNN

Accuracy of models w.r.t. different saturation levels



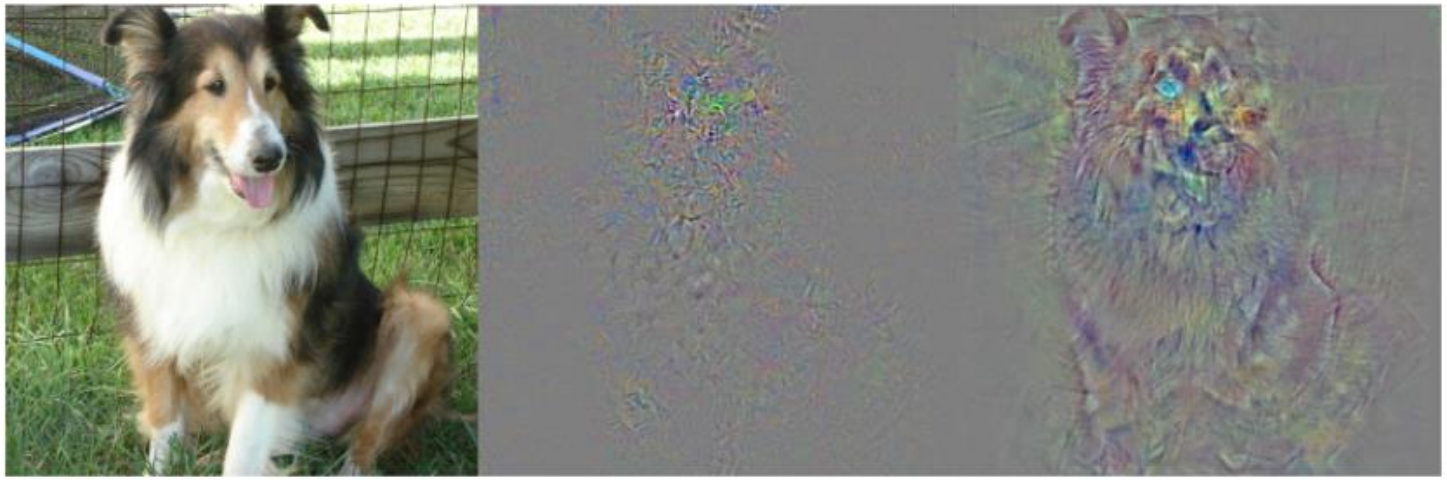
(a) Caltech-256



(b) Tiny ImageNet

On the Connection Between Adversarial Robustness and Saliency Map Interpretability

- Quantify “robust models exhibit more interpretable saliency maps” by considering the alignment between input image and saliency map
- *As the distance to the decision boundary grows, so does the alignment.*
- Try to confirm the idea based on models trained with a local Lipschitz regularization



- ReLU family's networks are locally affine -> the distance to the decision boundary can be computed

$$l(x) = \sup\{r \mid \forall i : \Psi^i \text{ affine in } B_r(x)\}$$

Definition 3 (Alignment, Multi-Class Case). *Let*

$$\Psi = (\Psi^1, \ldots, \Psi^n) : X \to \mathbb{R}^n$$

be differentiable in x. Then for an n-class classifier defined a.e. by

$$F(x) = \arg\max_i \Psi^i(x), \tag{6}$$

we call $\nabla\Psi^{F(x)}$ the saliency map of F. *We further call*

$$\alpha(x) := \frac{|\langle x, \nabla\Psi^{F(x)}(x)\rangle|}{\|\nabla\Psi^{F(x)}(x)\|}, \tag{7}$$

the alignment with respect to Ψ in x.

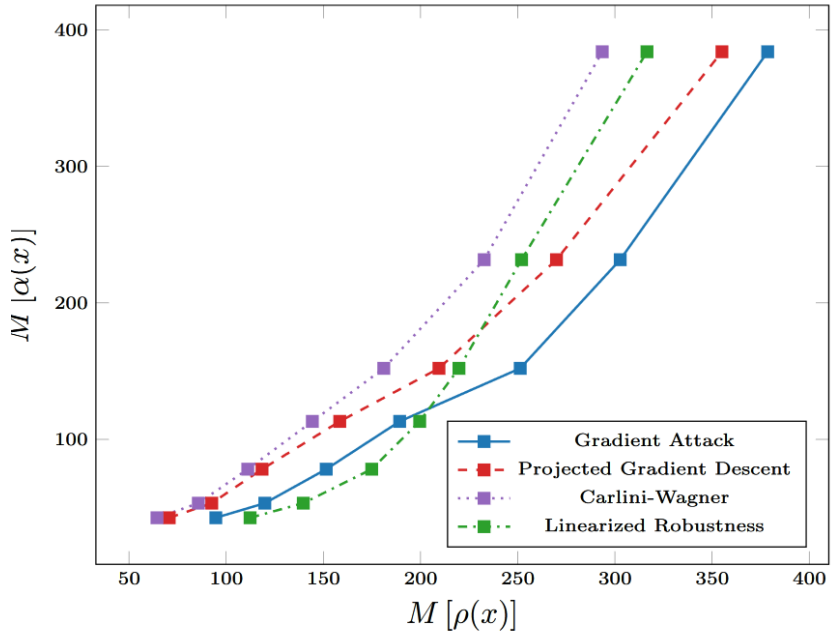
Definition 4 (Linearized Robustness). *Let $\Psi(x)$ be the differentiable score vector for the classifier F in x. We call*

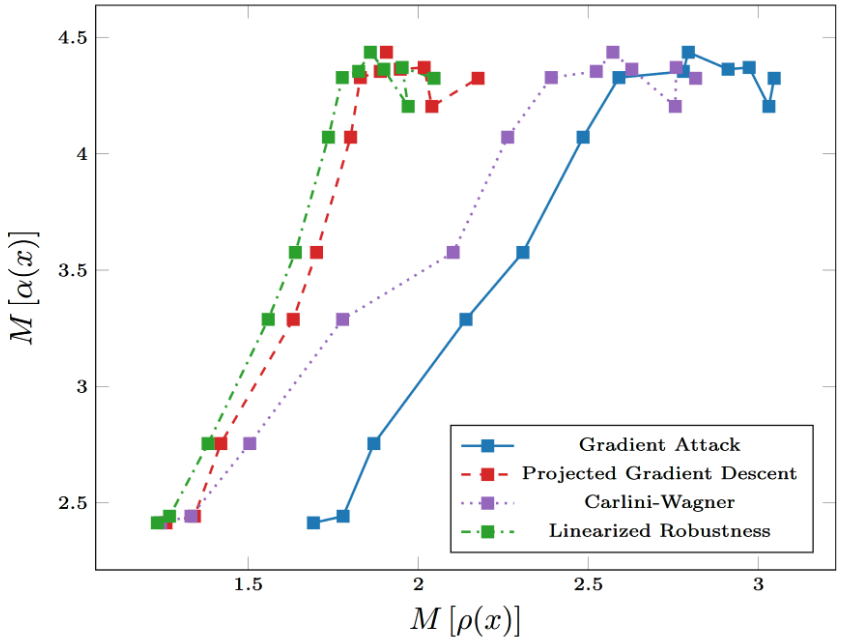
$$\tilde{\rho}(x) := \min_{j \neq i^*} \frac{\Psi^{i^*}(x) - \Psi^j(x)}{\|\nabla\Psi^{i^*}(x) - \nabla\Psi^j(x)\|}, \tag{9}$$

the linearized robustness *in x, where $i^* := F(x)$ is the predicted class at point x.*

- The above considerations demonstrate how an increase in robustness may induce an increase in the alignment between an input image and its respective saliency map
  - The explanation of the phenomenon
- Experiments：

$$\frac{1}{N}\sum_{i=1}^{N}\left[\mathcal{L}(f_\theta(x^{(i)}), y^{(i)}) + \lambda \cdot \|\nabla \mathcal{L}(f_\theta(x^{(i)}), y^{(i)})\|^2\right]$$





Leveraging Interpretability for Improved Adversarial Learning

- Spatially constrained one-pixel adversarial perturbation strategy

- Tips： An effective use of gradient-based interpretability methods for tasks outside of the purpose of purely explain-ability

Spatially constrained one-pixel adversarial per-turbations

- For everyindividual adversarial perturbation, a set of N vectors in $\mathbb{R}^5$
  - x, y coordinates & RGB values are randomly generated, giving the initial parent population
- In each iteration:
  - N children are generated and the fittest pixels providing the lowest probabilistic label for the correct class remain
- Hypothesize that the probability of susceptibility is highest in areas that are identified as highly sensitive viagradient-based interpretability
  - sensitivity maps generated via gradient-based interpretability

- Using SmoothGrad to generate a sensitivity map

$$s(x) = \frac{1}{n}\sum_{1}^{n} \hat{s}(x + \mathcal{N}(0, \sigma^2)); \quad S = \{s(x) > \tau\}$$

(a) Susceptibility set generation.

- *The pixels withhigher sensitivity in the generated sensitivity maps are moresusceptible to attack*

(b) One-pixel adversarial perturbation generation.

- Perturbation generation:

$$p(x) = \{X, Y, r, g, b\}; \quad (X, Y) \in S$$

$$\max_{p(x)} f_{adv}(x + p(x)) \ \vee ||p(x)||_0 \leq 1$$

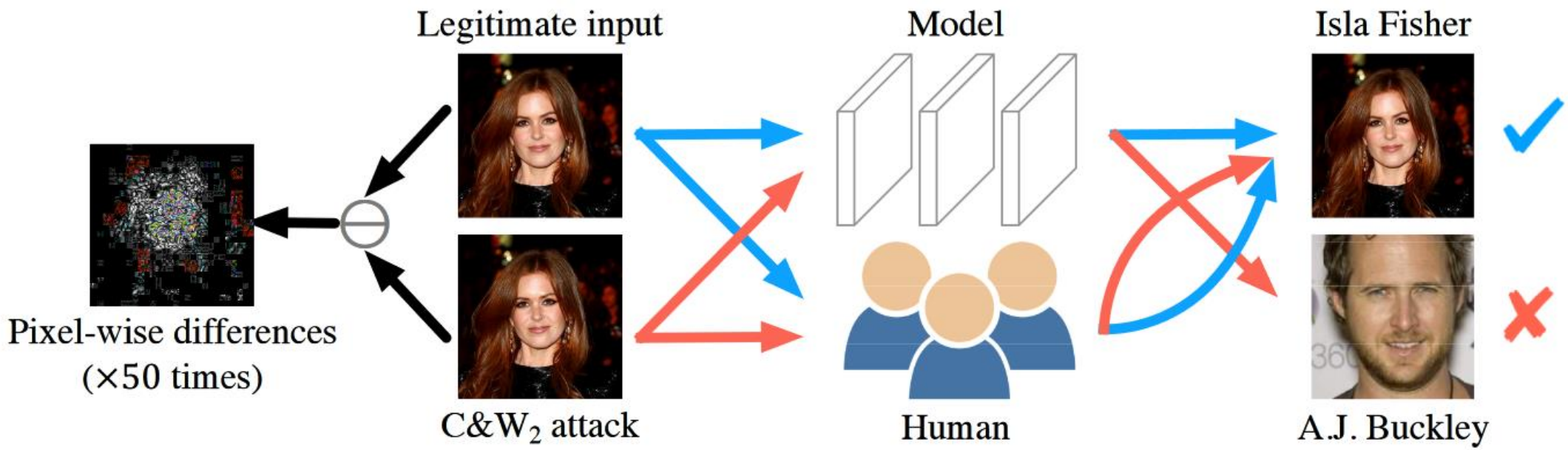
Experiments

- Interpretability helps a lot !

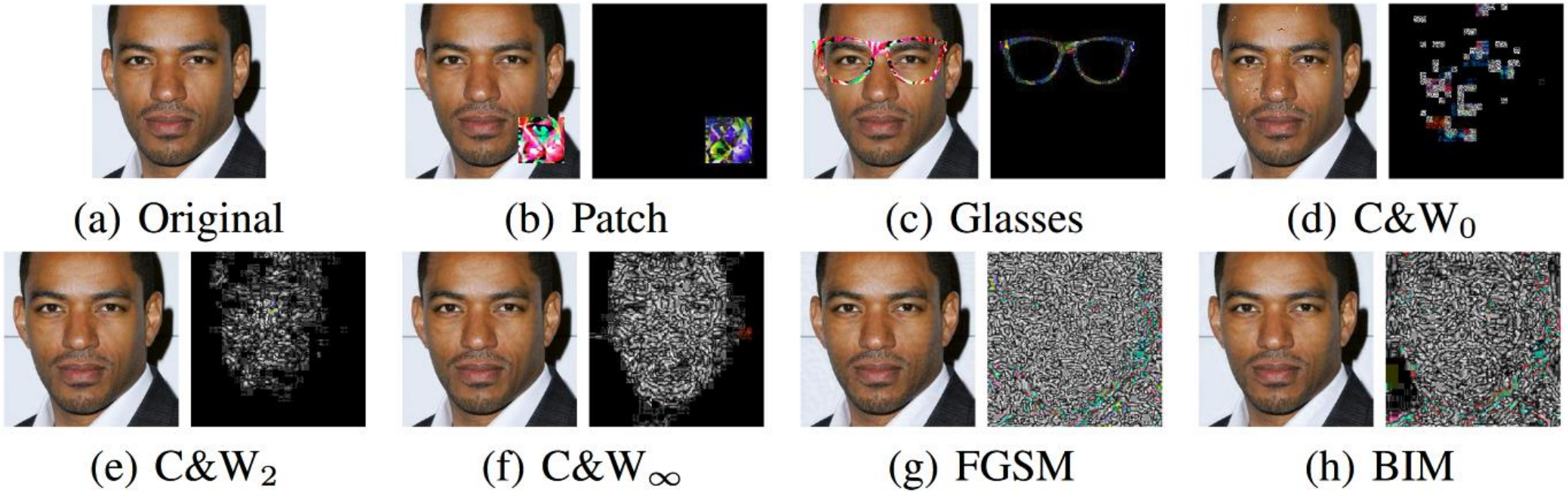
Unconstrained			Constrained (ours)		
T (s)	AVG T (s)	AVG # itr	T (s)	AVG T (s)	AVG # itr
207.98	2.08	35.61	37.15	0.56	2.52
216.00	2.16	37.40	35.36	0.54	2.33
212.10	2.12	36.72	34.16	0.52	2.26
210.71	2.11	36.39	37.03	0.58	2.52
210.72	2.11	36.24	36.05	0.55	2.40

Attacks Meet Interpretability: Attribute-steered Detection of Adversarial Samples

- *“While classification results on benign inputs can be reasoned based on the human perceptible features/, results on adversarial samples can hardly be explained” – adversarial example is related with interpretability*
- Identify neurons critical for individual attributes
  - features a novel bi-directional correspondence inference between attributes and internal neurons
  - Enhanced the critical neurons & suppress the uninterpretable parts
- Mainly focus on face recognition models

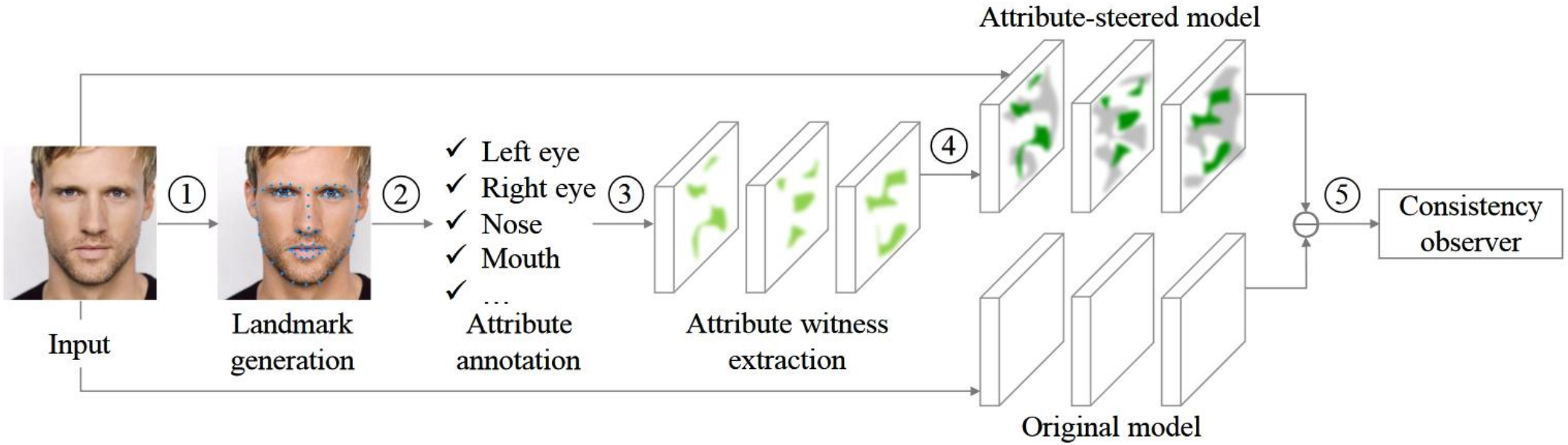

Legitimate input
Model
Isla Fisher
Pixel-wise differences
(×50 times)
C&W2 attack
Human
A.J. Buckley

Different attacks on FRSes

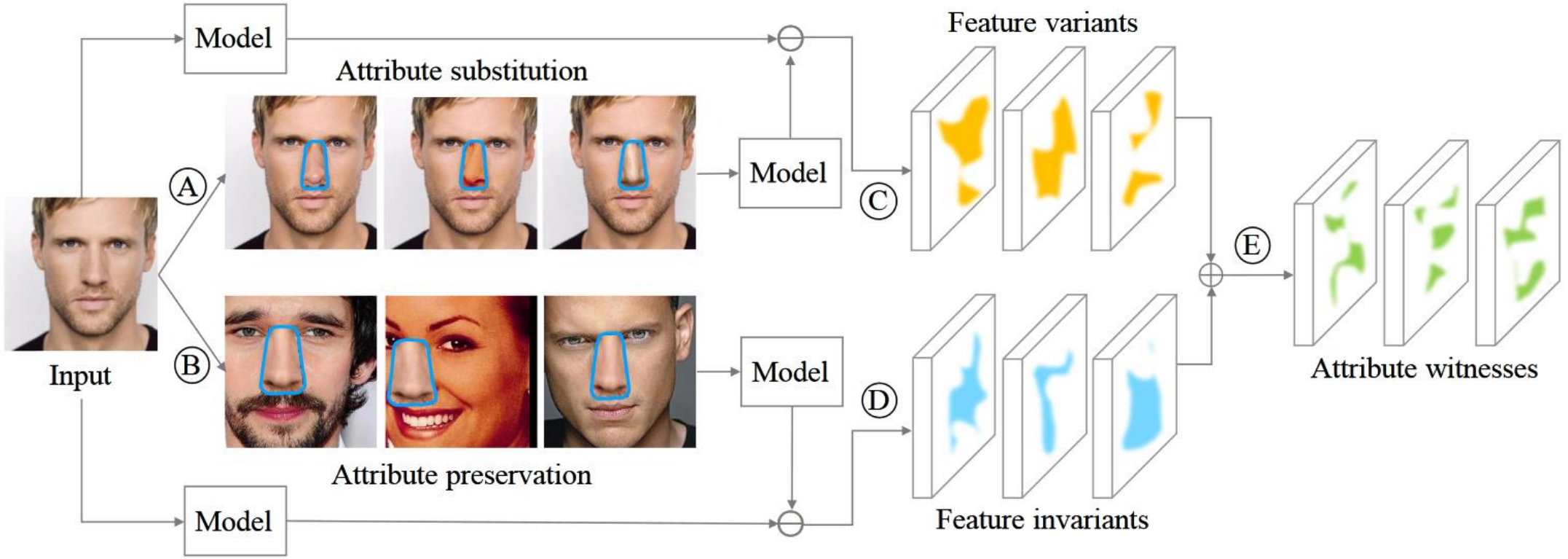


- Adversary detection: after feature squeezing, adversarial samples are likely to induce different classification results
- Key idea： *determine if the prediction results of a FRS DNN are based on human recognizable features/attributes*
- Step 1- attribute witnesses： identifying neurons that correspond to human perceptible attribute
- Step 2- : given a test image, observe the activation values of attribute witnesses and their comparison with the activation values of other neurons,

Workflow


Attribute-steered model
Input
1
Landmark generation
2
✓ Left eye
✓ Right eye
✓ Nose
✓ Mouth
✓ …
Attribute annotation
3
Attribute witness extraction
4
5
Consistency observer
Original model

Attribute Witness Extraction



- Bi-directional:

$$\textit{Attribute changes} \longrightarrow \textit{Neuron activation changes}$$

$$\textit{Neuron activation changes} \longrightarrow \textit{Attribute changes}$$

$$\textit{No attribute changes} \longrightarrow \textit{No neuron activation changes}$$

Some details

- Model：
$$F(x) = \text{softmax} \circ f^n \circ \cdots \circ f^1(x).$$
- Layer：
$$f^l(p) = \langle f_1^l(p), \cdots, f_m^l(p) \rangle.$$
- Step A：
$$\Delta f_{j,as}^l = |f_j^l(p) - f_j^l(p_{as})|$$
$$g_{as}^{i,l}(p) = \{f_j^l(p) \mid \Delta f_{j,as}^l > median_{j \in [1,m]}(\Delta f_{j,as}^l)\}$$
- Step B：
$$g_{ap}^{i,l}(p) = \{f_j^l(p) \mid \Delta f_{j,ap}^l \leq median_{j \in [1,m]}(\Delta f_{j,ap}^l)\}$$
$$g^i = \bigcup_{l=1}^{n} g^{i,l}(p) = \bigcup_{l=1}^{n} g_{as}^{i,l}(p) \cap g_{ap}^{i,l}(p)$$

Attribute-steered Model

- Strengthen the witness neurons and weaken the others
- Neuron weakening: $v' = e^{-\frac{v-\mu}{\alpha \cdot \sigma}} \cdot v$
- Neuron strengthening: $v' = \epsilon \cdot v + \left(1 - e^{-\frac{v-\min}{\beta \cdot \sigma}}\right) \cdot v,$

- *There isn't any form of re-training*

Layer Name #Neuron	conv1_1 64	conv1_2 64	pool1 64	conv2_1 128	conv2_2 128	pool2 128	conv3_1 256	conv3_2 256	conv3_3 256	pool3 256
#Left Eye	1	-	-	-	2	3	4	2	3	2
#Right Eye	1	-	-	-	3	3	4	3	2	3
#Nose	1	-	-	-	1	3	2	-	1	3
#Mouth	1	-	-	-	3	2	4	3	15	7
#Left Eye & Right Eye	1	-	-	-	2	3	3	1	-	-
#Left Eye & Nose	1	-	-	-	1	3	2	-	-	-
#Left Eye & Mouth	1	-	-	-	2	1	2	1	1	-
#Right Eye & Nose	1	-	-	-	1	3	1	-	-	-
#Right Eye & Mouth	1	-	-	-	3	1	2	2	1	1
#Nose & Mouth	1	-	-	-	1	1	1	-	-	-
#Shared	1	-	-	-	1	1	1	-	-	-

Layer Name #Neuron	conv4_1 512	conv4_2 512	conv4_3 512	pool4 512	conv5_1 512	conv5_2 512	conv5_3 512	pool5 512	fc6 4096	fc7 4096
#Left Eye	9	5	15	7	12	4	1	1	-	1
#Right Eye	7	3	10	9	9	1	-	-	-	-
#Nose	10	8	17	13	7	2	2	1	-	1
#Mouth	19	12	12	11	8	2	1	2	1	1
#Left Eye & Right Eye	5	1	3	4	2	-	-	-	-	-
#Left Eye & Nose	3	-	4	-	1	-	-	-	-	-
#Left Eye & Mouth	1	1	-	-	-	-	-	-	-	-
#Right Eye & Nose	3	-	1	1	1	-	-	-	-	-
#Right Eye & Mouth	2	-	2	-	-	-	-	-	-	-
#Nose & Mouth	5	1	2	2	-	-	-	-	-	-
#Shared	1	-	-	-	-	-	-	-	-	-

Detection of adversarial examples

Detector	FP	Targeted										Untargeted	
		Patch		Glasses		$C\&W_0$		$C\&W_2$		$C\&W_\infty$		FGSM	BIM
		First	Next	First	Next	First	Next	First	Next	First	Next		
FS [18]	23.32%	0.77	0.71	0.73	0.58	0.68	0.65	0.60	0.50	0.42	0.37	0.36	0.20
AS	20.41%	0.96	0.98	0.97	0.97	0.93	0.99	0.99	1.00	0.96	1.00	0.85	0.76
AP	30.61%	0.89	0.96	0.69	0.75	0.96	0.94	0.99	0.97	0.95	0.99	0.87	0.89
WKN	7.87%	0.94	0.97	0.71	0.76	0.83	0.89	0.99	0.97	0.97	0.96	0.86	0.87
STN	2.33%	0.08	0.19	0.16	0.19	0.90	0.94	0.97	1.00	0.76	0.87	0.46	0.41
AmI	9.91%	0.97	0.98	0.85	0.85	0.91	0.95	0.99	0.99	0.97	1.00	0.91	0.90
w/o Left Eye	18.37%	0.97	0.99	0.75	0.79	0.88	0.92	0.99	0.95	0.97	0.98	0.89	0.90
w/o Right Eeye	18.08%	0.93	0.96	0.73	0.80	0.86	0.91	0.99	0.96	0.98	0.98	0.86	0.87
w/o Nose	27.41%	0.97	0.99	0.78	0.84	0.91	0.94	0.98	0.97	0.99	0.98	0.94	0.90
w/o Mouth	20.99%	0.91	0.97	0.74	0.79	0.86	0.95	1.00	0.95	0.99	0.98	0.86	0.87

Summary

Reference:

- Zhang C , Bengio S , Singer Y . Are All Layers Created Equal?[J]. 2019.
- Veit A , Wilber M , Belongie S . Residual Networks Behave Like Ensembles of Relatively Shallow Networks[J]. Advances in Neural Information Processing Systems, 2016.
- Alvarezmelis D, Jaakkola T S. On the Robustness of Interpretability Methods.[J]. arXiv: Learning, 2018.
- Alvarezmelis D , Jaakkola T S . Towards Robust Interpretability with Self-Explaining Neural Networks[J]. 2018.
- Lee G H , Alvarez-Melis D , Jaakkola T S . Towards Robust, Locally Linear Deep Networks[J]. 2019.
- Zhang T , Zhu Z . Interpreting Adversarially Trained Convolutional Neural Networks[J]. 2019.
- Etmann C , Lunz S , Maass P , et al. On the Connection Between Adversarial Robustness and Saliency Map Interpretability[J]. 2019.
- Kumar D , Ben-Daya I , Vats K , et al. Beyond Explainability: Leveraging Interpretability for Improved Adversarial Learning[J]. 2019.
- Liu N, Yang H, Hu X, et al. Adversarial Detection with Model Interpretation[C]. knowledge discovery and data mining, 2018: 1803-1811.