

Detect and Repair Errors in DNN-based Software

Thesis Defense

Yuchi Tian

July 21st, 2021

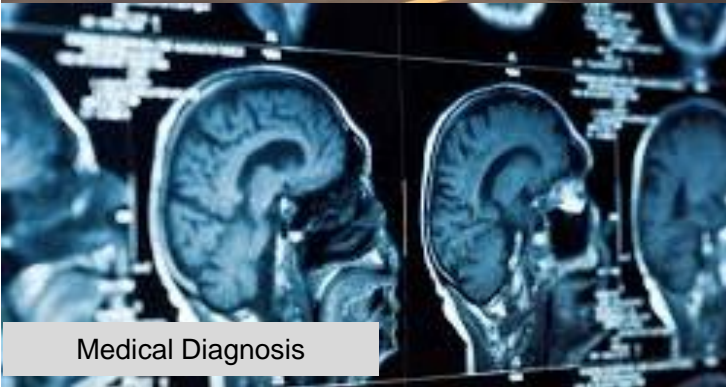




Self Driving Car



Malware Detection



Medical Diagnosis



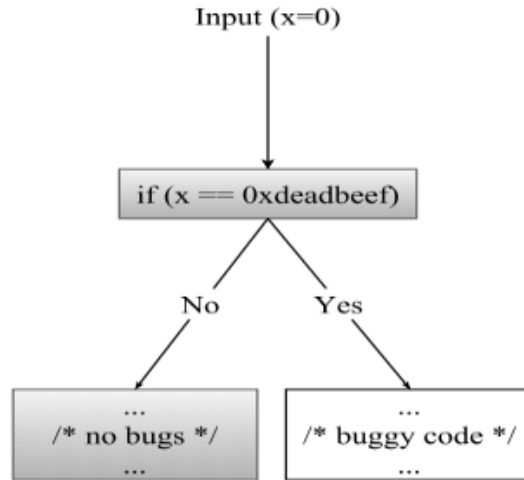
Aircraft Collision Avoidance

DNN is increasingly used in safety critical system

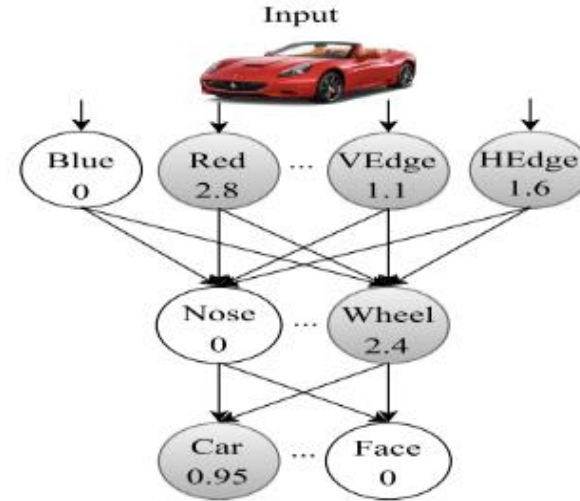
DNN suffers from corner cases



Traditional software vs. DNN



Traditional software



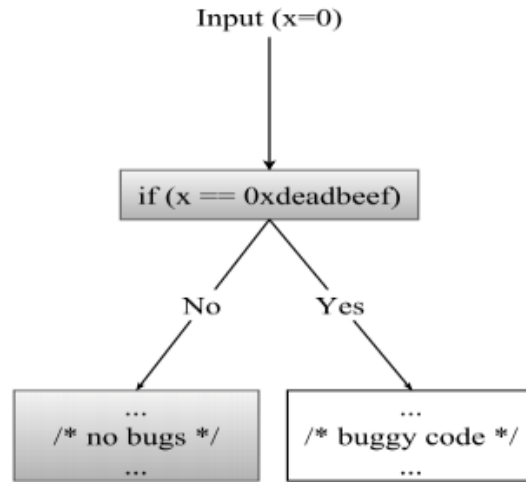
Deep Neural Network (DNN)

DNN: logic is not encoded with control flow

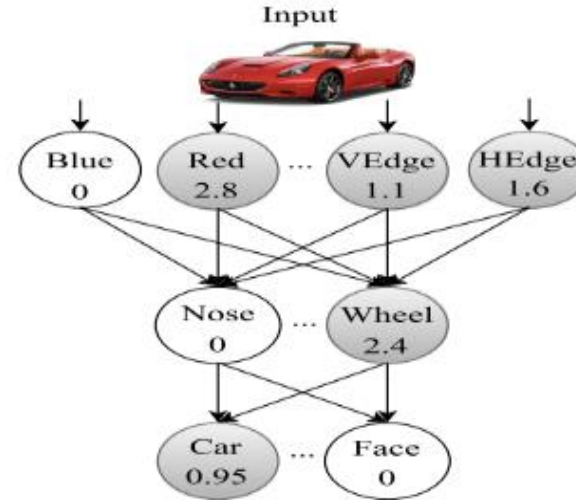


Code-coverage based testing will not work

Traditional software vs. DNN



Traditional software



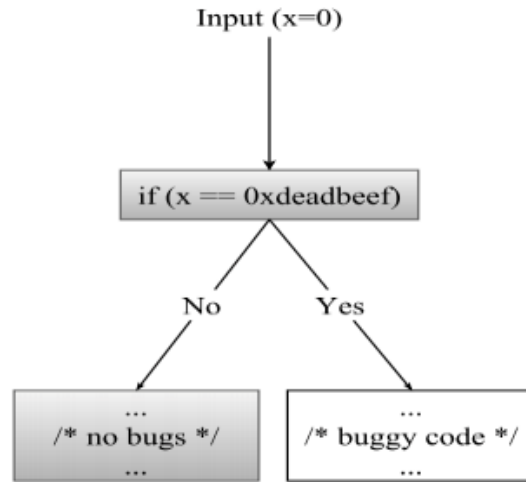
Deep Neural Network (DNN)

DNN models are highly non-linear

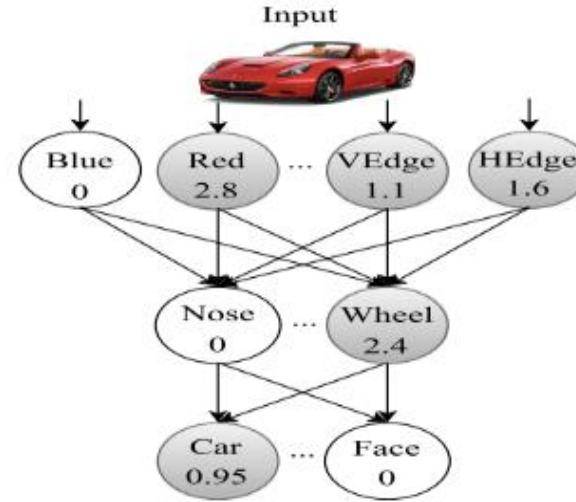


Symbolic analysis techniques will not work

Traditional software vs. DNN



Traditional software



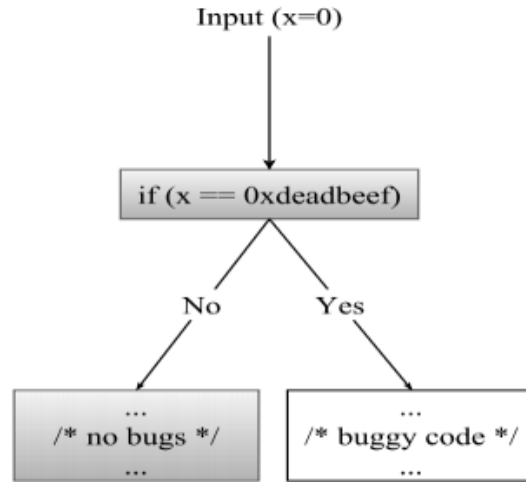
Deep Neural Network (DNN)

DNN is opaque

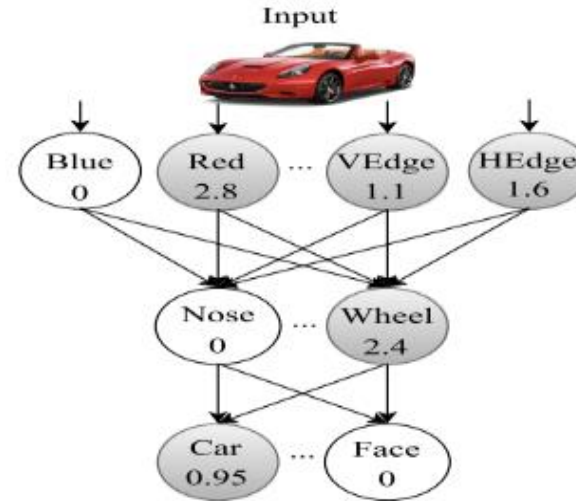


Existing debugging technique will not work

Traditional software vs. DNN



Traditional software



Deep Neural Network (DNN)

Traditional software testing, debugging and repairing techniques do not work well for DNN

My works – SE for AI

DNN
Testing

- DeepTest
- DeepRobust
- DeepInspect

DNN
Repair

- DeepTest
- Weighted
Regularization

My works – Error types



Instance wise error

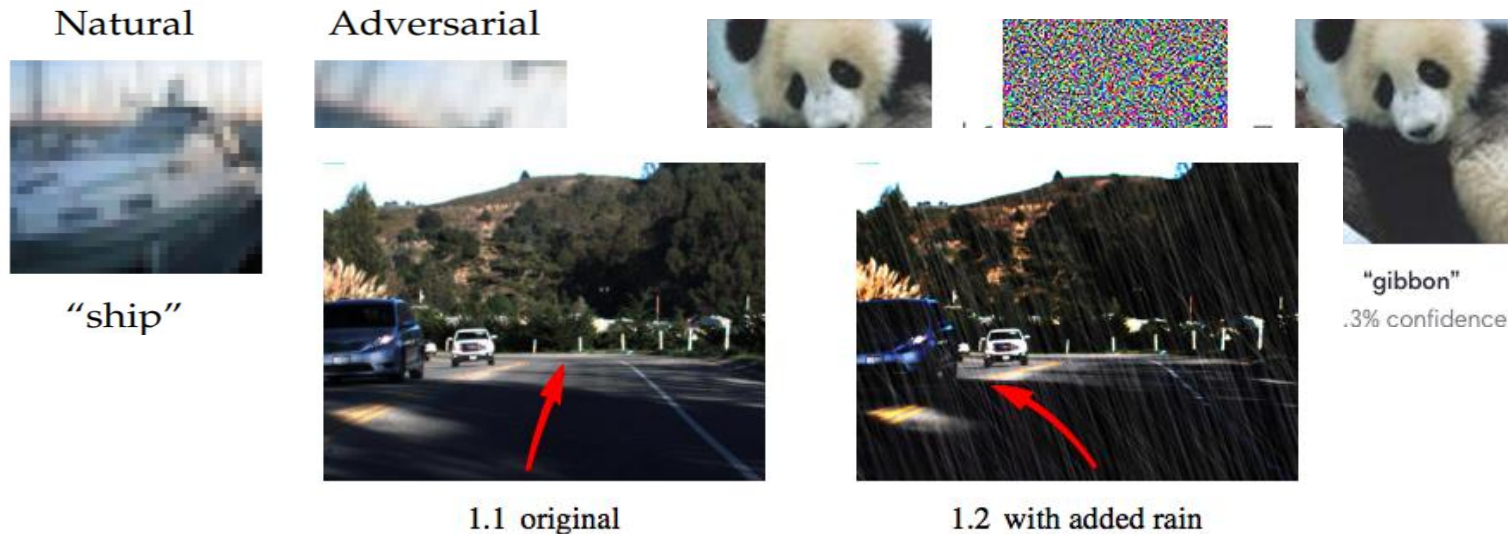


Group level error



Instance-wise errors

One single input leads to model's erroneous output



Instance-wise errors

Instance wise error



Norm
bounded
perturbation

[PGD, Madry,
ICLR'18]

...

Natural
transformation

[DeepXplore,
Pei, SOSP'17]
[DeepTest]
[DeepRobust]

...

Norm
bounded &
natural
transformation

[DeepHunter,
Xie, ISSTA'19]

...

Generative
adversary
network

[DeepRoad,
Zhang,
ASE'18]

...

Physical
attack

[Eykholt,
CVPR'18]

...

Group level errors

DNN model's weak performance on differentiating among certain classes or has inconsistent performance across classes.

Group confusion error

Group bias error

COCO gender (multi-label)

ImageNet (single-label)



cello -> violin

COCO (multi-label)



keyboard -> keyboard, mouse



women, skiing scenario ->
men, skiing scenario

12/96



Group level errors

Group level error



Group bias problem

[Fairness Testing, Galhotra,
ESEC/FSE'17]

[FairSquare, Albarghouthi, OOPSLA'17]

[DeepInspect, Tian&Zhong, ICSE'20]

[WR, Tian&Zhong, in submission]

Group confusion problem

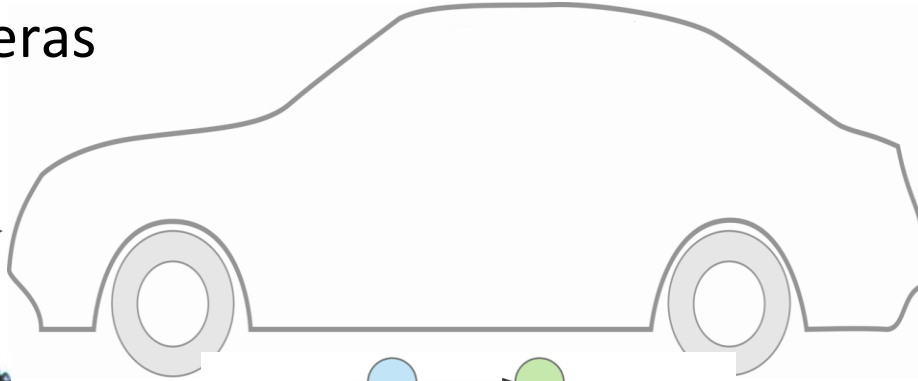
[DeepInspect, Tian&Zhong, ICSE'20]

[DNNrepair, Tian, short paper, FSE'20]

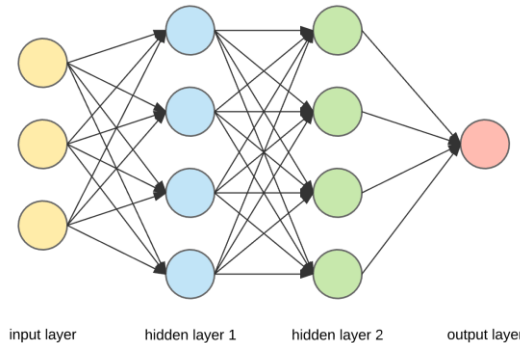
[WR, Tian&Zhong, in submission]

Study subject - DNN based autonomous car

Images from cameras



Steering angle



input layer

hidden layer 1

hidden layer 2










output layer



Study subject - DNN based Image Classification

There are two types of classification problems:

- Single-label: each input image has only one ground-truth label.
- Multi-label: each input image can have multiple ground-truth labels.

	Multi-Class	Multi-Label
C = 3		
  	<p>Samples</p>    <p>Labels (t)</p> <p>[0 0 1] [1 0 0] [0 1 0]</p>	<p>Samples</p>    <p>Labels (t)</p> <p>[1 0 1] [0 1 0] [1 1 1]</p>

Outline

- DeepInspect [ICSE' 20]
- Weighted Regularizations [ESEC/FSE' 20 short paper] [12 pages in submission]
- DeepTest [ICSE' 18]
- DeepRobust [FASE' 21]

Outline

- **DeepInspect [ICSE' 20]**
- Weighted Regularizations [ESEC/FSE' 20 short paper] [12 pages in submission]
- DeepTest [ICSE' 18]
- DeepRobust [FASE' 21]

Confusion and Bias Errors

- Group-level errors:
 - Confusion errors: a DNN model cannot distinguish between certain pair of classes
 - Bias errors: a DNN model is bias toward one class than the other class given a third class.

Confusion errors



Person



Bus

A DNN based classifier in a self-driving car may be confused between person and bus.

Confusion errors



Tiger



Cat

A DNN based classifier may be confused between cat and tiger.

Bias errors

Man



Bias toward

Woman

A DNN based classifier is more likely to predict the person to be a woman if there is a baby in the image.

Bias errors












A DNN based classifier is more likely to predict the person to be a man if the image is about outdoor sports.

DNN Image Classification

There are two types of classification problems:

- Single-label: each input image has only one ground-truth label.
- Multi-label: each input image can have multiple ground-truth labels.

	Multi-Class	Multi-Label
$C = 3$		
  	<p>Samples</p>    <p>Labels (t)</p> <p>[0 0 1] [1 0 0] [0 1 0]</p>	<p>Samples</p>    <p>Labels (t)</p> <p>[1 0 1] [0 1 0] [1 1 1]</p>

Formal Definition of Confusion Errors

We define two types of confusion measures:
given class x and y ,

$$\text{single-label_conf}(x, y) = \text{mean}(P(x|y), P(y|x))$$

$$\text{multi-label_conf}(x, y) = \text{mean}(P(x|x, y), P(y|x, y))$$

We refer these two measures as $\text{error}(x, y)$ in general and we call a model has a **confusion error** between x and y if:

$$\text{error}(x, y) > \text{confusion_threshold}$$

Formal Definition of Bias Errors

We define a bias measure Confusion Disparity (cd):

$$\text{cd}(x, y, z) = |\text{error}(x, z) - \text{error}(y, z)|$$

where error can be either single-label_conf or multi-label_conf

and Average Confusion Disparity (avg_cd):

$$\text{avg_cd}(x, y) := \frac{1}{|O| - 2} \sum_{z \in O, z \neq x, y} \text{cd}(x, y, z)$$

We call a model has a **bias error** on x and y if:

$$\text{avg_cd}(x, y) > \text{bias_threshold}$$

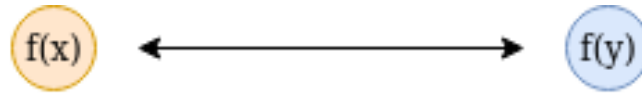
How do we detect the confusion and bias errors?



26/96

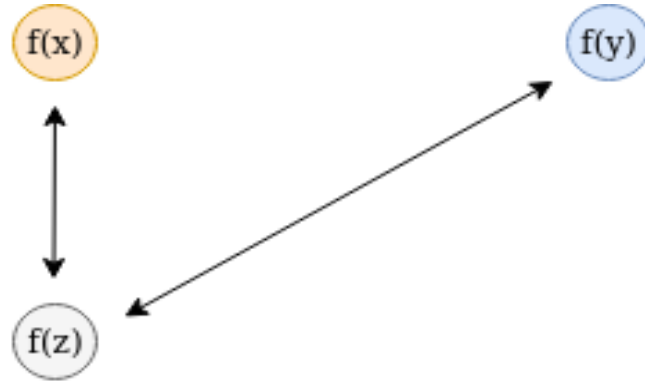
Leverage distance between two class representations

For each class x , find a good vector representation $f(x)$ such that:
 $f(x)$ reflects how a model perceives class x .



$$\text{distance}(f(x), f(y)) \propto \frac{1}{\text{error}(x, y)}$$

For the bias measure, it follows that:



$$|\text{distance}(f(x), f(z)) - \text{distance}(f(y), f(z))| \propto |\text{error}(x, z) - \text{error}(y, z)|$$

NAPVD

(Neuron Activation Probability Vector
Distance)

$$P(C) = [A(n_1)/N, A(n_2)/N, \dots, A(n_t)/N],$$

C is a class, N is total number of inputs with class C ,

$A(n_i)$ is activation frequency of neuron n_i

$$NAPVD(x, y) = L2_norm(P(x), P(y))$$

The smaller the distance, the larger confusion between x and y

Experiment Setup

Baselines:

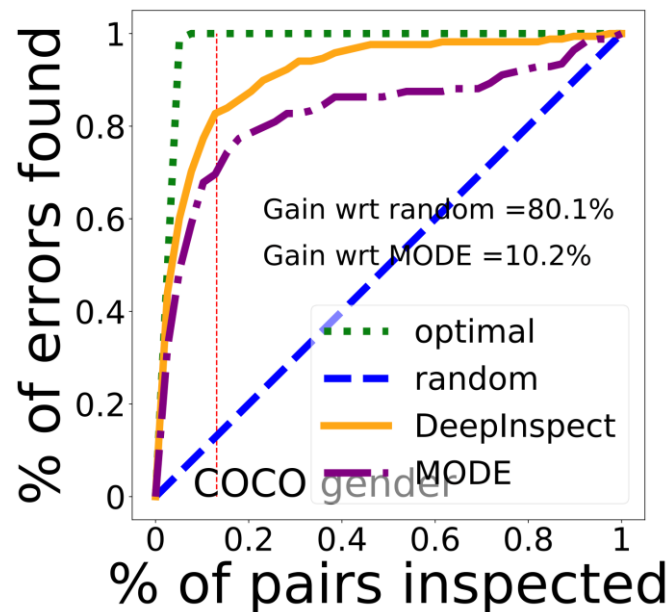
Classification Task	Dataset	#Classes	CNN Models
Multi-label classification	COCO	80	ResNet-50
	COCO gender	81	ResNet-50
	imSitu	205095	ResNet-34
Single-label classification	CIFAR-100	100	CNN
	Robust CIFAR-10	10	Small CNN
			Large CNN
			ResNet
	ImageNet	1000	ResNet-50

- Random: picks random class-pairs for inspection
- MODE-inspired: from the last linear layer before the output layer we extract per-class weight vectors and compute the pairwise distances between the weight vectors.

Results for Confusion Errors

NAPVD < mean-1std

Dataset	Method	Precision	Recall
COCO gender	DeepInspect	0.327	0.827
	MODE	0.248	0.744
	random	0.052	0.131

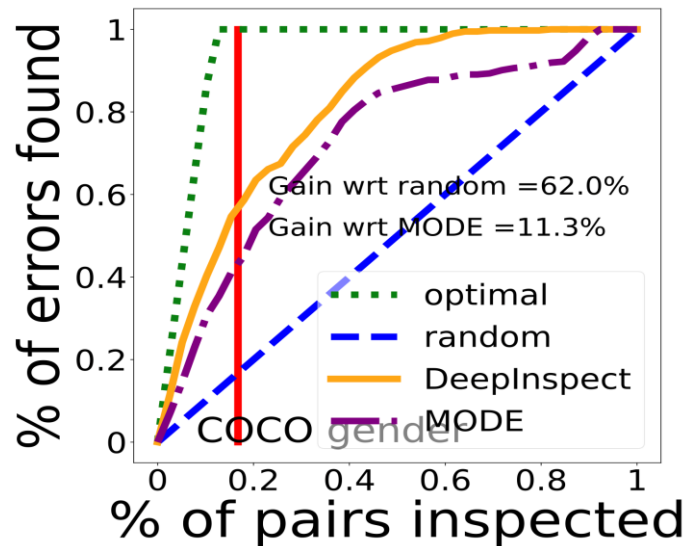


our method is better than the baselines

Results for Bias Errors

Avg_bias > mean+1std

Dataset	Method	Precision	Recall
COCO gender	DeepInspect	0.401	0.568
	MODE	0.315	0.393
	random	0.118	0.168



our method is better than the baselines

Identified Errors

Confusion error

ImageNet (single-label)



cello -> violin

COCO (multi-label)



keyboard -> keyboard, mouse

Bias error

COCO (multi-label)



women, skiing scenario ->
men, skiing scenario

Outline

- DeepInspect [ICSE' 20]
- **Weighted Regularizations [ESEC/FSE' 20 short paper]
[12 pages in submission]**
- DeepTest [ICSE' 18]
- DeepRobust [FASE' 21]

Confusion errors



Tiger



Cat

A DNN based software may be confused between cat and tiger.

Bias errors

Man ← Bias toward



Woman

A DNN model is more likely to predict the person to be a man if the image is about outdoor sports.

Definition

Confusion errors between class x and class y

$$\textit{single_label_conf}(x, y) = \textit{mean}(\textit{prob}(x|y), \textit{prob}(y|x))$$

$$\textit{multi_label_conf}(x, y) = \textit{mean}(\textit{prob}((x, y)|x), \textit{prob}((x, y)|y))$$

Definition

Bias errors between class x and class y , given class z .

$$cd(x, y, z) = |confusion(x, z) - confusion(y, z)|$$

cd means “confusion disparity”

Problem to solve

Reduce confusion between a pair of classes

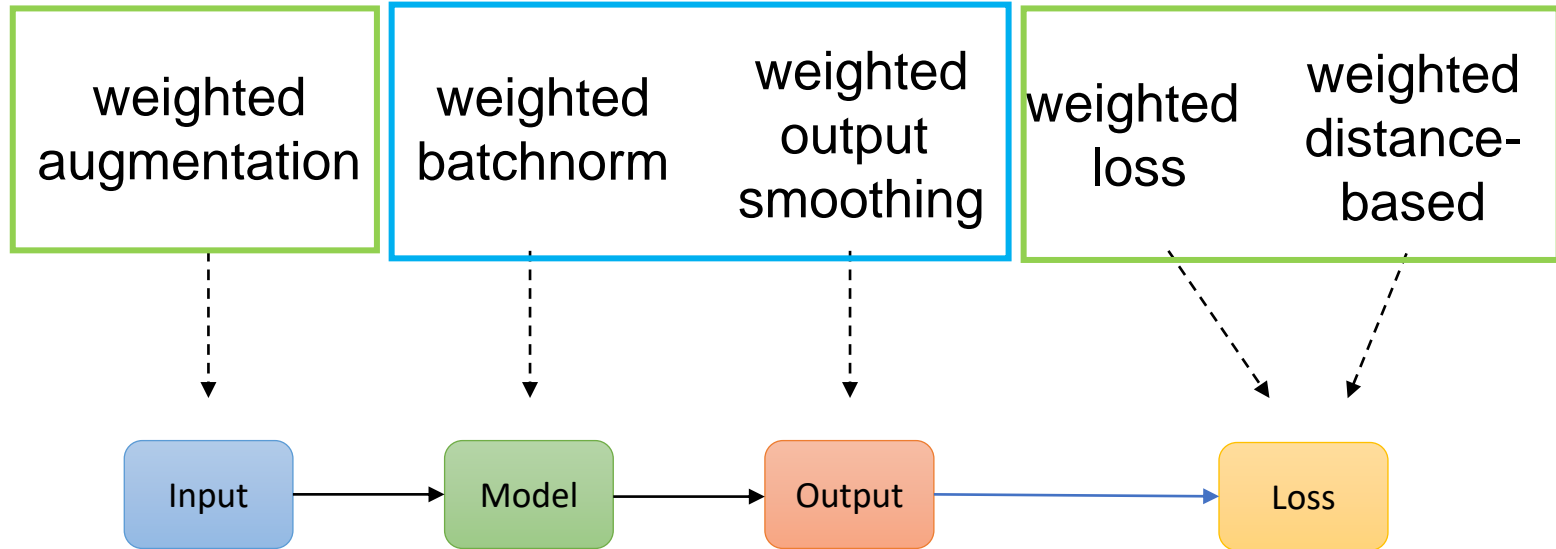
$$\textit{single_label_conf}(x, y) = \textit{mean}(\textit{prob}(x|y), \textit{prob}(y|x))$$

$$\textit{multi_label_conf}(x, y) = \textit{mean}(\textit{prob}((x, y)|x), \textit{prob}((x, y)|y))$$

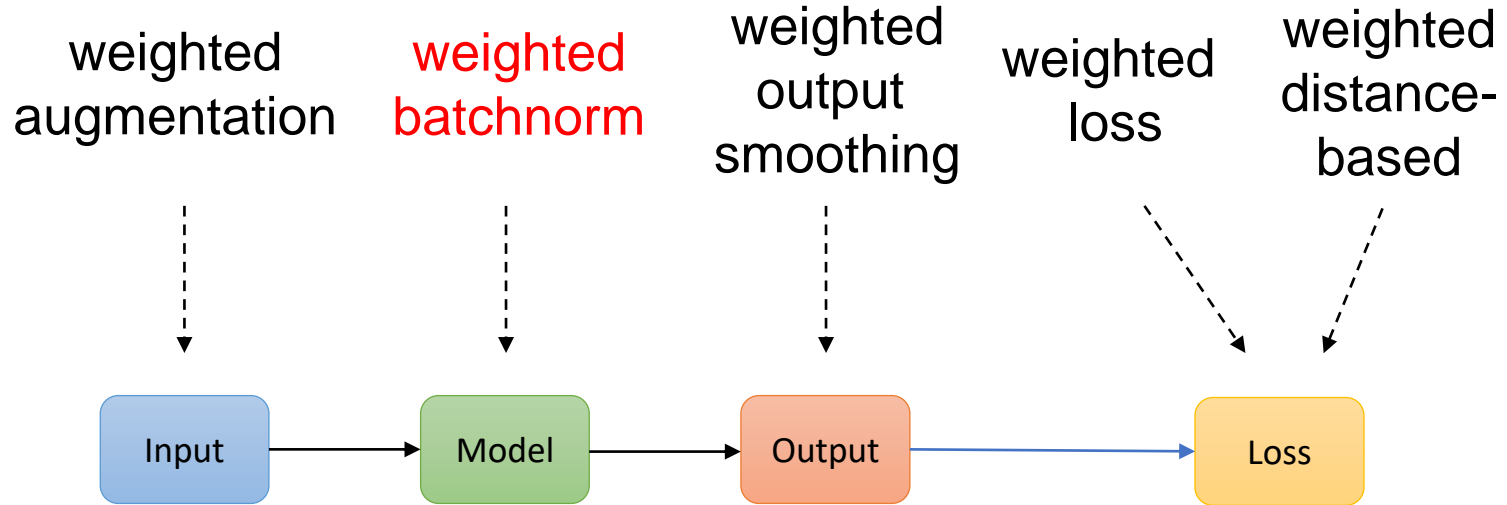
Reduce bias among a triplet

$$\textit{cd}(x, y, z) = |\textit{confusion}(x, z) - \textit{confusion}(y, z)|$$

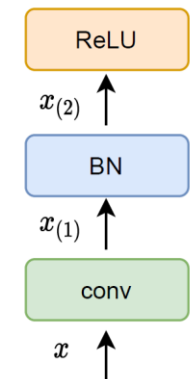
Methodology – Weighted Regularization



Methodology – Weighted Regularization

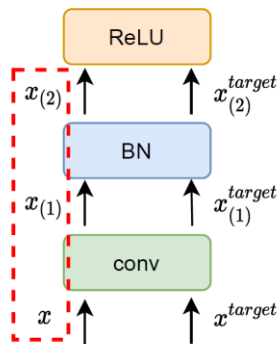


Weighted Batchnorm (w-bn)



(a) Traditional BN

$$x_{(2)} = \frac{x_{(1)} - E[x_{(1)}]}{\sqrt{\text{Var}[x_{(1)}] + \epsilon}} \times \gamma + \beta$$



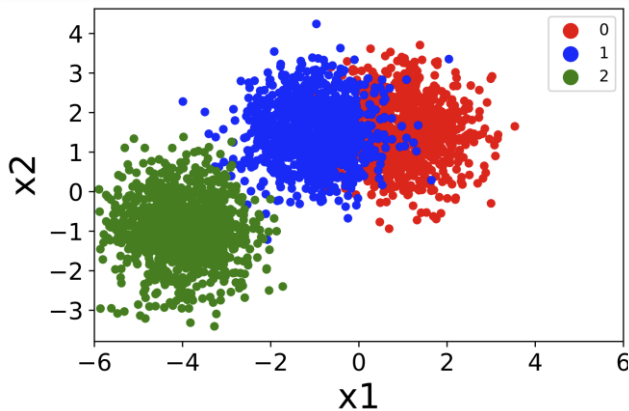
(b) proposed reweighted BN

$$x_{(2)} = \frac{x_{(1)} - \hat{E}[x_{(1)}, x_{(1)}^{target}]}{\sqrt{\hat{\text{Var}}[x_{(1)}, x_{(1)}^{target}] + \epsilon}} \times \gamma + \beta$$

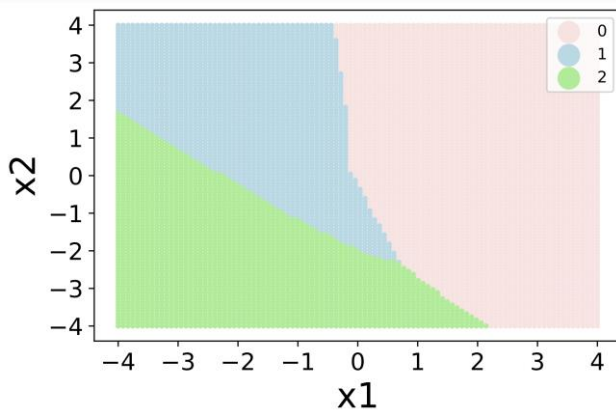
$$\hat{E}[x_{(1)}, x_{(1)}^{target}] := (1 - \rho)E[x_{(1)}] + \rho E[x_{(1)}^{target}]$$

$$\hat{\text{Var}}[x_{(1)}, x_{(1)}^{target}] := (1 - \rho)\text{Var}[x_{(1)}] + \rho\text{Var}[x_{(1)}^{target}]$$

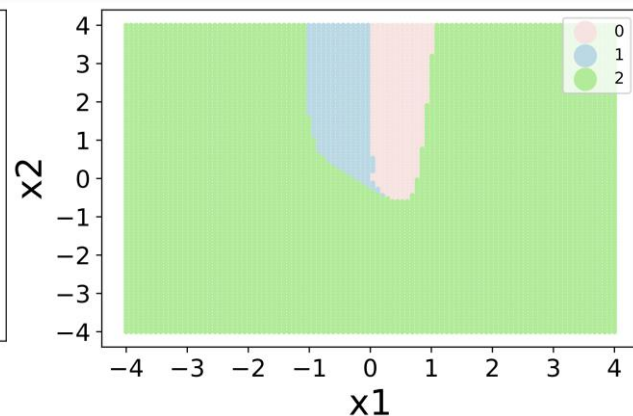
Weighted Batchnorm (w-bn)



Random generated 2d dataset

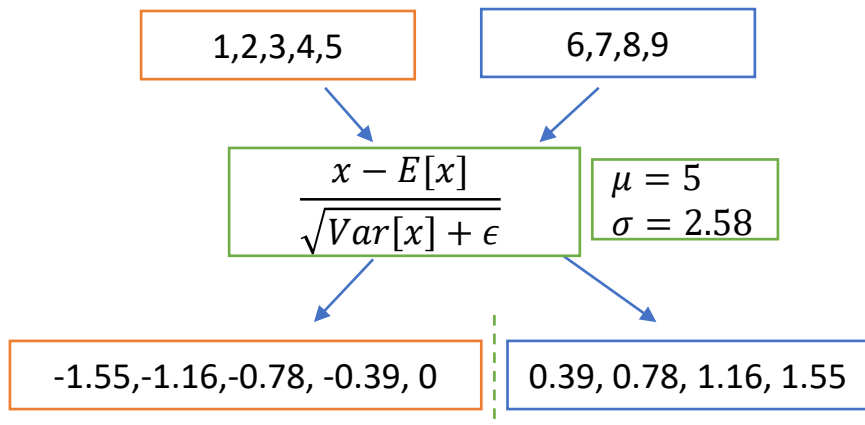


Decision boundary of DNN model with traditional batch normalization

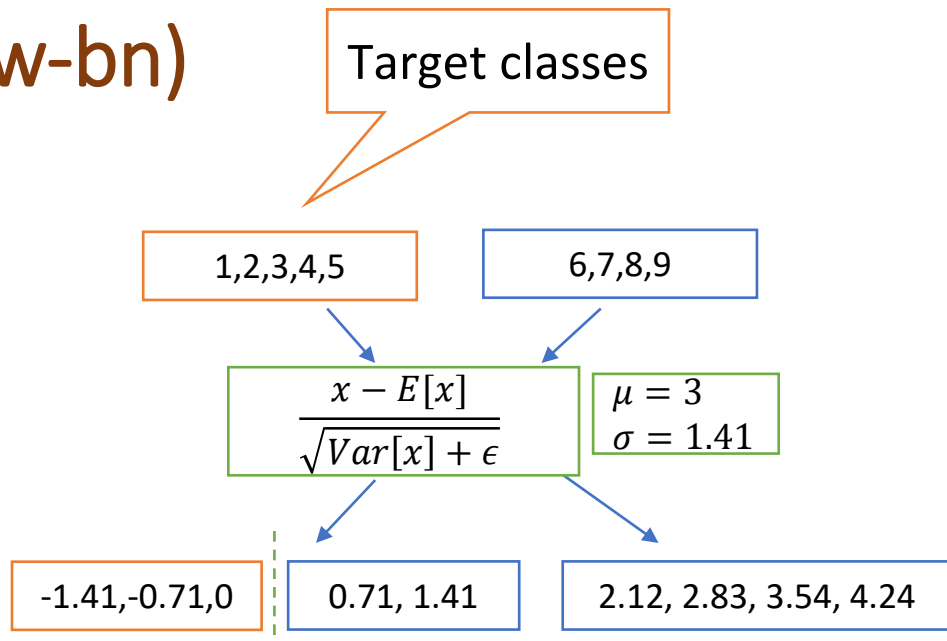


Decision boundary of DNN model with weighted batch normalization

Weighted Batchnorm (w-bn)

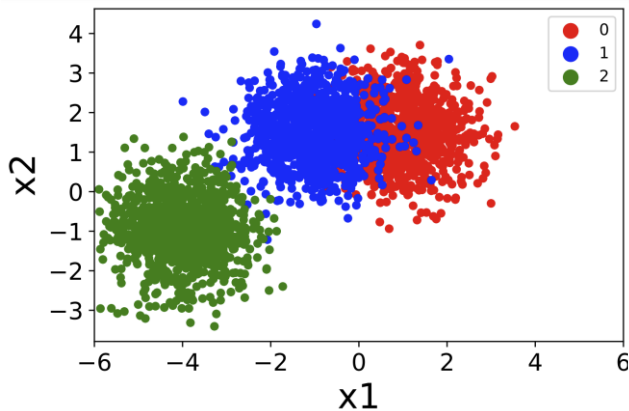


Original batchnorm

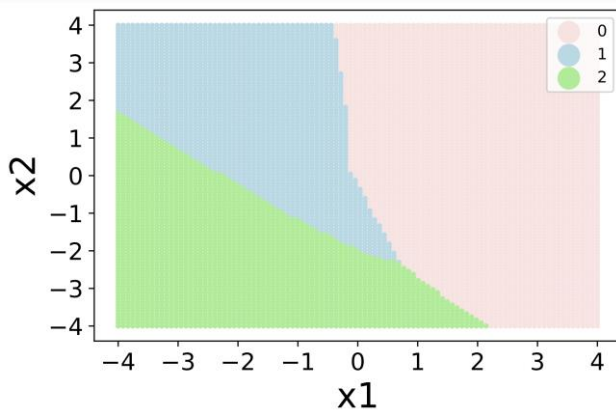


Weighted batchnorm

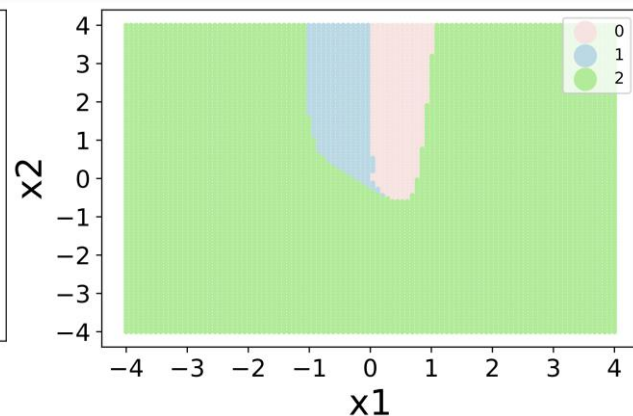
Weighted Batchnorm (w-bn)



Random generated 2d dataset

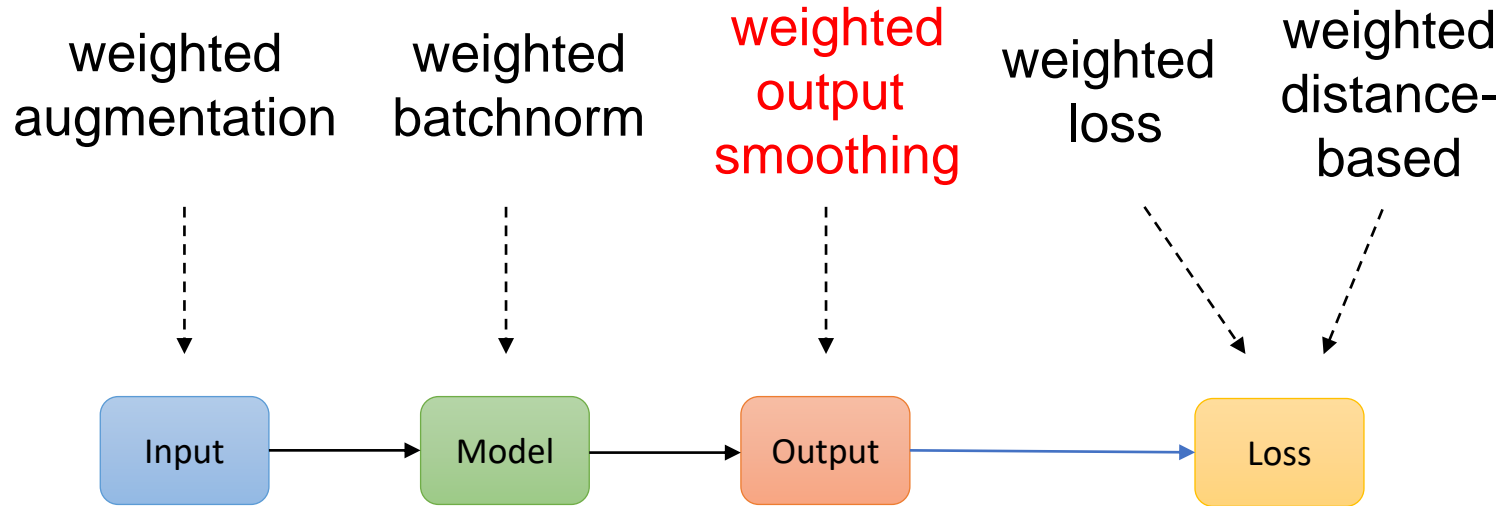


Decision boundary of DNN model with traditional batch normalization

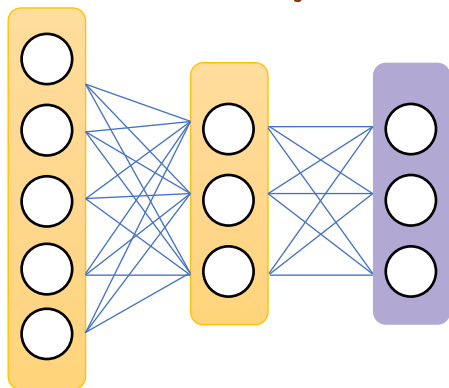


Decision boundary of DNN model with weighted batch normalization

Methodology – Weighted Regularization



Weighted Output Smoothing (w-os)



dog: 0.56

cat: 0.32

bear: 0.12

$\times 0.1$

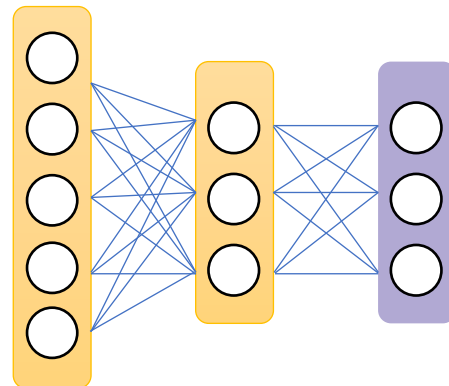
$\times 0.1$



dog: 0.056

cat: 0.032

bear: 0.12



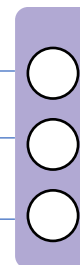
dog: 0.96

cat: 0.03

bear: 0.01

$\times 0.1$

$\times 0.1$

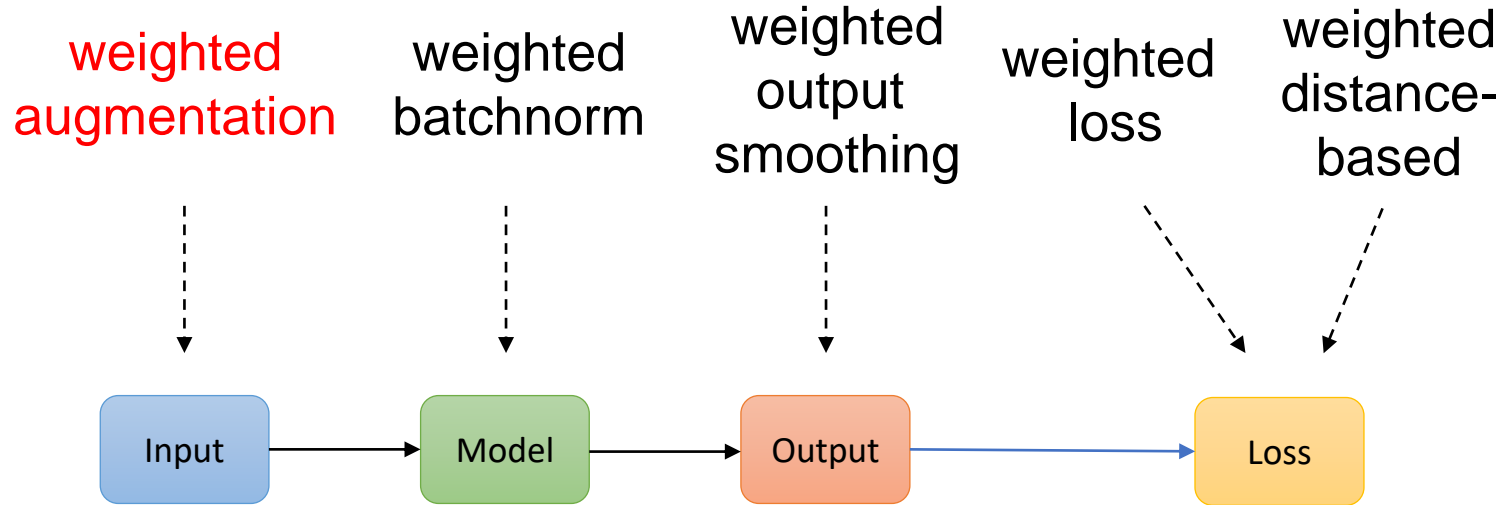


dog: 0.096

cat: 0.003

bear: 0.01

Methodology – Baselines



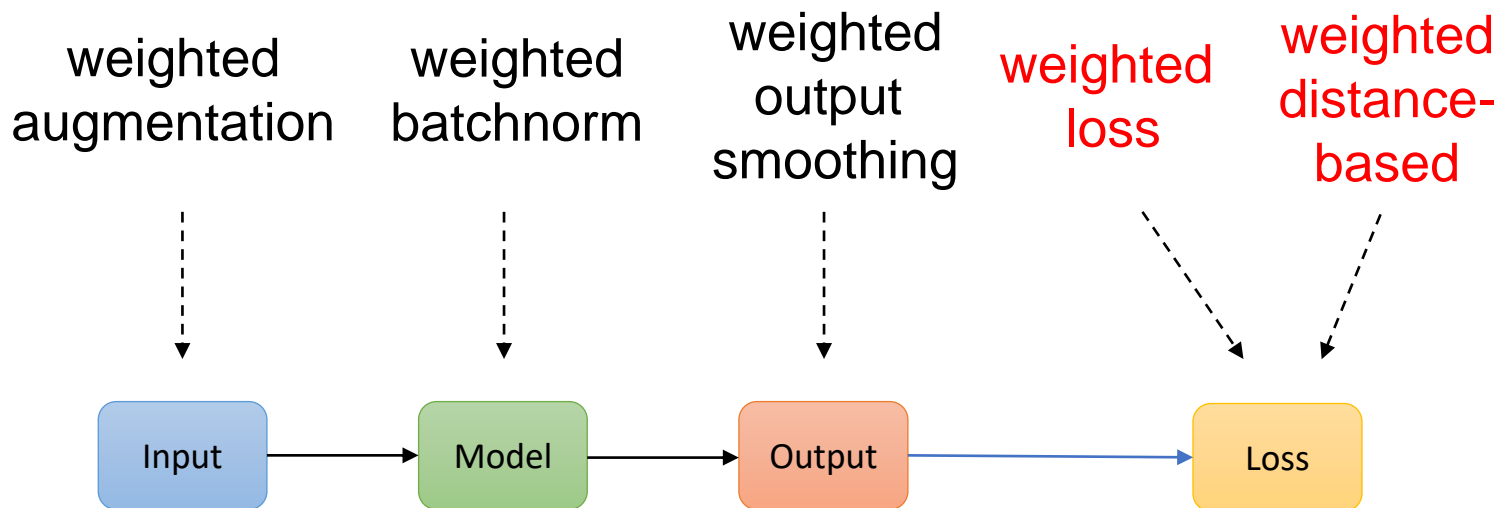
Weighted Augmentation (w-aug)

$$Loss_{orig} = \mathbb{E}_{(x,y) \sim \mathbb{D}} \mathbf{L}(f(x), y)$$

$$Loss_{aug} = \mathbb{E}_{(x,y) \sim \mathbb{D}'} L(f(x), y)$$

$$pdf'(X, Y) = \begin{cases} \rho \times pdf(X, Y), & \text{if } y \in Y_{target} \\ pdf(X, Y), & \text{otherwise} \end{cases}$$

Methodology – Baselines



Weighted Loss (w-loss)

Fix confusion errors:

$$Loss_{wl} = (1 - \rho) \mathbb{E}_{(x,y) \sim \mathbb{D}} \mathbf{L}(f(x), y) + \rho \mathbb{E}_{(x,y) \sim \mathbb{D}(Y_{target})} \mathbf{L}(f(x), y)$$

$$\mathbb{D}(Y_{target}): \quad pdf'(X, Y) = \begin{cases} pdf(X, Y), & \text{if } (x, y) \sim \mathbb{D} \text{ s.t. } y \in Y_{target} \\ & \text{and } f(x) \neq y \text{ and } f(x) \in Y_{target} \\ 0, & \text{otherwise} \end{cases}$$

Fix bias errors:

$$Loss_{wl} = (1 - \rho) \mathbb{E}_{(x,y) \sim \mathbb{D}} \mathbf{L}(f(x), y) + \rho \left(\mathbb{E}_{(x,y) \sim \mathbb{D}(Y_{target+})} \mathbf{L}(f(x), y) + \mathbb{E}_{(x,y) \sim \mathbb{D}(Y_{target-})} \mathbf{L}(f(x), y) \right)$$

Weighted Distance-based Regularization (w-dbr)

$$P(A) = \frac{[S(n_1), S(n_2), \dots S(n_t)]}{N},$$

$S(n_i)$ is the sum of each output of neuron n_i in second to last layer,
given N input images in class A.

$$D(A, B) = \|P_{new}(x), P_{new}(y)\|_2$$

Repair Confusion Errors:

$$Loss_{dbr-conf} = Loss_{orig} - \rho D_{new}(x, y)$$

Repair Bias Errors:

$$Loss_{dbr-bias} = Loss_{orig} + \rho \text{abs}(D_{new}(A, C) - D_{new}(B, C))$$

Study subject

- CIFAR-10:
 - 10 classes; 50,000 training and 10,000 testing images
- CIFAR-100:
 - 100 classes; 50,000 training and 10,000 testing images.
- MS-COCO:
 - 80 objects; 80783 training images and 40504 validation images.
- MS-COCO gender:
 - a subset of MS-COCO dataset with gender information. [Zhao et.al., 2017]

Research Question

1. Performance comparison in repairing single confusing pair
2. Redistribute confusion differently for different approaches.
3. Performance comparison in repairing multiple confusing pairs
4. Performance comparison in reducing a triplet's bias

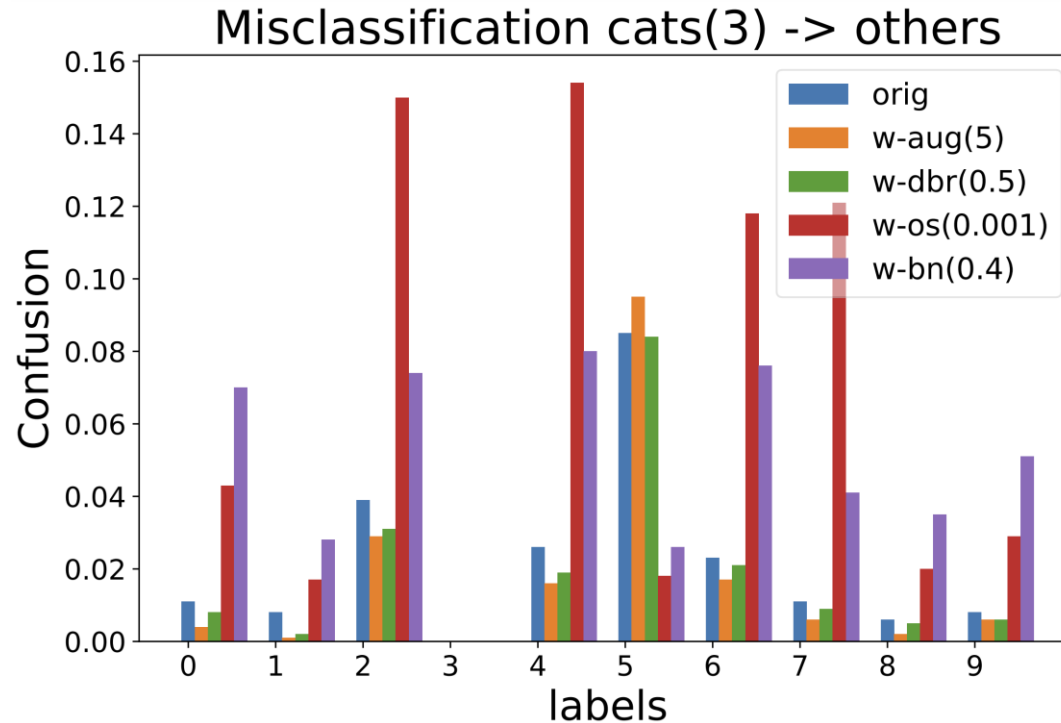
Reduce single confusing pair

Dataset	Target classes	Method	Accuracy	Confusion
COCO gender	Handbag, Woman	orig	0.6691	0.0394
		w-bn(0.4)	0.6689	0.00603
		w-os(0.5)	0.6686	0
		w-loss(0.4)	0.6697	0.02063
		w-aug(3)	0.6679	0.0643
		w-dbr(0.5)	0.6672	0.0676

Reduce single confusing pair

Dataset	Target classes	Method	Accuracy	Confusion
CIFAR-100	Girl, Woman	orig	0.6961	0.15
		w-aug (5)	0.7043	0.12
		w-bn (0.6)	0.6174	0.005
		w-os(0.001)	0.6901	0.01
		w-loss(0.4)	0.543	0.17
		w-dbr(0.1)	0.6628	0.125

Confusion redistributed comparison



Reduce two pairs' confusion

Dataset	Target classes	Method	Accuracy	Confusion
COCO	(Person, Bus), (Mouse, keyboard)	orig	0.6604	0.4025
		w-os (0.5)	0.6575	0
		w-loss (0.6)	0.6603	0.0723
		w-bn(0.6)	0.6574	0.2032
CIFAR-10	(Cat, Dog), (Automobile, Truck)	orig	0.8747	0.134
		w-os (0.1)	0.8628	0.1055
		w-dbr (0.1)	0.8778	0.128
		w-bn(0.6)	0.8328	0.082

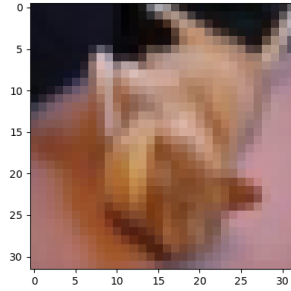
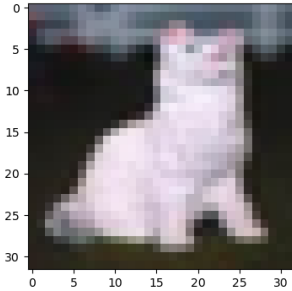
Reduce one triplet's bias

Dataset	Model	Target classes	Method	Accuracy	Bias
COCO gender	ResNet-50	Skis, Woman, Man	orig	0.6691	0.2630
			w-os (0.5)	0.6685	0
			w-loss (0.4)	0.6706	0.02472
			w-aug (3)	0.6648	0.2972
			w-bn (0.6)	0.6645	0.0861
			w-dbr (0.5)	0.6687	0.2606

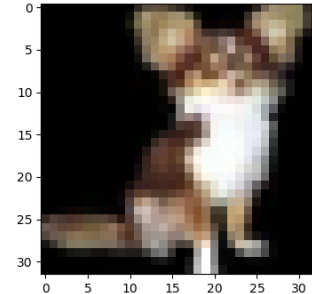
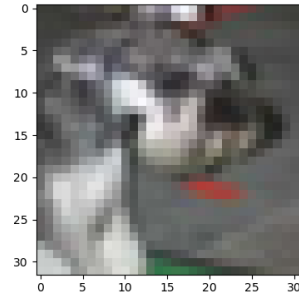
Reduce one triplet's bias

Dataset	Model	Target classes	Method	Accuracy	Bias
CIFAR-100	ResNet-34	Woman, Girl, Boy	orig	0.6961	0.09
			w-aug (5)	0.7059	0.075
			w-bn (0.2)	0.6892	0.040
			w-os (0.001)	0.688	0.01
			w-loss (0.4)	0.5804	0.07
			w-dbr (0.1)	0.6668	0.04

Fixed confusion errors



cat - > dog



dog - > cat

Fixed confusion errors



person - > bus



bus - > person

Fixed bias errors



woman, skis - > man, skis



woman, man, skis - > man, skis

Conclusion

- We proposed five weighted regularization techniques to repair confusion and bias errors. w-os and w-bn works very well in all settings and are able to reduce confusion or bias close to 0.
- w-bn supports fine-tuning with new data to increase accuracy, however it requires batch normalization layer.
- w-loss works better in multi-label classification while w-dbr works better in single-label classification. The reason will be explored in future work.

Outline

- DeepInspect [ICSE' 20]
- Weighted Regularizations [ESEC/FSE' 20 short paper] [12 pages in submission]
- **DeepTest [ICSE' 18]**
- DeepRobust [FASE' 21]

DeepTest: Automated Testing of Deep-Neural-Network-driven Autonomous Cars (ICSE'18)

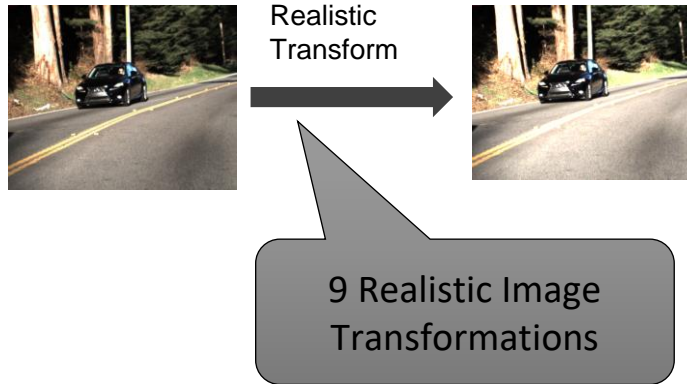


1.1 original



1.2 with added rain

Testing Self-Driving Car's DNN

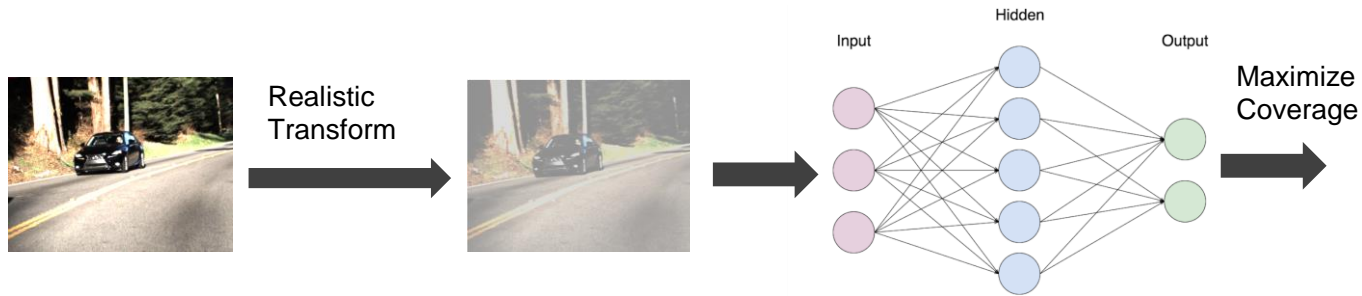


Alpha transformations	Translation, Scale, Shear, Rotation
Linear transformations	Contrast, Brightness
Blurring	Averaging, Gaussian, Median, Bilateral filter
Composite	Rain effect, Fog effect

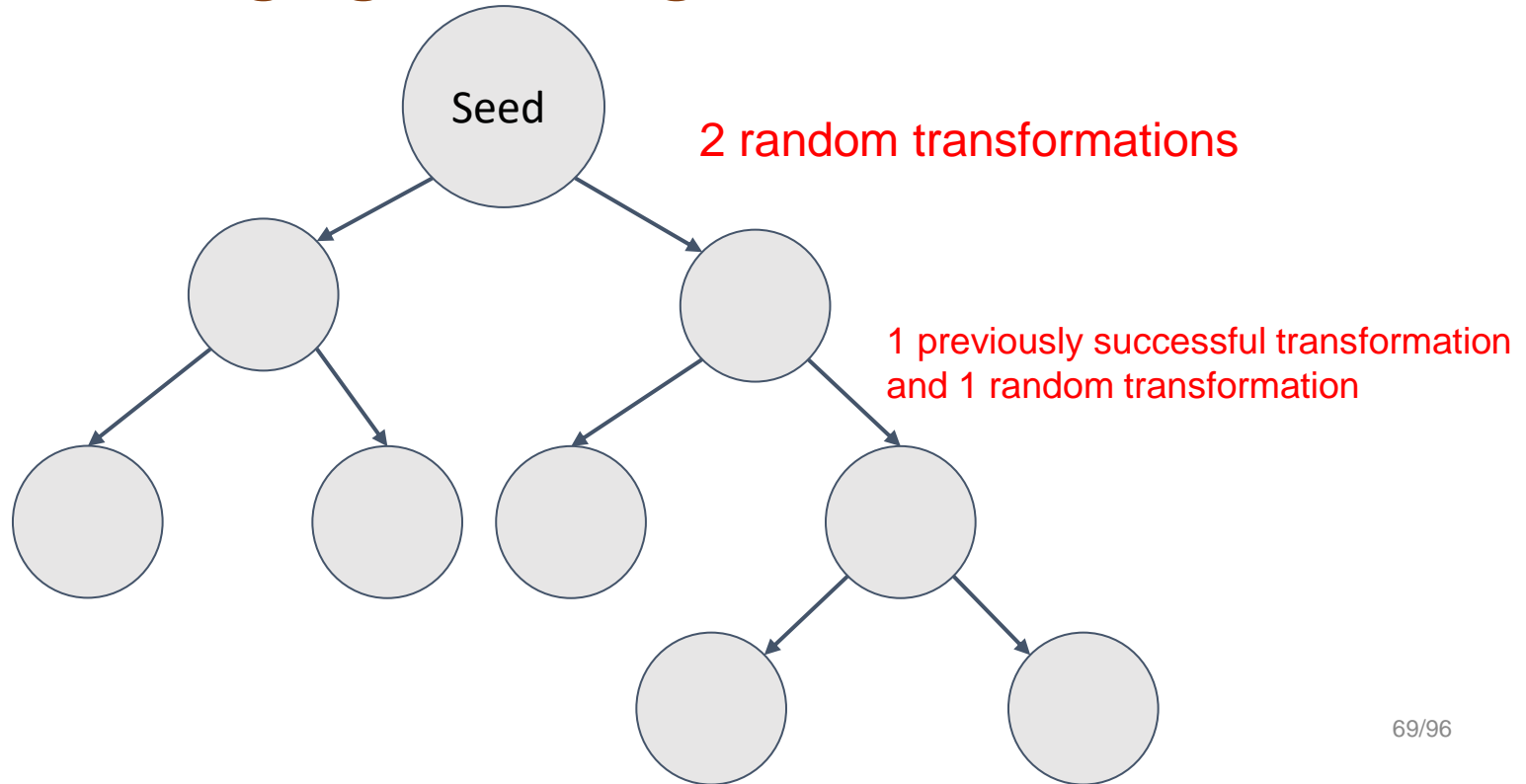
67/96

Testing Self-Driving Car's DNN

$$\text{Neuron Coverage} = \frac{|\text{Activated Neurons}|}{|\text{Total Neurons}|} \text{ [Pei et. al]}$$



Neuron coverage guided algorithm



Detection of erroneous behaviors

Original images: $\{I_{01}, I_{02}, \dots, I_{0n}\}$

Predicted outputs: $\{\theta_{01}, \theta_{02}, \dots, \theta_{0n}\}$

Original labels: $\{\theta_1, \theta_2, \dots, \theta_n\}$

Mean square error: $MSE_{orig} = 1/n \cdot \sum_{(i=1)}^n (\theta_i - \theta_{0i})^2$

Transformed images: $\{I_{t1}, I_{t2}, \dots, I_{tn}\}$

Predicted outputs: $\{\theta_{t1}, \theta_{t2}, \dots, \theta_{tn}\}$

Statistical Metamorphic relation: $(\theta_i - \theta_{ti})^2 \leq \lambda MSE_{orig}$

An error occurs when this relation is violated

Detected Erroneous Behaviour



Scale:1.3; Translation:74; Gaussian blur:3x3;Brightness:85;

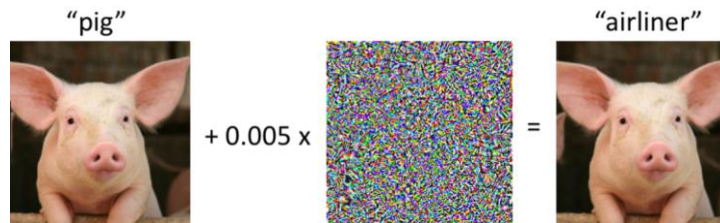
71/96

Outline

- DeepInspect [ICSE' 20]
- Weighted Regularizations [ESEC/FSE' 20 short paper] [12 pages in submission]
- DeepTest [ICSE' 18]
- **DeepRobust [FASE' 21]**

Robustness Problem

Norm-bounded perturbation



Physical Adversarial



Natural Variations (this work)



Robustness varies across Images



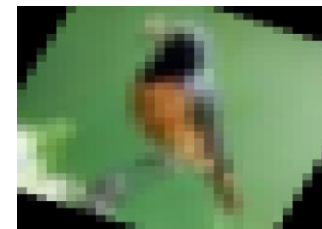
bird



bird



bird



bird



dog



bird



airplane



cat

A well-trained DNN performs well on variations of some images
but not others

74/96

Robustness varies across Images



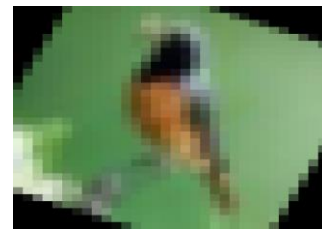
bird



bird



bird



bird



dog



bird



airplane



cat

We call DNN's performance on individual image **local robustness**, as opposed to robustness of the whole dataset, which is well studied in previous work^{75/96}

Problem to solve

Goal: automatically identify non-robust (a.k.a. weak) images from unlabeled test data

Implication: prioritizing labeling / testing, model fixing



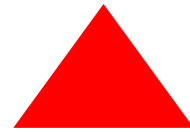
- How to define such vulnerable test cases?
- What properties they have?
- How to use the properties to identify them effectively?

Local Robustness

- **Original Data Point:** an original un-modified data instance (image in our case)



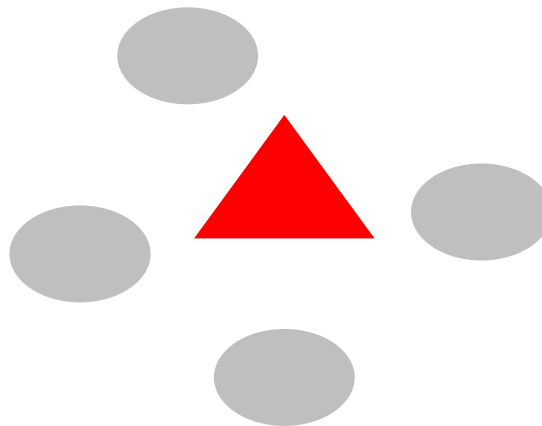
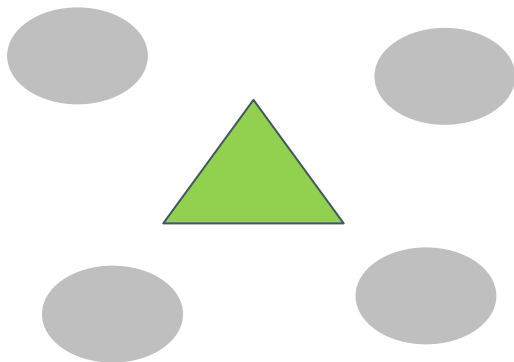
Correctly Classified



Misclassified

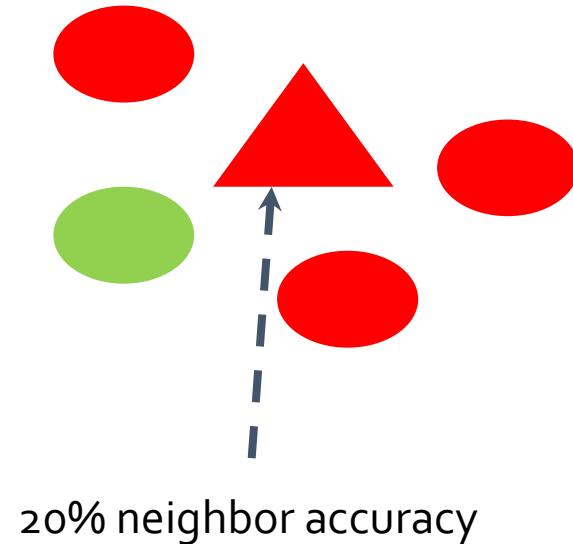
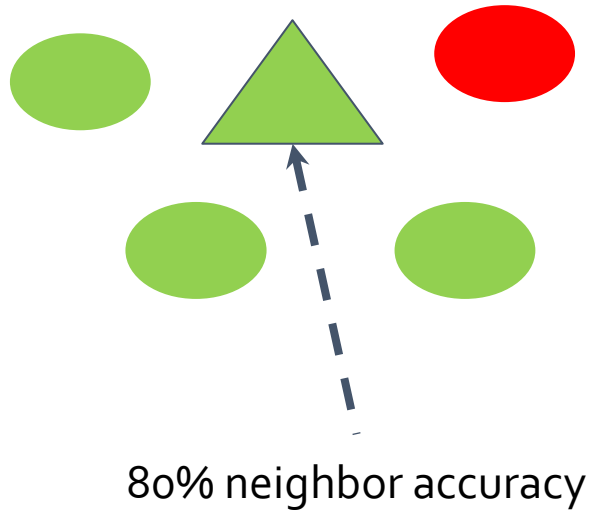
Local Robustness

- **Neighbors:** natural variations of the original data point



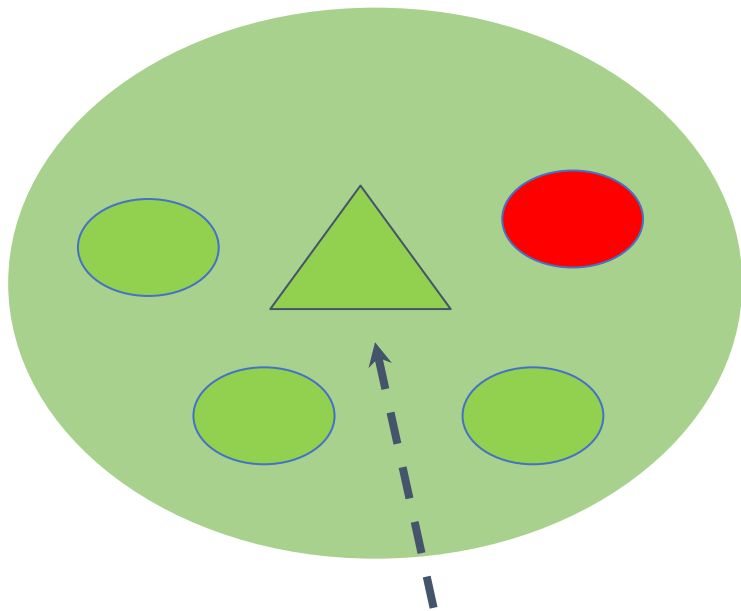
Local Robustness

- **Neighbor Accuracy:** percentage of its neighbors (including itself) that can be correctly classified

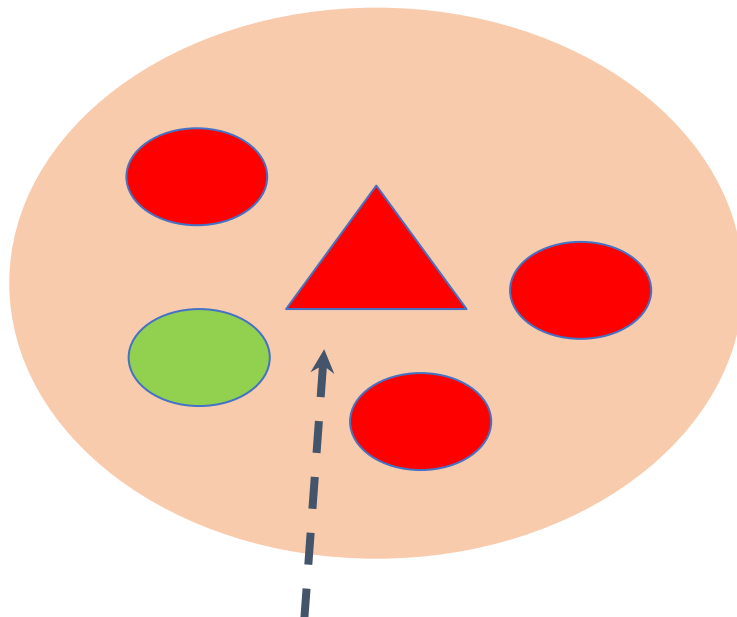


Local Robustness

- **Weak Point:** neighbor accuracy < a user-specified threshold (50% in this example)



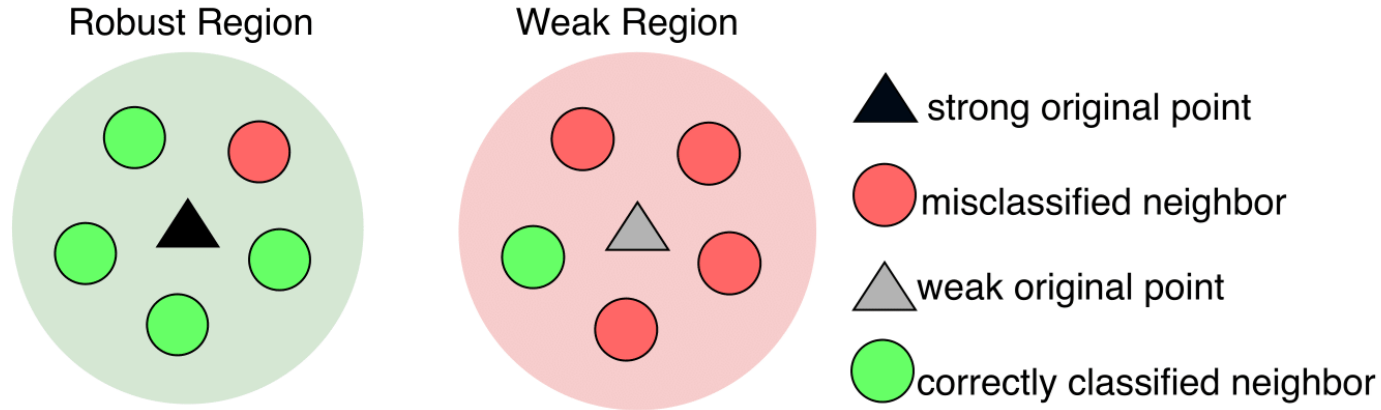
Strong(a.k.a Robust) Point(80%>50%)



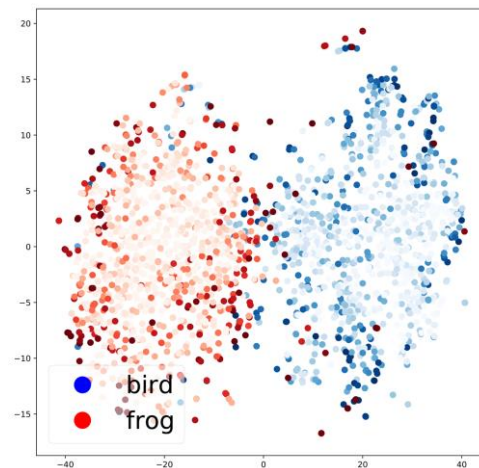
Weak Point(20%<50%)

80/96

Properties of Robust vs. Weak Points



Property-1: weak points are concentrated towards the class decision boundaries

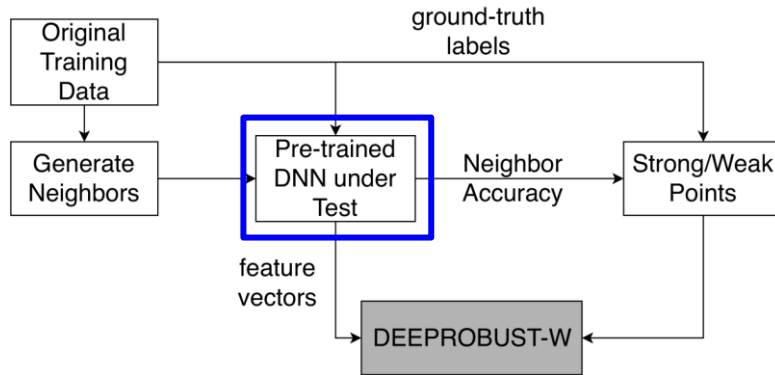


t-SNE graphs of two randomly sampled classes with darker color denoting lower neighbor accuracy

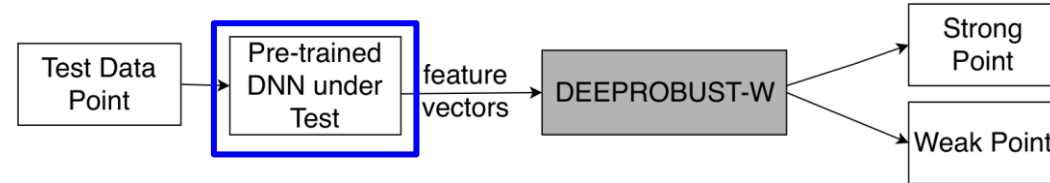
Implication: Weak points and robust points can be separated in the representation space

Property1 -> A White-box Method: DeepRobust-W

Training Stage:



Inference Stage:



Pros:

- Fast Inference (only query once)

Cons:

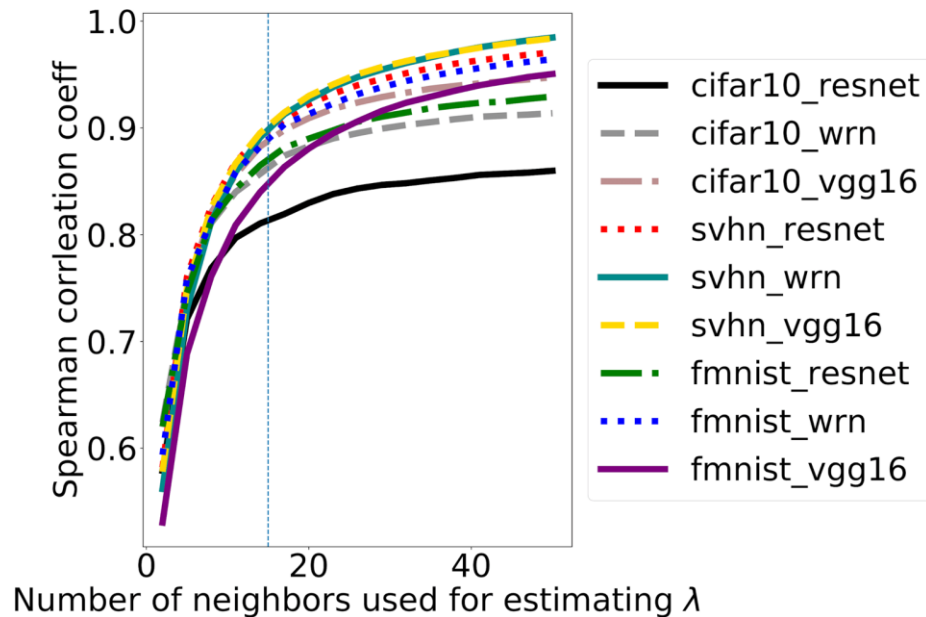
- Need to access the DNN internal activation values
- Need training

/96

Property2: weak points have more diverse neighbor predictions

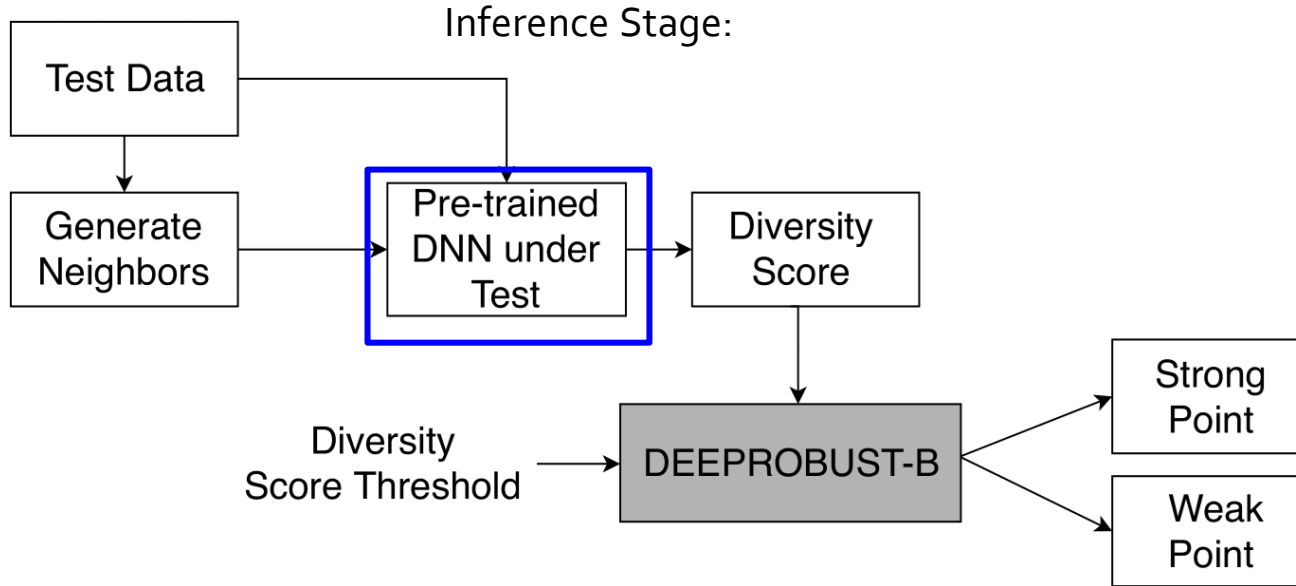
Simpson Diversity Index $\lambda = \sum_{i=1}^k p_i^2$

k: number of possible classes
p_i: the probability of neighbors
predicted to class i



Implication: diversity index (when # neighbors ≥ 15) is a good proxy for estimating neighbor accuracy

Property2 -> A Black-box Method: DeepRobust-B



Pros:

- No training needed
- Treating the DNN as a black-box

Cons:

- Need to query multiple times

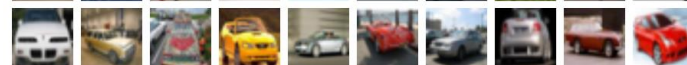
Experiment 1: Image Classification

- Datasets: CIFAR-10, SVHN, F-MNIST
- Model: ResN, WRN, VGG
- Methods:
 - DeepRobust-W,
 - DeepRobust-B,
 - Baselines: random, top1
- Natural Variations: rotation, shifting
- Neighbor Accuracy Thresholds: 0.75, 0.5

airplane



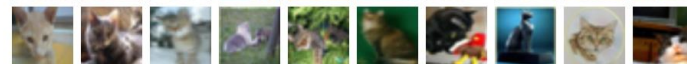
automobile



bird



cat



deer



dog



DeepRobust-W Result

Results on CIFAR-10 + ResNet + 0.5 Neighbor Accuracy Threshold

	F1	True-Positive	False-Positive
DeepRobust-W	0.79	3844	764
Top1	0.376	1218	206
Random	0.488	2372	2236

DeepRobust-W has much better performance than the baseline methods (Similar results hold for other settings)

87/96

DeepRobust-B Result - Classification

Results on CIFAR-10 + ResNet + 0.5 Neighbor Accuracy Threshold

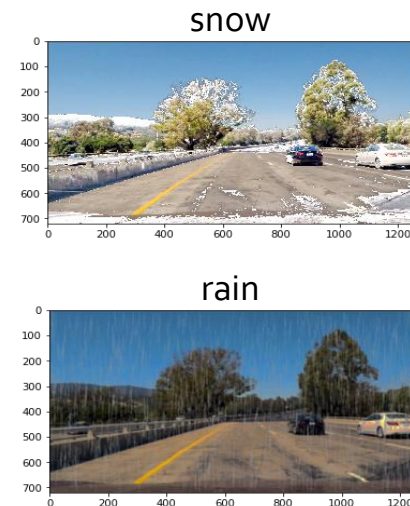
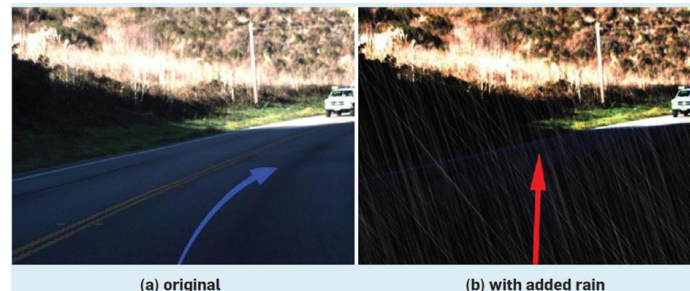
	f1	tp	fp
ours	0.939	4714	257
top1	0.376	1218	206
random	0.501	2516	2455

DeepRobust-B has much better performance than the baseline methods (Similar results hold for other settings)

88/96

Experiment 2: Steering Angle Prediction

- Methods: DeepRobust-W, random
- Natural Variations: rain, snow
- Datasets: Driving data from Udacity Simulator
- Model: NVIDIA DAVE-2, Epoch, and Chauffeur
- Neighbor Accuracy Thresholds: 0.75, 0.5

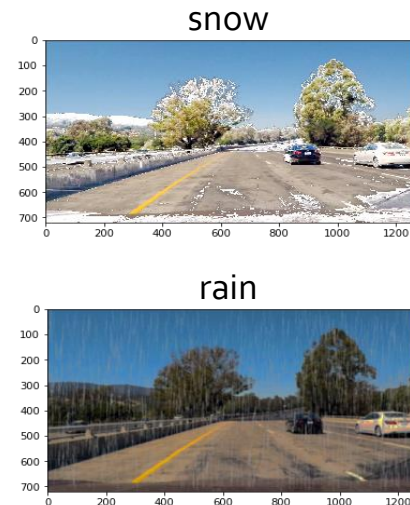
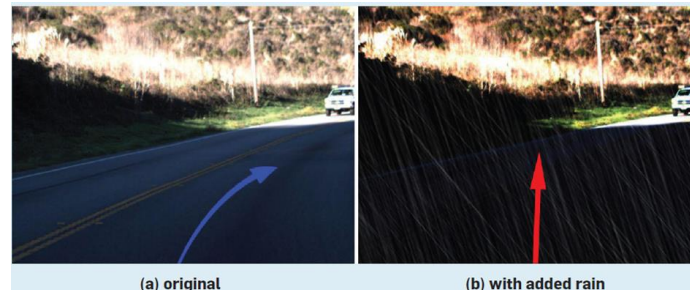


89/96

89

Experiment 2: Steering Angle Prediction

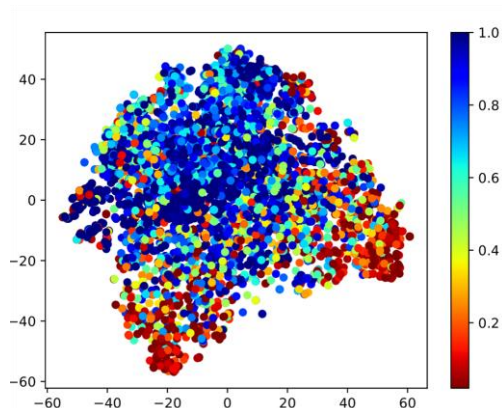
- Methods: DeepRobust-W, random
- Natural Variations: rain, snow
- Tasks: Regression task
- Neighbor correctness: $|\theta_{o1} - \theta_1|^2 < \lambda MSE$



90/96

90

DeepRobust-W Result



Finding: Many correctly predicted original images have low neighbor accuracies and they tend to cluster together.

Results with 0.5 Neighbor Accuracy Threshold

	f1	tp	fp
DeepRobust-W	0.789	4354	1112
random	0.586	3234	2232

Similar results hold for other settings.

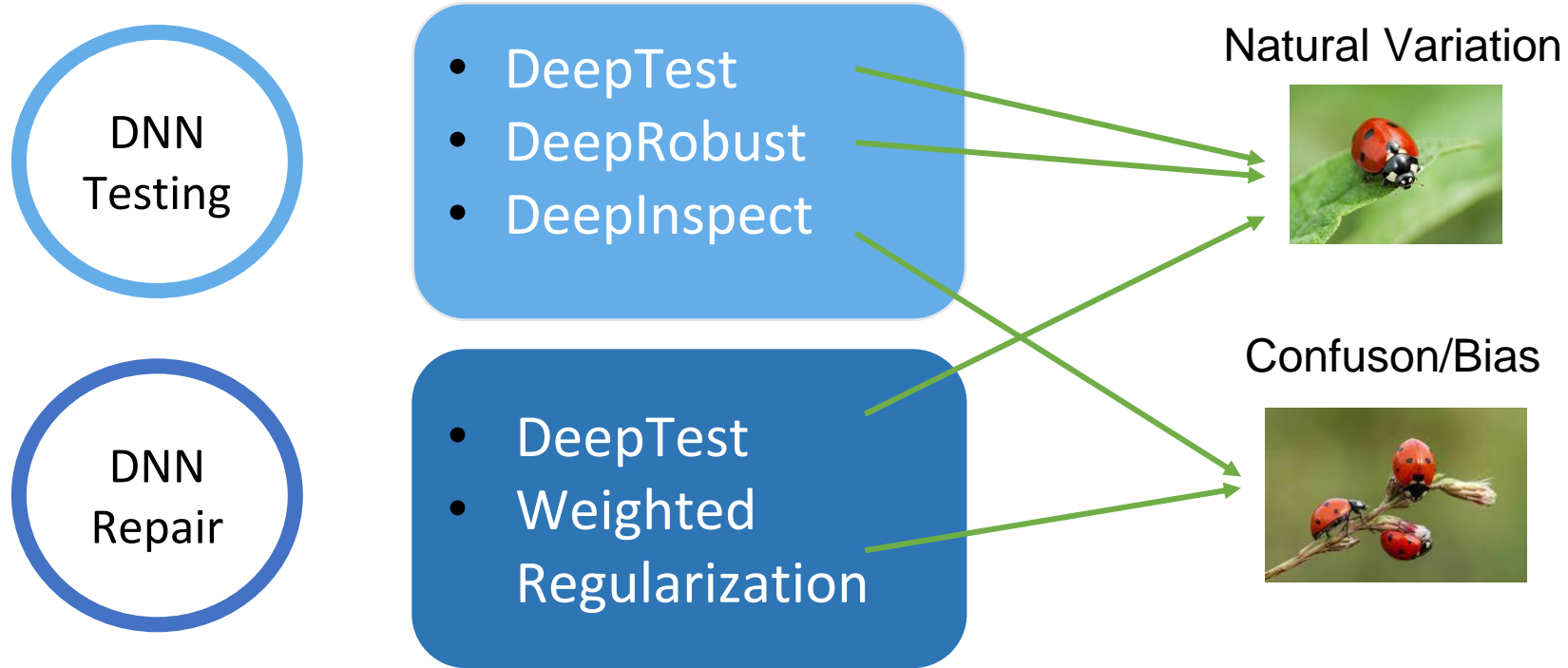
Future Work

- Other Natural Variations (and potentially extend to other types of perturbations)
- Other applications (e.g. NLP, medical imaging)
- Downstream Applications: Prioritizing Testing Cases, Model Fixing

Conclusion

- We conduct an empirical study to understand the local robustness properties of DNNs under natural variations.
- We implement a white-box and a black-box method that can effectively identify weak points.
- Our black-box model has higher performance than white-box approach. However, our white-box model can be applied to regression tasks such as self-driving cars and it can effectively identify those weak points in real time.

Contributions



Contributions & Accomplishments

1. DeepTest [ICSE' 18]

- [Tian, Pei, Jana, Ray]

2. DeepRobust [FASE' 21]

- [Zhong, Tian, Ray]

3. DeepInspect[ICSE' 20]

- [Tian, Zhong, Ordonez, Kaiser, Ray]

4. WR [ESEC/FSE' 20 short paper] [12 pages in submission]

- [Tian, Zhong, Sweeney, Ordonez, Ray]

Thank you!