PIGEON: Flying Beyond Strong Neuron Activation Coverage

https://github.com/Kangjuheon/project_pigeon

Juheon Kang (G202548001)
2715wngjs@uos.ac.kr

Hyunseo Shin (G202448003)
hseo98@uos.ac.kr

Eunkyung Choi (G202448018)
rmarud202@uos.ac.kr

Github, PPT

WTS

SNAC를 개선한 Coverage “PIGEON”을 고안했고,

강건성과의 대표적인 지표들과의 상관관계 비교에서

유의미한 진척을 확인했습니다.

1. SNAC Review

$$SNAC = \frac{|UCN|}{|N|}$$

- Strong Neuron Activation Coverage
- |N| : # neurons
- |UCN| : # UpperCornerNeuron = # training max 넘은 testing

1. SNAC Review

Comparison of coverage metrics

Coverage Metrics (%)	LeNet-1	LeNet-4	LeNet-5	ResNet20	TinyTaxiNet
KMNC (K: 10)	95.00	85.29	90.70	98.75	62.35
KMNC (K: 1000)	60.23	54.33	59.10	71.16	43.54
TKNC (K: 10)	88.57	81.59	82.40	65.09	52.06
TKNC (K: 1000)	1.00	3.27	4.93	3.90	0.59
NBC	0.87	0.66	0.58	5.55	2.01
SNAC	0.87	1.05	1.16	6.46	3.21
NC (Threshold: 0.00)	100.00	90.58	96.12	100.00	67.65
NC (Threshold: 0.20)	61.90	76.09	85.66	100.00	67.65
NC (Threshold: 0.50)	30.95	63.77	74.03	99.16	61.76
NC (Threshold: 0.75)	23.81	59.42	67.05	13.99	52.94
MC/DC (Sign-Sign)	5.77	44.17	10.62	24.66	27.60
MC/DC (Sign-Value)	63.46	35.68	8.40	51.58	28.65
MC/DC (Value-Sign)	23.08	58.20	13.71	52.20	41.67
MC/DC (Value-Value)	100.00	67.21	15.23	99.63	46.88

- NC(0) easiest to achieve, low coverage of NBC and SNAC (same train and test distributions), MC/DC hard to achieve (more constraints).

- 수업시간에 나온 coverage 수치 비교

1. SNAC Review

Comparison of coverage metrics

Coverage Metrics (%)	LeNet-1	LeNet-4	LeNet-5	ResNet20	TinyTaxiNet
KMNC (K: 10)	95.00	85.29	90.70	98.75	62.35
KMNC (K: 1000)	60.23	54.33	59.10	71.16	43.54
TKNC (K: 10)	88.57	81.59	82.40	65.09	52.06
TKNC (K: 1000)	1.00	3.27	4.93	3.90	0.59
NBC	0.87	0.66	0.58	5.55	2.01
SNAC	0.87	1.05	1.16	6.46	3.21
NC (Threshold: 0.00)	100.00	90.58	96.12	100.00	67.65
NC (Threshold: 0.20)	61.90	76.09	85.66	100.00	67.65
NC (Threshold: 0.50)	30.95	63.77	74.03	99.16	61.76
NC (Threshold: 0.75)	23.81	59.42	67.05	13.99	52.94
MC/DC (Sign-Sign)	5.77	44.17	10.62	24.66	27.60
MC/DC (Sign-Value)	63.46	35.68	8.40	51.58	28.65
MC/DC (Value-Sign)	23.08	58.20	13.71	52.20	41.67
MC/DC (Value-Value)	100.00	67.21	15.23	99.63	46.88

- NC(0) easiest to achieve, low coverage of NBC and SNAC (same train and test distributions), MC/DC hard to achieve (more constraints).

- 백분율임에도, SNAC의 값은 매우 낮았습니다.

1. SNAC Review

Comparison of coverage metrics

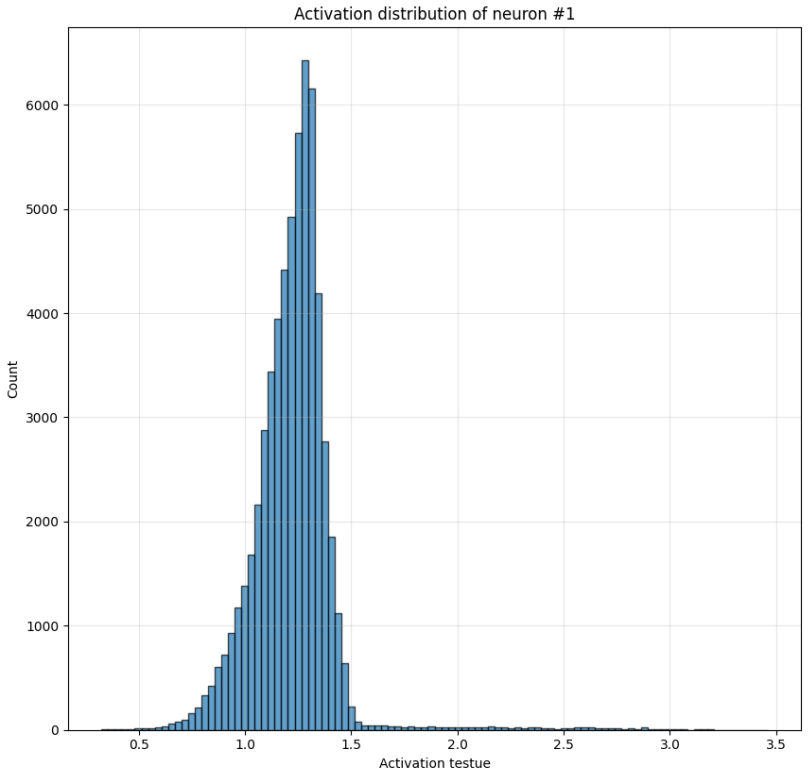
Coverage Metrics (%)	LeNet-1	LeNet-4	LeNet-5	ResNet20	TinyTaxiNet
KMNC (K: 10)	95.00	85.29	90.70	98.75	62.35
KMNC (K: 1000)	60.23	54.33	59.10	71.16	43.54
TKNC (K: 10)	88.57	81.59	82.40	65.09	52.06
TKNC (K: 1000)	1.00	3.27	4.93	3.90	0.59
NBC	0.87	0.66	0.58	5.55	2.01
SNAC	0.87	1.05	1.16	6.46	3.21
NC (Threshold: 0.00)	100.00	90.58	96.12	100.00	67.65
NC (Threshold: 0.20)	61.90	76.09	85.66	100.00	67.65
NC (Threshold: 0.50)	30.95	63.77	74.03	99.16	61.76
NC (Threshold: 0.75)	23.81	59.42	67.05	13.99	52.94
MC/DC (Sign-Sign)	5.77	44.17	10.62	24.66	27.60
MC/DC (Sign-Value)	63.46	35.68	8.40	51.58	28.65
MC/DC (Value-Sign)	23.08	58.20	13.71	52.20	41.67
MC/DC (Value-Value)	100.00	67.21	15.23	99.63	46.88

- NC(0) easiest to achieve, low coverage of NBC and SNAC (same train and test distributions), MC/DC hard to achieve (more constraints).

- 같은 분포로부터 나왔기 때문이라고 하는데요,
- SNAC는 분포로부터 나오는 값일까요?

2. Problem

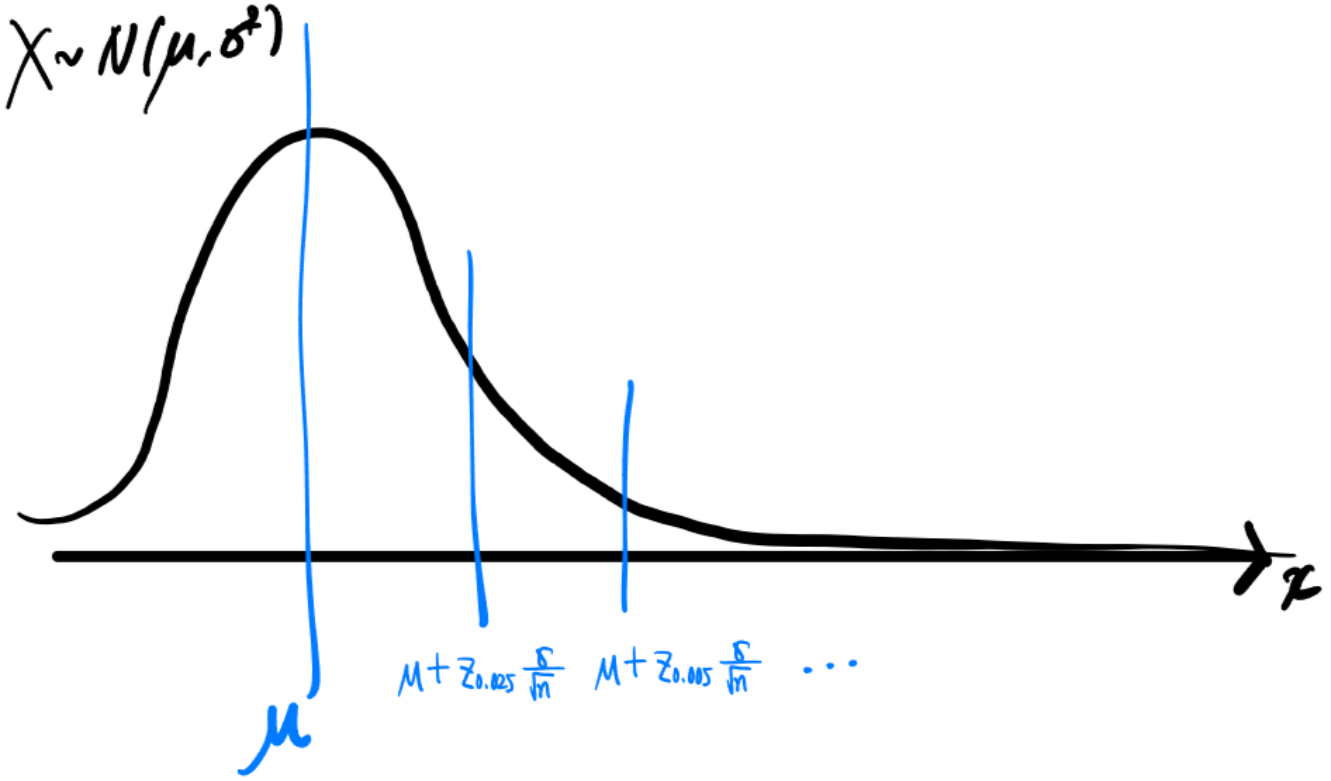
- 먼저, activation 값들이 따르는 분포가 있을 것이고, 정규분포라고 가정해보자.



실제 ResNet18의
First neuron;Second Layer
출력 분포

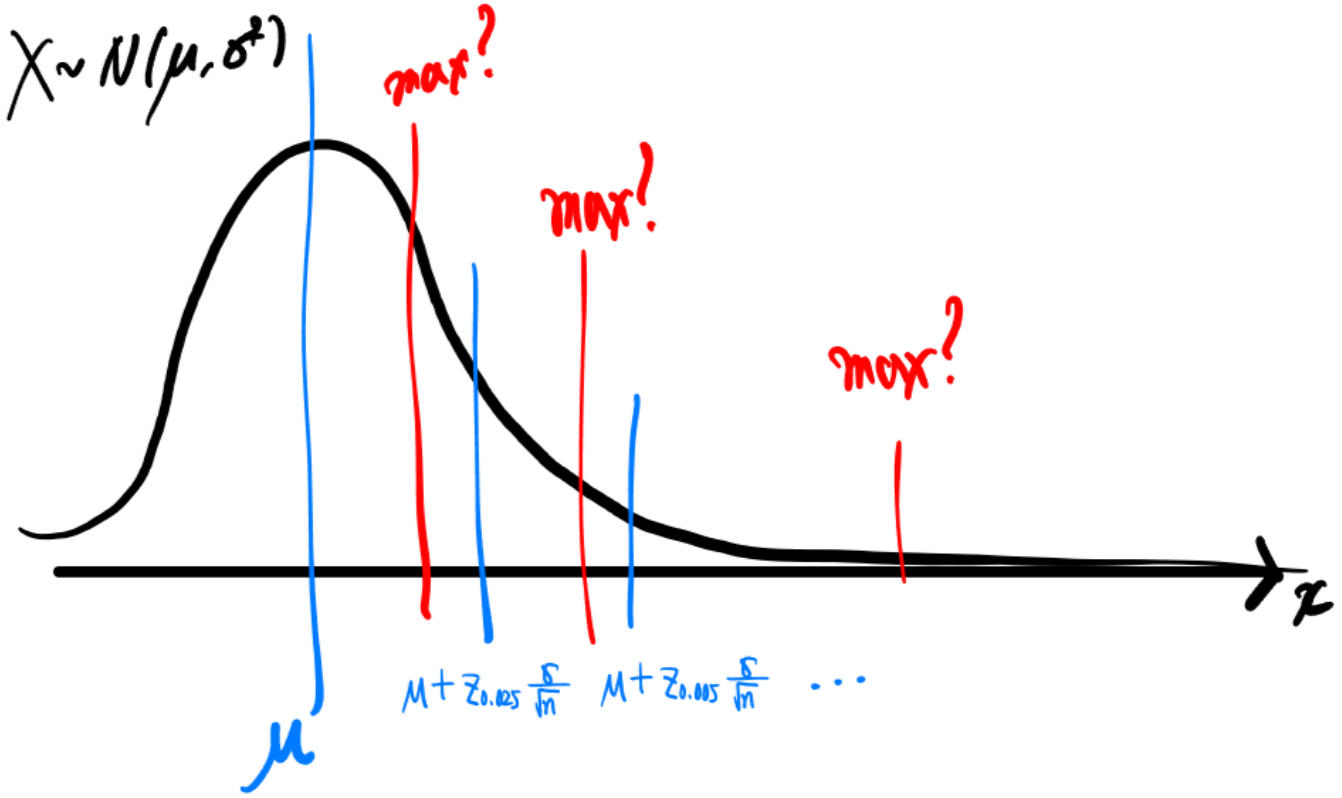
2. Problem

- 먼저, activation 값들이 따르는 분포가 있을 것이고, 정규분포라고 가정해보자.



2. Problem

- 먼저, activation 값들이 따르는 분포가 있을 것이고, 정규분포라고 가정해보자.



2. Problem

- 먼저, activation 값들이 따르는 분포가 있을 것이고, 정규분포라고 가정해보자.

- Max는 이상치(outlier)의 유력한 후보이다.

2. Problem

- 먼저, activation 값들이 따르는 분포가
있을 것이고, 정규분포라고 가정해보자.

- Max는 이상치(outlier)의 유력한 후보이다.

- 재수가 좋으면 적절한 기준치가 되지만,
- 재수가 나쁘면 달성하기 쉬운 값, 혹은 너무 어려운 값이 될 수 있다.

2. Problem

- 먼저, activation 값들이 따르는 분포가 있을 것이고, 정규분포라고 가정해보자.

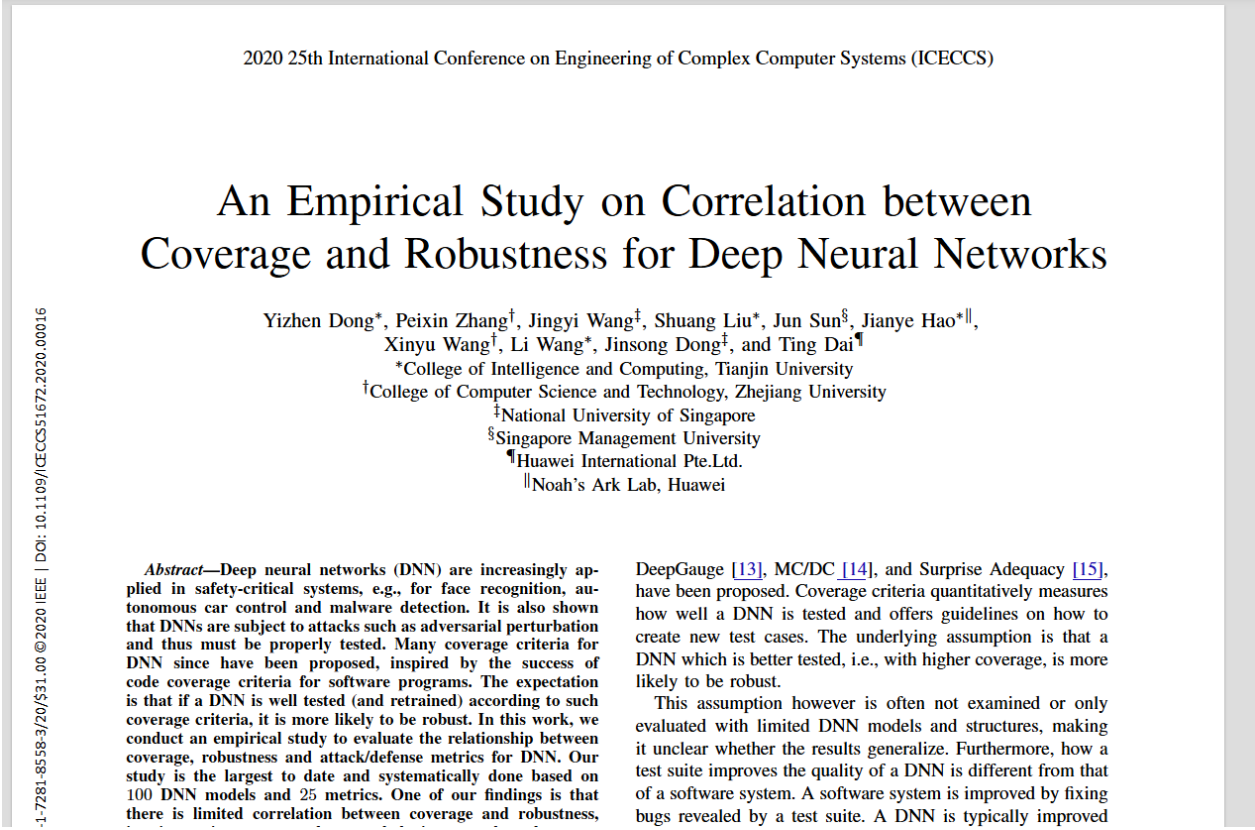
- Max는 이상치(outlier)의 유력한 후보이다.

- 재수가 좋으면 적절한 기준치가 되지만,
- 재수가 나쁘면 달성하기 쉬운 값, 혹은 너무 어려운 값이 될 수 있다.

기존의 SNAC의 기준치는 Train Dataset의 분포로부터 나오기보다는
Train Data 중 특정 데이터 하나, 특히 outlier에 지배적인 영향을 받는다.

2. Problem

- Dong et al. (2020)에서 Coverage들과 Robustness 사이의 상관관계를 분석함.



2020 25th International Conference on Engineering of Complex Computer Systems (ICECCS)

An Empirical Study on Correlation between Coverage and Robustness for Deep Neural Networks

Yizhen Dong*, Peixin Zhang†, Jingyi Wang‡, Shuang Liu*, Jun Sun§, Jianye Hao*‖,
Xinyu Wang†, Li Wang*, Jinsong Dong‡, and Ting Dai¶
*College of Intelligence and Computing, Tianjin University
†College of Computer Science and Technology, Zhejiang University
‡National University of Singapore
§Singapore Management University
¶Huawei International Pte.Ltd.
‖Noah's Ark Lab, Huawei

-1-7281-8558-3/20/$31.00 ©2020 IEEE | DOI: 10.1109/ICECCS51672.2020.00016

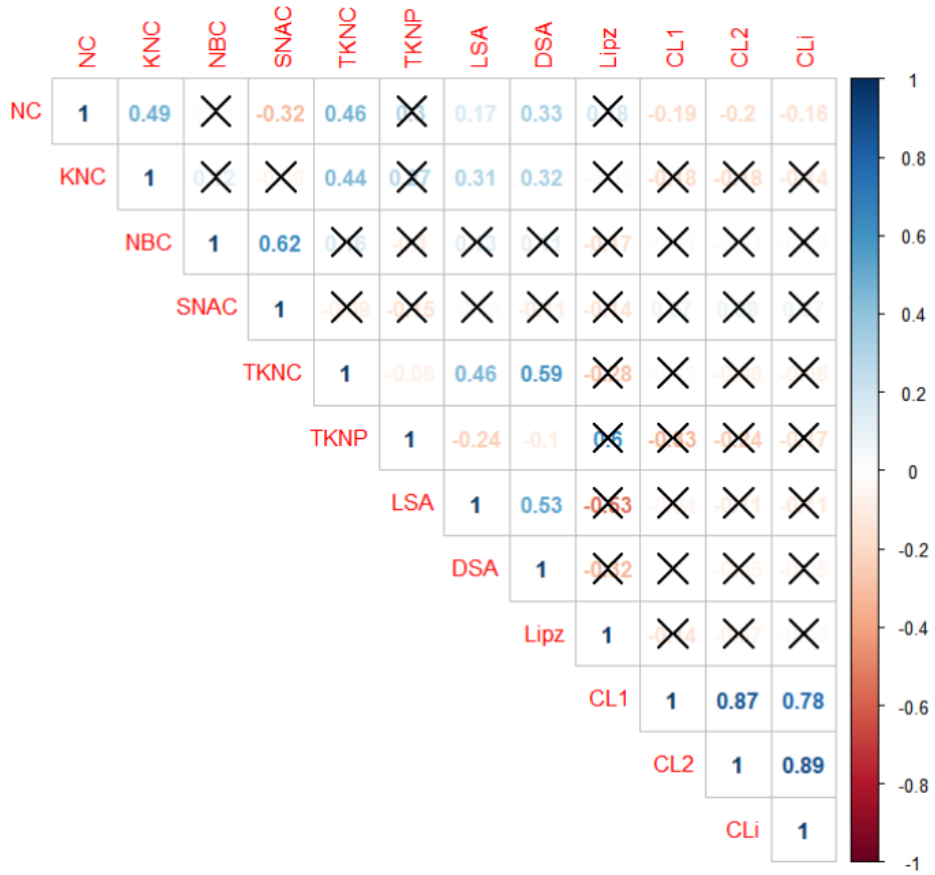
***Abstract*—Deep neural networks (DNN) are increasingly applied in safety-critical systems, e.g., for face recognition, autonomous car control and malware detection. It is also shown that DNNs are subject to attacks such as adversarial perturbation and thus must be properly tested. Many coverage criteria for DNN since have been proposed, inspired by the success of code coverage criteria for software programs. The expectation is that if a DNN is well tested (and retrained) according to such coverage criteria, it is more likely to be robust. In this work, we conduct an empirical study to evaluate the relationship between coverage, robustness and attack/defense metrics for DNN. Our study is the largest to date and systematically done based on 100 DNN models and 25 metrics. One of our findings is that there is limited correlation between coverage and robustness,**

DeepGauge [13], MC/DC [14], and Surprise Adequacy [15], have been proposed. Coverage criteria quantitatively measures how well a DNN is tested and offers guidelines on how to create new test cases. The underlying assumption is that a DNN which is better tested, i.e., with higher coverage, is more likely to be robust.

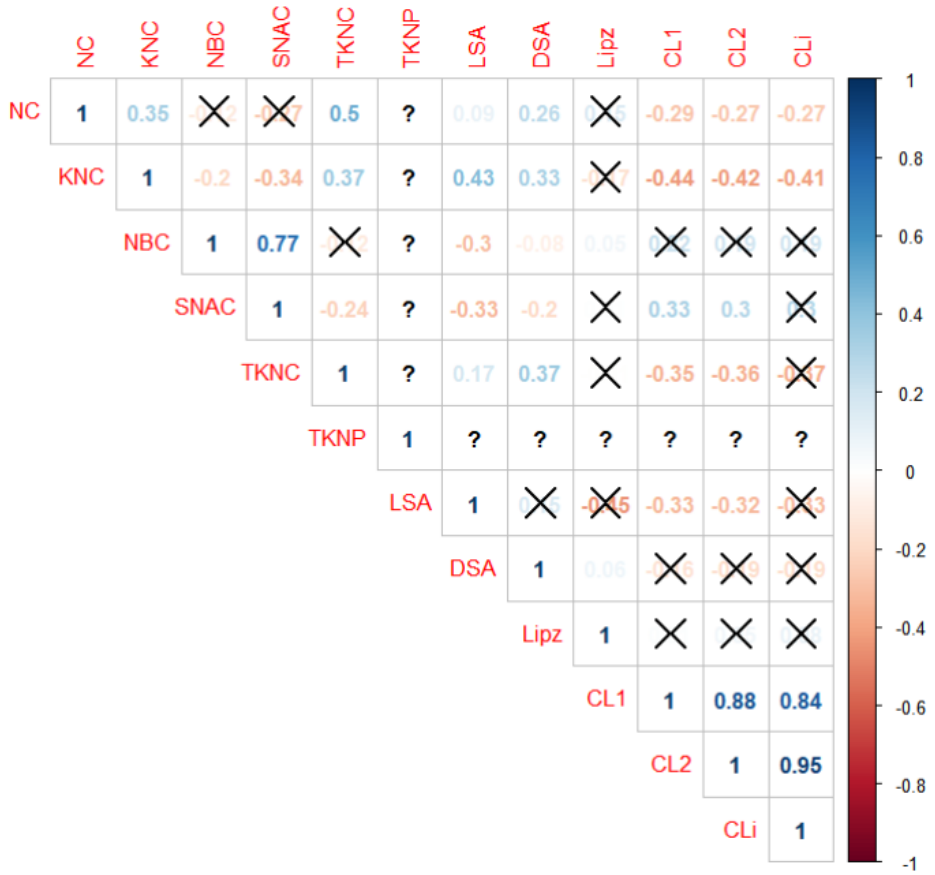
This assumption however is often not examined or only evaluated with limited DNN models and structures, making it unclear whether the results generalize. Furthermore, how a test suite improves the quality of a DNN is different from that of a software system. A software system is improved by fixing bugs revealed by a test suite. A DNN is typically improved

Dong et al. An Empirical Study on Correlation between Coverage and Robustness for Deep Neural Networks. IEEE, 2020.

2. Problem



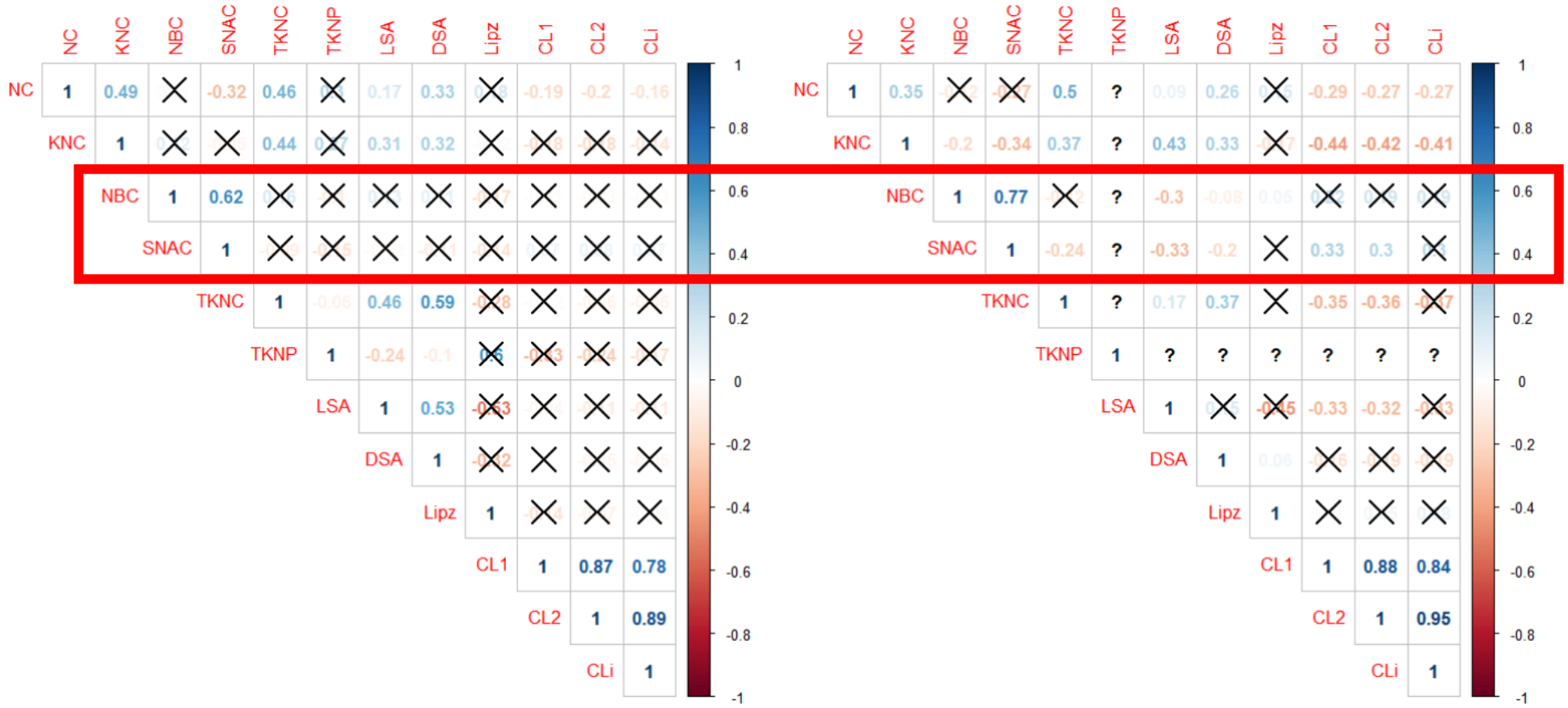
(a) MNIST



(b) CIFAR10

Dong et al. An Empirical Study on Correlation between Coverage and Robustness for Deep Neural Networks. IEEE, 2020.

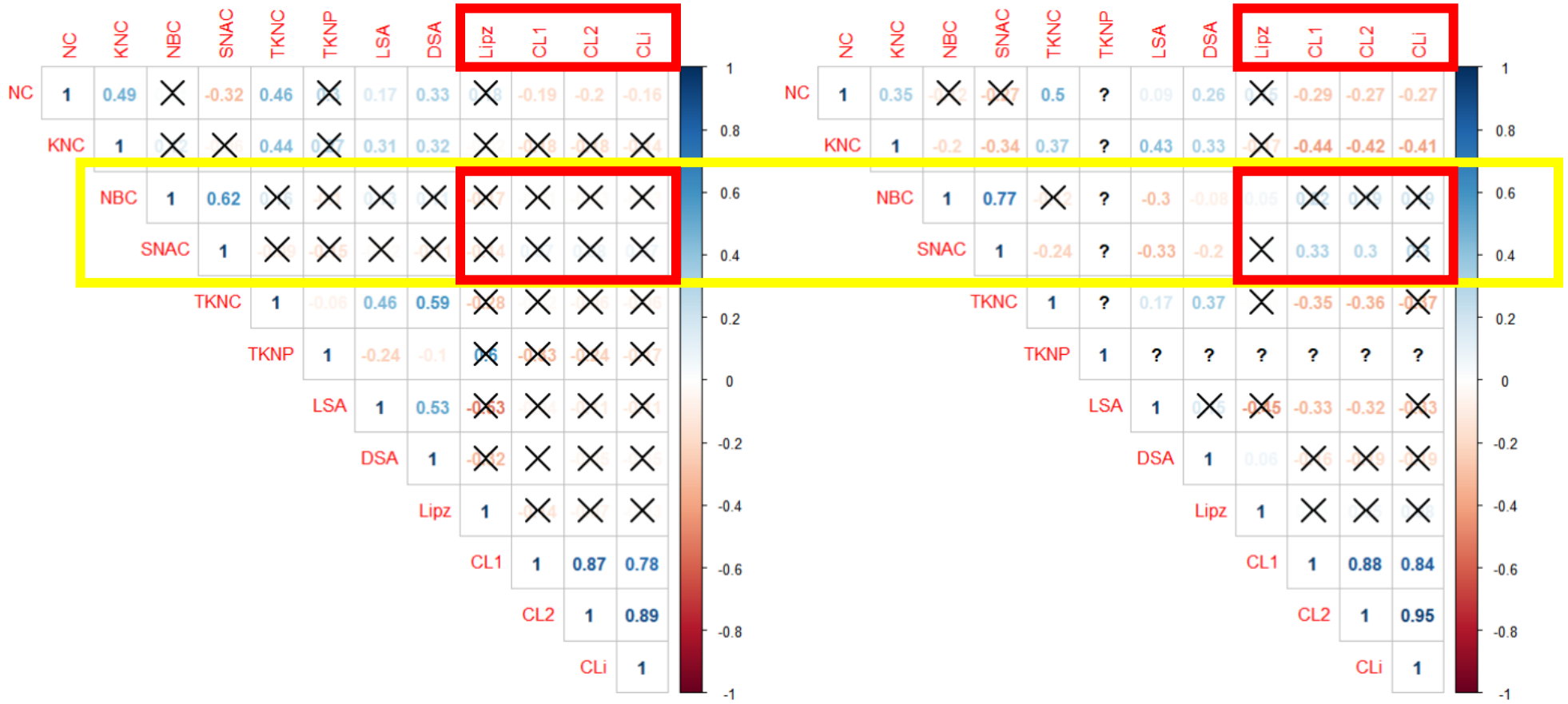
2. Problem



(a) MNIST **(b) CIFAR10**

Dong et al. An Empirical Study on Correlation between Coverage and Robustness for Deep Neural Networks. IEEE, 2020.

2. Problem



(a) MNIST **(b) CIFAR10**

Dong et al. An Empirical Study on Correlation between Coverage and Robustness for Deep Neural Networks. IEEE, 2020.

2. Problem

NBC	1	0.62	✗	✗	✗	✗	✗	✗	✗	✗
SNAC	1		✗	✗	✗	✗	✗	✗	✗	✗

We further investigate the correlation among all test coverage criteria themselves. It can be observed from Fig. 3 that NC, KNC, TKNC, LSA and DSA are positively correlated with each other. NBC and SNAC are correlated with each other with medium or high strength, whereas they have no (or weak negative) correlation with the other metrics. The results are consistent with observations reported in [13] and [15] which propose these coverage. This suggests that despite that different coverage criteria are defined differently, they are in general correlated (except for the boundary coverage).

Dong et al. An Empirical Study on Correlation between Coverage and Robustness for Deep Neural Networks. IEEE, 2020.

3. PIGEON

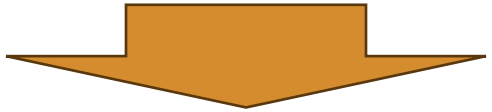
- 원래 SNAC의 도입 취지는, 과활성(over-activation) 뉴런의 비율을 확인

-> 취약한 뉴런의 비율 혹은 testing data의 적대성 평가

3. PIGEON

- 원래 SNAC의 도입 취지는, 과활성(over-activation) 뉴런의 비율을 확인

-> 취약한 뉴런의 비율 혹은 testing data의 적대성 평가



Training 분포를 반영하며 안정적이면서도,
over-activation에 대해 반응할 수 있는 Metric Design 필요

3. PIGEON

- 원래 SNAC의 도입 취지는, 과활성(over-activation) 뉴런의 비율을 확인

-> 취약한 뉴런의 비율 혹은 testing data의 적대성 평가

Training 분포를 반영하며 안정적이면서도,
over-activation에 대해 반응할 수 있는 Metric Design 필요

3. PIGEON

- Definition

$$\text{Pigeon}_i(x) = \begin{cases} 1 & \text{if } z_i(x) > \mu_i + k\sigma_i, \\ 0 & \text{otherwise} \end{cases}$$

$$\text{Pigeon} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}\left[\exists x \in \mathcal{D}_{\text{test}} : \text{Pigeon}_i(x) = 1\right],$$

- 가정: activation의 분포가 정규분포이다.
- 기준: training 때의 activation의 분포상의 95%, 99%, 99.99%, · · ·
- 95%, 99%, 99.99%, · · · 등은 Hyperparameter

3. PIGEON

- Why pigeon?
  - Symbol of peace
  - Repetition “9”(99.99 · · ·)

3. PIGEON

- Why pigeon?
  - Symbol of peace
  - Repetition “9”(99.99 · · ·)

- WHY PIGEON???
  - 철저하게 training 때의 분포로부터 계산된 통계량이므로, coverage 수치를 distribution-aware하게 분석할 수 있다.
  - 운에 따라 등장할 수도, 하지 않을 수도 있는 Outlier에 민감하게 반응하지 않게 된다. Outlier free!!

3. PIGEON

- Why pigeon?
  - Symbol of peace
  - Repetition “9”(99.99 · · ·)

- WHY PIGEON???
  - 철저하게 training 때의 분포로부터 계산된 통계량이므로, coverage 수치를 distribution-aware하게 분석할 수 있다.
  - 운에 따라 등장할 수도, 하지 않을 수도 있는 Outlier에 민감하게 반응하지 않게 된다. Outlier free!!
  - 무엇보다도, 기존의 NBC, SNAC보다 강건성과 상관관계가 유의미하게 개선되었다.

4. Experiment

Statement: SNAC는 분포와 무관하다.
= 쉽게 조작 가능하다.

- ResNet18 train with MNIST

4. Experiment

Statement: SNAC는 분포와 무관하다.
= 쉽게 조작 가능하다.

- ResNet18 train with MNIST
  - 학습하면서, 각 뉴런의 max 활성치와 그 때의 input data를 기록
  - 중복을 허용하여 이렇게 기록된 데이터셋은 MNIST의 subset임, 이것만 그대로 넣으면 뉴런의 max를 발현하는 것.

4. Experiment

Statement: SNAC는 분포와 무관하다.
= 쉽게 조작 가능하다.

- ResNet18 train with MNIST
  - 학습하면서, 각 뉴런의 max 활성치와 그 때의 input data를 기록
  - 중복을 허용하여 이렇게 기록된 데이터셋은 MNIST의 subset임, 이것만 그대로 넣으면 뉴런의 max를 발현하는 것.
  - 이 subset의 각 data가 max로 만드는 뉴런의 활성도를 Label로 생각하고, FGSM/PGD를 통해 활성도를 최대 ε=0.3 만큼 증가하도록 attack

4. Experiment

Statement: SNAC는 분포와 무관하다.
= 쉽게 조작 가능하다.

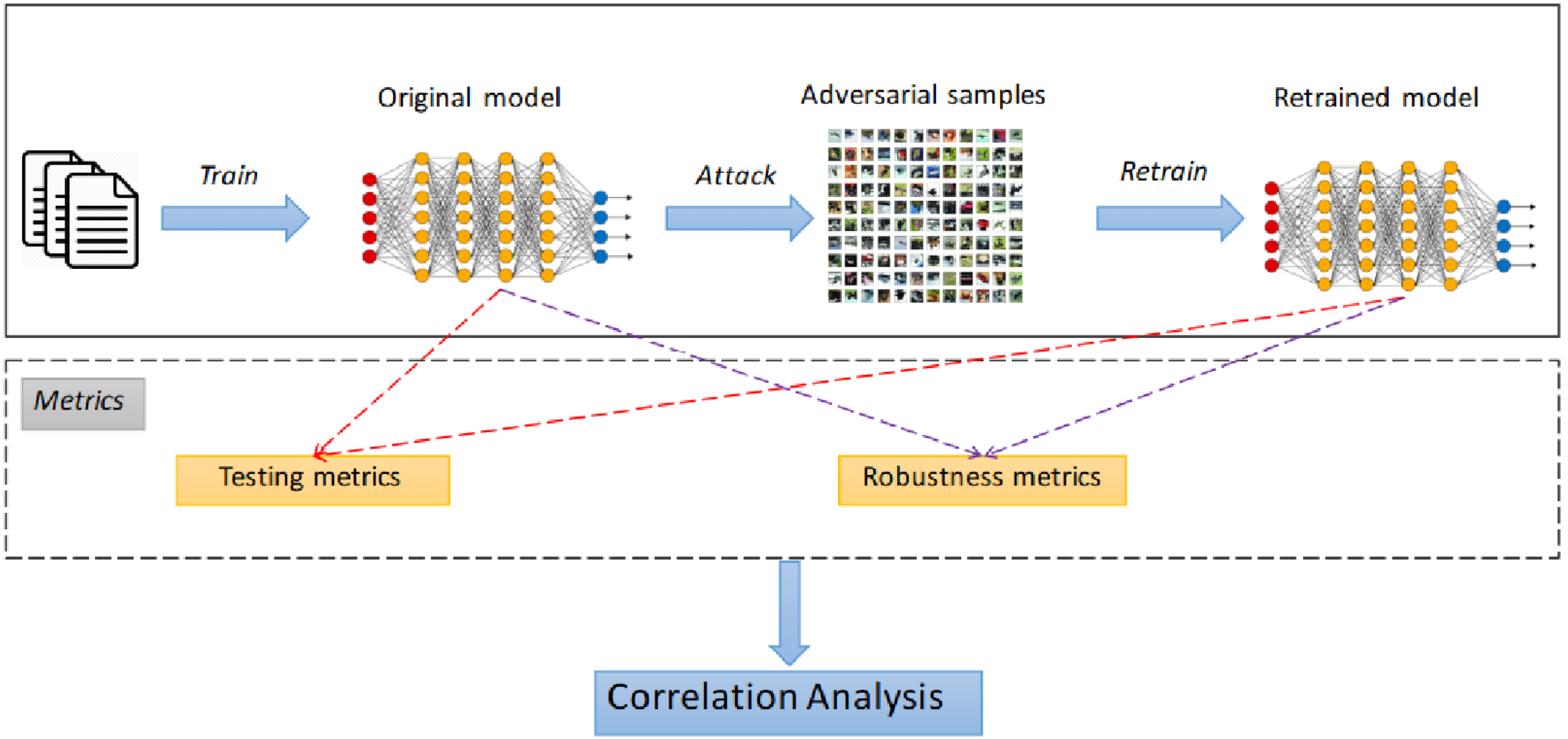
- ResNet18 train with MNIST
  - 학습하면서, 각 뉴런의 max 활성치와 그 때의 input data를 기록
  - 중복을 허용하여 이렇게 기록된 데이터셋은 MNIST의 subset임, 이것만 그대로 넣으면 뉴런의 max를 발현하는 것.
  - 이 subset의 각 data가 max로 만드는 뉴런의 활성도를 Label로 생각하고, FGSM/PGD를 통해 활성도를 최대 ε=0.3 만큼 증가하도록 attack
  - 그렇게 만들어진 adversarial examples는 분포상 큰 차이가 없음에도, 데이터의 개수를 조정하며 쉽게 SNAC 값을 결정할 수 있음.

4. Experiment

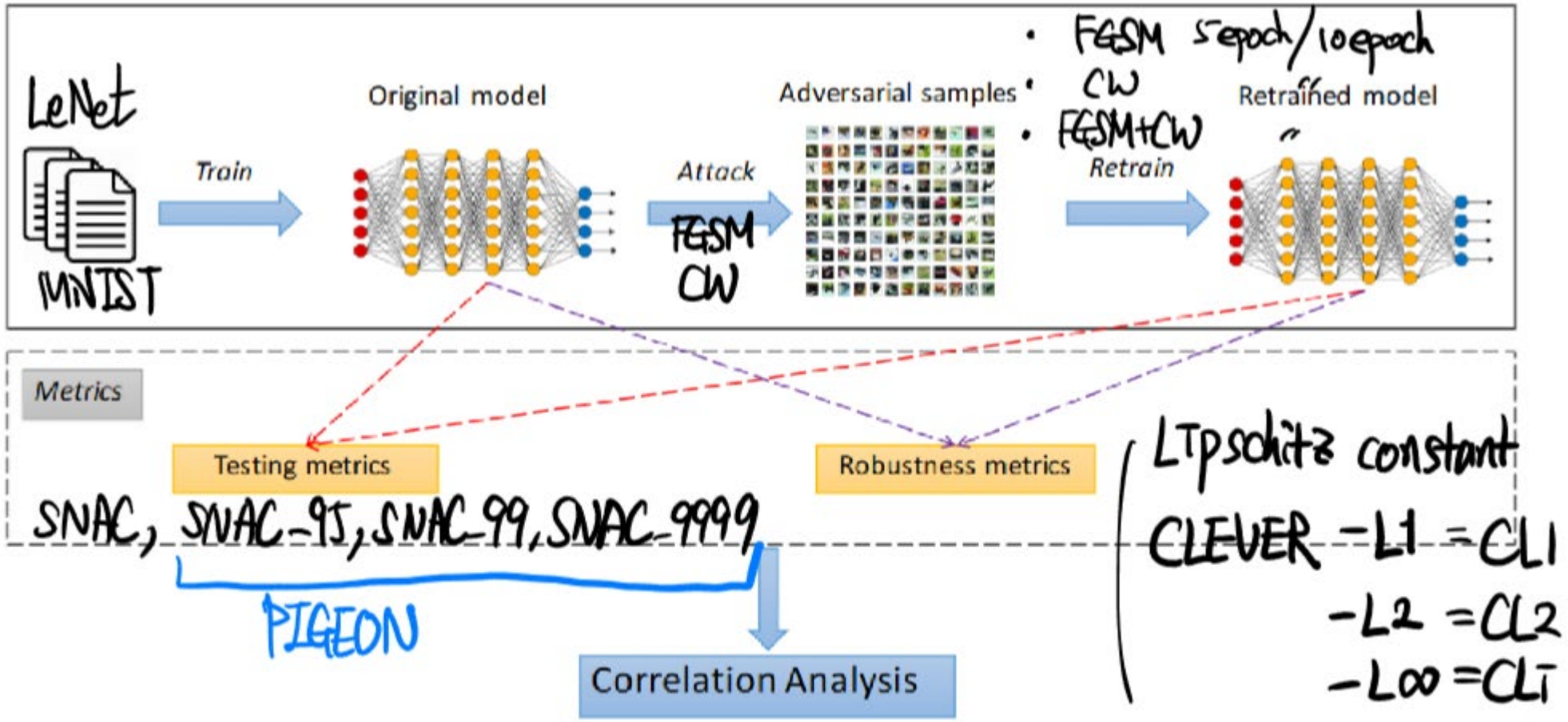
Statement: SNAC는 분포와 무관하다.
= 쉽게 조작 가능하다.

- ResNet18 train with MNIST
  - 간단하게 구현했을 때, PGD로 60% attack에 성공함

4. Experiment



4. Experiment



7 models x 21 Datasets = 147 cases

4. Experiment

- Dong et al. (2020)을 baseline으로함.
- FGSM, CW의 parameter를 바꿔가며 attack 재현. (성공률 99.9%)
- LeNet에 MNIST만 학습시킨 pure LeNet과
  - FGSM Datasets으로 5epoch, 10epoch retrain 시킨 LeNet
  - CW Datasets으로 5epoch, 10epoch retrain 시킨 LeNet
  - FGSM+CW Datasets으로 5epoch, 10epoch retrain 시킨 LeNet
    - = total 7 models
- Adversarial Datasets와 Original을 섞어가며 데이터 21 datasets

- 가능한 cases: 7models x 21 datasets = 147 cases

4. Experiment

- 가능한 cases: 7models x 21 datasets = 147 cases
- 각 case에 대해
  - coverage 계산: SNAC, SNAC_95, SNAC_99, SNAC_9999
  - 강건성 계산: Lipschitz constant, CL1, CL2, CLi

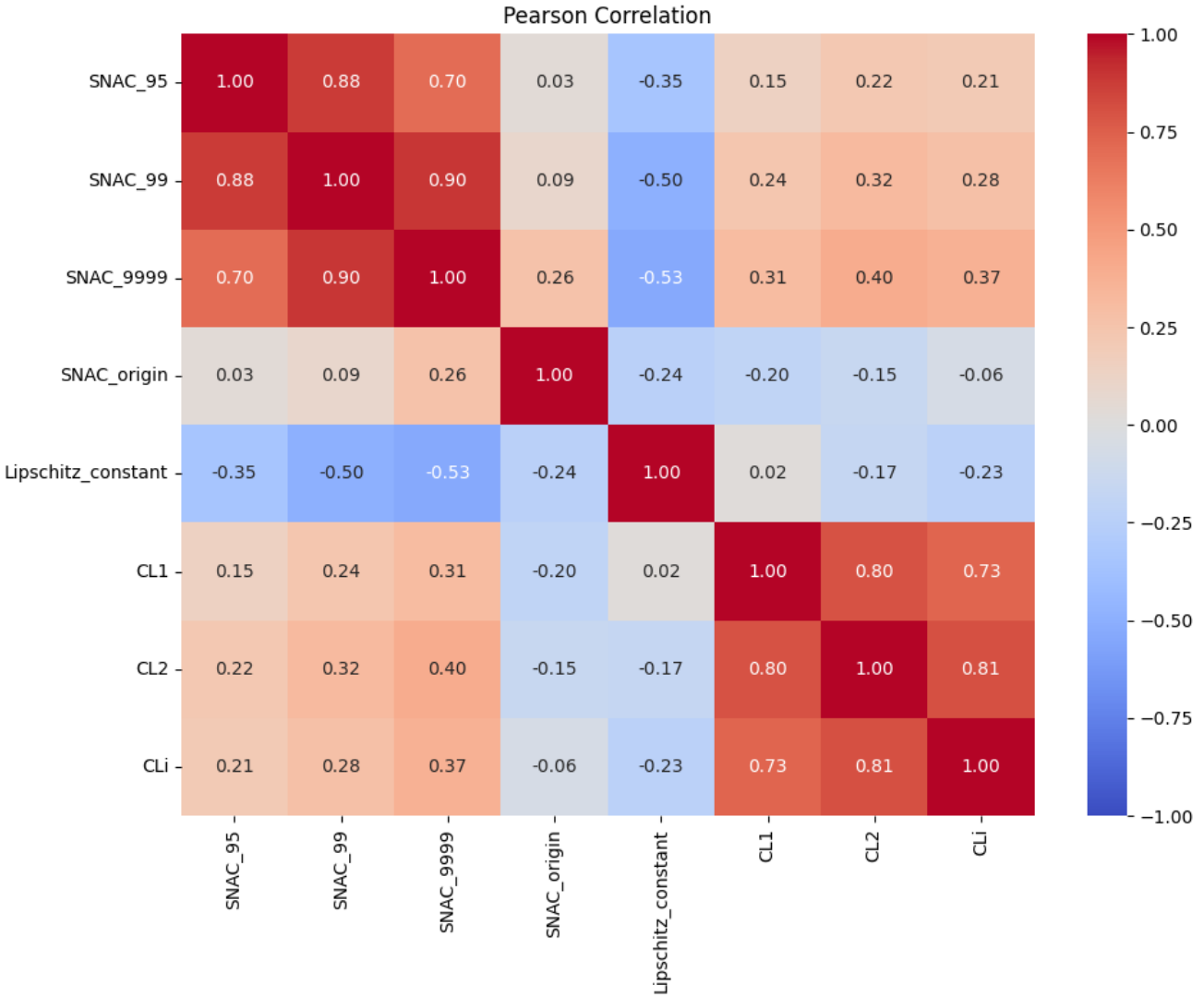
4. Experiment

- 가능한 cases: 7models x 21 datasets = 147 cases
- 각 case에 대해
  - coverage 계산: SNAC, SNAC_95, SNAC_99, SNAC_9999
  - 강건성 계산: Lipschitz constant, CL1, CL2, CLi
- 잠깐, Training 때 activation 분포를 어떻게 알 수 있나?
  - 처음엔 모두 기록하여 통계를 내려고 했으나..
  - 각 뉴런의 모든 활성값의 총합과, 제곱의 총합을 업데이트 하며 기록하면 평균과 분산을 구할 수 있다!
  - 정규분포를 가정했으므로 SNAC_95, SNAC_99, SNAC_9999를 구할 수 있다.

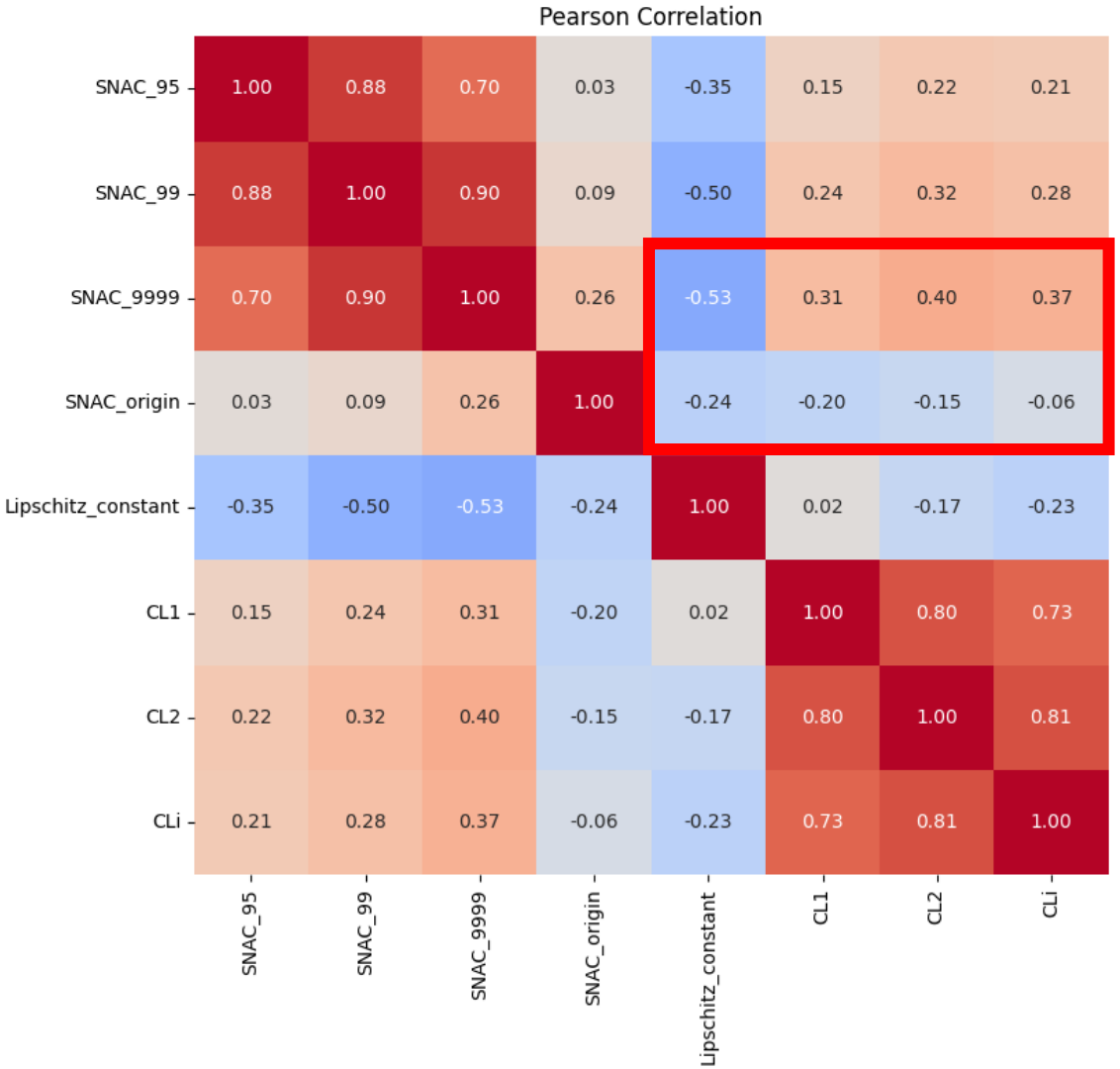
4. Experiment

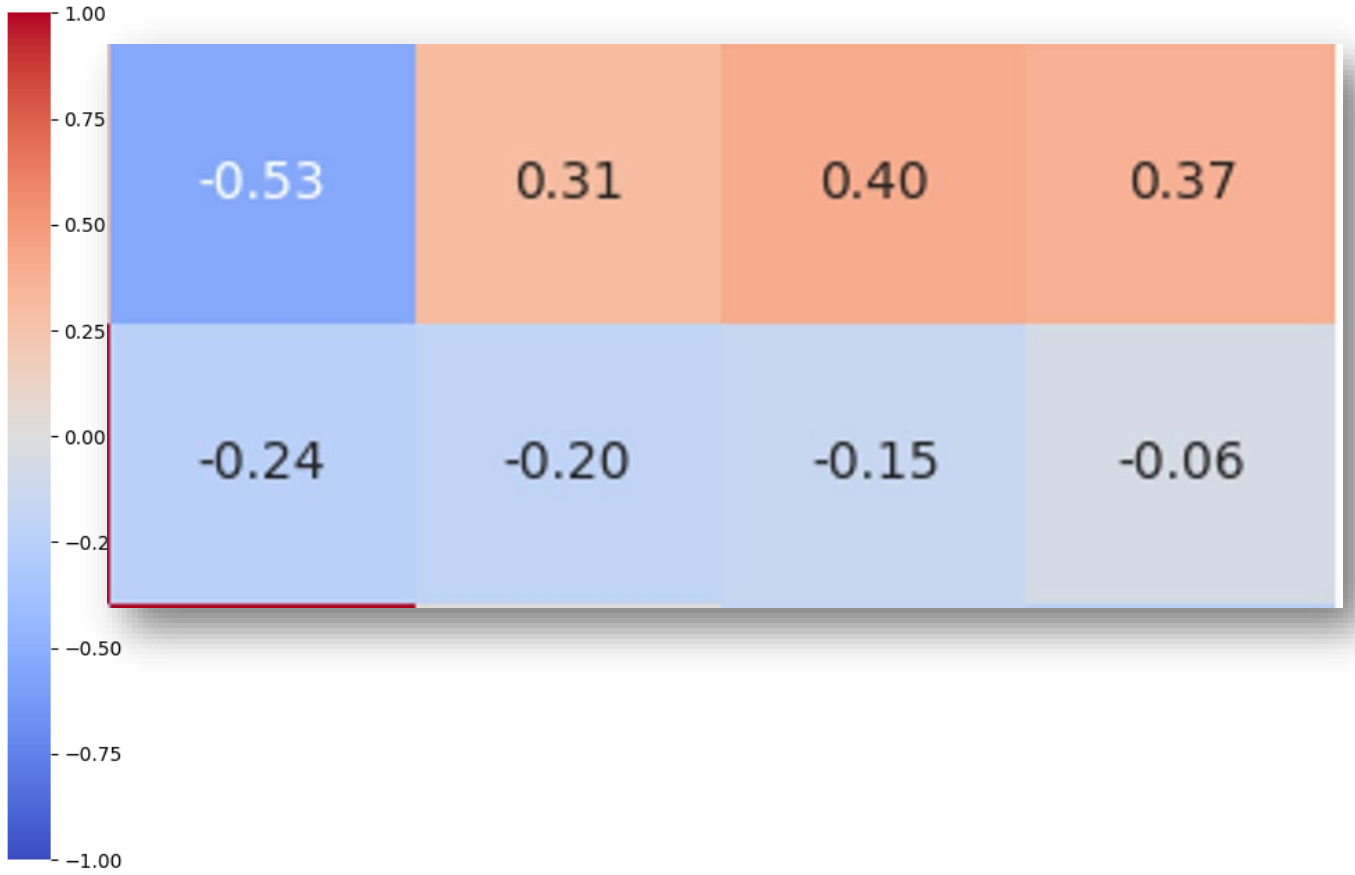
- 가능한 cases: 7models x 21 datasets = 147 cases
- 각 case에 대해
  - coverage 계산: SNAC, SNAC_95, SNAC_99, SNAC_9999
  - 강건성 계산: Lipschitz constant, CL1, CL2, CLi
- 구한 모든 지표의 상관관계(Pearson, Spearman, Kendall's Tau)를 분석한다.

5. Results

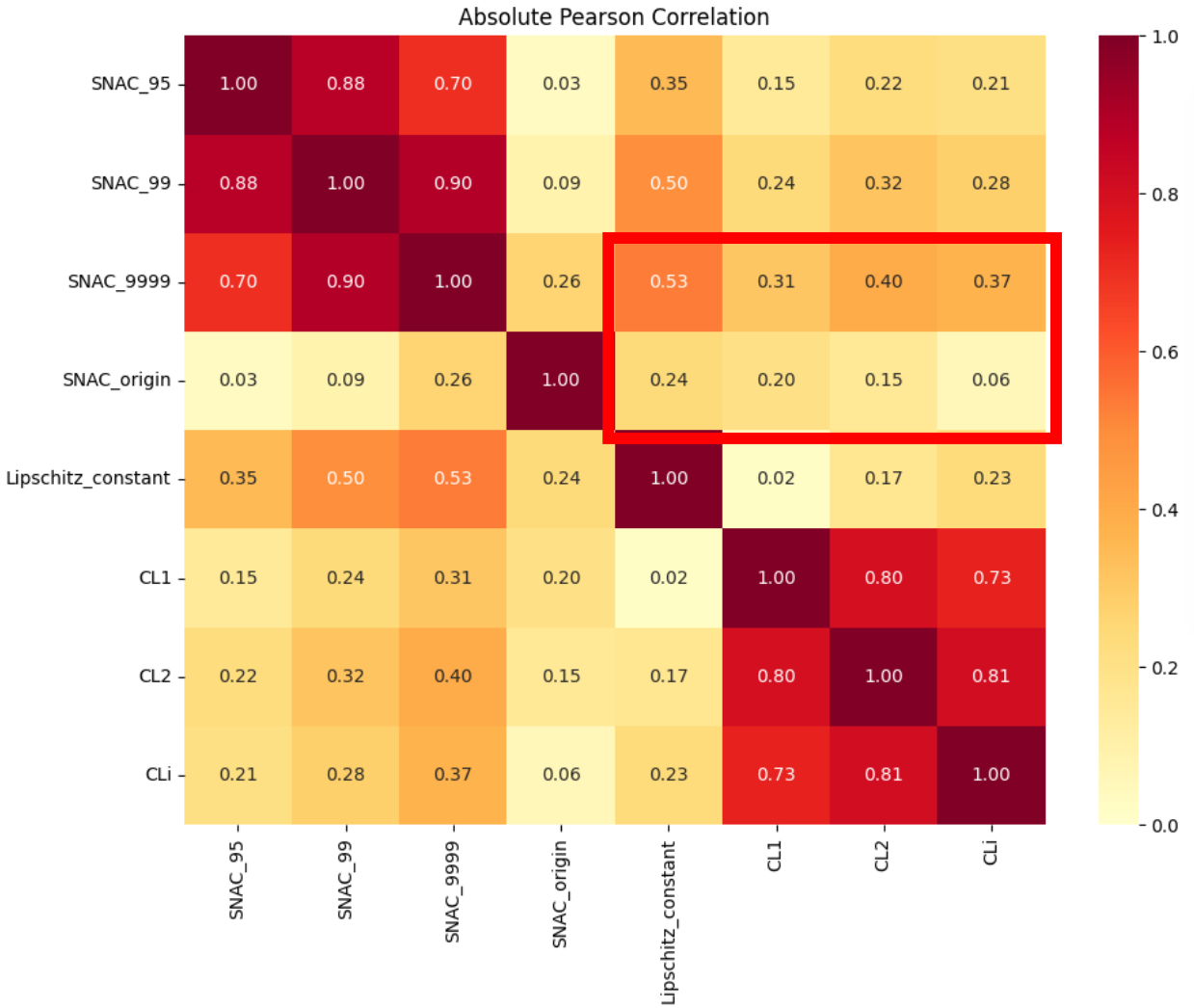

Pearson Correlation
SNAC_95
SNAC_99
SNAC_9999
SNAC_origin
Lipschitz_constant
CL1
CL2
CLi
1.00 0.88 0.70 0.03 -0.35 0.15 0.22 0.21
0.88 1.00 0.90 0.09 -0.50 0.24 0.32 0.28
0.70 0.90 1.00 0.26 -0.53 0.31 0.40 0.37
0.03 0.09 0.26 1.00 -0.24 -0.20 -0.15 -0.06
-0.35 -0.50 -0.53 -0.24 1.00 0.02 -0.17 -0.23
0.15 0.24 0.31 -0.20 0.02 1.00 0.80 0.73
0.22 0.32 0.40 -0.15 -0.17 0.80 1.00 0.81
0.21 0.28 0.37 -0.06 -0.23 0.73 0.81 1.00
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00

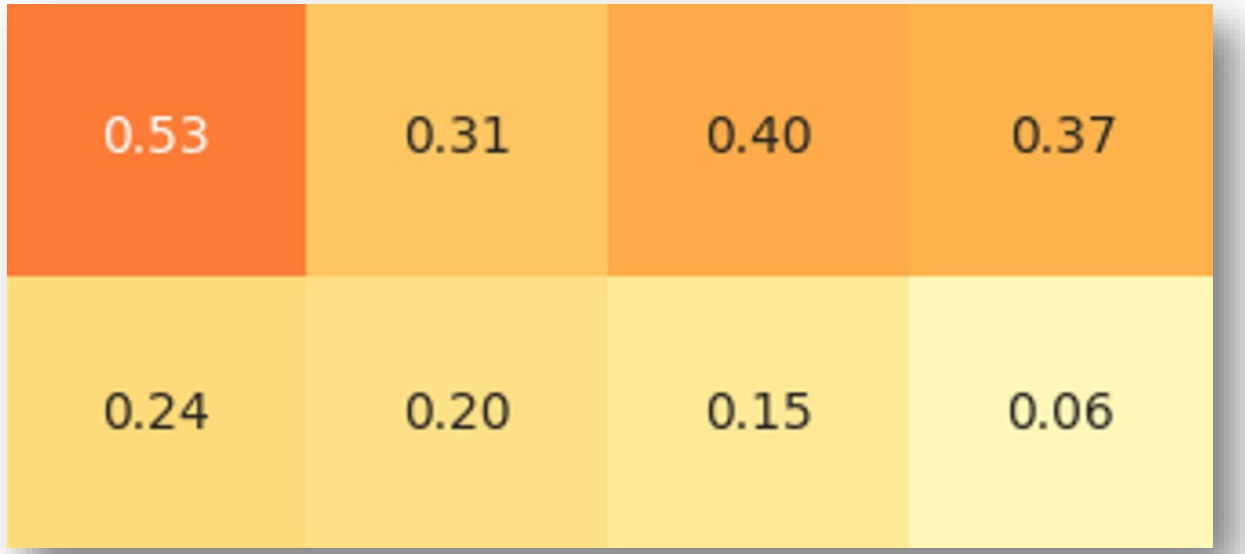
5. Results


Pearson Correlation
SNAC_95 1.00 0.88 0.70 0.03 -0.35 0.15 0.22 0.21
SNAC_99 0.88 1.00 0.90 0.09 -0.50 0.24 0.32 0.28
SNAC_9999 0.70 0.90 1.00 0.26 -0.53 0.31 0.40 0.37
SNAC_origin 0.03 0.09 0.26 1.00 -0.24 -0.20 -0.15 -0.06
Lipschitz_constant -0.35 -0.50 -0.53 -0.24 1.00 0.02 -0.17 -0.23
CL1 0.15 0.24 0.31 -0.20 0.02 1.00 0.80 0.73
CL2 0.22 0.32 0.40 -0.15 -0.17 0.80 1.00 0.81
CLi 0.21 0.28 0.37 -0.06 -0.23 0.73 0.81 1.00
SNAC_95
SNAC_99
SNAC_9999
SNAC_origin
Lipschitz_constant
CL1
CL2
CLi

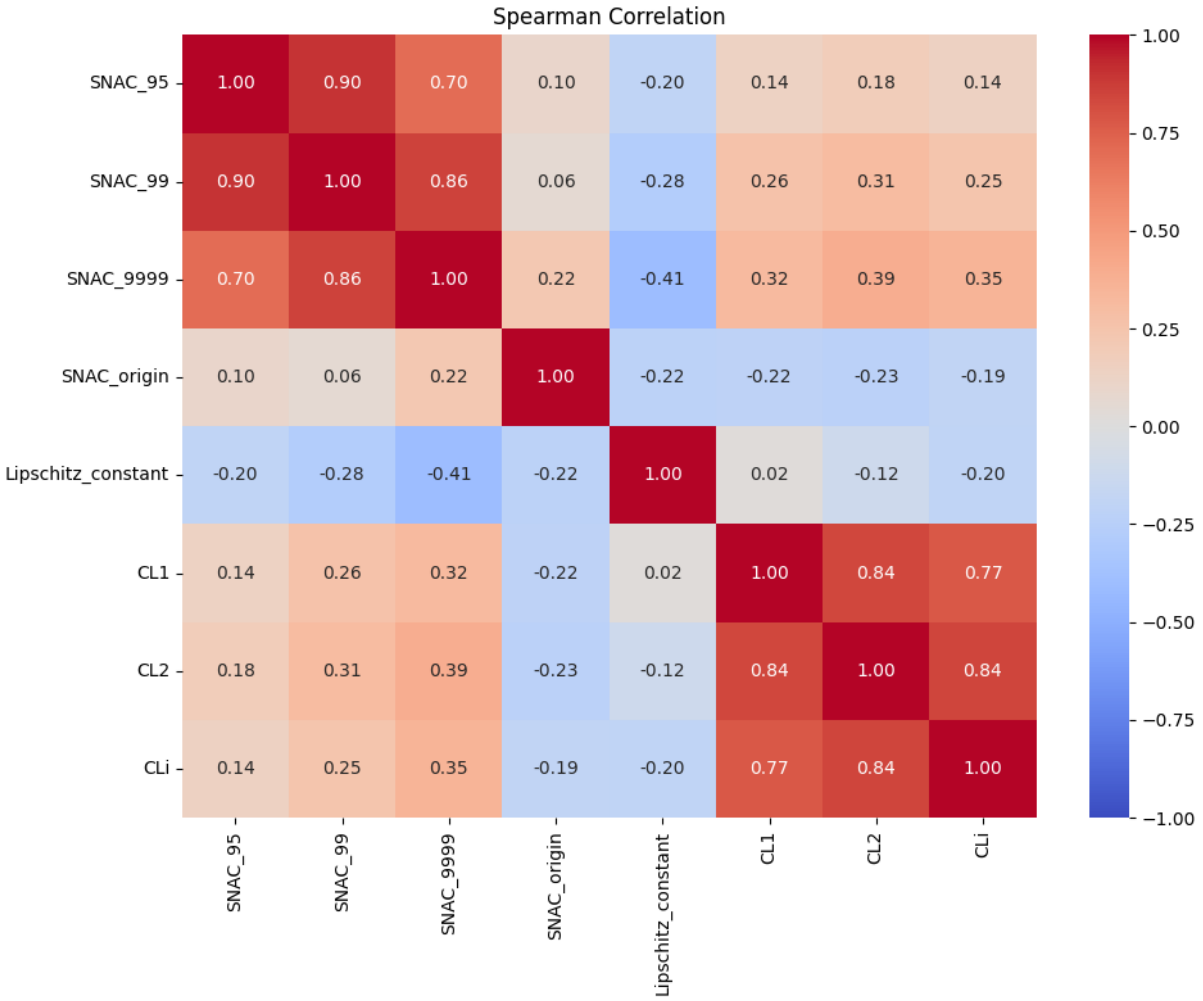

1.00
0.75
0.50
0.25
0.00
−0.2
−0.50
−0.75
−1.00
-0.53 0.31 0.40 0.37
-0.24 -0.20 -0.15 -0.06

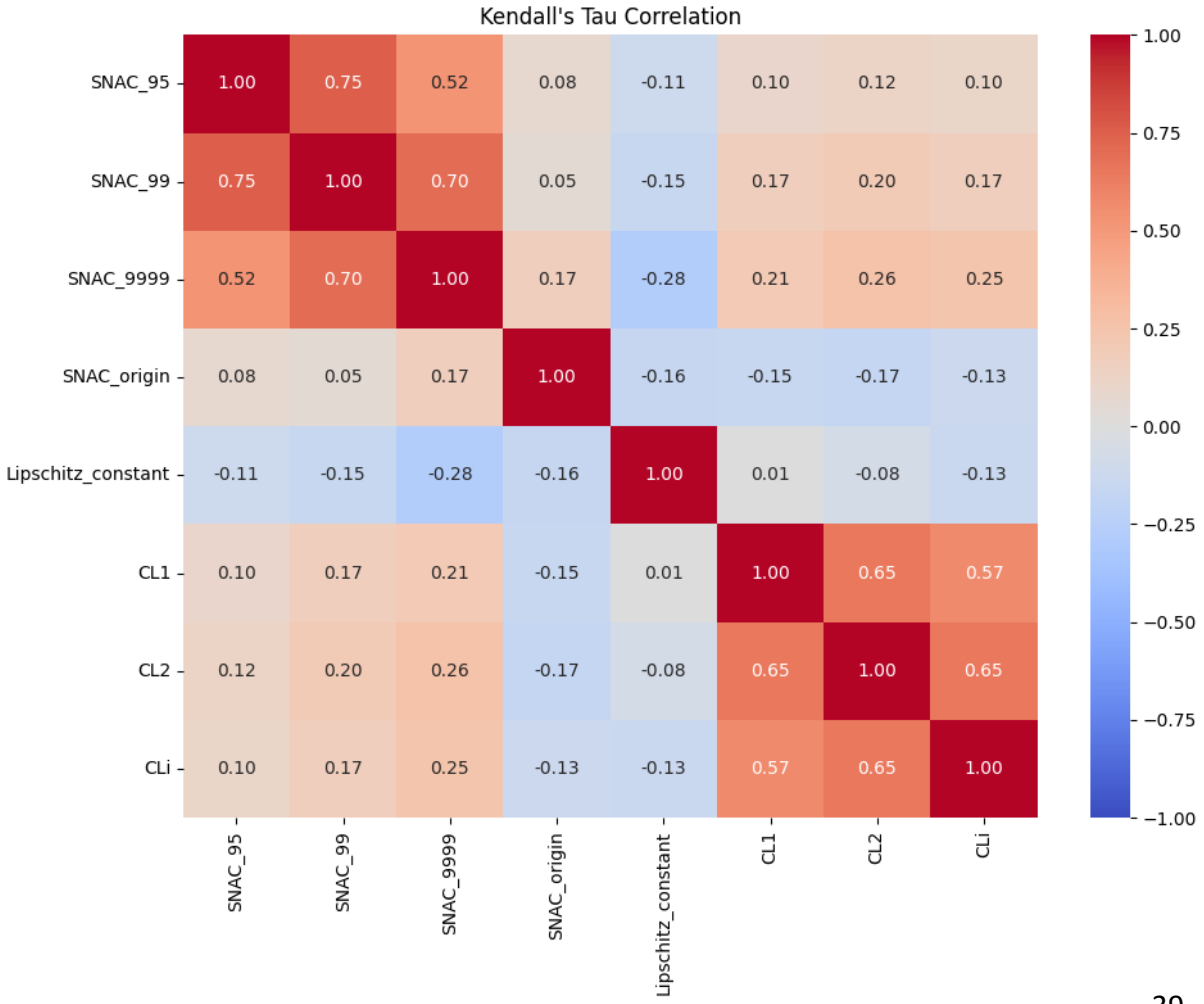
5. Results


Absolute Pearson Correlation
SNAC_95 1.00 0.88 0.70 0.03 0.35 0.15 0.22 0.21
SNAC_99 0.88 1.00 0.90 0.09 0.50 0.24 0.32 0.28
SNAC_9999 0.70 0.90 1.00 0.26 0.53 0.31 0.40 0.37
SNAC_origin 0.03 0.09 0.26 1.00 0.24 0.20 0.15 0.06
Lipschitz_constant 0.35 0.50 0.53 0.24 1.00 0.02 0.17 0.23
CL1 0.15 0.24 0.31 0.20 0.02 1.00 0.80 0.73
CL2 0.22 0.32 0.40 0.15 0.17 0.80 1.00 0.81
CLi 0.21 0.28 0.37 0.06 0.23 0.73 0.81 1.00
SNAC_95
SNAC_99
SNAC_9999
SNAC_origin
Lipschitz_constant
CL1
CL2
CLi
1.0
0.8
0.6
0.4
0.2
0.0


0.53 0.31 0.40 0.37
0.24 0.20 0.15 0.06

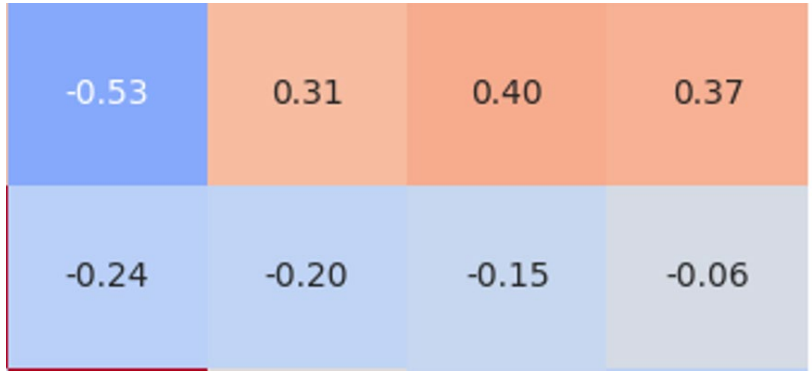
5. Results


Spearman Correlation
SNAC_95 1.00 0.90 0.70 0.10 -0.20 0.14 0.18 0.14
SNAC_99 0.90 1.00 0.86 0.06 -0.28 0.26 0.31 0.25
SNAC_9999 0.70 0.86 1.00 0.22 -0.41 0.32 0.39 0.35
SNAC_origin 0.10 0.06 0.22 1.00 -0.22 -0.22 -0.23 -0.19
Lipschitz_constant -0.20 -0.28 -0.41 -0.22 1.00 0.02 -0.12 -0.20
CL1 0.14 0.26 0.32 -0.22 0.02 1.00 0.84 0.77
CL2 0.18 0.31 0.39 -0.23 -0.12 0.84 1.00 0.84
CLi 0.14 0.25 0.35 -0.19 -0.20 0.77 0.84 1.00
SNAC_95 SNAC_99 SNAC_9999 SNAC_origin Lipschitz_constant CL1 CL2 CLi
1.00 0.75 0.50 0.25 0.00 −0.25 −0.50 −0.75 −1.00

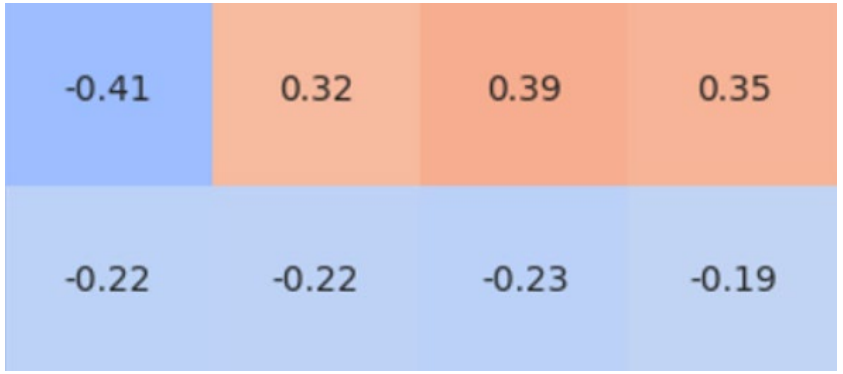

Kendall's Tau Correlation
SNAC_95 1.00 0.75 0.52 0.08 -0.11 0.10 0.12 0.10
SNAC_99 0.75 1.00 0.70 0.05 -0.15 0.17 0.20 0.17
SNAC_9999 0.52 0.70 1.00 0.17 -0.28 0.21 0.26 0.25
SNAC_origin 0.08 0.05 0.17 1.00 -0.16 -0.15 -0.17 -0.13
Lipschitz_constant -0.11 -0.15 -0.28 -0.16 1.00 0.01 -0.08 -0.13
CL1 0.10 0.17 0.21 -0.15 0.01 1.00 0.65 0.57
CL2 0.12 0.20 0.26 -0.17 -0.08 0.65 1.00 0.65
CLi 0.10 0.17 0.25 -0.13 -0.13 0.57 0.65 1.00
SNAC_95 SNAC_99 SNAC_9999 SNAC_origin Lipschitz_constant CL1 CL2 CLi
1.00 0.75 0.50 0.25 0.00 −0.25 −0.50 −0.75 −1.00

6. Analysis

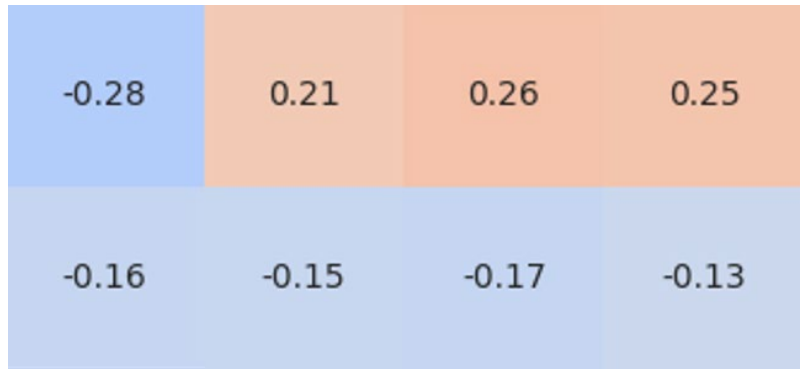
- 147가지의 경우에 대해 correlation을 살펴본 결과,
  - 대체로 강건성 메트릭과 상관관계가 기존의 SNAC에 비해 강해졌다.
  - 특히 SNAC_9999의 경우 모든 강건성 메트릭과 모든 종류의 상관관계에서 뚜렷하게 강해졌다.
  - Testing Dataset이 대체로 최소 2만 여개~최대 21만 여개로 분포의 정의상 세 PIGEON 중 SNAC_9999가 직관적으로 가장 합리적이었다.



- Pearson



- Spearman



- Kendall's Tau

6. Analysis

- PIGEON이 높으면?
  - testing 하는 모델의 train data가 정규화 되어있을 수 있다.
  - 모델이 강건하지 않을 수 있다.(과활성에 취약한 뉴런이 많다.)
  - testing 하는 dataset이 train data로부터 분포의 차이가 있을 수 있다.

- PIGEON이 낮으면?
  - testing 하는 모델의 train data의 분포가 넓을 수 있다.
  - 모델이 강건할 수 있다.(activation의 범위가 적절히 제한되고 있다.)
  - testing 하는 dataset이 정규화되어 있을 수 있다.

- testing하는 dataset에 대해서 우리가 확인할 수 있으므로, 모델의 강건성을 간접적으로 측정할 수 있게 됩니다.

7. Limitations & Future work

- Limitations
  - 뉴런 출력을 정규분포를 가정하고 했으나, 실제로 정규분포로부터 오차가 존재한다.
  - 문제가 되는 뉴런을 특정할 수 없다.
  - Train dataset의 분포에 밀접한 관계가 있어서, 학습 데이터의 분포를 알지 못할 경우 닫힌 해석을 하기 어렵다. (앞페이지 해석에 영향)
  - Testing Dataset의 scale에 영향을 받을 여지가 크다.
    - (데이터를 많이 넣으면, 분포상 기준점을 넘길 확률이 증가)
- Future work
  - Metric detail(Model Depth, Layer Width 반영하여 탈정규분포 오차를 반영한 정확한 분포)
  - 다양한 모델과 데이터 셋, 다양한 metric에 대해서 상관관계 확장
  - Metric –hyperparameter 실험 및 옵션으로..-> 95, 99, 99.99 값이 올라갈 수록 상관관계가 높아지긴 했지만, 의미론적으로는 그런 데이터가 등장하지 않을 확률이므로, coverage testing을 하는 모델과, data의 scale에 따라 조정하여 정할 수 있도록 하면 좋을 것 같습니다.
  - 혹은 여러 값을 측정하여(SNAC_99, SNAC_9999) 값을 동시 분석.
    - 예를 들어, SNAC_99, SNAC_9999의 값이 비슷하면 취약한 뉴런이 일부 뚜렷한 상황으로 해석 가능.

Thank You!

or QA

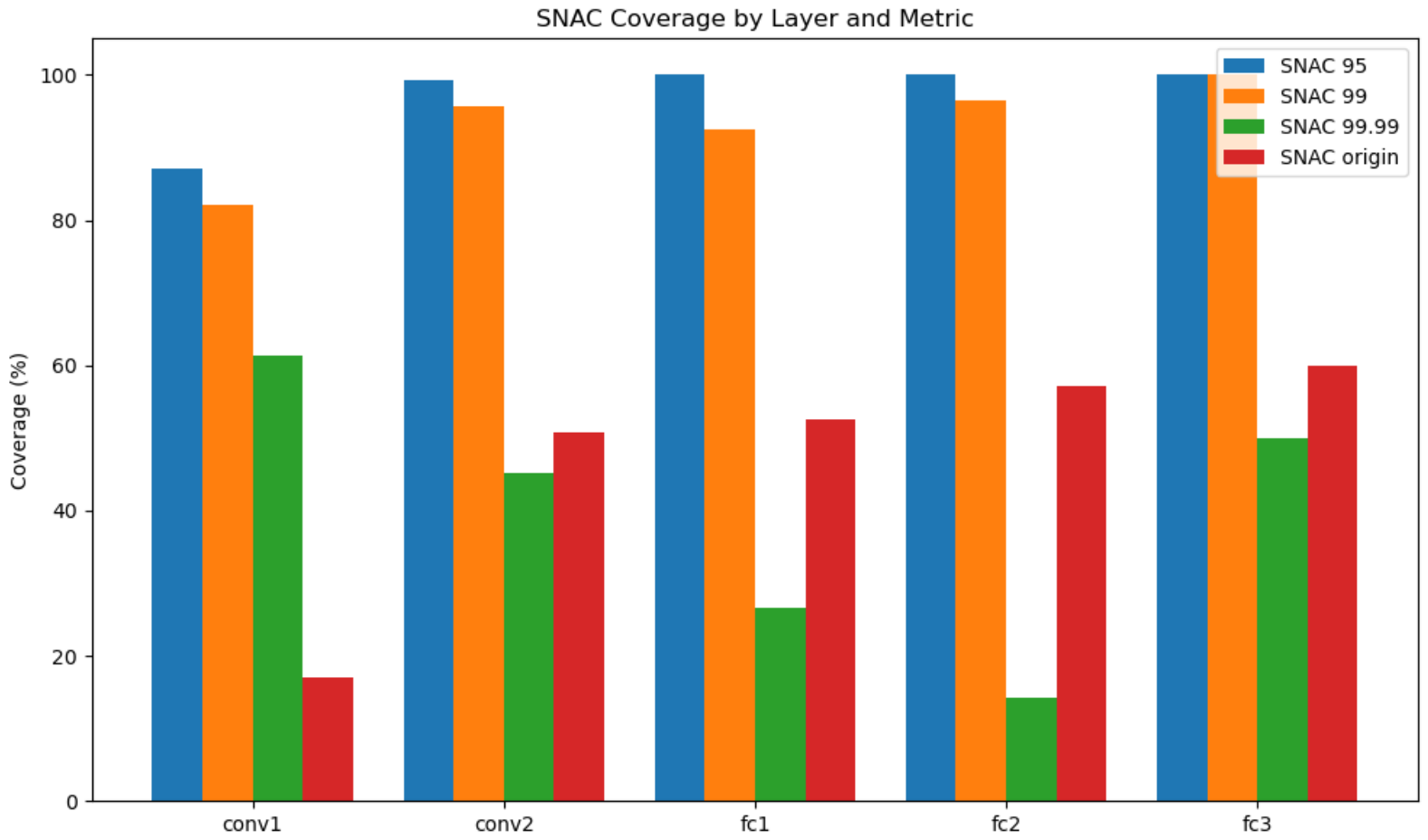
Juheon Kang (G202548001)
2715wngjs@uos.ac.kr

Hyunseo Shin (G202448003)
hseo98@uos.ac.kr

Eunkyung Choi (G202448018)
rmarud202@uos.ac.kr

Github, PPT

APPENDIX


SNAC Coverage by Layer and Metric
SNAC 95
SNAC 99
SNAC 99.99
SNAC origin
Coverage (%)
0
20
40
60
80
100
conv1
conv2
fc1
fc2
fc3