Instituto Superior Técnico | Feedzai

BENCHMARKING ROBUSTNESS METRICS

Clara Pereira - 99405
Joana Correia - 100412
Vasco Gameiro - 110881

WHAT IS IT ROBUSTNESS?

A model's ability to maintain accurate predictions under out-of-distribution inputs.

WHY IS IT IMPORTANT?

1. Machine learning systems are deployed in high-stakes domains.

2. Criminals exploit small perturbations to mislead these systems.

3. Models must be robust to these attacks while guaranteeing performance over long-term changes in the data.

OUR HYPOTHESIS

There is a single metric that could holistically measure robustness against these adversarial attacks and natural changes in the data

THE ROAD TO ROBUSTNESS

Hypothesis

Datasets

Models

Metrics

Results

DATASET

IMAGENET

- 200 classes (62 used)
- 30000 images used
- Higher resolution (64)
- Higher computational cost

OR

CIFAR-10

- 10 classes
- 30000 images used
- Lower resolution (32)
- Lower computational cost

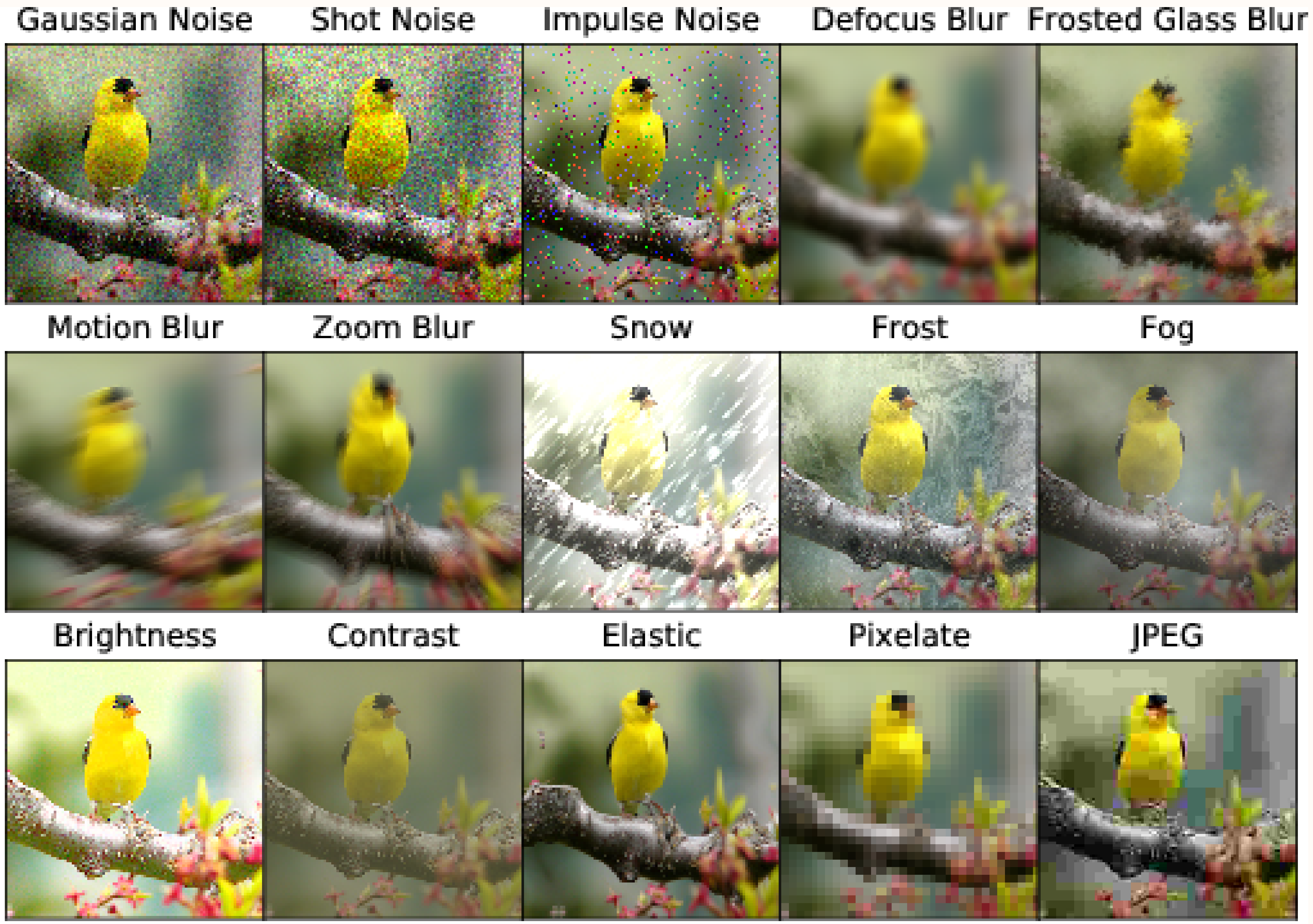
DATASET

OOD TEST SETS

CIFAR-10.1

CIFAR-10C

Attacked



DATASET

OOD TEST SETS

CIFAR-10.1

CIFAR-10C

Attacked

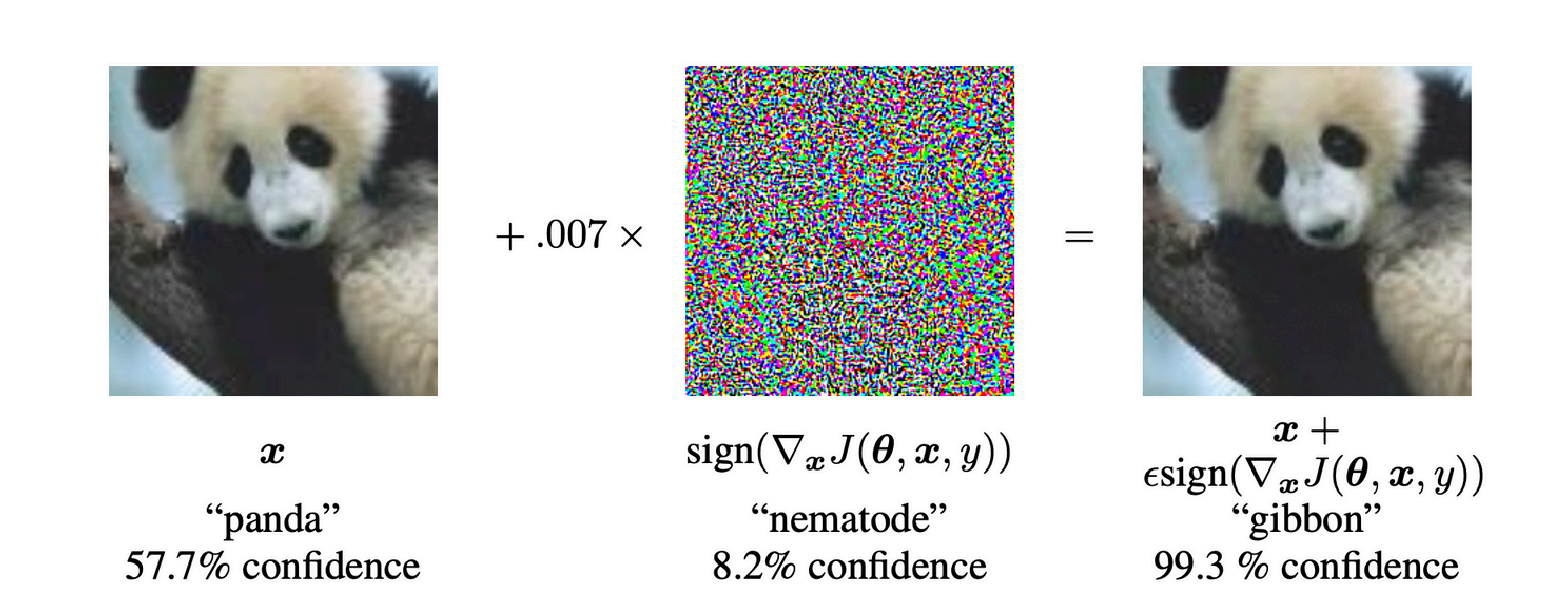
FSGM

PGD

BIM

DeepFool

CW



MODELS

Custom CNN

HYPERPARAMETERS

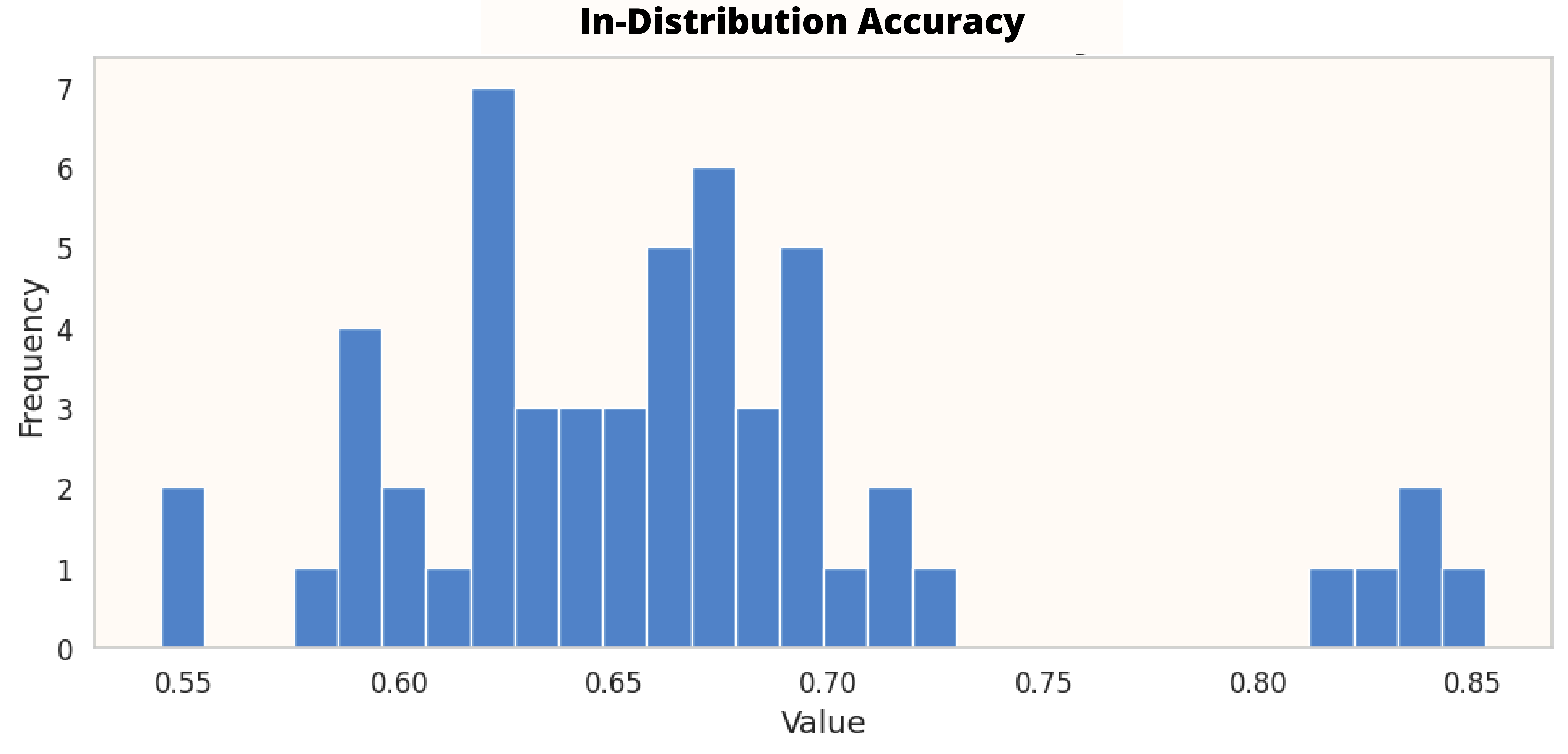
1. Optimizer
2. Learning Rate
3. Depth
4. Width
5. Dropout

VGG Inspired

HYPERPARAMETERS

1. Optimizer
2. Learning Rate

Result: > 50 models


In-Distribution Accuracy
Frequency
Value
0
1
2
3
4
5
6
7
0.55
0.60
0.65
0.70
0.75
0.80
0.85

METRICS

Performance

Sharpness

Norm-Based

METRICS
PERFORMANCE

Directly drawn from the network output

Classification:

- Accuracy
- Precision
- Recall
- F1-Score

Uncertainty?

METRICS

IMPORTANCE OF UNCERTAINTY

"All models are wrong but some are useful."
George Box

METRICS

NORM-BASED

Weight norms reveal insights into model behavior

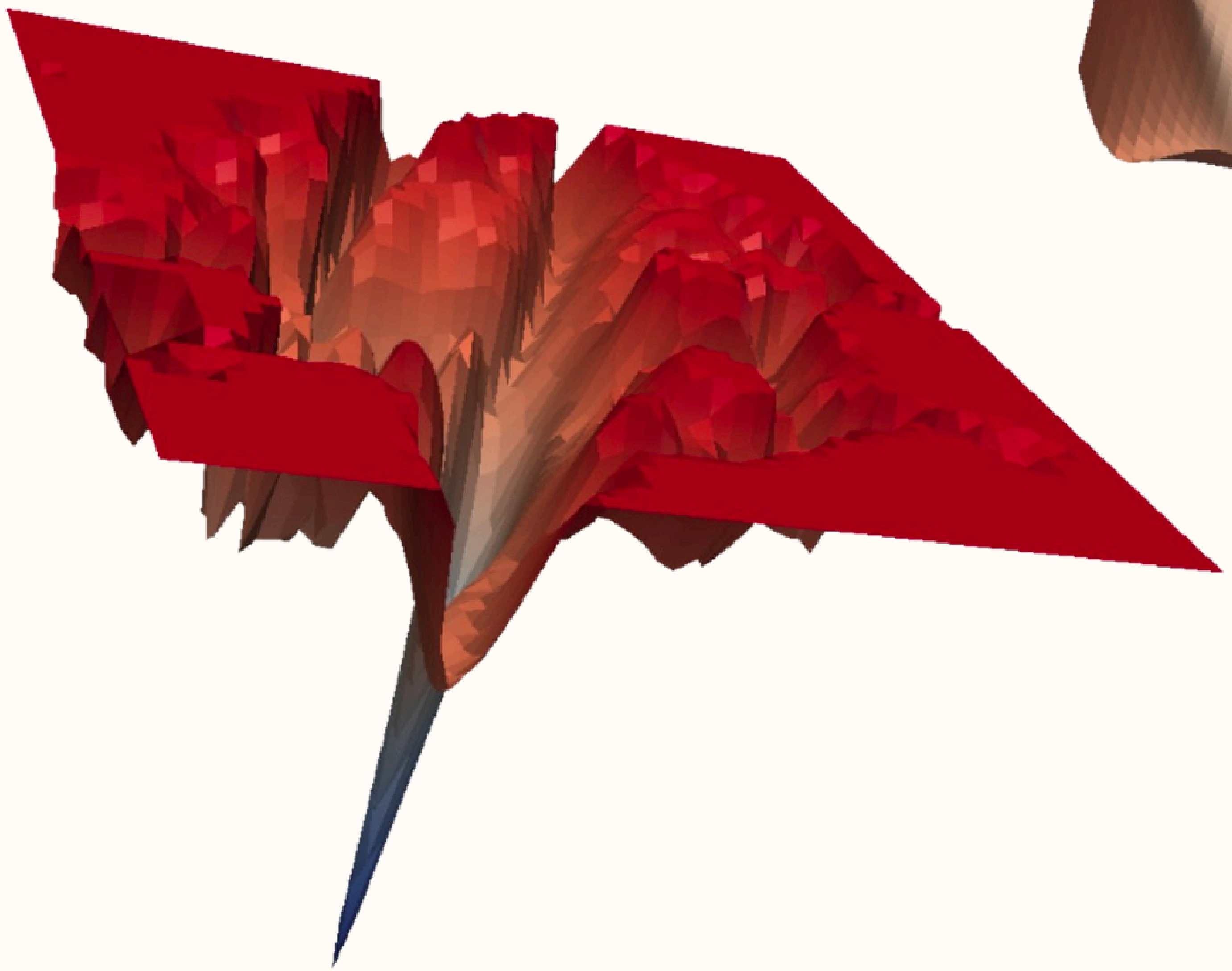
- Spectral Norm
- Frobeneous Norm
- Trace Norm
- Path Norm

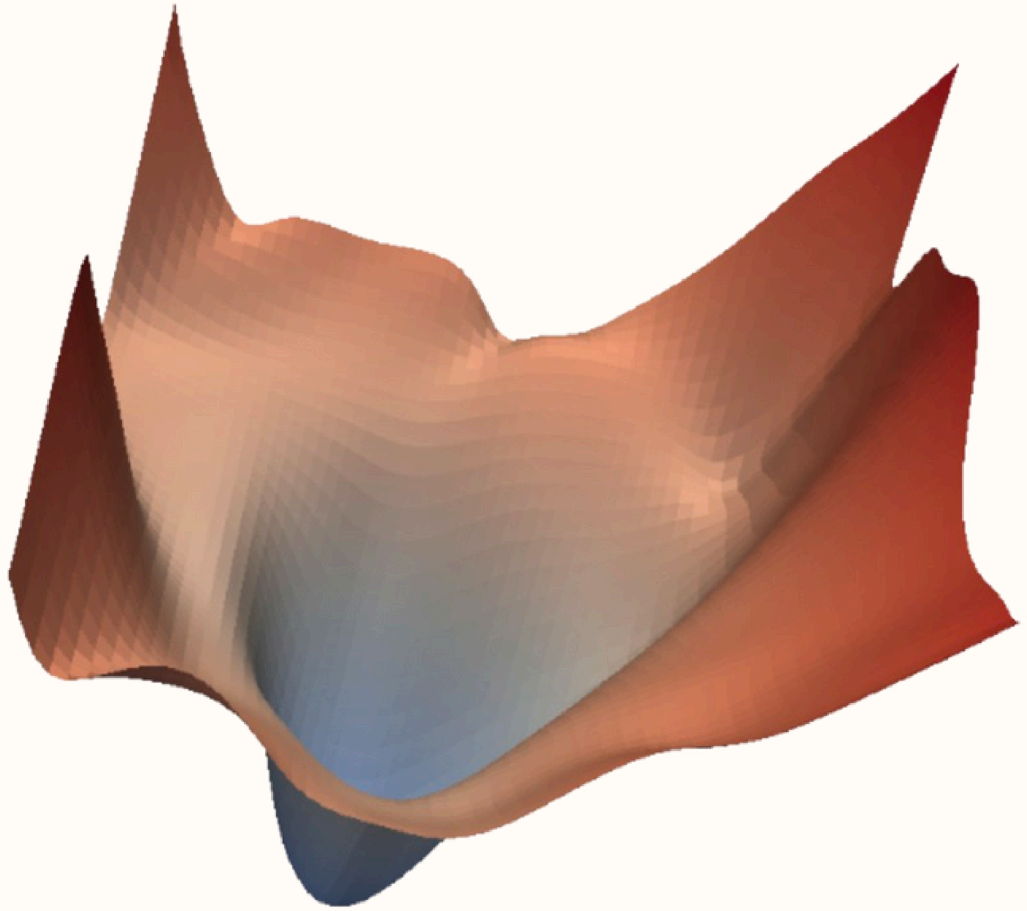
Over the Margin....

METRICS

SHARPNESS

Looking for the right sigmas





METRICS

SHARPNESS

A CRITIQUE

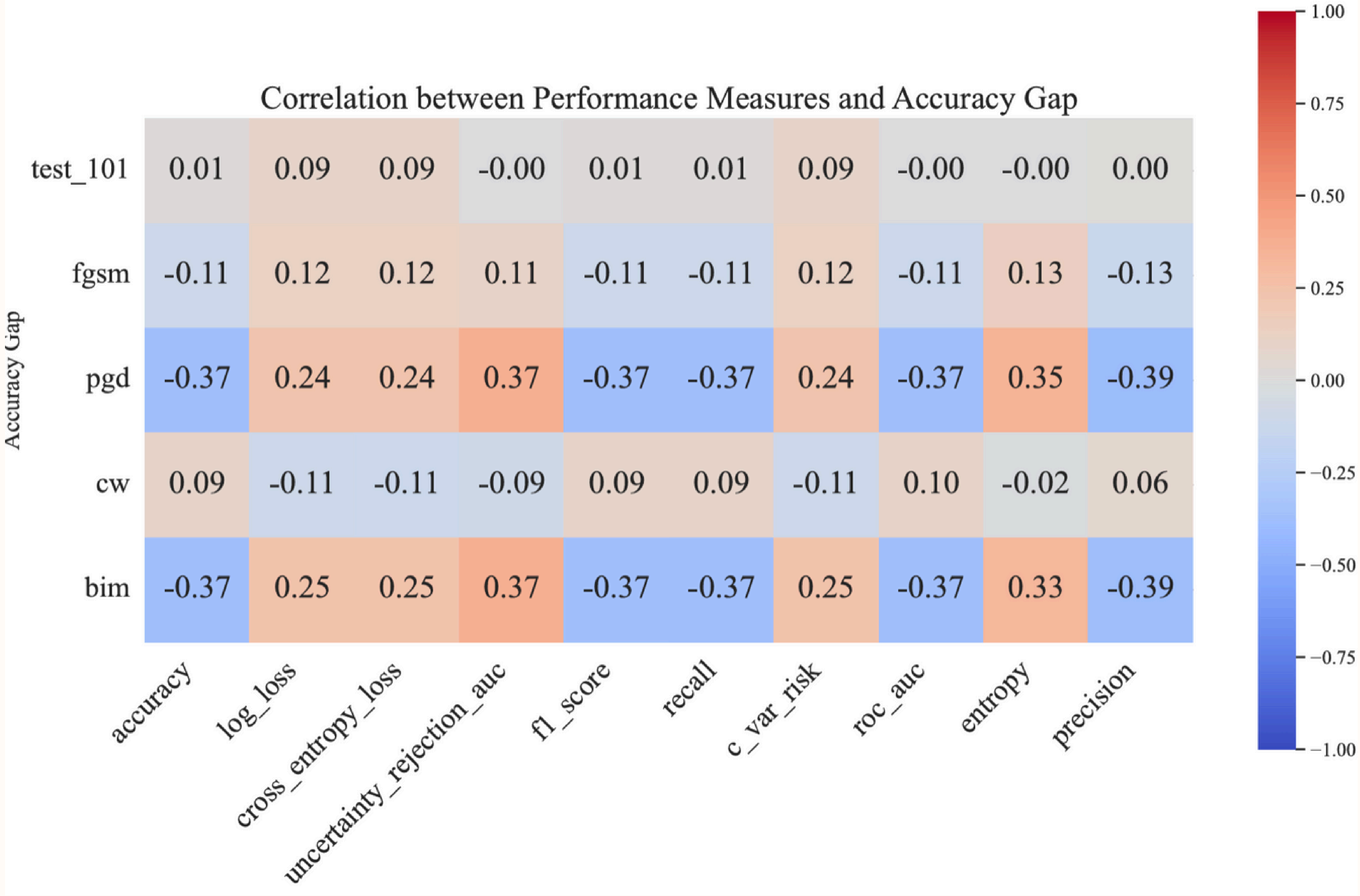
- Requires arbitrary hyperparameters
- "Source" code and literature Pseudo-code don't match
- **LONG** computational times (~2h per model)

```python
"""Algorithms for searching sigmas'"""

import numpy as np
import tensorflow as tf

################################################################################
###################################Utils######################################
################################################################################

def add_noise_to_variables(variables):
  """Create tf ops for adding noise to a list of variables."""
  perturbation_ph = {}
  add_perturbation_op = []
  subtract_perturbation_op = []
  for v in variables:
    perturbation_ph[v] = tf.placeholder(
        tf.float32, shape=v.get_shape().as_list())
    add_perturbation_op.append(tf.assign_add(v, perturbation_ph[v]))
    subtract_perturbation_op.append(tf.assign_add(v, -perturbation_ph[v]))
  return perturbation_ph, add_perturbation_op, subtract_perturbation_op


def get_gaussian_noise_feed_dict(ph_list, scale):
  """Get noise with standard deviation of scale."""
  feed_dict = {}
  for ph in ph_list:
    feed_dict[ph] = np.random.normal(
        scale=scale, size=ph.get_shape().as_list())
  return feed_dict


def flatten_and_concat(variable_list):
  variable_list = [tf.reshape(v, [-1]) for v in variable_list]
  return tf.concat(variable_list, axis=0)


def norm_of_weights(weights):
  flat_weights = [np.reshape(w, -1) for w in weights]
  concat_weight = np.concatenate(flat_weights)
  weight_norm = np.linalg.norm(concat_weight)
  return weight_norm

################################################################################
##################################PacBayes####################################
################################################################################
```

RESULTS

HOW DID ROBUSTNESS PERFORM?

Not well....



RESULTS

However!

The standard notion of robustness - Accuracy Gap - is limiting.
We decided to assess robustness in **two** alternative ways

RESULTS

UPDATED DEFINITION

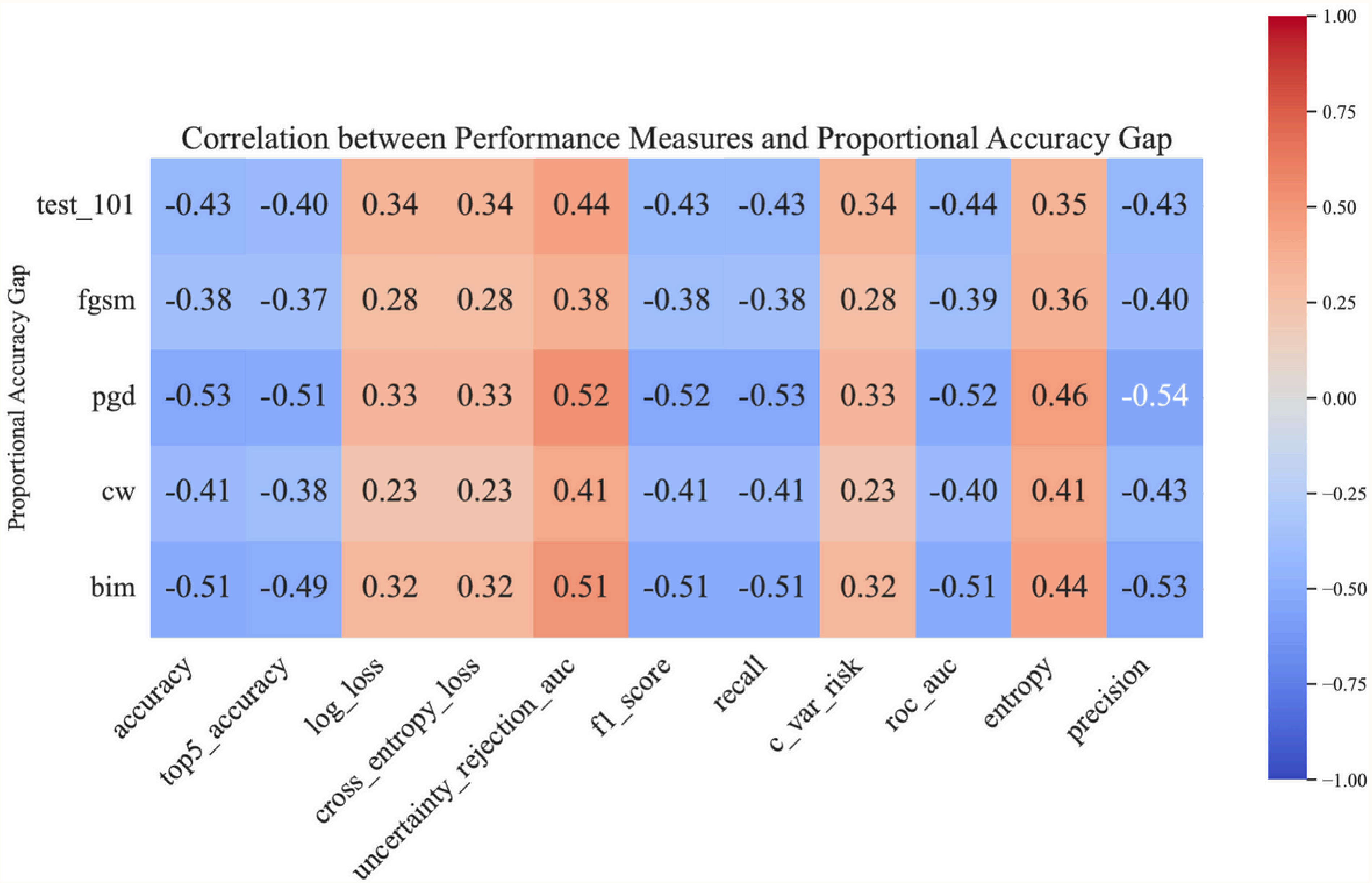
Uncertainty Robustness: How does the model certainty (Log-Loss) change in out-of-distribution scenarios?

Relative Robustness: What is the proportional drop in Accuracy/Certainty?

RESULTS

NEW INSIGHTS!

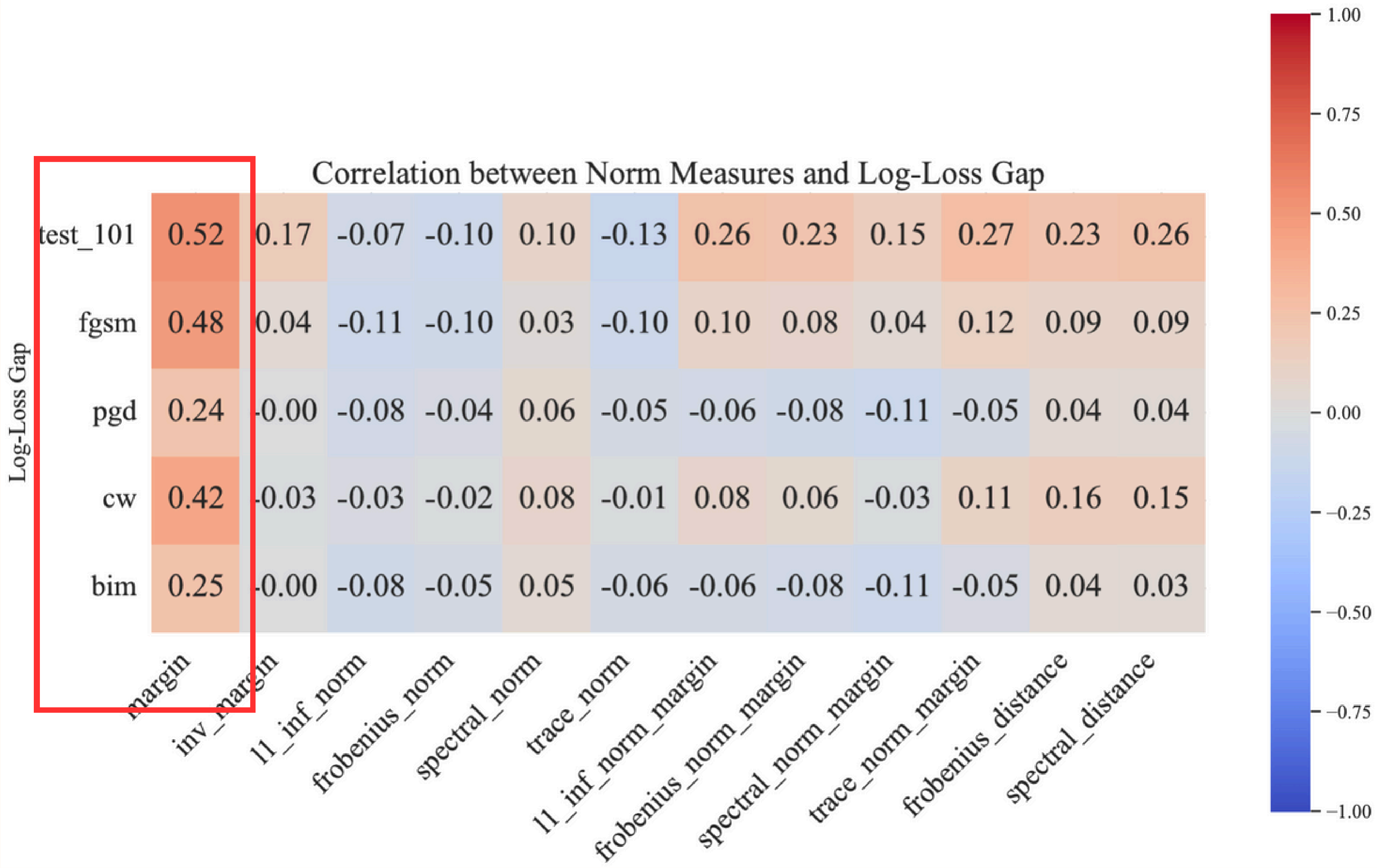
1. In-distribution Performance and Model Weight Norms over Margin moderately predict relative robustness.



RESULTS

NEW INSIGHTS!

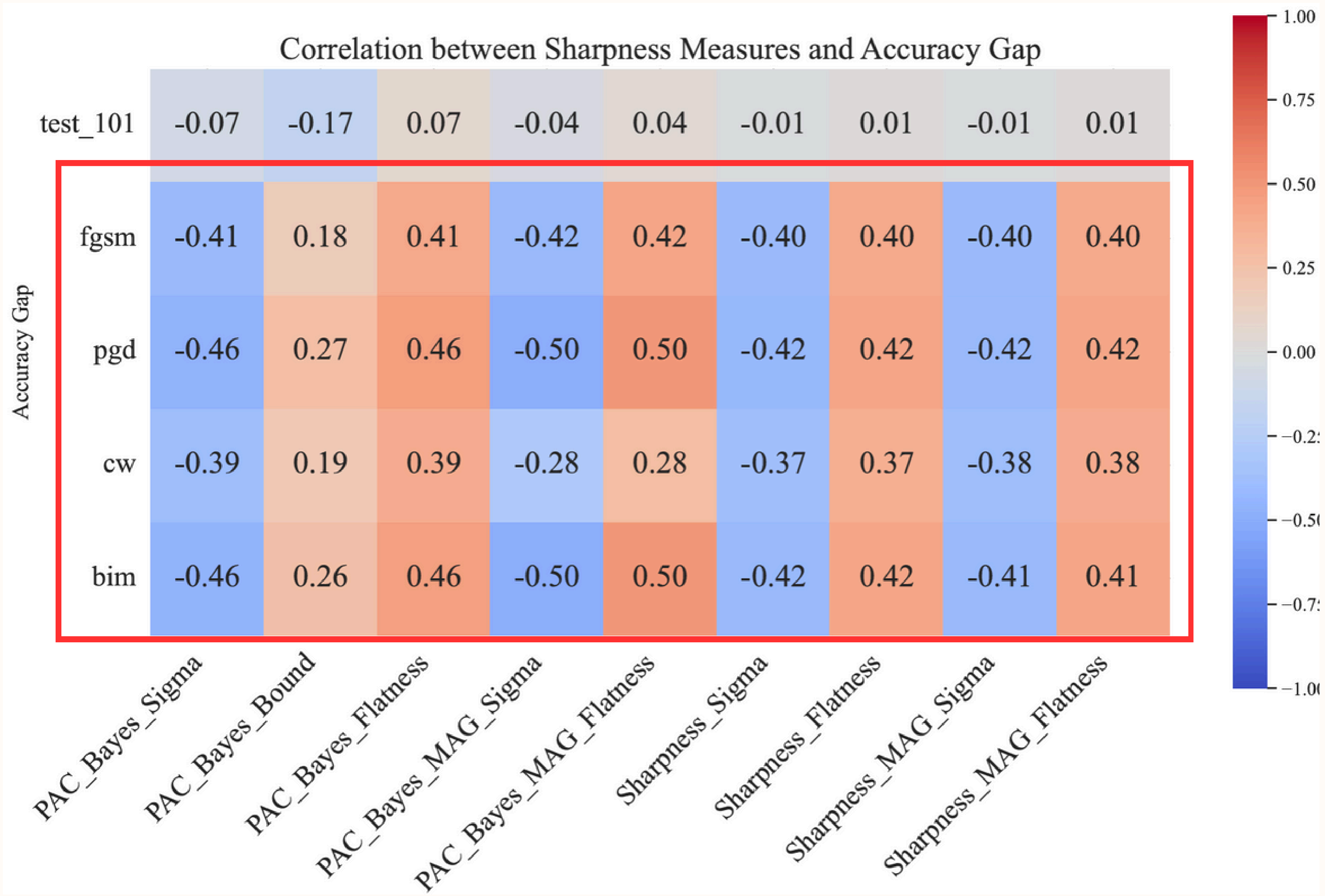
2. Margin and Entropy are moderately correlated with uncertainty robustness.



RESULTS

NEW INSIGHTS!

3. Sharpness moderately predicts relative robustness to adversarial attacks.



NEXT STEPS?

New data

More Models

Different Metrics

Black-Box Attacks

THANK YOU!

QUESTIONS?