MaNo: Exploiting Matrix Norm for Unsupervised Accuracy Estimation

Renchunzi Xie[*1] **Ambroise Odonnat**[*23] **Vasilii Feofanov**[*2] **Weijian Deng**[4] **Jianfeng Zhang**[2] **Bo An**[1]

[*]Equal contribution [1]NTU [2]Huawei Noah's Ark Lab [3]Inria [4]ANU

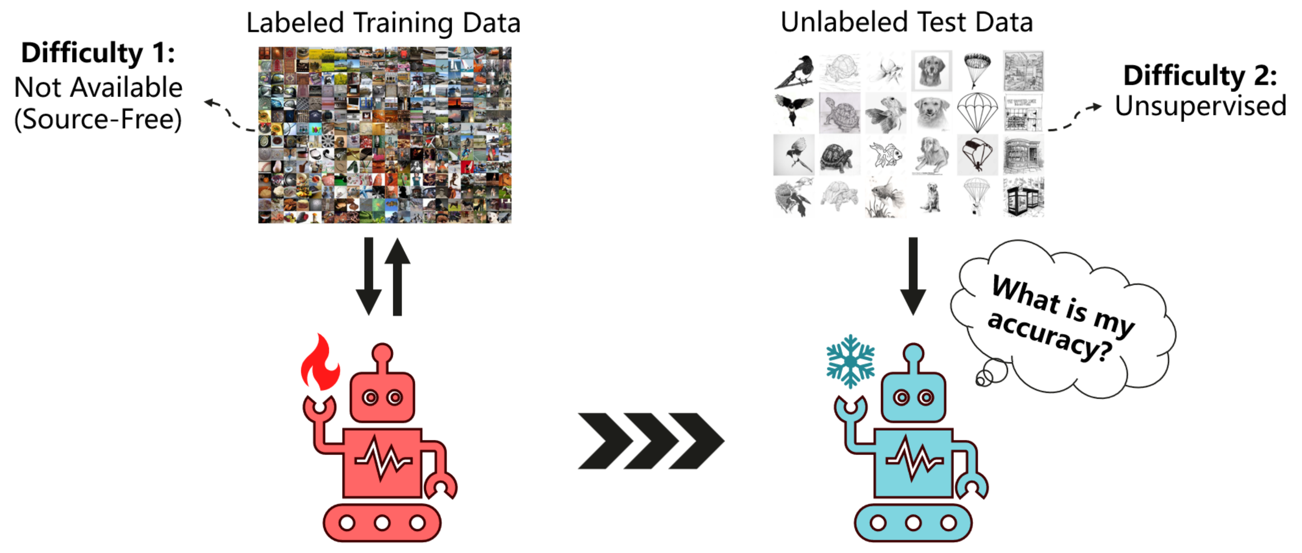
TL;DR

- Predicting generalization performance under distribution shifts is challenging
- Most methods use logits without dealing with miscalibration cases
- We propose MaNo, a **theoretically grounded** estimation approach
- It automatically takes into account miscalibration scenarios
- It can be applied to ResNets, ConvNext, and ViT architectures
- Benefits: **SOTA**, **efficient**, **architecture agnostic**, **robust**

Problem Setup

Goal: given a pre-trained model f, predict its performance on a test set $\mathcal{D}_{\text{test}}$.

- *Input:* a pre-trained model f and test data $\mathcal{D}_{\text{test}}$.
- *Distribution shift:* $p_S \neq p_T$ where training data $\sim p_S$ and test data $\sim p_T$.
- *Output:* an estimation score $\mathcal{S}(f, \mathcal{D}_{\text{test}})$ that linearly correlates the true accuracy.



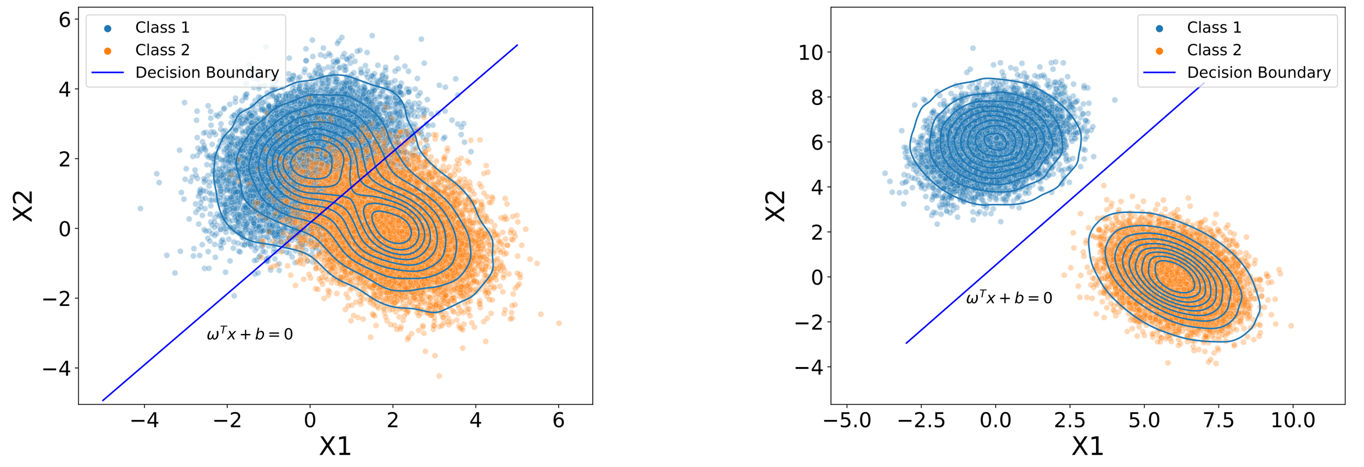
This is a challenging task often occurring in real-world scenarios.

Motivation

Question 1: *Why are logits informative of generalization performance?*
Question 2: *How to alleviate the overconfidence issues of logits-based methods?*

Logits Reflect Distances to Decision Boundaries

- Decision boundary of class k is the hyperplane $\{\mathbf{z}' \in \mathbb{R}^q \mid \boldsymbol{\omega}_k^\top \mathbf{z}' = 0\}$.
- Distance from a point $\mathbf{z}$ this hyperplane is $\mathrm{d}(\boldsymbol{\omega}_k, \mathbf{z}) = |\boldsymbol{\omega}_k^\top \mathbf{z}| / \|\boldsymbol{\omega}_k\|$.
- Logits reflects decision to decision boundary as $|\mathbf{q}_k| = |\boldsymbol{\omega}_k^\top \mathbf{z}| \propto d(\boldsymbol{\omega}_k, \mathbf{z})$.
- **Low-density assumption**: misclassified samples are closer to decision boundaries.



High-density region Low-density region

Logits (in absolute values) positively correlated to generalization performance.

MaNo: A Simple Three-Step Recipe

- *Input:* Pre-trained model f, test dataset $\mathcal{D}_{\text{test}} = \{\mathbf{x}_i\}_{i=1}^N$.
- *Inference:* Recover logits $\mathbf{q}_i = f(\mathbf{x}_i)$.
- *Criterion:* $\Phi(\mathcal{D}_{\text{test}}) = \mathrm{KL}(\text{uniform}||\text{softmax proba})$

$$1)\quad v(\mathbf{q}_i) = \begin{cases} 1 + \mathbf{q}_i + \frac{\mathbf{q}_i^2}{2}, & \text{if } \Phi(\mathcal{D}_{\text{test}}) \leq \eta \\ \exp(\mathbf{q}_i), & \text{if } \Phi(\mathcal{D}_{\text{test}}) > \eta \end{cases}$$

$$2)\quad \sigma(\mathbf{q}_i) = \frac{v(\mathbf{q}_i)}{\Sigma_{k=1}^K v(\mathbf{q}_i)_k} \in \Delta_K$$

$$3)\quad \mathcal{S}(f, \mathcal{D}_{\text{test}}) = \frac{1}{\sqrt[p]{NK}} \|\mathbf{Q}\|_p = \left(\frac{1}{NK} \sum_{i=1}^{N} \sum_{k=1}^{K} |\sigma(\mathbf{q}_i)_k|^p \right)^{\frac{1}{p}}$$

MaNo is simple yet efficient and we prove that it captures the model's uncertainty.

Challenging Setting: Natural Shift

- *Natural shift:* most difficult and most realistic benchmarks.
- The normalization we proposed corrects the issues with softmax's overconfidence.
- MaNo significantly outperforms other baseline methods.

Dataset	Model	ConfScore R^2	ConfScore ρ	Nuclear R^2	Nuclear ρ	MaNo R^2	MaNo ρ
PACS	ResNet18	0.594	0.755	0.609	0.874	**0.827**	**0.909**
	ResNet50	0.070	0.069	0.611	0.888	**0.923**	**0.958**
	WRN-50-2	0.646	0.678	0.607	0.867	**0.924**	**0.972**
	Average	0.437	0.501	0.609	0.876	**0.891**	**0.946**
Office-Home	ResNet18	0.795	0.909	0.692	0.783	**0.926**	**0.930**
	ResNet50	0.769	0.895	0.731	0.895	**0.838**	**0.916**
	WRN-50-2	0.741	0.874	0.766	0.874	**0.800**	**0.895**
	Average	0.768	0.892	0.730	0.850	**0.854**	**0.913**
DomainNet	ResNet18	0.670	0.736	0.758	0.789	**0.902**	**0.937**
	ResNet50	0.570	0.706	0.809	0.879	**0.910**	**0.950**
	WRN-50-2	0.774	0.874	0.850	0.911	**0.893**	**0.978**
	Average	0.671	0.722	0.805	0.895	**0.899**	**0.949**
RR1-WILDS	ResNet18	0.951	**1.000**	0.885	**1.000**	**0.983**	**1.000**
	ResNet50	0.918	**1.000**	0.906	**1.000**	**0.978**	**1.000**
	WRN-50-2	0.941	**1.000**	0.840	**1.000**	**0.969**	**1.000**
	Average	0.937	**1.000**	0.877	**1.000**	**0.977**	**1.000**

MaNo significantly outperforms competitors under natural shift.

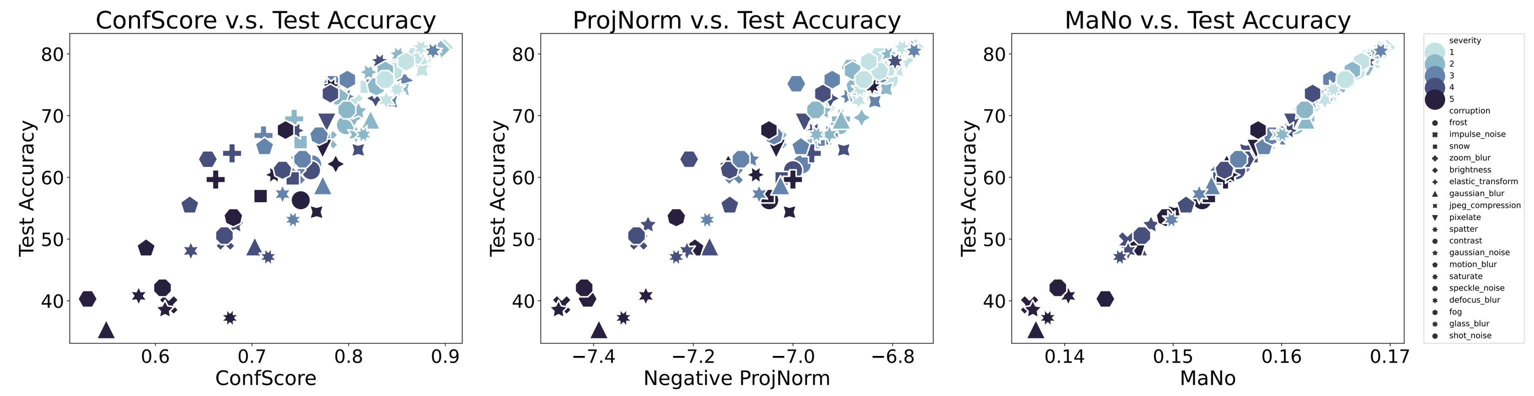
Experimental Results: Better, Faster, Stronger

- Comparison between MaNo and its competitors with metrics ρ and R^2,
- Comparison across several architectures: ResNets, ConvNext, ViT,
- Extensive evaluation with common benchmarks on various distribution shifts.

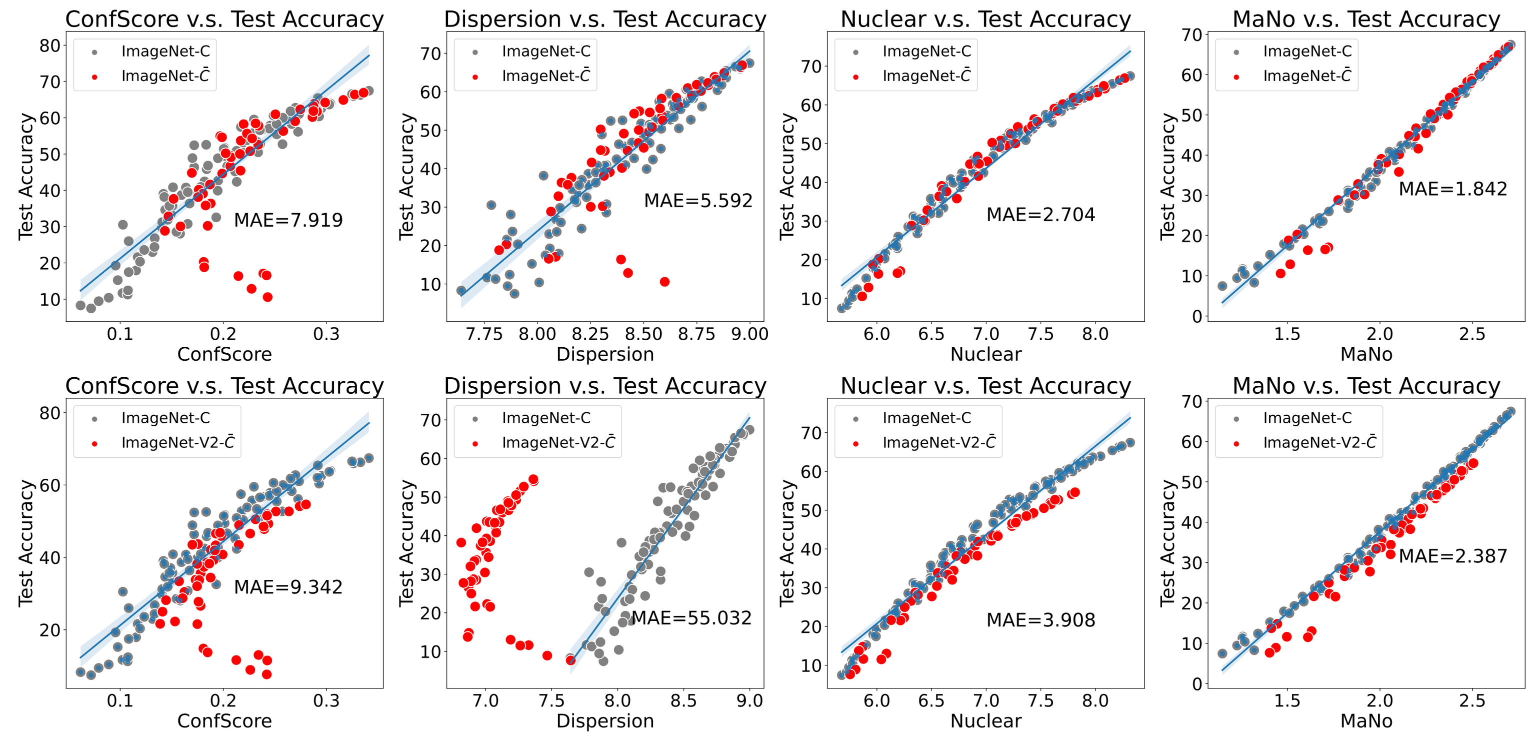
Shift	MaNo -	COT 2024	MDE 2024	Nuclear 2023	Dispersion 2023	ProjNorm 2022
Synthetic	**0.991**	0.988	0.947	0.982	0.960	971
Subpopulation	**0.983**	0.962	0.920	0.973	0.909	897
Natural	**0.905**	0.871	0.436	0.455	0.410	382
Overall improvement		2%	25%	6%	26%	28%

MaNo outperforms all the baselines while being training-free.

Entity-18 (Subpopulation Shift): MaNo linearly correlates with the ground-truth test.



Corruptions on ImageNet (Synthetic Shift): MaNo significantly surpasses its competitors.



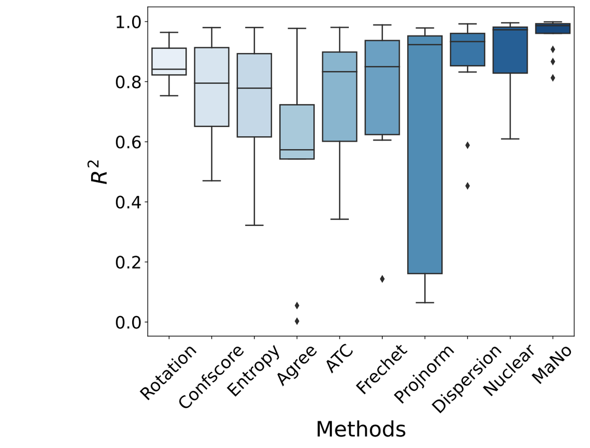
Main References

- **Odonnat et al.** - AISTATS 2023 *$\mathcal{T}$-similarity*
- **Deng et al.** - ICML 2023 *Nuclear*
- **Xie et al.** -**NeurIPS 2024 (this work)** MaNo

Renchunzi Xie Ambroise Odonnat Vasilii Feofanov

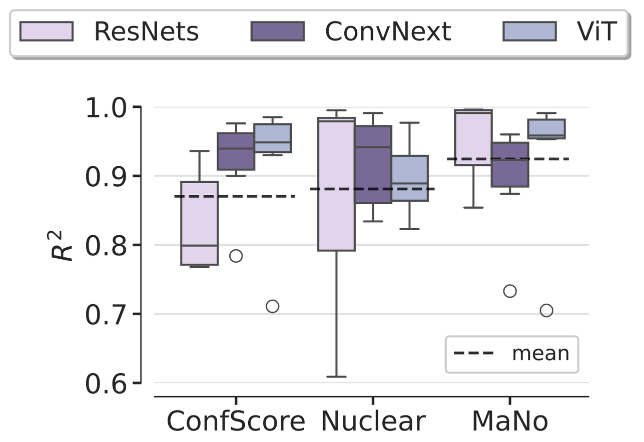
Robustness Analysis

We conducted large-scale experiments and ablations on all the distribution shifts.



Overall, MaNo leads to the best and most robust estimations!

We tested our approach's efficiency and versatility with 3 SOTA architectures.



MaNo is the best approach to use with SOTA architectures!

Take Home Message

Predicting generalization performance under distribution shifts is challenging.
→ Start using MaNo for an **efficient** and **accurate** estimation!

Want to Know More?

paper code

Correspondence to renchunzi.xie@ntu.edu.sg and ambroiseodonnattechnologie@gmail.com