A Brief Talk on Distribution Shift

Qingyao Sun Yucong Liu Minxuan Duan

May 26, 2022

THE UNIVERSITY OF CHICAGO

Outline

1. Problem statement
   - Different Types of Distribution Shift
   - Accuracy Drops and Why
2. Theoretical Explanation
   - Theory of Concept Shift: Information-theoretic Perspective
   - Theory of Covariate Shift: Minimal Stable Variable Set
   - Connection with Adversarial Robustness
3. Methods and Experiments
   - Simple Fixed Shift
   - Our Improvement

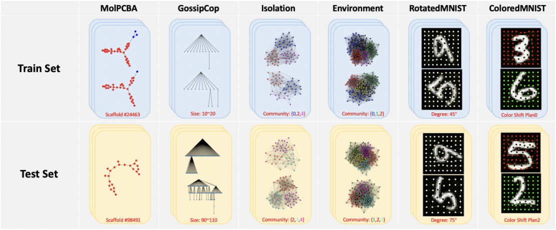
Problem statement

- One of the most common assumptions for machine learning models is that the training and test data are IID.
- In practice, this assumption does not hold.

Distribution Shift

A model is deployed on a data distribution $P_{\text{test}}(X, Y)$ related but different from what it was trained on $P_{\text{train}}(X, Y)$.

- It poses significant robustness challenges.



[1]

[1] Ding, et al. "A Closer Look at Distribution Shifts and Out-of-Distribution Generalization on Graphs." (2021).

Different Types of Distribution Shift

1. **Covariate Shift**: Assume the testing distribution differs from the training distribution in covariate shift only.

$$P_{\text{test}}(X, Y) = P_{\text{test}}(X) P_{\text{train}}(Y \mid X)$$

2. **Label Shift**: Assume that the label marginal can change but the class-conditional distribution remains fixed.

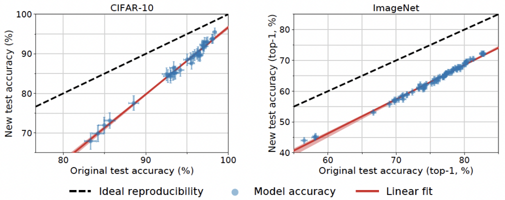
$$P_{\text{test}}(X, Y) = P_{\text{test}}(Y) P_{\text{train}}(X \mid Y)$$

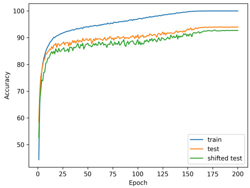
   In some degenerate cases, the label shift and covariate shift assumptions can hold simultaneously.

3. **Concept Shift**: $P(Y \mid X)$ changes across domains.

Accuracy Drops

- [2] builds new test sets for the CIFAR-10 and ImageNet. The accuracy drops range from 3% to 15% on CIFAR-10 and 11% to 14% on ImageNet.
- We also try some naive distribution shifts, e.g. simple bias, simple beautification, filter method used for photos.





[2]Recht, Benjamin, et al. "Do imagenet classifiers generalize to imagenet?." ICML. PMLR, 2019.

Why Accuracy Drops

Take the problem of classification as an example:

- We aim to find a model $\hat{f}$ that minimizes the population loss

$$L_{\mathcal{D}}(\hat{f}) = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \mathbb{I}[\hat{f}(x) \neq y]$$

- Since we do not know the true distribution $\mathcal{D}$, we instead measure the performance via a *test set* $\mathcal{S}$ drawn from $\mathcal{D}$:

$$L_{\mathcal{S}}(\hat{f}) = \frac{1}{|\mathcal{S}|} \sum_{(x,y)\in\mathcal{S}} \mathbb{I}[\hat{f}(x) \neq y]$$

- If we test by collecting a new test set $\mathcal{S}'$ from a data distribution $\mathcal{D}'$ that we carefully control to resemble the original distribution $\mathcal{D}$.

$$L_{\mathcal{S}} - L_{\mathcal{S}'} = \underbrace{(L_{\mathcal{S}} - L_{\mathcal{D}})}_{\text{Adaptivity gap}} + \underbrace{(L_{\mathcal{D}} - L_{\mathcal{D}'})}_{\text{Distribution Gap}} + \underbrace{(L_{\mathcal{D}'} - L_{\mathcal{S}'})}_{\text{Generalization gap}}$$

Theory of Concept Shift [3]

Using the chain rule of mutual information, one can express distribution shift as the sum of two separate components:

$$\underbrace{I(XY;t)}_{\text{Distribution shift}} = \underbrace{I(X;t)}_{\text{Covariate shift}} + \underbrace{I(Y;t \mid X)}_{\text{Concept shift}}$$

where $I(XY;t) = D_{\mathrm{KL}}(P(X,Y|t) \mid P(X,Y))$, $t = \mathrm{test}$ or train.

Proposition

For any model $Q(Y \mid X)$ and $\alpha := \min\{P(t=\mathrm{test}), P(t=\mathrm{train})\}$

$$\underbrace{D_{\mathrm{KL}}\left(P_{Y|X}^{t=\mathrm{train}} \| Q_{Y|X}\right)}_{\text{Train error}} + \underbrace{D_{\mathrm{KL}}\left(P_{Y|X}^{t=\mathrm{test}} \| Q_{Y|X}\right)}_{\text{Test error}} \geq \frac{1}{1-\alpha} I(Y;t \mid X)$$

Thus, whenever the selection induces concept shift, any sufficiently flexible model must incur in strictly positive test error.

[3] Federici, Marco, Ryota Tomioka, and Patrick Forré. "An information-theoretic approach to distribution shifts." Advances in Neural Information Processing Systems 34 (2021).

Theory of Covariate Shift[4]

For common loss functions, the covariate shift generalization problem can be tackled by the **minimal stable variable set** which satisfies the condition of the following theorem.

Theorem (Informal Version)

A subset of variables $S \subseteq X$ that can approximate the target $\mathbb{E}_{P_{\text{test}}}[Y|X]$ if and only if it satisfies $\mathbb{E}_{P_{\text{train}}}[Y|S] = \mathbb{E}_{P_{\text{train}}}[Y|X]$.

The existence and uniqueness of such variables are guaranteed.

Theorem (Informal Version)

Under ideal conditions (perfectly learned sample weights and infinite samples),

- if X_i is not in the minimal stable variable set, stable learning algorithms could filter it out, and
- otherwise, there exists sample weighting functions with which stable learning algorithms could identify X_i.

[4] Xu, et al. "Why Stable Learning Works? A Theory of Covariate Shift Generalization." arXiv(2021).

Connection with Adversarial Robustness[5]

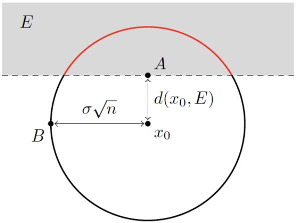
Two types of robustness

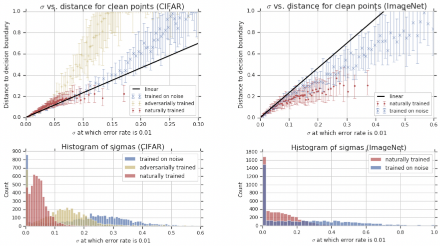
- Adversarial robustness to small-worst case perturbations of the input

$$\mathbb{P}_{x\sim p}[d(x, E) > \epsilon]$$

- Corruption robustness to distributional shift

$$\mathbb{P}_{x\sim q}[x \notin E]$$



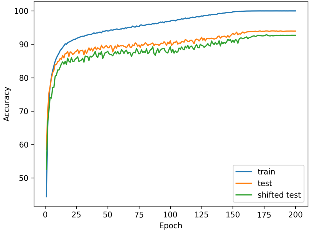


[5] Ford, Nic, et al. "Adversarial examples are a natural consequence of test error in noise." arXiv(2019).

Simple Fixed Shift

Intuition

What may happen under a simple distribution shift on test data?



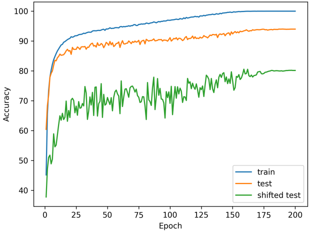


Figure: Performance of Resnet-18 on CIFAR-10

Simple Fixed Shift

- Robustness under distribution shift exists.
- Models are not able to undertake every distribution shift.

Open Question

How to define/check if one distribution shift is a reasonable shift?
How to measure a distribution shift which doesn't destroy the original structure of data?

A Fine-Grained Analysis on Distribution Shift

Framework

A classical Bayesian structure

$$z \sim p(z) \qquad y^i \sim p(y^i|z) \quad i = 1 \dots K \qquad \boldsymbol{x} \sim p(\boldsymbol{x}|z) \tag{1}$$

By a simple refactorization, we can write

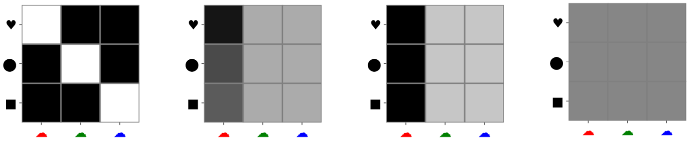
$$p(y^{1:K}, \boldsymbol{x}) = p(y^{1:K}) \int p(\boldsymbol{x}|z) p(z|y^{1:K}) dz = p(y^{1:K}) p(\boldsymbol{x}|y^{1:K}).$$

The distribution shift discussed in this paper is **label shift**

$$\begin{aligned} p(y^{1:K}) \neq & p_{\text{train}}(y^{1:K}) \neq & p_{\text{test}}(y^{1:K}) \\ p(\boldsymbol{x}|y^{1:K}) = & p_{\text{train}}(\boldsymbol{x}|y^{1:K}) = & p_{\text{test}}(\boldsymbol{x}|y^{1:K}) \end{aligned} \tag{2}$$

A Fine-Grained Analysis on Distribution Shift

Training distribution



(a) p_{train}: **SC.** (b) p_{train}: **LDD.** (c) p_{train}: **UDS.** (d) p_{test}: y^l, y^a **are IID.**

Figure 1: Visualization of the joint distribution for the different shifts we consider on the DSPRITES example. The lighter the color, the more likely the given sample. figure 1a-1c visualise different shifts over $p_{\text{train}}(y^l, y^a)$ discussed in 2.2: *spurious correlation* (SC), *low-data drift* (LDD), and *unseen data shift* (UDS). figure 1d visualises the test set, where the attributes are uniformly distributed.

Figure: Spurious correlation, Low-data drift, and Unseen data shift

Test distribution

We assume that the attributes are distributed uniformly. This is desirable, as all attributes are represented and a-priori independent.

A Fine-Grained Analysis on Distribution Shift

Deepmind has a lot of computing power, so they can afford to do a grid search over everything

Experiment

We provide a holistic analysis of current state-of-the-art methods by evaluating 19 distinct methods grouped into five categories across both synthetic and real-world datasets. Overall, we train more than 85K models.

......

We evaluate the 19 different methods across these six datasets, three distribution shifts, varying label noise, and dataset size.

Not discussing the details, but see results below.

A Fine-Grained Analysis on Distribution Shift

Impact of Spurious Correlation

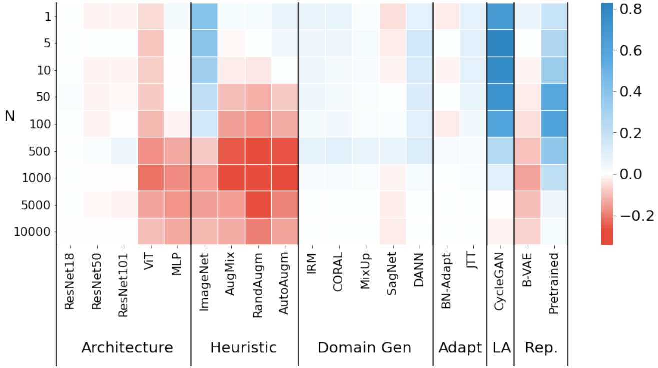


Figure: We use all correlated samples and vary the number of samples N from the true, uncorrelated distribution. We plot the percentage change over the baseline ResNet, averaged over all seeds and datasets.

A Fine-Grained Analysis on Distribution Shift

Impact of Low-data Drift

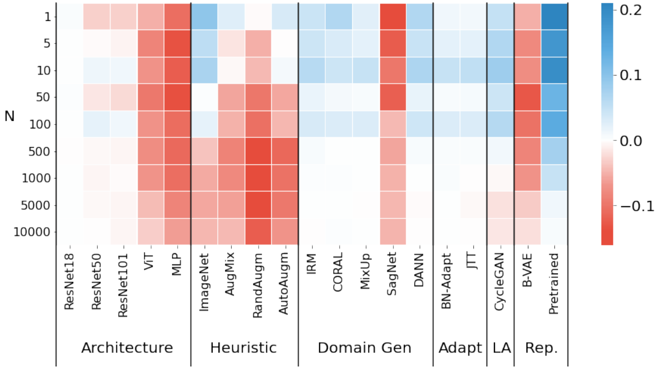


Figure: We use all samples from the high data regions and vary the number of samples N from the low-data region. We plot the percentage change over the baseline ResNet, averaged over all seeds and datasets.

A Fine-Grained Analysis on Distribution Shift

Impact of Unseen-data Drift

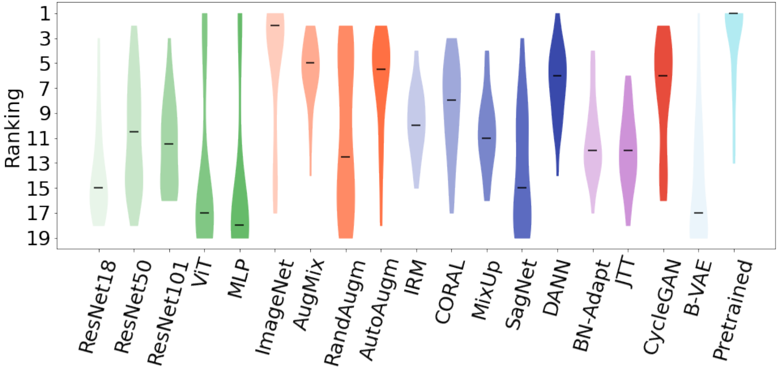


Figure: We rank the methods (where best is 1, worst 19) for each dataset and seed and plot the rankings, with the overall median rank as the black bar.

A Fine-Grained Analysis on Distribution Shift

Takeways

- Pretraining is a powerful tool across different shifts and datasets.
- Heuristic augmentation improves generalization if the augmentation describes an attribute.
- Learned data augmentation is effective across different conditions and distribution shifts.

What can be improved?

We find the uniform assumption on the test set unrealistic, e.g.

- The types of equipment may not distribute equally across all hospitals,
- The proportion of patients with a tumor is not necessary 50%,
- We are forced to consider pregnant men if "sex" and "pregnancy" are two of the attributes.

Their assumption can be used to avoid bias and improve fairness, though.

Our experiment

Model

We took inspiration from the paper, but put a distribution shift on z instead of y.

$$z \sim p(z) \qquad y^i \sim p(y^i|z) \quad i = 1 \dots K \qquad \boldsymbol{x} \sim p(\boldsymbol{x}|z) \tag{3}$$

where $p(z)$ is different on the training and test set, but both $p(y^i|z)$ and $p(\boldsymbol{x}|z)$ stays the same. Note that we have covariate shift, label shift, and concept shift.

Specifically, $p(y^i|z)$ is a point-mass distribution and $p(\boldsymbol{x}|z)$ is a normal distribution.

Experiment setup

Dataset

We used the **3dshapes** dataset from Deepmind, which contains the following latent factor values, with no noise added.

floor hue	10 values linearly spaced in [0, 1]
wall hue	10 values linearly spaced in [0, 1]
object hue	10 values linearly spaced in [0, 1]
scale	8 values linearly spaced in [0, 1]
shape	4 values in [0, 1, 2, 3]
orientation	15 values linearly spaced in [-30, 30]

This gives us a total of $10 \times 10 \times 10 \times 8 \times 4 \times 15 = 480000$ images.

In our experiment, z is the latent factor, $\boldsymbol{x}$ is the corresponding image with a Gaussian noise $\mathrm{N}(0, \sigma^2)$, and $\boldsymbol{y}$ is a pre-determined component of z, e.g. scale.

Experiment setup

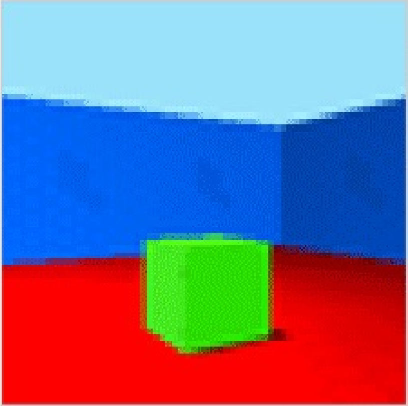


Figure: A sample image from 3dshapes

Experiment setup

Overview

1. Train a neural network with training distribution $\mathcal{D}_{\mathrm{train}\,i}, i = 1, \cdots 480000$.
2. Evaluate the accuracy on each of the 480000 images, i.e. population. Denote it with A_i, for $i \in \{1, \cdots 480000\}$.
3. Calculate the expected test accuracy if the test distribution was $\mathcal{D}_{\mathrm{test}}$ with $\sum_{i=1}^{480000} A_i \mathcal{D}_{\mathrm{test}\,i}$.

Which $\mathcal{D}_{\mathrm{train}}$ and $\mathcal{D}_{\mathrm{test}}$ are we going to use?

Experiment setup

Distributions

Each of the following distributions is used as $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{test}}$, i.e. $12 \times 12 = 144$ pairs.

```
weights = {
    'uniform': np.ones(480000),
    'scale': scale,
    'shape': shape,
    'orientation': orientation,
    'scale+shape': scale + shape,
    'scale+orientation': scale + orientation,
    'shape+orientation': shape + orientation,
    'scale+shape+orientation': scale + shape + orientation,
    'scale*shape': scale * shape,
    'scale*orientation': scale * orientation,
    'shape*orientation': shape * orientation,
    'scale*shape*orientation': scale * shape * orientation,
}
```

Figure: Distribution of attributes

Experiment setup

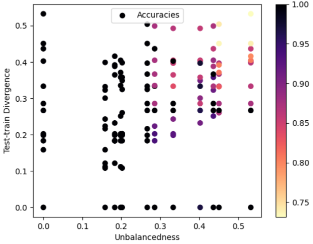
Hyperparameters

Label	Scale
Noise Scale (σ)	0.01
Device count	8
Local batch size	2048
Training batches	512
Learning rate	0.001

We note that using a σ of 0.1 and 0.001 produces roughly the same result.

Experiment result

ResNet18



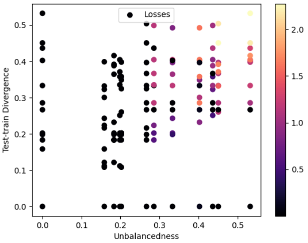


Figure: Robustness of ResNet18

Here **Unbalancedness** means the total variation distance between a uniform distribution and $\mathcal{D}_{\text{train}}$, and **Test-train Divergence** means the total variation distance between $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{test}}$.

Experiment result

ResNet34

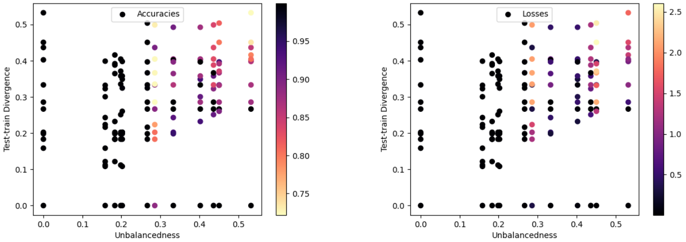


Figure: Robustness of ResNet34

Experiment result

ResNet50

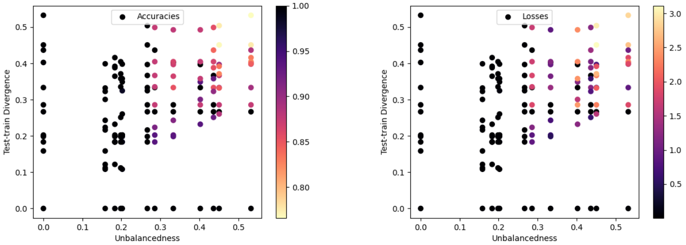


Figure: Robustness of ResNet50

Thank you

Thank you for listening