# Testing Generalization in Alternative Causal Structures on DomainBed

**Soren Dunn**[1] **Noah Fischer**[1]
[1]University of Chicago
sorendunn@uchicago.edu
nfischer@uchicago.edu

## Abstract

This project investigates the generalization of deep correlation alignment (Deep CORAL), deep domain generalization via conditional invariant adversarial networks (C-DANN), and group distributionally robust optimization (GroupDRO) algorithms on two novel synthetic datasets. Unlike most commonly used benchmark datasets, both of these datasets have spurious features caused by the image labels, mimicking potential causal structures in the medical imaging domain. The results of these training runs indicate that even though these algorithms may achieve similar performance on popular domain generalization benchmarks, their performance in practice may drastically depend on the causal structure of the environments used for training and deployment. [1]

## 1 Introduction and Related Work

Machine learning systems frequently fail to generalize to domains not tested during training. Reliance on features only spuriously related to the prediction target results in degraded performance when testing in domains which exhibit alternate relationships between the spurious features and prediction target. This failure can manifest in domains as diverse self-driving cars (Volk et al. [2019]), medical systems (Castro et al. [2020]), and image classification (Geirhos et al. [2022]).

Cognizant of these issues, many algorithms have been developed to help improve domain generalization. DomainBed is a PyTorch suite containing many of the proposed algorithms for domain generalization and datasets to test them on (Gulrajani and Lopez-Paz [2020]). All of the datasets in the suite contain several varied environments and all of the algorithms currently contained in the suite perform to within 2 percent accuracy of each other on the available set of seven benchmarks in the suite. Due to limited compute access, we selected three of the algorithms in this suite to test based on high performance on the DomainBed suite as well as having a reputation for performing well on other domain generalization benchmarks. We tested these algorithms on two variations of the colored Modified National Institute of Standards and Technology (MNIST) dataset not contained in the original suite.

Deep CORAL (Sun and Saenko [2016]), the first of the algorithms tested, is an adaptation of the correlation alignment (CORAL) method to deep neural networks. It trains a deep neural network with the following loss:

$$l_{CLASS} + \sum_{i=1}^{t} \lambda_i l_{CORAL}$$

[1]We publicly release our code at https://github.com/sorendunn/Testing-Generalization-in-Alternate-Causal-Structures-On-DomainBed

Preprint. Under review.

where $l_{CLASS}$ is the classification loss, t denotes the number of CORAL loss layers in the network, and $\lambda$ is a weight that trades off the adaptation with classification accuracy.

$l_{CORAL}$ is the specific CORAL loss specified below:

$$l_{CORAL} = \frac{1}{4d^2}\|C_S - C_T\|_F^2$$

where F denotes the Frobineous norm and $C_S$ and $C_T$ are the covariance matrices of the source and target (new environment) data respectively.

C-DANN (Li et al. [2018]) (the second of these algorithms tested) focuses on solving the problem that the invariance of the conditional distribution P(Y|X) isn't guaranteed, causing a possible mismatch between source and target distributions.

The key idea of this approach is the application of a minimax game strategy, leveraging two kinds of minimax values: the class-conditional minimax value and the class prior-normalized marginal minimax value.

The class-conditional minimax value is computed based on the examples from different domains but the same class. By optimizing this minimax value, the invariance of class-conditional distributions among domains can be ensured, thus making the feature distribution of each class similar across domains.

The class prior-normalized marginal minimax value, on the other hand, aims to normalize the class prior distribution, which is especially helpful when the sample size is not large and overfitting can occur. It can help to improve learning of domain-invariant features by ensuring the equality of class prior-normalized marginal distributions across domains.

The method employs a conditional invariant adversarial network that consists of four components: a representation learning network, an image classification network, a class-conditional domain network, and a class prior-normalized domain network. These networks are trained to minimize the joint loss that includes image classification loss, class-conditional domain loss, and class prior-normalized domain loss.

The last algorithm tested was group distributionally robust optimization (GroupDRO) (Li et al. [2018]). This approach consists of coupling training a method to improve the accuracy of the training on the worst-case domain with increased regularization, which has lead to better empirical performance than distributionally robust optimization methods alone.

Almost all of the datasets in the DomainBed suite feature the causal structure depicted in Figure 1.

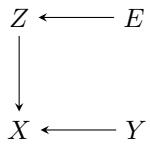


Figure 1: Typical Domainbed Dataset Causal Structure

E represents the different environments, X represents the image being trained on, Y represents the target label, and Z represents spurious features in the image which depend on the environment. For example, Z represents the rotation in the RotatedMNIST dataset, the style of image in the PACS, Office-Home, and DomainNet datasets, and the background of the image in the TerraIncognita and SVIRO datasets. None of these attributes directly cause or are clearly caused by the underlying classification of the image (thus there is no definite case where an arrow should be drawn between Z and Y). Though this type of causal structure may correspond to some cases with domain generalization issues, it would be useful to test if the algorithms in the test suite work for generalizations with different forms of causal structure.

The only dataset in the suite which clearly breaks out of this causal structure is the ColoredMNIST handwriting dataset. The ColoredMNIST dataset explicitly creates a causal effect between the image label and the spurious color feature. The colored MNIST causal structure is shown in Figure 2.

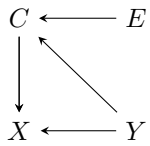


Figure 2: ColoredMNIST Causal Structure

The only difference between this structure and the structure in Figure 1 is the causal pathway between Y and Z. Though this dataset clearly breaks from the type of causal structure present in Figure 1, it only tests a particularly simple instance of having spurious features caused by the image label. Therefore we set out to create datasets with more complicated causal structures based on ColoredMNIST.

First to ground the motivation for worrying about such causal structures, consider the case of trying to identify patients who have a tumor where the spurious feature is the quality of machine being used for imaging. In this case, a certain severity of tumor may cause a patient to have more severe symptoms which may cause them to spend the time to drive to a more expensive hospital which may have a more expensive imaging device which leaves artifacts on the final image. Thus the target of prediction (the severity of an individual's tumor) may effect a spurious feature in the image (artifacts left by the type of machine used for imaging) which if varying across environments (hospitals). In situations such as this, domain generalization techniques which only work for spurious features which are mostly unrelated to the prediction target may fail since the spurious feature is clearly a causal descendant of the prediction target.

To test algorithms on this sort of causal structure we created a dataset called ColoredRotatedMNIST1 (CRM1) built off of the original ColoredMNIST. First recall how ColoredMNIST was created: each image in MNIST was labeled with whether the digit depicted was under 5, this label was flipped with probability 0.25, the digit was colored green if the label was 1 and 0 otherwise, and finally the color was flipped with a probability depending on the environment (the environments used here have probability 0.1, 0.2, and 0.9 respectively). CRM1 adds a rotation to ColoredMNIST; each digit has a 0.75 probability of being rotated with a number of degrees depending on the environment (15 degrees for environment 1, 50 degrees for environment 2, and 90 degrees for environment 3). This additional structure tests if the algorithms can account for multiple features which are different across environments where some are and others aren't effected by the digit classification Y. The resulting directed acyclic graph (DAG) is shown in Figure 3.

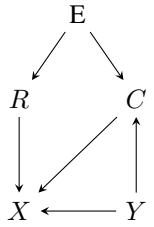


Figure 3: ColoredRotatedMNIST1 Causal Structure

Here, the environment causally affects both color and rotation, and the label affects color as well. E, X, and Y are defined as in Figure 1, R is the image roation, and C is the image color. The second iteration we called ColoredRotatedMNIST2 which adds an additional causal relationship between rotation and color. The probability of the color being flipped is modified to be the product of the probability of that digit being flipped based on the environment and the angle that digit was rotated at divided by 100.

Figure 4 displays the same causal structure as Figure 3 with an additional causal arrow between the rotation and the color.

With these new datasets, we hope to study the effects of layered spurious features on predictive accuracy.

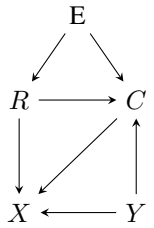


Figure 4: ColoredRotatedMNIST2 Causal Structure

## 2 Methodology

For each dataset, we trained three convolutional neural networks from scratch for each algorithm tested. Each network trained on two of the three domains for each dataset, testing on the third. In order to determine the batch size, learning rate, weight decay, and other hyperparameters for the networks we used the best-performing hyperparameters from the original DomainBed training run on the ColoredMNIST dataset. These hyperparameters were available for all three of the algorithms tested. Training these models took two hours using 27 CPU's. Each model trained for 5000 steps with the final oracle accuracy (accuracy using data from the testing environment) and accuracy exported from each run.

## 3 Results

The full results for these experiments can be found in the appendix, however the plot below summarize the accuracy of the different algorithms across datasets.

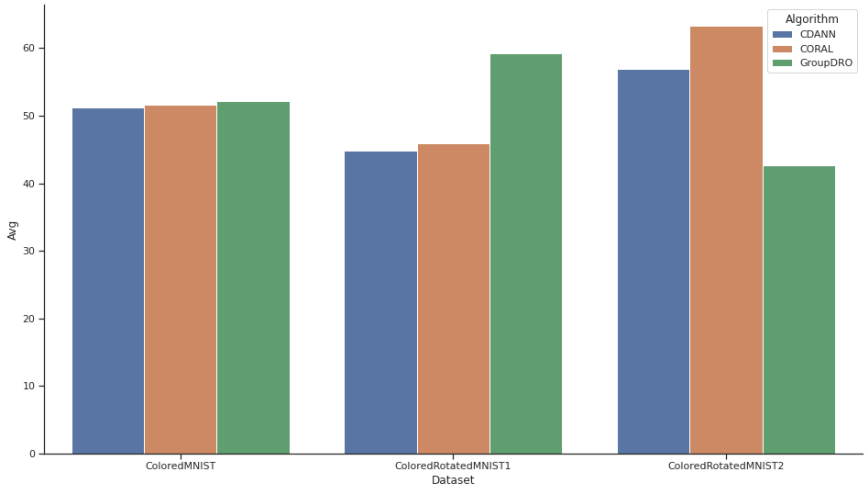


Figure 5: Average Accuracy of Selected Algorithms on ColoredRotatedMNIST1 and ColoredRotatedMNIST2

All the models performed almost identically on all three test environments for the original ColoredMNIST dataset. However the three models achieved wildly differing accuracy on the new datasets. We find it unlikely that the models didn't converge since the accuracy of the methods on ColoredMNIST matched the accuracy from the original DomainBed run and the methods tended to achieve still higher accuracy on the modified datasets than the original one despite the more complicated causal structure.

As can be seen in the full run results in Appendix A, in the original ColoredMNIST dataset holding out the third training environment (with a 90 percent probability of flipping the the color of the image) decreased the models' accuracy from around 70 percent to around 10 percent for all algorithms. However on the new datasets holding out the third environment did not always result in the lowest out-of-sample accuracy. One reason for this in CRM2 was that since the probability of flipping the color (dependent on the environment) was adjusted based on the number of degrees the image was rotated (including 0 for images that were not rotated), the third environment was not dramatically from the other two. This fact likely resulted in holding out that environment not leading to as significant a drop in accuracy.

Each of the algorithms used had unique reasons for failing to generalize to the novel causal structure. In the case of Deep CORAL, the lack of generalization between domains in the new datasets is likely because Deep CORAL's process of attempting to minimize the difference in the covariances between the different environments was unable to account for both a spurious feature which was a causal descendant of Y along with another spurious feature which was not a causal descendant of Y.

The GroupDRO method had particularly high variation between sets of test and train environments. Most dramatically, for CRM2 GroupDRO went from 100 percent accuracy to 9.2 percent accuracy when holding out environment 1 instead of environment 2. This dramatic difference in accuracy can't only be attributed to environment 1 randomly having many more flipped labels than environment 2 because the Deep CORAL method actually performs at a staggering 98.2 percent accuracy for the same configuration of training/test environments that GroupDRO only achieved 9.2 percent accuracy.

However, environment 1 being harder is likely part of the story: the Deep CORAL method doesn't explicitly optimize for improving the worst-case accuracy so is likely more robust to the hardest environment being assigned for testing. The GroupDRO method on the other hand fundamentally relies on optimizing the worst case accuracy across the testing domains and so is more likely to catastrophically fail when the hardest environments are left for testing. This hypothesis also aligns with the method achieving the high observed accuracy when the second environment is held out. Now that the hard environment is no longer being used for testing optimizing the accuracy on it leads to extremely high performance even on the test set.

The C-DANN algorithm also exhibited high variability and low accuracy for the modified datasets. This was likely because its strategy for accounting for a different conditional distribution failed to perform well when using both spurious features which were causal descendants of the target labels and spurious features which were not.

Overall, it is unclear whether the models performed better or worse overall on the new datasets. The relative performance between models varied wildly depending on the specific environments held out and dataset used. These results indicate the susceptibility of methods for domain generalization to alternate causal structures.

## 4 Conclusion

We investigated the effectiveness of the Deep CORAL, C-DANN, and GroupDRO algorithms on causal structures involving spurious features as causal descendants of the prediction target using variations of the MNIST dataset. Although all three methods tested exhibited high accuracy on the original DomainBed datasets, they each achieved wildly varying accuracy on the modified MNIST variants. These results are instructive of the need to closely analyze causal structure when choosing between domain generalization algorithms.

## References


Daniel C. Castro, Ian Walker, and Ben Glocker. Causality matters in medical imaging. *Nature Communications*, 11(1), jul 2020. doi: 10.1038/s41467-020-17478-w. URL https://doi.org/10.1038%2Fs41467-020-17478-w.

Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness, 2022.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization, 2020.

Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. Deep domain generalization via conditional invariant adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation, 2016.

Georg Volk, Stefan Müller, Alexander von Bernuth, Dennis Hospach, and Oliver Bringmann. Towards robust cnn-based object detection through augmentation with synthetic rain variations. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 285–292, 2019. doi: 10.1109/ITSC.2019.8917269.

# A Appendix: Full DomainBed results for all models

## A.1 Model selection: training-domain validation set

### A.1.1 ColoredMNIST

| Algorithm | +90% | +80% | -90% | Avg |
|---|---|---|---|---|
| C-DANN | 71.5 ± 0.0 | 72.2 ± 0.0 | 9.8 ± 0.0 | 51.2 |
| CORAL | 71.5 ± 0.0 | 73.5 ± 0.0 | 9.8 ± 0.0 | 51.6 |
| GroupDRO | 72.7 ± 0.0 | 73.7 ± 0.0 | 10.1 ± 0.0 | 52.2 |

### A.1.2 ColoredRotatedMNIST1

| Algorithm | +90% | +80% | -90% | Avg |
|---|---|---|---|---|
| C-DANN | 89.2 ± 0.0 | 35.5 ± 0.0 | 9.7 ± 0.0 | 44.8 |
| CORAL | 15.9 ± 0.0 | 71.5 ± 0.0 | 50.5 ± 0.0 | 45.9 |
| GroupDRO | 89.2 ± 0.0 | 78.3 ± 0.0 | 10.2 ± 0.0 | 59.3 |

### A.1.3 ColoredRotatedMNIST2

| Algorithm | +90% | +80% | -90% | Avg |
|---|---|---|---|---|
| C-DANN | 98.4 ± 0.0 | 50.5 ± 0.0 | 21.7 ± 0.0 | 56.9 |
| CORAL | 98.2 ± 0.0 | 72.6 ± 0.0 | 19.0 ± 0.0 | 63.3 |
| GroupDRO | 9.2 ± 0.0 | 100.0 ± 0.0 | 18.9 ± 0.0 | 42.7 |

### A.1.4 Averages

| Algorithm | ColoredMNIST | Avg |
|---|---|---|
| C-DANN | 40.9 ± 21.9 | 40.9 |
| CORAL | 56.6 ± 10.6 | 56.6 |
| GroupDRO | 52.2 ± 0.0 | 52.2 |

| Algorithm | ColoredRotatedMNIST1 | Avg |
|---|---|---|
| C-DANN | 36.0 ± 18.6 | 36.0 |
| CORAL | 38.4 ± 16.0 | 38.4 |
| GroupDRO | 59.3 ± 0.0 | 59.3 |

| Algorithm | ColoredRotatedMNIST2 | Avg |
|---|---|---|
| C-DANN | 48.1 ± 18.6 | 48.1 |
| CORAL | 72.0 ± 18.5 | 72.0 |
| GroupDRO | 42.7 ± 0.0 | 42.7 |

### A.2 Model selection: test-domain validation set (oracle)

#### A.2.1 ColoredMNIST

| Algorithm | +90% | +80% | -90% | Avg |
|---|---|---|---|---|
| C-DANN | 74.9 ± 0.0 | 74.4 ± 0.0 | 30.2 ± 0.0 | 59.8 |
| CORAL | 72.4 ± 0.0 | 77.8 ± 0.0 | 48.9 ± 0.0 | 66.4 |
| GroupDRO | 73.7 ± 0.0 | 73.6 ± 0.0 | 49.0 ± 0.0 | 65.4 |

#### A.2.2 ColoredRotatedMNIST1

| Algorithm | +90% | +80% | -90% | Avg |
|---|---|---|---|---|
| C-DANN | 89.9 ± 0.0 | 67.2 ± 0.0 | 14.5 ± 0.0 | 57.2 |
| CORAL | 49.6 ± 0.0 | 74.7 ± 0.0 | 51.6 ± 0.0 | 58.6 |
| GroupDRO | 89.3 ± 0.0 | 79.6 ± 0.0 | 49.5 ± 0.0 | 72.8 |

#### A.2.3 ColoredRotatedMNIST2

| Algorithm | +90% | +80% | -90% | Avg |
|---|---|---|---|---|
| C-DANN | 98.5 ± 0.0 | 90.1 ± 0.0 | 49.0 ± 0.0 | 79.2 |
| CORAL | 98.4 ± 0.0 | 84.1 ± 0.0 | 21.0 ± 0.0 | 67.9 |
| GroupDRO | 50.3 ± 0.0 | 100.0 ± 0.0 | 50.5 ± 0.0 | 66.9 |

#### A.2.4 Averages

| Algorithm | ColoredMNIST | Avg |
|---|---|---|
| C-DANN | 52.4 ± 15.7 | 52.4 |
| CORAL | 67.9 ± 3.2 | 67.9 |
| GroupDRO | 65.4 ± 0.0 | 65.4 |

| Algorithm | ColoredRotatedMNIST1 | Avg |
|---|---|---|
| C-DANN | 46.5 ± 22.6 | 46.5 |
| CORAL | 56.3 ± 4.8 | 56.3 |
| GroupDRO | 72.8 ± 0.0 | 72.8 |

| Algorithm | ColoredRotatedMNIST2 | Avg |
|---|---|---|
| C-DANN | 71.7 ± 16.0 | 71.7 |
| CORAL | 75.5 ± 16.2 | 75.5 |
| GroupDRO | 66.9 ± 0.0 | 66.9 |