Domain Generalization using causal and anti-causal features

Ahmed Elsayed, elsayeaa@gmail.com

DREU 2021, University of Illinois at Urbana Champaign

Abstract:

Modern machine learning methods often assume that the observed data are independent and identically distributed (i.i.d). While it is one of the most well-understood and well-researched principles in machine learning, the i.i.d assumption fails awfully in many real-world applications, specifically due to the identically distributed assumption. In some domains like medicine and behavioral science, one can relax the i.i.d assumption by using causal and anti-causal features. i.e., the existence of reliable causal structures (SCMs) can be utilized to ensure robustness to certain types of distribution shifts. . Our work explores this approach by predicting estimators using causal and anti-causal features across distinct domains -- showing that causal methods can improve the reliability of our predictions. We are investigating ways in which anti-causal features can be wisely used to improve prediction results while maintaining close to the same generalization of causal transfer learning.

Keywords: Causal-inference; transfer-learning; empirical risk minimization; domain generalization

1. Introduction

Empirical Risk Minimization (ERM) traditionally assumes that training and test environments follow the same data distribution. During settings of distribution or dataset *shift* (Quiñonero-Candela et al., 2009), the performance of the standard ERM methods deteriorates rapidly – making a 50-50 chance guess better than adopting a standard ERM learning framework. Consider a handwriting classification system (Zhang et al., 2021), that is trained on a dataset of user data, but it has never seen the new end users. Each new end user induces a never-seen data distribution – some of them having unique handwriting, causing extreme shifts in the global training distribution. Consequently, the model performance deteriorates rapidly.

Many previous algorithms are interested in discovering invariance; that is, estimating an invariant, causal predictors from multiple training environments, ignoring all spurious correlations within a data distribution (e.g., Arjovsky et al., 2020). Such algorithms can alleviate the excessive reliance of machine learning systems on data biases, enabling an out-of-distribution (**OOD**) generalization on new test distributions. However, these methods face limitations when the input-output relationship varies across data distributions. It also disregards all non-causal information, consequently affecting the invariance risk that it might not achieve good performance.

In Structural Causal Models **SCMs**, causal and anti-causal features are identified. In this report, we wisely investigate conditions when to use anti-causal information at test time to adapt to data shifts. To do so, we study problems in which the model does not know the causal and the anti-causal features. For example, in medical observational datasets, we have access to disease causes and symptoms, e.g., LUCAS and LUCAP lung cancer toy datasets from the causality challenge (Guyon et al., 2008). However, our model would not label *Genetics* as causal and *Coughing* as anti-causal. Instead, our model would estimate the weights assigned to each feature, making optimal use of causal and anti-causal features. We apply a set transformer to allow inputs to be *permutation invariant* and of any size. In addition, the Set Transformer allows our model to take advantage of group properties of data distributions and make our neural network more robust. (see Lee et al., 2018). We investigate different train conditions and analyze the behavior of our testing environments in relation to these conditions.

2. Related Work

Several prior works have studied distribution shifts, domain generalization, adaptation, and causal discovery.

Causal Invariance. As aforementioned, there are several learning frameworks that leverage invariant relations among training domains to address data distribution during test time (e.g., Arjovsky et al., 2020). While prior work typically assumes it is helpful and safe to ignore anti-causal features**,** in this report, we investigate ways in which anti-causal features can be helpful for out-of-domain generalization (OOD). We test this method along with our algorithm and identify settings where our algorithm outperforms ERM and other causal-information-only-reliant algorithms.

Adaptation to distribution shifts. The Adaptive Risk Minimization (ARM) Algorithm (Zhang et al., 2021) is an unlabeled adaptation learning framework that is robust to data shifts. This model adapts by using unlabeled data batches to handle dataset or data distribution shifts. This work makes the connection between adaption to dataset shift and meta-learning explicit, enabling more amenable methods to expressive models (e.g., deep neural networks) – this leads to study real world problems with raw observational data.

Distribution shifts have been studied before. For instance, domain adaptation algorithms that assume access to test examples at training time (Wilson & Cook, 2020). Most problems addressed in domain adaptation algorithms were of a single test distribution – having difficulty to being applied to multiple test distributions or domains.

The Set Transformer. The *Set Transformer* is an attention-based neural network module (Lee et al., 2018), which aggregates inputs, ignoring the relative order of samples. For example, a set of causal and anti-causal features would depend on the input data not the relative order of causal Vs. anti-causal data. The *Set Transformer* is an effective tool in our process.

3. Approach

3.1 The model

We are interested in predicting the target variable Y in unseen domains using causal and anti-causal features in source domains. Given a Structural Causal Model (SCM), our model – without knowing the causal relations – uses all available information to figure out a learning pattern that outperforms standard ERM and behaves like causal ERM (which depends on the invariants in source domains).

3.2 The Algorithm

We devised an algorithm that follows an adaptation learning framework while taking advantage of the robustness of a Set Transformer to take advantage of group properties. We have our SCM generative process as $Y^k = \alpha^t X^k + \epsilon^k$, $Z^k = \gamma^k Y^k + \eta^k$, where $\epsilon^k \sim \mathcal{N}(0, \sigma^2), \eta^k \sim \mathcal{N}\left(0, \sigma_\eta^2\right)$, In this model, η^k is the parameter responsible for the tasks. Our input data is $\hat{X}^k$, where $\hat{X}^k = X^k \cup Z^k$. At training time, we have N tasks available. We first calculate the *ordinary least squares* (OLS) estimators $\hat{\beta}_S$ for the input data $\hat{X}$. Then, we train the Set Transformer to predict optimal linear regression coefficients β_S^* (i.e., the best estimators for the weights of causal and anti-causal information). Then, we use the predicted estimators to calculate Y, such that the expected loss function for β^* verifies: $\mathbb{E}\left(\left(Y - \beta^* \hat{X}\right)^2\right)$. Finally, we introduce a regularization term to account for the difference between $\hat{\beta}$ and β^* using a hyperparameter λ – the weight of the regularization term, making our final expected loss $\mathcal{L}$ as:

$$\mathcal{L} = \mathbb{E}\left(\left(Y - (\beta^*)^t \hat{X}\right)^2\right) + \lambda \cdot (\frac{1}{N} \cdot \Sigma \mid \hat{\beta} - \beta^* \mid) \text{ , where } N \text{ is total number of features.}$$

We show our algorithm for training and testing this algorithm as the following:

```
Algorithm: domain generalization using causal and anti-causal features
// Training procedure
Require: batch size $N$, # training steps $T$, L1 weight $\lambda$
Initialize: $\mathcal{L}$, $\mathrm{ST}$ , $\mathcal{L}$ is the loss function, ST is the Set Transformer
for $t = 1, \cdots, T$ do
    Sample corresponding $(\bar{x}_k, \hat{\beta}_k) \leftarrow ST(\hat{X}_S, \hat{\beta}_S)$ for $k = 1, \dots, N$ from training groups
    $\beta^* \leftarrow g(\bar{x}_k, \hat{\beta}_k)$ predict optimal estimators from $\hat{X}$
    calculate $\hat{y}_k$ from $(\bar{x}_k, \hat{\beta}_k)$ for $k = 1, \dots, N$
    $\mathcal{L} \leftarrow \mathcal{L}(y_k, \hat{y}_k) + \lambda \cdot (\frac{1}{\mathrm{N}} \Sigma_{k=1}^{\mathrm{K}} |\beta_k - \bar{\beta}_k|)$ where $y_k$ is true target variable, $\hat{y}_k$ is predicted
    back propagation
// Testing procedure
Require: test batch $\hat{X}_T$: $x_1, \dots, x_K$, $\theta$
$\beta^* \leftarrow ST(\theta, x_1, \dots, x_K; \hat{\beta}_T)$ calculate optimal coefficients
predict $y$ from $\beta^*$ and $\hat{X}_T$
```

4. Experiments and Methods

Our experiments were designed to answer the following:

1. When does our model behave in a better, worse, or equal fashion compared to an i.i.d model or a causal model?
2. Do certain properties of the source domains influence the efficacy of our approach?
3. Is our model robust enough to handle different number of inputs, that is, multiple causal and anti-causal features?

To answer these questions, we proposed the following design:

For each arbitrary dataset, we run four or three models and compare their losses with respect to the exogenous influence eta η. We run a standard ERM model, a causal ERM model (a model which only pools causal information as its input for ERM), a i.i.d model (a model in which we assume train and test follow the same data distribution), and our algorithm. For each experiment, we define D_S as the number of domains for train data, D_T number of domains for test data, n as the sample size in any domain, ϵ, η as aforementioned in *section 3.1*.

4.1 First Experiment: Changing number of domains

For this experiment, we investigate the effect of changing the number of domains on our error, while also taking into the consideration the interval of source domains' exogenous influence η_S

Models used: causal ERM, i.i.d model, Our Algorithm

4.1.1 η of Source domains follows: $\boldsymbol{max(\eta_S) < min(\eta_T)}$

Fixed parameters: $D_T = 300, n = 1000, \eta_T \in [1,4], \eta_S \in [0,2]$

Changing parameter: $D_S \in \{50, 100, 150, \dots, 500\}$

Results:

At all trials, the i.i.d model outperformed both our algorithm and the causal ERM, but there are trials where our algorithm outperformed the causal ERM at some intervals during some trials.

The following table shows the results of this experiment:

Trial #	$\boldsymbol{D_S}$	Interval of $\boldsymbol{\eta_T}$ where $\mathbb{E}_{\boldsymbol{our\ algorithm}} \leq \mathbb{E}_{\boldsymbol{causal\ ERM}}$
1	50	[1, 3]
2	100	[1, 2.5]

3	150	[1, 2.35]
4	200	[1, 1.2]
5	250	[1, 2.1]
6	300	[1, 2.2]
7	350	[1, 2.53]
8	400	[1, 3.1]
9	**450**	**[1, 3.2]**
10	500	[1, 1.75]

Table 1 - the results of performing our second experiment where the number of source domains is variant cetris paribus.

Also, the following is the produced plot for the highlighted trial (# 9):

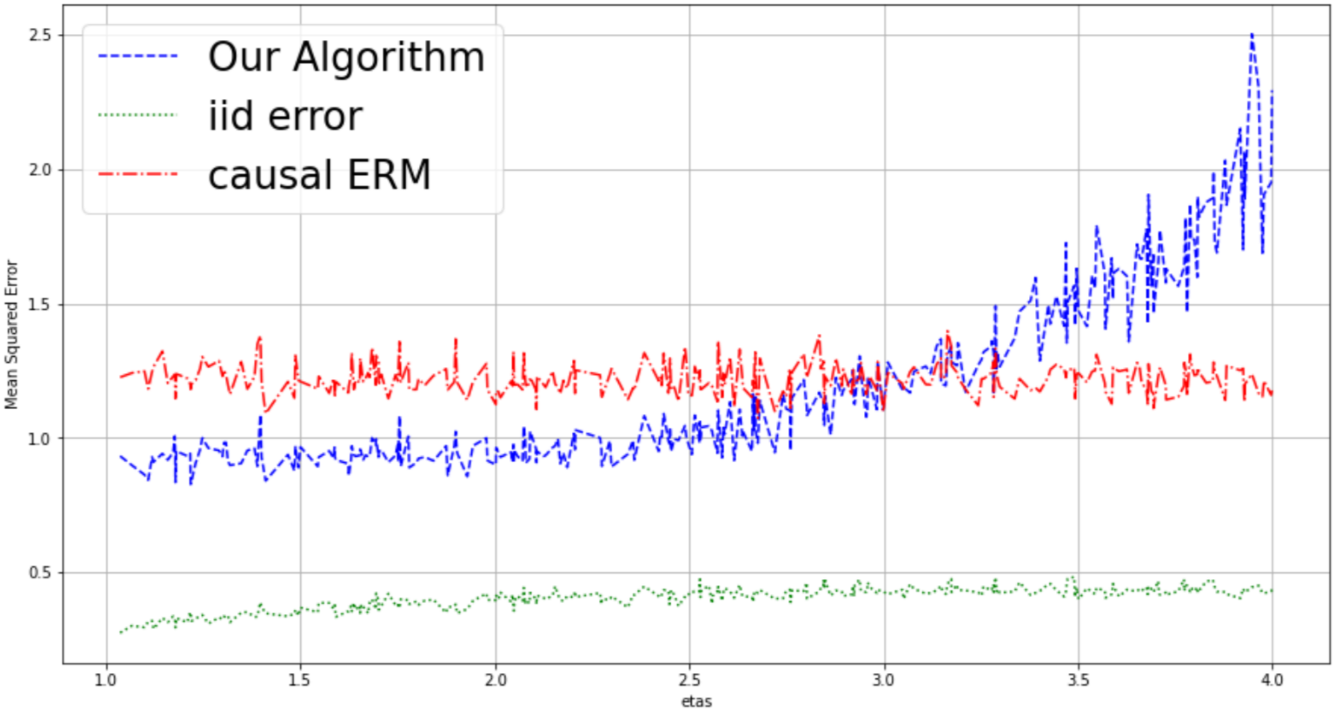

Figure 1- the plotted errors of our models at trial 9

4.1.2 η of Source domains follows: $max(\eta_T) < min(\eta_S)$

Fixed parameters: $D_T = 300, n = 1000, \eta_T \in [1,4], \eta_S \in [4.1, 7.1]$

Varying parameter: $D_S \in \{50, 100, 150, \dots, 500\}$

Results: At all trials, our algorithm behaved worse than the causal ERM and the i.i.d model. The following shows a plot at a trial where $D_S = 450$:

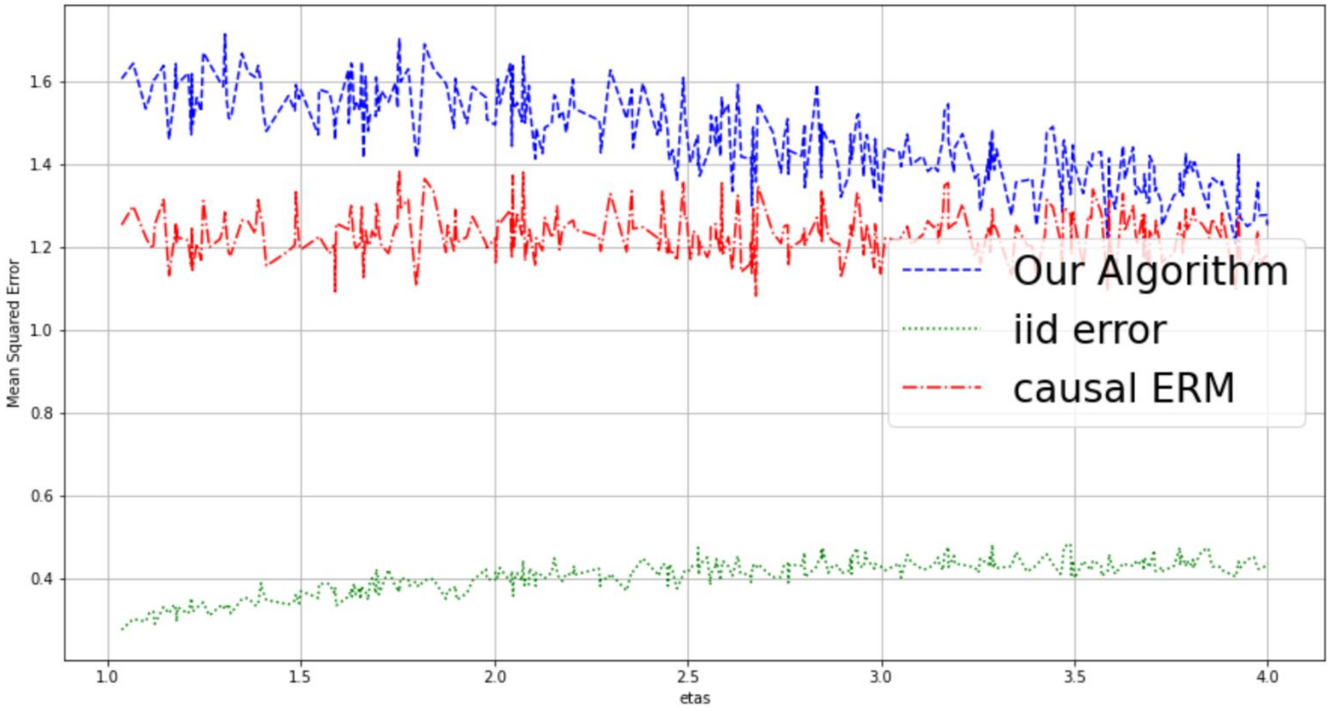

Figure 2 - the plotted errors of our models at $D_S = 450$

4.1.3 Analysis:

We find that when the training domains exogenous influence η_S is way bigger than our test domains exogenous influence η_T, our algorithm consistently behaves worse than the causal ERM; however, when the training domains have $\eta_S \leq n_T$, we can find intervals where our algorithm outperforms the causal ERM.

4.2 Second Experiment: Different error intervals for test and train domains.

For this experiment, we consider the effect of the source domains' exogenous influence η_S on our error.

Models used: Causal ERM, i.i.d model, our Algorithm

Fixed parameters: $D_S = D_T = 250, \eta_T \in [1, 5], n = 2000$

Varying parameter: the interval of η_S

Results:

At all trials, the i.i.d model outperformed both our algorithm and the causal ERM model, but there are trials where our algorithm consistently outperformed the causal ERM model.

The following table shows the results of this experiment:

Trial	$\boldsymbol{\eta_S\ interval}$	Interval of $\boldsymbol{\eta_T}$ where $\mathbb{E}_{\boldsymbol{our\ algorithm}} \leq \mathbb{E}_{\boldsymbol{causal\ ERM}}$	Error scale
1	[1,4]	[1, 1.8]	[0, 2]
2	[1.5, 4.5]	[1, 2.47]	[0, 0.6]
3	[2, 5]	[1, 4.16]	[0, 0.6]
4	[2.5, 5.5]	[1, 3.2]	[0, 0.4]
5	[3, 6]	CONSISTENTLY BETTER	[0, 0.8]
6	[3.5, 6.5]	[1, 3.94]	[0, 0.3]
7	[4, 7]	CONSISTENTLY BETTER	[0 , 0.7]
8	[4.5, 7.5]	CONSISTENTLY BETTER	[0, 0.35]
9	[5, 8]	CONSISTENTLY BETTER	[0, 0.3]
10	[5.5, 8.5]	CONSISTENTLY BETTER on a decreasing trend	[0, 0.25]
11	[6, 9]	[3.9, 5] on a decreasing trend	[0, 0.7]

The following plot was produced at trial 10:

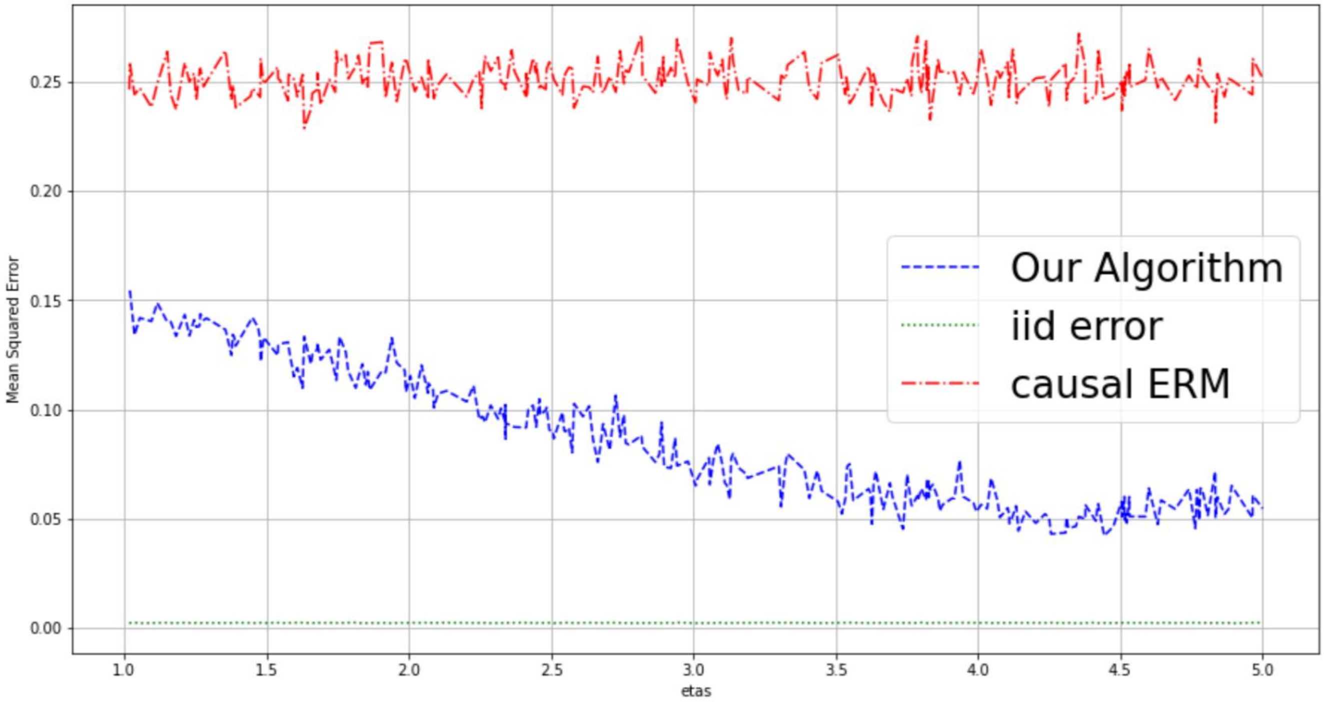

Figure 3 -the plot of our models at trial 10

Analysis:

We find that our model performs better on intervals where the exogenous influence of the training domains is higher than the exogenous influence of the test domains; however, when the training domains exogenous influence is ridiculously higher than the test domains, our algorithm performs worse the causal ERM. At trial 6, where our algorithm fails after $\eta_T \approx 3.94$, we find that the error scale was very small, making the difference between causal ERM and our algorithm negligible. Even though our algorithm didn't perform better, the error difference is too small.

5. Conclusion

Based on the previous experiments, we conclude that our approach outperforms than the Standard ERM algorithm. We also find that we can optimize our algorithm if we find that our training and test datasets meet certain conditions. We find that our algorithm consistently outperforms the standard ERM; however, our algorithm is highly dependent on our hyper-parameters; that is, the number of domains, the effect of the exogenous influence, and the sample distribution influence the performance of our algorithm.

6. Future Applications

We can use these results to create a zero-shot anti-causal transfer model. This can help in creating a meta-learning model in which the algorithm first identifies the properties of the source and transfer domains. Consequently, the model can run an algorithm with the most optimal error outcome, improving upon traditional causal-inference algorithms.

7. References

1. Arjovsky, M., Bottou, L., Gulrajani, I., & Lopez-Paz, D. (2019). Invariant risk minimization. arXiv preprint arXiv:1907.02893.
2. Guyon, I., Aliferis, C., Cooper, G., Elisseeff, A., Pellet, J. P., Spirtes, P., & Statnikov, A. (2008, December). Design and analysis of the causation and prediction challenge. In Causation and Prediction Challenge (pp. 1-33). PMLR.
3. Lee, J., Lee, Y., Kim, J., Kosiorek, A., Choi, S., & Teh, Y.W. (2018) Set Transformer: A Framework for Attention-based Permutation-Invariant Neural Networks.
4. Quiñonero Candela, J., Sugiyama, M., Schwaighofer, A., & Lawrence, N. Dataset Shift in Machine Learning. *The MIT Press*, 2009
5. Wilson, G. and Cook, D. A survey of unsupervised deep domain adaptation. ACM Transactions on Intelligent Systems and Technology (TIST), 2020.
6. Zhang, M., Marklund, H., Dhawan, N., Gupta, A., Levine, S., & Finn, C., (2021_4) Adaptive Risk Minimization: A Meta-Learning Approach for Tackling Group Distribution Shift. arXiv preprint arXiv:2007.02931, 2021