Conditional Distributional Invariance through Implicit Regularization

Anonymous Authors[1]

Abstract

A significant challenge faced by models trained via standard Empirical Risk Minimization (ERM) is that they might learn features of the input X which help it predict label Y in the training set which shouldn't matter, i.e. associations which might not hold in test data. Causality lends itself very well to separate such spurious correlations from genuine, causal, ones. In this paper, we present a simple causal model for data and a method using which we can train a classifier to predict a category Y from an input X, while being invariant to a variable Z which is spuriously associated with Y. Notably, this method is just a slightly modified ERM problem without any explicit regularization. We empirically demonstrate that our method does better than regular ERM on standard metrics on benchmark datasets.

1. Introduction

Neural networks are usually trained via Empirical Risk Minimization (ERM), where we want to find model parameters which minimize a loss function $\mathcal{L}$ on a training dataset. A key challenge here is that this optimization problem gives the model an incentive to learn any and all associations in the training set which reduce the loss. This is not always desirable in practice. Suppose that we wish to classify product reviews as positive or negative. It might be that the positive reviews in the training set are disproportionately of one type of product – say books. This gives the model the incentive to learn the association between books and positive reviews, even though this association is not meaningful in predicting the sentiment of previously unseen reviews.

These are known as *spurious* associations, and they are present because an additional variable Z is correlated with Y and can be "seen" in X. Causality offers a useful framework to think about this problem. In particular, we want to think of *potential outcomes* (Rubin, 2005) which represent counterfactual distributions of X when Z takes on different values. Let $X(z)$ denote the potential outcome of X when $Z = z$. The idea behind invariant learning is that we want our predictor to only look at the sentiment Y and not the product category Z. In other words, we want the model to make the same prediction for all potential outcomes $X(z)$. This can be formalized as *counterfactual invariance* (Veitch et al., 2021): A predictor F is said to be counterfactually invariant to Z if for all z, z', we have $F(X(z)) = F(X(z'))$.

This leads us to the question of how to learn counterfactually invariant predictors in practice. Modelling the causal relationship between X, Y, and Z often allows us to develop methods to do this. In this paper, we present one such case where the causal model is plausible in many important applications. We first explain the causal setting we're interested in and then describe a simple method to learn counterfactually invariant predictors in this setting. The beauty of this method is that it does not depend on solving a harder optimization problem than regular ERM, but just does ERM in a slightly modified setting. We demonstrate successful empirical performance and conclude with a discussion on avenues for future research.

2. Methodology

2.1. Causal Structure of Data

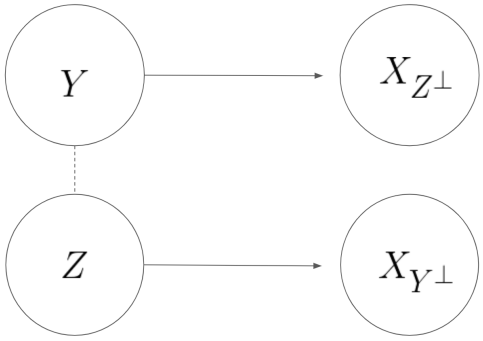

Figure 1. Causal structure of the data for the proposed method. The dotted line between Y and Z represents a spurious association.

The proposed method only works when the data has a spe-

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the SCIS workshop.

cific causal structure. This is shown as a causal graph in Figure 1. This is the setting of an *anti-causal* problem, where we have a label Y which causes the predictor X. Many standard text and image classification problems are anti-causal. For example, in sentiment analysis, the text X is written to explain the sentiment Y, so an intervention on Y will alter the text, but an intervention on the text will not alter Y. The same holds for standard image classification problems, such as classifying whether an image is of a cat or a dog. We also have an additional variable Z which is spuriously correlated with Y, as indicated by the dotted line between them. We assume that the predictor X decomposes in a certain way, i.e. there is a part of X (which we call $X_{Z^\perp}$) which is only affected by Y and a part of X (which we call $X_{Y^\perp}$) which is only affected by Z. An important implication is that the effects of Y and Z on X are "separable" in the sense that interventions on Y and Z will alter disjoint parts of the predictor X. This is commonly known as a *purely spurious* association between Y and Z (Veitch et al., 2021), since in such a case Z is merely a distraction and does not offer any information which might be useful in predicting Y on new data.

2.2. Method

We further restrict ourselves to classification problems, and – for ease of exposition – to binary classification problems. We also assume that Z is discrete. Thus, we are in a setting where we have a binary random variable Y, a discrete random variable Z, and a random variable X. Given that the data follows the structure in Figure 1, we are interested in learning $P(Y = 1 \mid X_{Z^\perp})$. Our main claim, under this setting, is as follows. Suppose X is supported on S_X and Z is supported on S_Z. Let G be a function on $S_X \times S_Z$ and F be a function on S_X. Further suppose that we have the following relationship:

$$G(X, Z) = \left(\frac{F(X)P(Z|Y=1)}{F(X)P(Z|Y=1) + (1-F(X))P(Z|Y=0)}\right) \tag{1}$$

Then, if we train a neural network to learn G from X and Z as inputs, Y as labels, and using cross-entropy loss, then the learned function F will be $F(X) = P(Y = 1 \mid X_{Z^\perp})$, exactly what we are interested in.

The result above is an implication of the fact that if we use binary labels Y and want to predict them using a function $G(X, Z)$, then $P(Y = 1 \mid X, Z) = \arg\min_G \mathcal{L}(G(X, Z), Y)$ when $\mathcal{L}$ is the population cross-entropy loss function. Further note that from Figure 1,

$$P(Y = 1 \mid X, Z) = P(Y = 1 \mid X_{Z^\perp}) \left(\frac{P(Z|Y=1)}{\sum_{y=0}^{1} P(Y = y \mid X_{Z^\perp})P(Z|Y=y)}\right) \tag{2}$$

Algorithm 1 Training Procedure

Input: Training dataset $\mathcal{D}_{\text{train}}$ with observations of the form (x_i, y_i, z_i), a neural network F which takes X as an input and outputs in $(0, 1)$, and the empirical cross-entropy loss function ℓ

1. For all y in the support of Y and z in the support of Z, compute $P(Z = z \mid Y = y)$ as its MLE in $\mathcal{D}_{\text{train}}$
2. Compute $F(x_i)$ from the neural network
3. Modify this output to obtain $G(x_i, z_i)$ as a function of $F(x_i)$ and the probabilities as in (1)
4. Compute loss as $\ell(G(x_i, z_i), y_i)$.

where we simply use Bayes' rule and the fact that $Y \perp X_{Y^\perp} \mid Z$ (which follows from d-separation). We see a clear connection between (1) and (2): if $G(X, Z) = P(Y = 1 \mid X, Z)$, then $F(X) = P(Y = 1 \mid X_{Z^\perp})$.

This result suggests a simple algorithm for training a classifier to learn $P(Y = 1 \mid X_{Z^\perp})$. The naïve way to train a classifier is to learn a function $F(X)$ by ERM with the cross-entropy loss function. But in this case we obtain $F(X) = P(Y = 1 \mid X) = P(Y = 1 \mid X_{Z^\perp}, X_{Y^\perp})$, which depends on $X_{Y^\perp}$, which is not desirable due to the spurious correlation between Y and Z. If we make a simple modification to this procedure and instead perform ERM on the right-hand-side of (1), then (2) implies that the learned function will be $F(X) = P(Y = 1 \mid X_{Z^\perp})$. Note that when Z is discrete, the probabilities $P(Z = z \mid Y = y)$ can be computed offline for all $z \in S_Z$ and $y \in S_Y$, provided that our training dataset is large enough. Algorithm 1 clearly outlines this procedure.

3. Experiments

In our experiments, we look at synthetic and natural datasets where Y and Z are spuriously correlated. We are mainly concerned with two metrics:

1. **Subgroup accuracy**: If the model learns through spurious correlations, then we would expect it to perform poorly on some subgroups of the data. For example, if Y and Z are both binary and Y and Z are highly (positively) correlated, then we would expect a standard ERM model to do poorly on the $(Y = 1, Z = 0)$ and $(Y = 0, Z = 1)$ subgroups. The standard set by (Koh et al., 2021) is to look at *worst subgroup accuracy*. Counterfactually invariant prediction should have a higher worst subgroup accuracy than standard ERM.
2. **Conditional independence**: If a predictor F is coun-

terfactually invariant to Z, then we should have $F(X) \perp Z \mid Y$ (Veitch et al., 2021). To measure this, we will be looking at the correlation between $F(X)$ and Z within every Y subgroup. For a counterfactually invariant predictor F, these correlations should be lower than those for regular ERM.

In the following experiments, we empirically demonstrate that the method outlined in Section 2 increases worst subgroup accuracy and decreases correlations within Y subgroup compared to an ERM baseline.

3.1. Synthetic Text Data

We are interested in comparing the predictions the model makes for the potential outcome $X(z)$ for different values of z. The challenge in doing this on a real dataset is that counterfactuals are not observed, so we first look at look at a text classification example where we synthetically perturb X in different ways based on Z to observe all potential outcomes $X(z)$.

Data: We used data from the Amazon product reviews repository published by (Ni et al., 2019). We used product reviews under the 'Appliances' category and binarized ratings so that 1 star and 2 star ratings were one class and 4 star and 5 star ratings were the other class (3 star ratings were omitted). We balanced the classes so that we had 18000 product reviews in each class. Next, we synthetically generated a variable Z, which was correlated with the rating, as follows:

1. For every observation, we drew Z_i so that $Z_i|Y_i = 0 \sim \text{Bernoulli}(0.9)$ and $Z_i|Y_i = 1 \sim \text{Bernoulli}(0.1)$.
2. If $Z_i = 1$, we perturbed X_i (review text) so that every "and" token in the review became "andxxxx" and every "the" token became "thexxxx". If $Z_i = 0$, we did the same thing with "yyyy" suffixes.

Here, the correlation between Z and Y is purely spurious, and the text X can clearly be decomposed into $X_{Y^\perp}$ (the "xxxx"/"yyyy" suffixes) and $X_{Z^\perp}$ (everything else).

Importantly, we used two test sets for evaluation. The first test set was randomly held out from the data generated as above. The second test set modified the first to change all Z_i to $1-Z_i$ and perturn X_i accordingly ("xxxx" became "yyyy" and vice versa). In this way, we observed the potential outcomes $X(0)$ and $X(1)$ for *all* reviews.

Model architecture and training: We use a pre-trained DistilBERT model (Sanh et al., 2019) with a classification head which output class probabilities. We fine-tune the model with the AdamW optimizer (Loshchilov & Hutter, 2017), learning rate 1×10^{-5} and weight decay 0.01, for 5 epochs. This is done for two models: the baseline (naïve ERM) and the proposed model (ERM with modified output). To limit the signal available to the models and nudge them to learn the shortcut, we only use the first 20 tokens of each review text for training. The batch size is 16.

Results: In addition to subgroup accuracies and correlations, we compute a flip rate for both methods – the proportion of text examples for which the model made different predictions on the two test sets. Thus, we would expect a more counterfactually invariant predictor to have a lower flip rate. In Table 1, we see that our method outperforms ERM on all metrics. On Flip 1, the worst subgroup accuracy increases from 73.5% to 83.2%. On Flip 2, it increases from 75.5% to 83.6%. The correlations between $F(X)$ and Z for both values of Y and in both test datasets, and the flip rate is (more than) halved.

Method	**ERM**		**Model**	
Orientation	Flip 1	Flip 2	Flip 1	Flip 2
$Y=0, Z=0$	**0.735**	0.950	0.855	0.945
$Y=1, Z=0$	0.948	0.817	0.912	**0.836**
$Y=0, Z=1$	0.947	**0.755**	0.933	0.853
$Y=1, Z=1$	0.852	0.959	**0.832**	0.888
$\lvert\text{Corr}(f(X), Z)\rvert(Y=0)\rvert$	0.191	0.250	0.059	0.105
$\lvert\text{Corr}(f(X), Z)\rvert(Y=1)\rvert$	0.240	0.222	0.082	0.076
flip rate	0.163		0.078	

Table 1. Results of the experiment with the synthetic dataset. 'ERM' refers to the training with regular ERM, 'Model' refers to our proposed method. 'Flip 1' and 'Flip 2' correspond to the two test sets described previously. The first block of the table shows subgroup accuracies and the worst subgroup accuracies are in bold. The second block shows within-Y group correlations between $f(X)$ and Z, and the third gives the flip rate for both methods.

3.2. Amazon Product Reviews

We now extend the previous experiment to a related natural setting. We use Amazon product review data from the WILDS repository (Koh et al., 2021). In particular, our task to predict whether a product review is for a book or for another product. The training data has been subsampled so that book reviews tend to be negative, while reviews of other products tend to be positive and we want to be invariant to this.

Data: We subsample the dataset so that we have 10000 product reviews for books $(Y = 1)$ and 10000 reviews for other products $(Y = 0)$. For product reviews of books, we subsample the data so that 7500 of these reviews are negative $(Z = 0)$ and 2500 are positive $(Z = 1)$. On the other hand, for $Y = 0$, we subsample so that $Z = 0$ in 2500 examples and $Z = 1$ in 7500 examples.

Model architecture and training: As in the synthetic example, we fine-tune a DistilBERT model with a classification

head for 5 epochs. For optimization we use the AdamW optimizer with learning rate 1×10^{-5} and weight decay 0.01. The batch size is 16 and the input sequence maximum length is 300.

Results: See table 2 for results. ERM and our model both perform poorly positive reviews of books, but our model brings up the accuracy from 49.1% to 58.7%. Our model also does significantly better on negative reviews of other products while retaining good accuracy on the other two subgroups. The conditional correlations are low and similar for both models – our model has a slightly higher correlation between $F(X)$ and Z when $Y = 0$ than ERM (though both correlations are below 0.1), and a slightly lower correlation when $Y = 1$.

Method	**ERM**	**Model**
$Y = 0, Z = 0$	0.785	0.846
$Y = 1, Z = 0$	0.939	0.857
$Y = 0, Z = 1$	0.890	0.857
$Y = 1, Z = 1$	**0.491**	**0.587**
$\lvert\text{Corr}(f(X), Z) \rvert (Y = 0)\rvert$	0.026	0.040
$\lvert\text{Corr}(f(X), Z) \rvert (Y = 1)\rvert$	0.191	0.120

Table 2. Results of the Amazon experiment. Same format as Table 1.

3.3. Waterbirds

We next move to image classification. For this experiment, we use the Waterbirds dataset (Sagawa et al., 2019; Koh et al., 2021). The task is to classify images of birds as either land birds or water birds. The type of bird Y is spuriously correlated with the background in the image Z, i.e. land birds tend to appear with land backgrounds and water birds with water backgrounds. Our goal is to train a classifier which can accurately classify birds without relying on background information.

Data: The training dataset consists of a total of 4795 images, out of which 3498 are land birds with a land background $(Y = 0, Z = 0)$, 184 are land birds with a water background $(Y = 0, Z = 1)$, 56 are water birds with a land background $(Y = 1, Z = 0)$, and 1057 are water birds with a water background $(Y = 1, Z = 1)$.

Model architecture and training: We fine-tune a ResNet-50 model (He et al., 2016) with a classification head. We disable batch normalization and train both the baseline and the proposed model 50 epochs with the Adam optimizer (Kingma & Ba, 2014) with a learning rate of 3×10^{-5}. This learning rate was chosen after line search from $3 \times 10^{-4}, 1 \times 10^{-5}, 2 \times 10^{-5}, 3 \times 10^{-5}$, and 1×10^{-6}.

Results: Results of the Waterbirds experiment are given in Table 3. We see that our proposed method improves the worst subgroup accuracy from 63.4% to 74.3% on the water bird + land background subgroup. It performs marginally better on the land bird + water background category, and marginally worse on the land bird + land background and water bird + water background categories, as expected. We also see greater independence between $f(X)$ and Z for both values of Y as the correlations decrease significantly compared to the baseline ERM model.

Method	**ERM**	**Model**
$Y = 0, Z = 0$	0.994	0.988
$Y = 1, Z = 0$	**0.634**	**0.743**
$Y = 0, Z = 1$	0.848	0.867
$Y = 1, Z = 1$	0.930	0.919
$\lvert\text{Corr}(f(X), Z) \rvert (Y = 0)\rvert$	0.724	0.536
$\lvert\text{Corr}(f(X), Z) \rvert (Y = 1)\rvert$	0.567	0.365

Table 3. Results of the Waterbirds experiment. Same format as Table 1.

4. Discussion

In this preliminary work, we have presented a frequently encountered causal setting and a method which achieves strong performance on standard invariant prediction tasks in this setting. This method does not add an explicit regularizer to the optimization problem and merely solves a slightly modified ERM problem. To conclude, we briefly comment on possible directions for future research:

1. **More complex datasets**: So far, we have only looked at examples with low-dimensional Z. The value of methods like ours comes in when Z is high-dimensional, such as in the Civil Comments and full Amazon datasets from the WILDS repository (Koh et al., 2021). We would like to see this method succeed on such examples as well.

2. **Calibration**: It is well known that neural networks which output probabilities can be poorly calibrated (Guo et al., 2017; Desai & Durrett, 2020). Our method only works if the output we get during training time is a reasonable estimate of $P(Y = 1 \mid X, Z)$. Thus, it will be worthwile to explore methods and architectures for better calibration since this has so far not been our main concern.

3. **Domain shift**: We would like models to perform well across different domains, i.e. not learn domain idiosyncrasies. If Z captures this domain specific association, and we assume that the causal relationship between X, Y, and Z does not change across domains, we can extend our method to this setting.

References

Desai, S. and Durrett, G. Calibration of pre-trained transformers. *arXiv preprint arXiv:2003.07892*, 2020.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In *International Conference on Machine Learning*, pp. 1321–1330. PMLR, 2017.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Koh, P. W., Sagawa, S., Marklund, H., Xie, S. M., Zhang, M., Balsubramani, A., Hu, W., Yasunaga, M., Phillips, R. L., Gao, I., et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pp. 5637–5664. PMLR, 2021.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Ni, J., Li, J., and McAuley, J. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 188–197, 2019.

Rubin, D. B. Causal inference using potential outcomes: Design, modeling, decisions. *Journal of the American Statistical Association*, 100(469):322–331, 2005.

Sagawa, S., Koh, P. W., Hashimoto, T. B., and Liang, P. Distributionally robust neural networks. In *International Conference on Learning Representations*, 2019.

Sanh, V., Debut, L., Chaumond, J., and Wolf, T. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

Veitch, V., D'Amour, A., Yadlowsky, S., and Eisenstein, J. Counterfactual invariance to spurious correlations in text classification. *Advances in Neural Information Processing Systems*, 34, 2021.