

A PROOFS

Proof of Proposition 1. First let us present the DA-ERM solution:

$$f_{\text{DA-ERM}}(w) = \mathbb{E} [(w_1x + w_2y - y)^2 + (w_1x + w_2(ay + n) - y)^2] \quad (1)$$

$$\begin{aligned} &= \mathbb{E} [w_1^2x^2 + (w_2 - 1)^2y^2 + 2w_1(w_2 - 1)xy] \\ &\quad + \mathbb{E} [w_1^2x^2 + (w_2a - 1)^2y^2 + w_2^2n^2] \\ &\quad + \mathbb{E} [2w_1(w_2a - 1)xy + 2w_1w_2xn + 2w_2(w_2a - 1)yn] \end{aligned} \quad (2)$$

$$\begin{aligned} &= w_1^2\sigma_x^2 + (w_2 - 1)^2(\sigma_x^2 + \sigma_\varepsilon^2) + 2w_1(w_2 - 1)\sigma_x^2 \\ &\quad + w_1^2\sigma_x^2 + (w_2a - 1)^2(\sigma_x^2 + \sigma_\varepsilon^2) + w_2^2\sigma_n^2 \\ &\quad + 2w_1(w_2a - 1)\sigma_x^2 \end{aligned} \quad (3)$$

$$\begin{aligned} &= (w_1 + w_2 - 1)^2\sigma_x^2 + (w_2 - 1)^2\sigma_\varepsilon^2 \\ &\quad + (w_1 + w_2a - 1)^2\sigma_x^2 + (w_2a - 1)^2\sigma_\varepsilon^2 + w_2^2\sigma_n^2. \end{aligned} \quad (4)$$

Hence, the solution of $w^* = \arg \min_w f_{\text{DA-ERM}}(w)$ is given by

$$\begin{aligned} 2w_1^* + (1 + a)w_2^* - 2 &= 0, \\ (w_1^* + w_2^* - 1)\sigma_x^2 + (w_2^* - 1)\sigma_\varepsilon^2 + a(w_1^* + w_2^*a - 1)\sigma_x^2 + a(w_2^*a - 1)\sigma_\varepsilon^2 + w_2^*\sigma_n^2 &= 0. \end{aligned} \quad (5)$$

Subsequently,

$$w_{\text{DA-ERM}}^* = \begin{pmatrix} \frac{a^2(\sigma_x^2 + \sigma_\varepsilon^2) - 2a(\sigma_x^2 + \sigma_\varepsilon^2) + \sigma_x^2 + \sigma_\varepsilon^2 + 2\sigma_n^2}{a^2(\sigma_x^2 + 2\sigma_\varepsilon^2) - 2a\sigma_x^2 + \sigma_x + 2(\sigma_\varepsilon^2 + \sigma_n^2)} \\ \frac{2(a+1)\sigma_\varepsilon^2}{a^2(\sigma_x^2 + 2\sigma_\varepsilon^2) - 2a\sigma_x^2 + \sigma_x + 2(\sigma_\varepsilon^2 + \sigma_n^2)} \end{pmatrix}. \quad (6)$$

$$\begin{aligned} w_{\text{DAIR}}^* &= \arg \min_w f_{\text{DAIR}}(w) \\ &= \arg \min_w \mathbb{E} [(w_1x + w_2y - y)^2 + (w_1x + w_2(ay + n) - y)^2] \\ &\quad + [\lambda(|w_1x + w_2y - y| - |w_1x + w_2(ay + n) - y|)^2]. \end{aligned}$$

When $\lambda \rightarrow \infty$, we have $w_{\text{DAIR}_2}^* = 0$ and hence:

$$w_{\text{DAIR}}^* = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

We then evaluate the testing loss assuming the spurious feature is absent, i.e., $\mathbf{x}_{\text{test}} = (x, s = 0)$.

$$\begin{aligned} \ell_{\text{DAIR}}(\mathbf{x}_{\text{test}}; w_{\text{DAIR}}^*) &= \mathbb{E} [(w_{\text{DAIR}}^* \top \mathbf{x}_{\text{test}} - y)^2] \\ &= \mathbb{E} [(x - (x + \varepsilon))^2] \\ &= \sigma_\varepsilon^2. \end{aligned}$$

$$\begin{aligned} \ell_{\text{DA-ERM}}(\mathbf{x}_{\text{test}}; w_{\text{DA-ERM}}^*) &= \mathbb{E} [(w_{\text{DA-ERM}}^* \top \mathbf{x}_{\text{test}} - y)^2] \\ &= \mathbb{E} \left[\left(\frac{a^2(\sigma_x^2 + \sigma_\varepsilon^2) - 2a(\sigma_x^2 + \sigma_\varepsilon^2) + \sigma_x^2 + \sigma_\varepsilon^2 + 2\sigma_n^2}{a^2(\sigma_x^2 + 2\sigma_\varepsilon^2) - 2a\sigma_x^2 + \sigma_x + 2(\sigma_\varepsilon^2 + \sigma_n^2)} x - (x + \varepsilon) \right)^2 \right] \\ &= \sigma_\varepsilon^2 + \frac{(a+1)^4\sigma_\varepsilon^4\sigma_x^2}{(a^2(\sigma_x^2 + 2\sigma_\varepsilon^2) - 2a\sigma_x^2 + \sigma_x + 2(\sigma_\varepsilon^2 + \sigma_n^2))^2} \\ &\geq \ell_{\text{DAIR}}. \end{aligned}$$

Proof of Lemma 1 We proceed as follows:

$$r_{\ell_1}(z, \tilde{z}; \theta) - r_{\text{sq}}(z, \tilde{z}; \theta) = 2\sqrt{\min\{\ell(z; \theta), \ell(\tilde{z}; \theta)\}} \left| \sqrt{\ell(\tilde{z}; \theta)} - \sqrt{\ell(z; \theta)} \right|,$$

We break it into two cases: if $\ell(\tilde{z}; \theta) > \ell(z; \theta)$:

$$\begin{aligned} r_{\ell_1}(z, \tilde{z}; \theta) - r_{\text{sq}}(z, \tilde{z}; \theta) &= \ell(\tilde{z}; \theta) - \ell(z; \theta) - (\sqrt{\ell(\tilde{z}; \theta)} - \sqrt{\ell(z; \theta)})^2 \\ &= \ell(\tilde{z}; \theta) - \ell(z; \theta) - \ell(\tilde{z}; \theta) - \ell(z; \theta) + 2\sqrt{\ell(\tilde{z}; \theta)}\sqrt{\ell(z; \theta)} \\ &= -2\ell(z; \theta) + 2\sqrt{\ell(\tilde{z}; \theta)}\sqrt{\ell(z; \theta)} \\ &= 2\sqrt{\ell(z; \theta)}(\sqrt{\ell(\tilde{z}; \theta)} - \sqrt{\ell(z; \theta)}) \end{aligned}$$

If $\ell(\tilde{z}; \theta) \leq \ell(z; \theta)$:

$$\begin{aligned} r_{\ell_1}(z, \tilde{z}; \theta) - r_{\text{sq}}(z, \tilde{z}; \theta) &= \ell(z; \theta) - \ell(\tilde{z}; \theta) - (\sqrt{\ell(\tilde{z}; \theta)} - \sqrt{\ell(z; \theta)})^2 \\ &= \ell(z; \theta) - \ell(\tilde{z}; \theta) - \ell(\tilde{z}; \theta) - \ell(z; \theta) + 2\sqrt{\ell(\tilde{z}; \theta)}\sqrt{\ell(z; \theta)} \\ &= -2\ell(\tilde{z}; \theta) + 2\sqrt{\ell(\tilde{z}; \theta)}\sqrt{\ell(z; \theta)} \\ &= 2\sqrt{\ell(\tilde{z}; \theta)}(\sqrt{\ell(z; \theta)} - \sqrt{\ell(\tilde{z}; \theta)}) \end{aligned}$$

If we combine the two cases, we have:

$$r_{\ell_1}(z, \tilde{z}; \theta) - r_{\text{sq}}(z, \tilde{z}; \theta) = 2\sqrt{\min\{\ell(z; \theta), \ell(\tilde{z}; \theta)\}} \left| \sqrt{\ell(\tilde{z}; \theta)} - \sqrt{\ell(z; \theta)} \right|.$$

B MODEL ARCHITECTURE AND TRAINING PARAMETERS FOR MNIST EXPERIMENTS

Layer Type	Shape
Convolution + ReLU	$4 \times 4 \times 6$
Max Pooling	2×2
Convolution + ReLU	$4 \times 4 \times 16$
Max Pooling	2×2
Convolution + ReLU	$4 \times 4 \times 96$
Fully Connected + ReLU	64
Fully Connected	C

Table 5: Model Architecture, $C = 1$ for Colored MNIST and $C = 10$ for Rotated MNIST

Parameter		
Learning Rate	0.005	0.0005
Epochs	20	20
Batch-size	64	

Table 6: Training parameter

C SETUP AND ADDITIONAL RESULTS FOR VQA

We modify the official code released by [Agarwal et al. \(2020\)](#) to suit our formulation. All the methods are trained for 40 epochs with a learning rate of 0.001 and a batch size of 96.

Lambda	VQA v2 val (%)	Predictions flipped (%)	pos \rightarrow neg (%)	neg \rightarrow pos (%)	neg \rightarrow neg (%)
0.37	58.52	11.92	4.48	5.28	2.17
0.72	58.21	11.28	4.13	5.08	2.07
1.39	57.54	10.37	3.80	4.65	1.91
2.68	56.24	9.68	3.56	4.39	1.73
5.18	54.19	8.75	3.40	3.66	1.69
10	51.32	7.94	3.01	3.40	1.53

Table 7: Accuracy-Consistency Tradeoff on VQA v2 val and IV-VQA test set controlled by λ

Scheme	z	Color $y = 0$
C1	with $p = 0.8, z = y$	Red
	with $p = 0.2, z = 1 - y$	Green
C2	with $p = 0.9, z = y$	Red
	with $p = 0.1, z = 1 - y$	Green
C3	with $p = 0.1, z = y$	Red
	with $p = 0.9, z = 1 - y$	Green
C4	$z = 2$	Random

Table 8: Color schemes in Colored MNIST. Random color means that the value of each channel of the image is uniformly random chosen from 0 to 255.

Table 7 indicates a tradeoff between the accuracy on the VQA v2 ‘val’ set and the consistency metrics. As the λ value increases, the consistency between the predictions increases, while the accuracy on original examples decreases. For instance, A λ value of 10 strongly boosts consistency thus lowering the ‘Predictions flipped’ percentage to only 7.9% but sacrifices the predictive power causing the accuracy to drop to 51.3 %

D MNIST SETUP

We apply the proposed loss function (DAIR) on the following two datasets: Colored MNIST and Rotated MNIST. We compare the performance of DAIR with plain data augmentation, and invariant risk minimization (IRM) as a strong baseline. One crucial difference between our work and IRM is the motivation. IRM is designed to take two examples from two different environments and learn representations that are invariant to the environment, e.g., in cases where we are aggregating multiple datasets. On the other hand, we are interested in promoting invariance when we have a single dataset. As such, we artificially generate the second environment in IRM using data augmentation. For a given example z , we design an augmenter $A(\cdot)$ and use it to generate additional samples that adhere to the invariance we have in mind. Hence, IRM will be applied in the same way that examples from different environments are augmenting pairs.

Our Colored MNIST is an extension of the original Colored MNIST (Arjovsky et al., 2019). The label is a noisy function of both digit and color. The digit has a correlation of 0.75 with the label and a certain correlation with the label depending on the color scheme. Besides the two colors in the original dataset, we introduce fully random colored scheme to the dataset, which is the best augmenter one can think of. The three color schemes are detailed in Table 8.

Our Rotated MNIST is a variant of the original Rotated MNIST (Ghifary et al., 2015). The original dataset contains images of digits rotated d degrees, where $d \in \mathcal{D} \triangleq \{0, 15, 30, 45, 60, 75\}$. Similarly, we introduce the random degree scheme here to serve as the best possible augmenter. To further exploit the potential of the proposed algorithm, we make this dataset more difficult by introducing more challenging degree scheme; The rotation schemes are summarized in Table 9.

Note all the augmented images are generated on the fly. Examples of images from some transformation schemes are shown in Figures 8 to 13.

Scheme	Rotation
R1	0°
R2	90°
R3	0°, 180°
R4	90°, 270°
R5	[0°, 360°]
R6	[22.5°, 67.5°], [202.5°, 247.5°]

Table 9: Rotation schemes in Rotated MNIST. $[a, b]$ means that degrees are uniformly random chosen between a and b .

Setup Name	Train	Aug	Test	λ
Adv. Aug.	C1	C2	C3	1000
Rnd. Aug.	C1	C4	C3	100

Table 10: Training procedure of Colored MNIST.



Figure 8: C2 Figure 9: C3 Figure 10: C4 Figure 11: R4 Figure 12: R5 Figure 13: R6

Setup: We train a model consisted of three convolutional layers and two fully connected layers with 20,000 examples. For each dataset we are defining several different schemes on how the dataset could be modified: Table 8 (Colored MNIST) and Table 9 (Rotated MNIST). Then, we define several *setups*. Each setup is consisted of one original dataset, one augmentation dataset, and one test dataset, each of which is selected among the defined schemes. These setups are provided in Table 10 (Colored MNIST) and Table 11 (Rotated MNIST). For each setup, we train the model with the following four algorithms and compare their performances: ERM, DA-ERM, DAIR and Invariant Risk Minimization (IRM). Each experiment is repeated for 10 times; the mean and the standard derivation are reported. The value of λ are chosen base on the validation results. Detailed architectures and training parameters can be found in Appendix B.

D.1 COLORED MNIST

We conduct two sets of experiments for this dataset: Adversarial Augmentation Setup (Table 10) follows the exact same color schemes from the original Colored MNIST [Arjovsky et al. (2019)]. For Random Augmentation Setup, we train the model with the strongest possible augmenter: uniformly random color. The entire procedure is summarized in Table 10.

Setup	Train	Aug	Test	λ
Strong Aug.	R1	R5	R2	1
Weak Aug.	R4	R6	R3	10

Table 11: Training procedure of Rotated MNIST

D.2 ROTATED MNIST

We start with the strongest augmenter case. One may notice that there is a chance that the augmented images bear the same rotation degrees as the testing set. To make the task more difficult, we will use R6 as the augmented test to test how the trained model generalize to entirely unseen domain. The training procedure is summarized in Table [II](#).