LEARNING ENVIRONMENT DISTRIBUTIONS FOR ROBUST IMAGE CLASSIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Machine learning models often exploit spurious correlations, such as using background to recognize a bird in Waterbirds or the color of a digit to classify it in Colored MNIST. One approach to reducing this reliance is to train across multiple environments: data distributions that differ, for example, in their color-label correlation or in how training examples are grouped. However, the choice of these distributions and their contribution to training can strongly affect the resulting predictor. In this paper, we introduce Universal Adaptive Environment Discovery (UAED), a framework that jointly learns a predictor and a distribution over a specified family of environment constructions, regularized by a KL penalty toward a reference. For transformation-based training, this distribution determines how different transformations contribute to the objective; for group-based training, it weights alternative groupings of examples, which we construct from a reference predictor's prediction difficulty. Our analysis characterizes the resulting policy preferences: learned cooperatively with the predictor, the distribution tilts toward environments the predictor already handles well and collapses onto a shortcut environment below a computable KL threshold, whereas reversing the sign of its update yields a robust counterpart whose objective upper-bounds the expected training score over a KL ball of environment distributions. Experiments distinguish the effects of broader coverage from those of learning its weighting. Training over the family improves worst-case accuracy over restricted environments on Colored MNIST and worst-group accuracy over ERM on Waterbirds with ResNet-50 without group annotations, while with ViT-B/16 the grouping objective destabilizes fine-tuning at the shared learning rate. Learned policies move as predicted, away from the annotated grouping when cooperative and toward it when robust, and weak KL penalties reinforce color shortcuts; on the official GroupDRO code, where the grouping family reaches 81.8% worst-group accuracy without group annotations, cooperative learning lowers accuracy in every seed and the robust policy removes this loss, matching but not exceeding a fixed weighting. These findings identify both the potential and the limitations of jointly optimizing predictors and environment distributions.

1 INTRODUCTION

A digit classifier trained on Colored MNIST (Arjovsky et al., 2019) sees digits whose color agrees with a (noisy) label about 85% of the time, while the digit's shape agrees with it only 75% of the time. The color is the easier cue, so the classifier learns it, and its accuracy collapses once the color–label agreement is reversed at test time. A bird classifier trained on Waterbirds (Sagawa et al., 2020) makes the analogous mistake with the image background. Such *spurious correlations* are predictive during training but unreliable at deployment.

A standard remedy trains across several *environments*. In Colored MNIST an environment is specified by a single number e, the probability that the color *disagrees* with the label: the benchmark trains on $e = 0.1$ and $e = 0.2$ and tests on $e = 0.9$. Other environments rotate images by a chosen angle, or group the training examples in a chosen way, for instance by how hard they are for a reference model. Invariant Risk Minimization (IRM) asks one classifier to be optimal in every environment (Arjovsky et al., 2019), risk extrapolation (REx) penalizes differences between environment risks (Krueger et al., 2021), and GroupDRO minimizes the worst group risk (Sagawa et al.,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

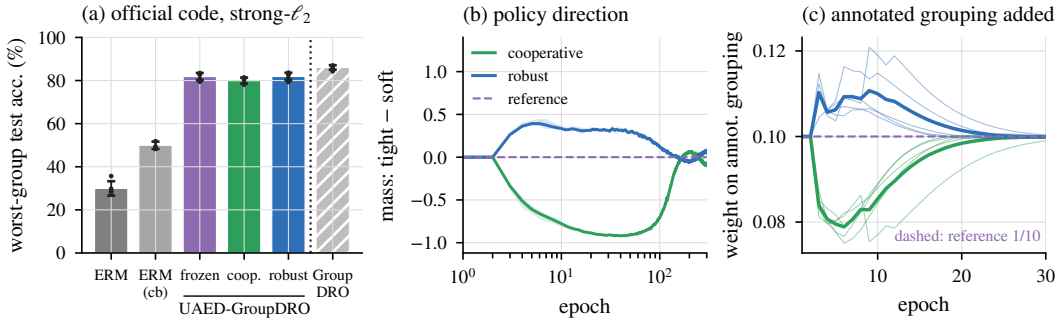

Figure 1: Learned policies on Waterbirds (ResNet-50, 5 seeds). **(a)** Official GroupDRO code: worst-group test accuracy (validation-selected; dots: seeds). Only GroupDRO (hatched) uses group labels. **(b)** Policy mass on the tightest minus the softest groupings (0: uniform reference). The cooperative policy moves toward easy groupings, the robust one toward hard ones. **(c)** Main protocol, annotated grouping added to the family: its weight is below $1/10$ (cooperative) or above it (robust) in every seed.

2020). All of them take the environments as given, yet *which* environments are used, and *how much* each contributes, strongly affects the resulting predictor (Gulrajani & Lopez-Paz, 2021; Rosenfeld et al., 2021).

This paper studies what happens when that choice is learned. We introduce Universal Adaptive Environment Discovery (UAED), which jointly trains a predictor and an *environment policy*: a distribution over a supplied family of environment constructions, regularized by a Kullback–Leibler (KL) penalty toward a reference distribution. The family is supplied by the practitioner (for example, all recolorings indexed by $e \in [0, 1]$, or several candidate groupings of the training set); UAED learns how its members contribute to the training objective. We instantiate it for transformations (recoloring) and for groupings built from a reference predictor's margins.

Our central observation is that, when the policy minimizes the same objective as the predictor, it acts as a *soft minimum* over environments: at the optimum, it shifts mass toward environments in which the current predictor already performs well (Section 3). This has two consequences, and our experiments are consistent with both. First, once the predictor behaves identically across the family, the optimal policy for that predictor is the reference itself, so learning the weights adds nothing beyond the coverage that the reference already provides. Second, when the KL penalty is weak, the objective prefers a shortcut predictor paired with the environment in which the shortcut is perfectly predictive. Therefore, the KL coefficient determines which of the two regimes occurs (Fig. 2). Reversing the sign of the policy update gives a robust counterpart, namely a KL-regularized adversary over the family, whose objective bounds the expected score under every environment distribution in a KL ball around the reference. On Waterbirds, the two policies move in opposite directions, as the analysis predicts (Fig. 1).

We examine these consequences with controlled comparisons that separate three effects: training on a restricted set of environments, training on a broader family with a *fixed* weighting, and *learning* the weighting over that same family. On Colored MNIST, UAED-IRM improves worst-case accuracy over restricted IRM by +7.5 pp at the final checkpoint (95% CI [+4.4, +10.5], positive in 5/5 seeds). Training on the same family with fixed weights gives the same gain (+7.6 pp with the policy frozen at the reference, +7.2 pp with uniform weights), and the learned policy ends at the reference. With a weak KL penalty ($\beta \leq 0.01$) the policy collapses onto the shortcut environment in every seed, and in-distribution validation ranks these models highest. On Waterbirds, training over nine groupings built from a reference model's margins improves worst-group accuracy over ERM with ResNet-50 by +6.4 pp (positive in 5/5 seeds) without group annotations; frozen and learned weightings again perform alike, and on the official GroupDRO code cooperative learning is worse than the frozen policy in every seed. When the annotated grouping is added to the family, the cooperative policy moves weight away from it and the robust policy toward it, in every seed, as predicted; on the official code, the robust policy removes the cooperative policy's loss in every seed and matches, but does not exceed, the frozen policy. With ViT-B/16, the grouping objective destabilizes fine-tuning at the shared learning rate, and diagnostic runs show that a lower learning rate removes this instability.

Our main contributions are as follows.

Methodology: We introduce UAED, a framework that learns a KL-regularized distribution over a supplied family of environment constructions jointly with the predictor, instantiated for transformation and partition families, with a cooperative and a robust policy.

Theory: We characterize the optimal policy for a fixed predictor (Section 3): it is a Gibbs tilt of the reference; the cooperative objective is a soft minimum over environments, its optimal policy equals the reference once the predictor is invariant, and it prefers a shortcut below a computable KL threshold; reversing the sign yields the entropic risk and its KL-ball bound.

Empirical validation: Controlled experiments on Colored MNIST and Waterbirds, including the official GroupDRO code, separate coverage from learned weighting, confirm the predicted direction of both policies, and diagnose when the grouping objective fails. Codes are available in the supplementary material.

2 Learning Environment Distributions

Environments, families and policies. Let $S = \{(x_i, y_i)\}_{i=1}^n$ be the training data, f_θ a predictor and ℓ a loss. An *environment construction* $\gamma \in \Gamma$ turns S into an environment. We consider two kinds. A *transformation* changes the observations: in Colored MNIST, $\gamma = e$ colors each digit so that the color disagrees with the label with probability e. A *partition* keeps the observations but groups them: $\gamma = k$ selects one of K candidate groupings. The family Γ is supplied; it encodes what the practitioner is willing to vary. An environment policy $q = \Pi_\phi$ is a distribution over Γ with parameters ϕ, and $p = \Pi_0$ is a fixed reference distribution with $q \ll p$.

Objective. Each method assigns every construction a *score* $a_\gamma(\theta) \geq 0$, the per-environment training criterion (for example, a risk plus an invariance penalty). UAED solves

$$\min_{\theta,\phi} \; \mathcal{J}(\theta, \phi) = \mathbb{E}_{\gamma\sim\Pi_\phi}\big[a_\gamma(\theta)\big] + \beta\, D_{\mathrm{KL}}(\Pi_\phi \,\|\, \Pi_0), \qquad \beta > 0. \tag{1}$$

Fixing $\Pi_\phi = \Pi_0$, the $\beta \to \infty$ limit of Eq. (1) (Corollary 1), gives UAED with a *frozen* policy, which we report as an ablation of the learned weighting; replacing Γ by a few hand-picked environments gives *restricted* training. Both parameters minimize the same objective, so the policy is *cooperative*: it is not an adversary searching for hard environments. The frozen policy starts and stays at the reference, whereas the learned policy may be initialized elsewhere, so the two share the family but not necessarily the initial policy. The original formulation of UAED also covers REx- and CORAL-type penalties that are nonlinear in the policy (Krueger et al., 2021; Sun & Saenko, 2016); the analysis and experiments below concern scores that enter linearly, as in Eq. (1).

Robust policy. The same family and KL penalty also define a *robust* counterpart in which the policy maximizes the score,

$$\min_{\theta} \max_{\phi} \; \mathbb{E}_{\gamma\sim\Pi_\phi}\big[a_\gamma(\theta)\big] - \beta\, D_{\mathrm{KL}}(\Pi_\phi \,\|\, \Pi_0). \tag{2}$$

Only the direction of the policy update changes: for a given policy the predictor receives the same gradient as in Eq. (1), and the policy ascends rather than descends the score. We refer to the two versions as *cooperative* and *robust* UAED; the frozen (fixed) policy is the $\beta \to \infty$ limit of both.

Transformation family: exact IRM for hard colors. For Colored MNIST, each image i has exactly two colored versions, x_i^0 (color agrees with the label) and x_i^1 (color disagrees). In environment e the version is drawn independently per image, so the environment risk on a minibatch B is available in closed form, $R_e = \frac{1}{|B|}\sum_{i\in B}[(1-e)\,\ell_i^0 + e\,\ell_i^1]$, and it is differentiable in e. For the IRMv1 penalty (Arjovsky et al., 2019) let $g_i^c = \partial_w \ell(w f_\theta(x_i^c), y_i)|_{w=1}$ be the scale gradient of example i under version c. We use the *expected squared* minibatch scale gradient,

$$G_e = \Big(\tfrac{1}{|B|}\textstyle\sum_{i\in B}\big[(1-e)g_i^0 + e\,g_i^1\big]\Big)^2 + \tfrac{e(1-e)}{|B|^2}\textstyle\sum_{i\in B}\big(g_i^1 - g_i^0\big)^2, \tag{3}$$

whose second term is the variance of the random color draws (Section B); dropping it would change the objective. The score is $a_e(\theta) = R_e + \lambda G_e$ for IRM and $a_e = R_e$ for ERM. The policy samples $e = \sigma(\mu + \exp(s)\, z)$ with $\mu \sim \mathcal{N}(m_\mu, \exp(2l_\mu))$, $s \sim \mathcal{N}(m_s, \exp(2l_s))$, $z \sim \mathcal{N}(0,1)$, and the KL term is computed between the Gaussian latents (μ, s). The reference has $m_\mu = m_s = l_\mu = l_s = 0$; it covers all of $[0,1]$ but is not uniform.

Partition family: groupings from a reference model. For Waterbirds we do not transform images. A reference predictor, trained by ERM on the training set only, provides each example's correct-class margin m_i. For each class c, a candidate partition splits the class into a lower- and a higher-margin stratum with soft memberships $s_{i,\mathrm{low}} = \sigma\big((t_c(\rho) - m_i)/(T\,\mathrm{sd}_c)\big)$ and $s_{i,\mathrm{high}} = 1 - s_{i,\mathrm{low}}$, where $t_c(\rho)$ and sd_c are the ρ-quantile and the standard deviation of the class-c margins. Three quantiles $\rho \in \{0.05, 0.1, 0.2\}$ and three temperatures $T \in \{0.05, 0.25, 1\}$ give $K = 9$ candidates with $G = 4$ soft groups each. On a minibatch, group risks $R_{kg} = \sum_i s_{ikg}\ell_i / \sum_i s_{ikg}$ are aggregated by the entropic score $A_k = \tau \log\big(\frac{1}{G}\sum_g e^{R_{kg}/\tau}\big)$, which lies within $\tau \log G$ of the worst group risk (Section A). The aggregation matters: weighting the group risks by group size recovers the ordinary empirical risk for every grouping, so such a score could not distinguish the candidates and the policy would have nothing to choose. The entropic score is nonlinear in the group risks and therefore depends on how the examples are grouped, and this dependence is what Section 3 analyzes. The score is $a_k(\theta) = \mathbb{E}_B[A_k(\theta; B)]$ and the policy is a softmax over the K candidates with a uniform reference. Because A_k is a soft worst-group risk, we call this instantiation UAED-GroupDRO; it differs from GroupDRO (Sagawa et al., 2020) in that its groups come from the reference model and its weights over groupings are learned. The groups describe difficulty under the reference model, not verified background categories; no background annotation is used in training.

3 What Does a Learned Environment Distribution Do?

Because the score term of Eq. (1) is linear in the policy, the policy step can be solved exactly for a fixed predictor. Throughout, Γ is a measurable space, p a probability measure on it, and $a_\theta : \Gamma \to [0, \infty)$ measurable; proofs are in Section A.

Proposition 1 (Gibbs policy). *For $\beta > 0$, the problem $\min_{q \ll p}\{\mathbb{E}_q[a_\theta] + \beta D_{\mathrm{KL}}(q\|p)\}$ has the unique minimizer $\frac{dq_\theta^\star}{dp}(\gamma) = e^{-a_\theta(\gamma)/\beta}/Z_\theta$ with $Z_\theta = \mathbb{E}_p[e^{-a_\theta/\beta}]$, and for every q, $\mathbb{E}_q[a_\theta] + \beta D_{\mathrm{KL}}(q\|p) = -\beta \log Z_\theta + \beta D_{\mathrm{KL}}(q\|q_\theta^\star)$.*

Proposition 1 is the Gibbs variational principle (Donsker & Varadhan, 1983; Catoni, 2007): the learned policy reweights the reference by $e^{-a_\theta/\beta}$, so environments with *low* score gain mass. Eliminating the policy yields the objective actually seen by the predictor.

Corollary 1 (Soft minimum over environments). $\inf_{\theta,q} \mathcal{J} = \inf_\theta J_\beta(\theta)$ *with* $J_\beta(\theta) = -\beta \log \mathbb{E}_{\gamma\sim p}\, e^{-a_\gamma(\theta)/\beta}$. *The function $\beta \mapsto J_\beta(\theta)$ is nondecreasing, $J_\beta(\theta) \le \mathbb{E}_p[a_\gamma(\theta)]$ with equality if and only if $a_\gamma(\theta)$ is p-almost surely constant, $J_\beta \to \mathbb{E}_p[a_\gamma(\theta)]$ (the frozen-policy objective) as $\beta \to \infty$ whenever $\mathbb{E}_p[a_\gamma(\theta)] < \infty$, and $J_\beta \to \operatorname{ess\,inf}_p a_\gamma(\theta)$ as $\beta \to 0$. Where differentiation under the integral is allowed, $\nabla_\theta J_\beta(\theta) = \mathbb{E}_{q_\theta^\star}[\nabla_\theta a_\gamma(\theta)]$.*

A cooperative policy therefore interpolates between the frozen-policy average and the *easiest* environment in the family, and the predictor is trained on a mixture that emphasizes environments it already handles well. The robust policy of Eq. (2) reverses this preference.

Corollary 2 (Robust policy). *For $\beta > 0$ with $\mathbb{E}_p e^{a_\theta/\beta} < \infty$, $\max_{q \ll p}\{\mathbb{E}_q[a_\theta] - \beta D_{\mathrm{KL}}(q\|p)\} = J_\beta^+(\theta) := \beta \log \mathbb{E}_p e^{a_\theta/\beta}$, attained by $dq_\theta^+/dp \propto e^{a_\theta/\beta}$, and for every $\rho \ge 0$, $\sup_{q: D_{\mathrm{KL}}(q\|p) \le \rho} \mathbb{E}_q[a_\theta] \le \beta\rho + J_\beta^+(\theta)$.*

The robust policy thus tilts the reference toward environments with a *high* score, and its objective, the entropic risk, bounds the expected score under every environment distribution within KL distance ρ of the reference (Ben-Tal et al., 2013). The bound concerns the training scores over the supplied family, not arbitrary test environments. How far either policy can move is controlled by the spread of the scores.

Proposition 2 (Departure from the reference). *If $d_\theta = \operatorname{ess\,sup}_p a_\theta - \operatorname{ess\,inf}_p a_\theta < \infty$, then $D_{\mathrm{KL}}(q_\theta^\star\|p) \le d_\theta^2/(8\beta^2)$ and $\mathrm{TV}(q_\theta^\star, p) \le \min\{1, d_\theta/(4\beta)\}$. In particular, $q_\theta^\star = p$ whenever a_θ is p-a.s. constant.*

On Colored MNIST the constant case has a concrete meaning. If f_θ ignores color, then $\ell_i^0 = \ell_i^1$ and $g_i^0 = g_i^1$, so R_e and G_e in Eq. (3) do not depend on e: *the better the predictor becomes at invariance, the less reason the policy has to leave the reference*. The opposite regime can also be made explicit. Compare an idealized color-invariant predictor, whose score $a^\star$ is the same in every environment, with a predictor that uses a shortcut.

Proposition 3 (When the cooperative objective prefers a shortcut)**.** *Let $\mathcal{Q}$ be a policy class containing p, let θ_{inv} satisfy $a_\gamma(\theta_{\mathrm{inv}}) = a^\star$ for all γ, and let θ_{s} be a predictor that the frozen policy does not prefer, $\mathbb{E}_p[a_\gamma(\theta_{\mathrm{s}})] \geq a^\star$. Then $\inf_{q\in\mathcal{Q}} \mathcal{J}(\theta_{\mathrm{s}}, q) < \min_{q\in\mathcal{Q}} \mathcal{J}(\theta_{\mathrm{inv}}, q) = a^\star$ if and only if $\beta < \beta^\star(\mathcal{Q}) = \sup\big\{\big(a^\star - \mathbb{E}_q[a_\gamma(\theta_{\mathrm{s}})]\big)/D_{\mathrm{KL}}(q\|p) : q \in \mathcal{Q},\, 0 < D_{\mathrm{KL}}(q\|p) < \infty\big\}$, with $\sup\emptyset = 0$ and $\beta^\star = \infty$ allowed. For unrestricted policies, if $a_\gamma(\theta_{\mathrm{s}}) \leq \varepsilon < a^\star$ on a set A with $0 < p(A) < 1$, then $\beta^\star \geq (a^\star - \varepsilon)/\log(1/p(A))$.*

In Colored MNIST, a color-only predictor attains risk and IRM penalty near zero at $e \approx 0$, where color *equals* the label, whereas no color-blind predictor can go below the label-noise entropy. Proposition 3 thus implies a KL weight below which the shortcut predictor, paired with a policy that favors small e, attains a lower joint objective than the invariant predictor. Under the frozen policy the same shortcut is never preferred: the reference is symmetric around $e = 0.5$, so the color-only predictor's average score is at least $\log 2 > a^\star$ (Section C). Finally, on Colored MNIST the latent KL upper-bounds the KL over e (data-processing inequality), so the latent policy family can only make concentrating mass more expensive (Section A).

Implications. The analysis has four implications for the experiments. **(P1)** When training makes the predictor nearly invariant over the family, the optimal policy is close to the reference and the learned policy should behave like the frozen one. **(P2)** Below a KL threshold, the policy can concentrate on environments in which a shortcut is most predictive, and the predictor then adopts the shortcut. **(P3)** On a partition family, the exact policy step moves weight toward groupings with a lower entropic score, by an amount bounded by Proposition 2; the learned policy only approximates this step, so we check the direction and report the observed departure. **(P4)** The robust policy moves weight the other way, toward groupings with a high score; if the annotated grouping is added to the family, the cooperative policy should give it less than its reference weight and the robust policy more. We developed this analysis after inspecting the main Colored MNIST runs, and the KL sweep and threshold calculation of Section 4.2 were carried out afterwards; for P1 and P2 the experiments are therefore consistency checks rather than blind tests. P3 was written down before any Waterbirds result of Section 4.3 was available. The robust policy and P4 were introduced after the cooperative results; P4 and the corresponding experiments were declared before they were run.

4 Experiments

The experiments ask three questions: **(Q1)** how much of the benefit of environment-based training comes from broader coverage rather than from learning the weighting; **(Q2)** whether the learned policies behave as Section 3 describes; and **(Q3)** whether learning the weighting over candidate groupings helps on a natural-image benchmark, with a cooperative or a robust policy. Table 3 (Section F) lists the information each method receives. All results use five training seeds (42, 10042, 20042, 30042, 40042) on a fixed data split; we report mean $\pm$ sample standard deviation and, for paired comparisons, the mean difference in percentage points (pp) with a 95% Student-t interval. Because checkpoint selection can itself favor shortcuts, every result is reported for two predeclared rules: the *final* checkpoint and the checkpoint selected on validation data.

4.1 Colored MNIST: coverage versus learned weighting

Setup. We use the data construction of Arjovsky et al. (2019): binary label "digit < 5" with 25% label noise, $2 \times 14 \times 14$ hard-colored inputs, the first 50,000 MNIST training images for training and the remaining 10,000 for validation, and the official test set. A color-blind predictor can reach at most 75% accuracy in every environment. All methods share an MLP with two hidden layers of width 256, Adam (learning rate 10^{-3}, policy 0.1), batch size 1024, 2001 updates of which the first 1000 are a warm-up with IRM weight 1, and IRM weight $\lambda = 100$ afterwards. As an external baseline over the same recoloring family, *uniform* ERM/IRM draws $e \sim \mathcal{U}[0, 1]$, a hand-chosen broad

Table 1: Colored MNIST test accuracy (%), $\lambda = 100$, 5 seeds. "Worst": minimum over $e \in \{0, 0.1, \ldots, 1\}$.

	Final checkpoint		Validation-selected	
Method	$e = 0.9$	Worst	$e = 0.9$	Worst
Restricted ERM	41.2 ± 1.4	36.4 ± 1.7	28.7 ± 2.1	21.8 ± 2.5
Restricted IRM	65.4 ± 2.0	64.1 ± 2.4	62.0 ± 1.7	60.1 ± 2.0
Uniform ERM	68.2 ± 2.2	65.3 ± 1.3	65.6 ± 4.0	64.4 ± 4.8
Uniform IRM	72.0 ± 1.0	71.4 ± 0.9	70.5 ± 1.1	70.1 ± 1.4
UAED family				
UAED-ERM, frozen ($\beta = \infty$)	69.0 ± 4.5	60.9 ± 5.5	64.5 ± 1.9	63.1 ± 2.2
UAED-ERM, learned ($\beta = 1$)	69.2 ± 5.3	59.9 ± 5.6	63.8 ± 2.8	62.4 ± 3.2
UAED-IRM, frozen ($\beta = \infty$)	72.4 ± 0.6	71.7 ± 0.5	71.3 ± 0.8	71.1 ± 1.0
UAED-IRM, learned ($\beta = 1$)	72.5 ± 0.6	71.6 ± 0.6	71.6 ± 0.7	71.4 ± 0.7

distribution without a policy or KL term. Transformation methods draw five environments per update; frozen-policy UAED uses the reference throughout, whereas learned-policy UAED starts from a policy concentrated near $e \approx 0.18$ and uses $\beta = 1$. Accuracy is evaluated on $e \in \{0, 0.1, \ldots, 1\}$; we report the standard test environment $e = 0.9$, the worst accuracy over this grid, and its mean. The validation rule selects, among post-warm-up checkpoints, the one with the highest worst validation accuracy at $e \in \{0.1, 0.2\}$. Full details are in Section D.

Results (Q1). Table 1 and Fig. 3 separate three levels. *Restricted versus broad.* Learned-policy UAED-IRM improves over restricted IRM by +7.1 pp at $e = 0.9$ and +7.5 pp in worst-case accuracy (CI [+4.4, +10.5]) at the final checkpoint, and by +11.3 pp (CI [+9.1, +13.5]) under validation selection; the difference is positive in every seed. Its accuracy is nearly flat in e and close to the 75% ceiling of color-blind predictors, whereas restricted IRM degrades steadily as e increases. *Which broad distribution.* Every IRM variant trained on the whole recoloring family reaches similar accuracy: relative to restricted IRM, UAED-IRM with its policy frozen at the reference gains +7.6 pp (final) and +10.9 pp (validation), and uniform IRM gains +7.2 pp and +10.0 pp. Coverage of the family, rather than the particular distribution over it, provides most of the robustness. *Learned versus fixed weights.* At the final checkpoint, learning the policy changes worst-case accuracy by -0.1 pp relative to the frozen policy (CI [-0.7, +0.4]) and by +0.2 pp relative to uniform IRM (CI [-1.5, +1.9]); under validation selection the differences are +0.3 pp (CI [-0.1, +0.8]) and +1.3 pp (CI [-0.5, +3.1]). No interval excludes zero; under validation selection the learned policy has the highest mean and the smallest spread across seeds of the three broad-family variants (Table 1). Learning the weights is therefore neutral at $\beta = 1$: it matches the fixed broad distributions but does not add robustness beyond them. For ERM, learning does not improve on either fixed distribution (-1.0 pp versus frozen, -5.4 pp versus uniform, final checkpoint).

The learned policy returns to the reference (P1). Starting from mean $e \approx 0.18$, the UAED-IRM policy reaches the reference within the first 70 updates of warm-up in every seed and stays there after the IRM penalty is activated (latent KL at most 0.02 during the following 100 updates; Fig. 2a). It ends with mean $e = 0.50$ and latent KL 0.0007. The early return is driven by the KL term; that the policy does not leave the reference later is consistent with Proposition 2: the final predictors are nearly color-invariant (Fig. 3a), so their scores vary little with e.

4.2 The KL weight decides between coverage and shortcut collapse

Setup. To examine the regime of Proposition 3, we reran UAED-IRM with $\beta \in \{0.01, 0.1\}$ and without the KL penalty ($\beta = 0$), for $\lambda \in \{100, 1000\}$, keeping everything else fixed. This sweep was run after the main results had been inspected and is exploratory; its grid and its analysis plan (compare settings only on the validation criterion) were fixed before it ran. All settings are reported (Table 5).

Results (Q2, P2). For $\beta \in \{0, 0.01\}$ the policy collapses onto small e in every seed within the first 50 warm-up updates (mean $e \leq 0.014$ at update 50; Fig. 2a). There color nearly equals the label, so a color-only predictor drives the training risk and the IRM penalty toward zero, and the final predictors

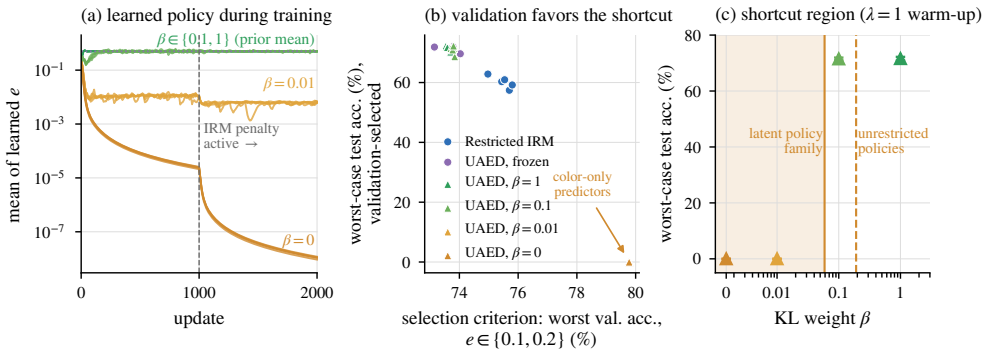

Figure 2: KL weight and shortcut collapse on Colored MNIST (UAED-IRM, $\lambda = 100$). **(a)** Mean e of the learned policy, every seed. **(b)** Validation criterion versus worst-case test accuracy of the selected checkpoint: validation favors the collapsed runs. **(c)** Worst-case test accuracy (final checkpoint) versus β; shaded: below the warm-up shortcut threshold of the latent policy family (Proposition 3); dashed: the same for unrestricted policies.

are color-only: $9.8 \pm 0.0\%$ at $e = 0.9$ and $0.0 \pm 0.0\%$ worst-case accuracy for $\beta = 0.01$. For $\beta = 0.1$ the policy stays near its initialization for about 50 updates and then returns to the reference by update 270 in every seed, and its results match $\beta = 1$ (Table 5). The same pattern holds with $\lambda = 1000$, and when the policy starts at the reference instead of near $e \approx 0.18$: for $\beta \in \{0, 0.01\}$ it leaves the reference in every seed, to $e \approx 0$ in two seeds and to $e \approx 1$, where reversed color predicts the label, in three, with worst-case accuracy $0.0 \pm 0.0\%$ for $\beta = 0.01$; for $\beta \in \{0.1, 1\}$ it stays at the reference (Section H). This is consistent with Proposition 3. For an idealized pair of a color-only and a color-invariant predictor, a numerical calculation shows that, within the latent policy family, the color-only predictor attains the lower objective for $\beta < 0.059$ during warm-up and $\beta < 0.038$ afterwards; the collapsed settings lie below these values and $\beta \in \{0.1, 1\}$ above them (Fig. 2c). This calculation was performed after the sweep. For unrestricted policies the corresponding value is higher (0.19), so the parametrization of the policy, and not only β, limits how cheaply mass can concentrate.

Validation selection rewards the shortcut. The collapsed runs attain the highest value of the validation criterion among all configurations, 79.8% compared with 73.6% for $\beta = 1$ (Fig. 2b), because validation environments with $e \in \{0.1, 0.2\}$ reward color; a practitioner tuning β on in-distribution validation data would select the worst models. Checkpoint selection also matters within runs: at $\lambda = 1000$ the learned policy is ahead of the frozen one under validation selection in every seed, but all broad-family variants degrade by the final checkpoint (Section F), in line with the observation that model selection is central to domain generalization (Gulrajani & Lopez-Paz, 2021).

4.3 Waterbirds: learned weights over candidate groupings

Setup. Waterbirds (Sagawa et al., 2020; Wah et al., 2011) places land- and waterbirds on land or water backgrounds that agree with the label for 95% of training images; the four (label, background) groups define worst-group accuracy. We fine-tune ImageNet-pretrained ResNet-50 (He et al., 2016) and ViT-B/16 (Dosovitskiy et al., 2021) with SGD (learning rate 10^{-3}, momentum 0.9, weight decay 10^{-4}) for 30 epochs without data augmentation. The reference predictor for the partition family and for JTT is the ERM model of the same seed after two epochs. UAED-GroupDRO uses the nine groupings of Section 2 with $\tau = 0.1$, class-balanced minibatches, two warm-up epochs with the plain loss, and, for learned policies, $\beta = 0.05$ with a uniform reference; the cooperative and the robust policy share every setting. These settings were fixed before the runs reported here, informed by an earlier implementation, and were not re-tuned for ViT-B/16; exploratory ResNet-50 runs with other settings ran alongside the main runs and are not reported. *Official protocol.* To rule out artifacts of our pipeline, we also run the official GroupDRO code (Sagawa et al., 2020), patched only to add UAED, with its strong-ℓ_2 setting for ResNet-50 (learning rate 10^{-5}, weight decay 1, batch 64, 300 epochs), which is also the final-model setting of JTT (Liu et al., 2021a); the reference is the ERM model after 60 epochs, as in JTT (Section G). All methods select checkpoints by worst-group validation accuracy; Sections F and G report the final checkpoint. Average accuracy is weighted by the training group proportions.

Table 2: Waterbirds test accuracy (%), validation-selected checkpoint, 5 seeds. UAED rows use the nine reference-model groupings and no group labels; GroupDRO (grey) trains on the true groups. "ℓ_2": official GroupDRO code, strong-ℓ_2 setting (Section G).

	ResNet-50		ViT-B/16		ResNet-50, ℓ_2	
Method	Worst	Average	Worst	Average	Worst	Average
ERM	65.1 ± 1.3	97.2 ± 0.1	63.3 ± 2.2	96.9 ± 0.2	29.9 ± 3.3	95.9 ± 0.1
ERM (class-bal.)	67.1 ± 1.6	97.1 ± 0.1	64.5 ± 1.7	96.6 ± 0.3	49.9 ± 1.7	95.8 ± 0.0
JTT	75.7 ± 4.1	90.5 ± 4.0	63.1 ± 2.7	96.5 ± 0.4	–	–
UAED, frozen	72.0 ± 1.3	97.0 ± 0.3	26.3 ± 2.0	82.4 ± 1.9	81.8 ± 1.7	97.1 ± 0.3
UAED, cooperative	71.5 ± 0.7	97.0 ± 0.4	27.3 ± 4.2	85.3 ± 2.3	80.0 ± 1.4	97.1 ± 0.1
UAED, robust	71.5 ± 1.8	97.0 ± 0.4	–	–	81.9 ± 1.7	97.1 ± 0.2
GroupDRO (true group label)	85.3 ± 0.7	94.9 ± 0.5	79.6 ± 3.2	95.6 ± 0.7	86.1 ± 1.1	92.7 ± 2.3

Results (Q3): groupings versus ERM. With ResNet-50, UAED-GroupDRO improves worst-group accuracy over ERM by +6.4 pp with the cooperative policy (95% CI [+5.0, +7.8], positive in 5/5 seeds) and by +6.8 pp with the frozen policy (CI [+4.7, +9.0]), and over class-balanced ERM by +4.4 pp (CI [+2.8, +6.1], 5/5 seeds), so the class-balanced sampler alone does not explain the gain. At the final checkpoint the gains are +8.2 pp and +2.5 pp (Table 6). Against JTT, which uses the same reference model, the sign depends on the rule: -4.2 pp under validation selection (CI [-9.8, +1.4]) and +3.2 pp at the final checkpoint (CI [-2.0, +8.4]). GroupDRO with annotated groups remains clearly stronger, as does EIIL as reported under a 300-epoch protocol (78.7%; Creager et al., 2021).

Official protocol. The official code reproduces the published strong-ℓ_2 pattern: ERM reaches 29.9 $\pm$ 3.3% worst-group accuracy (Sagawa et al. (2020) report 21.3%) and GroupDRO 86.1 $\pm$ 1.1% (84.6%). Without group annotations, frozen UAED-GroupDRO reaches 81.8 $\pm$ 1.7% (Fig. 1a), -4.2 pp relative to GroupDRO with annotations. Class balancing explains only part of the gain over ERM: class-balanced ERM reaches 49.9 $\pm$ 1.7% (+20.0 pp over ERM), and the groupings add +31.9 pp more (CI [+28.5, +35.3], positive in 5/5 seeds). Published results for this setting are in Section G.

Cooperative versus frozen weights (P1, P3). Learning the weights cooperatively over the same nine groupings changes worst-group accuracy by -0.4 pp with ResNet-50 (CI [-2.3, +1.4], positive in 1/5 seeds) and by +1.0 pp with ViT-B/16 (CI [-6.2, +8.2]); as on Colored MNIST, the learned weighting is neutral. Under the official protocol it is harmful: -1.9 pp (CI [-2.7, -1.0], positive in 0/5 seeds). The logged policies explain this difference. After warm-up the policy moves toward the softest groupings ($T = 1$), the direction P3 predicts: at its largest departure from the uniform reference (epoch 5, KL 0.06) it places 0.47 of its mass on them, against $1/3$ under the reference. As the training loss approaches zero, all scores shrink and the policy returns to the reference (KL $<$0.001 at epoch 30), as Proposition 2 implies for a vanishing spread of the scores (P1). Under the official protocol the policy departs much further (Fig. 1b), to KL 0.98 at epoch 38 with 0.94 of its mass on the softest groupings, and trains the model on this easier, optimistic objective (Corollary 1) for most of the run before returning to the reference.

Robust policy (P4). Reversing the sign of the policy update reverses the direction of its departure, as P4 predicts. With ResNet-50 in the main protocol, the robust policy moves toward the tightest groupings ($T = 0.05$), with 0.41 of its mass on them at its largest departure (epoch 5, KL 0.06), where the cooperative policy moves toward the softest; both return to the reference as the training loss vanishes. When the annotated (label, background) grouping is added to the family, the cooperative policy lowers its weight below the reference value $1/10$ in every seed (0.08 at its largest departure) and the robust policy raises it above $1/10$ in every seed (0.11; Fig. 1c). The resulting accuracy differences in this pipeline are within its run-to-run variation (up to 0.75 pp, from non-deterministic cuDNN autotuning; Table 8). Under the official protocol, where the cooperative policy costs -1.9 pp, the robust policy moves toward the tightest groupings (0.51 of its mass at epoch 5, KL 0.25) before returning to the reference, and it removes this cost: it improves on the cooperative policy by +1.9 pp (CI [+0.9, +2.9], positive in 5/5 seeds) and matches the frozen policy (+0.1 pp, CI [-0.2, +0.3]). This code is bitwise reproducible, so these differences are not run-to-run

variation. On Colored MNIST with $\beta = 1$ neither policy leaves the reference, and the robust policy matches the frozen one (-0.2 pp worst-case, final checkpoint; Section H).

A failure with ViT-B/16. With ViT-B/16 both frozen and cooperative UAED fall far below ERM, by -37.1 pp and -36.1 pp (0/5 seeds positive), and further by the final checkpoint; we found no implementation error (Section I). The failure starts when the partition score replaces the plain loss after warm-up: the training loss rises by a factor of 6.6 in every seed, while it keeps decreasing with ResNet-50 and for every other ViT-B/16 method. In exploratory diagnostic runs designed after observing this failure (Section I), lowering the learning rate from 10^{-3} to 10^{-4} removes the loss increase and gives the frozen policy $64.2 \pm 1.3\%$ worst-group accuracy, the same as class-balanced ERM at the same lower learning rate ($64.4 \pm 1.4\%$), whose accuracy does not depend on the learning rate ($64.5 \pm 1.7\%$ at 10^{-3}). The near-maximum over small groups defined by the reference model thus makes fine-tuning unstable at a learning rate that is safe for the plain loss; once stable, the grouping objective matches but does not improve on class-balanced ERM with this backbone.

5 Related Work

Environments, groups and augmentation. IRM, REx and GroupDRO (Arjovsky et al., 2019; Krueger et al., 2021; Sagawa et al., 2020) and feature alignment (Sun & Saenko, 2016) assume given environments or groups, and their behavior depends strongly on them (Rosenfeld et al., 2021; Kamath et al., 2021; Gulrajani & Lopez-Paz, 2021); IRM can also fail when an overparameterized model fits its training environments (Lin et al., 2022). EIIL infers the split that maximally violates an invariance criterion under a reference model (Creager et al., 2021), HRM alternates between inferring heterogeneity and learning invariant predictors (Liu et al., 2021b), Huang et al. (2022) diversify environments with auxiliary heads, and LfF, JTT and CnC use a biased or early-stopped reference model to find examples that contradict the shortcut (Nam et al., 2020; Liu et al., 2021a; Zhang et al., 2022). Our partition family also starts from a reference model but keeps several candidate groupings and weights them under an explicit KL budget; the robust policy is a KL-regularized adversary over this supplied family rather than over all splits, and both policies are Gibbs tilts of the reference (Catoni, 2007). AutoAugment and RandAugment search for augmentation policies (Cubuk et al., 2019; 2020), Augerino and OPTIMA learn transformation distributions jointly with the model (Benton et al., 2020; Matymov et al., 2026), and targeted augmentation improves out-of-domain robustness (Gao et al., 2023; Yao et al., 2022); for transformation families UAED is augmentation learning with an invariance penalty, and its frozen version, broad fixed augmentation, already accounts for the gain over restricted environments.

6 Discussion and Limitations

Across both benchmarks, the benefit of UAED comes from the family it is trained over rather than from learning the weights, as the soft-minimum analysis predicts. On the official GroupDRO code, the direction of the policy update determines whether learning the weights costs accuracy: the cooperative policy lowers worst-group accuracy in every seed, whereas the robust policy does not. In practice, we recommend a broad reference distribution, reporting the policy's departure from it, and not tuning the KL weight on in-distribution validation data, which rewards the shortcut.

Limitations. The transformation family must be supplied, and on Colored MNIST it encodes the knowledge that color is the nuisance; the partition family needs no group labels but depends on a reference model and on nine groupings chosen with ResNet-50. The analysis concerns the exact policy step and idealized predictors, with no guarantee for arbitrary test environments. The β sweep, the $\lambda = 1000$ runs, the ViT-B/16 diagnosis and the robust policy were added after the main results, each declared before it ran; all seeds share one data split.

REPRODUCIBILITY STATEMENT

Code, configurations, per-seed results and the script that generates every number and figure in this paper from the raw result files are included in the supplementary material. Sections B and D specify the estimators, architectures, schedules, seeds and selection rules; every configuration is stored with its run, and checkpoints are selected by predeclared rules that use validation data only.

AI USE STATEMENT

We used generative AI tools to assist with writing and debugging experiment code, producing plotting scripts, and editing the manuscript text. All generated code was tested (including gradient checks against enumeration and finite differences), all reported numbers are computed by scripts from raw result files, and the authors reviewed all AI-assisted content and take full responsibility for it.

ETHICS STATEMENT

This work uses public benchmark datasets (MNIST, Waterbirds) and raises no specific ethical concerns beyond the general risk that robustness methods can give a false sense of reliability; we therefore report negative and null results alongside positive ones.

REFERENCES

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant Risk Minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Aharon Ben-Tal, Dick den Hertog, Anja De Waegenaere, Bertrand Melenberg, and Gijs Rennen. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357, 2013.

Gregory Benton, Marc Finzi, Pavel Izmailov, and Andrew Gordon Wilson. Learning Invariances in Neural Networks from Training Data. In *Advances in Neural Information Processing Systems*, volume 33, pp. 17605–17616, 2020.

Olivier Catoni. PAC-Bayesian supervised classification: The thermodynamics of statistical learning. *Institute of Mathematical Statistics Lecture Notes–Monograph Series*, 56, 2007.

Elliot Creager, Jörn-Henrik Jacobsen, and Richard Zemel. Environment Inference for Invariant Learning. In *International Conference on Machine Learning*, pp. 2189–2200. PMLR, 2021.

Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. AutoAugment: Learning Augmentation Strategies From Data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 113–123, 2019.

Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. RandAugment: Practical Automated Data Augmentation with a Reduced Search Space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 702–703, 2020.

Monroe D Donsker and SR Srinivasa Varadhan. Asymptotic evaluation of certain Markov process expectations for large time. *Communications on Pure and Applied Mathematics*, 36(2):183–212, 1983.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Irena Gao, Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Out-of-Domain Robustness via Targeted Augmentations. In *International Conference on Machine Learning*, pp. 10799–10820. PMLR, 2023.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Bo-Wei Huang, Keng-Te Liao, Chang-Sheng Kao, and Shou-De Lin. Environment diversification with multi-head neural network for invariant learning. In *Advances in Neural Information Processing Systems*, volume 35, pp. 32370–32383, 2022.

Pritish Kamath, Akilesh Tangella, Danica Sutherland, and Nathan Srebro. Does invariant risk minimization capture invariance? In *International Conference on Artificial Intelligence and Statistics*, pp. 4069–4077. PMLR, 2021.

David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-Distribution Generalization via Risk Extrapolation (REx). In *International Conference on Machine Learning*, pp. 5815–5826. PMLR, 2021.

Yong Lin, Hanze Dong, Hao Wang, and Tong Zhang. Bayesian invariant risk minimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16021–16030, 2022.

Evan Z Liu, Behzad Haghgoo, Annie S Chen, Aditi Raghunathan, Pang Wei Koh, Shiori Sagawa, Percy Liang, and Chelsea Finn. Just Train Twice: Improving Group Robustness without Training Group Information. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6781–6792. PMLR, 18–24 Jul 2021a. URL https://proceedings.mlr.press/v139/liu21f.html.

Jiashuo Liu, Zheyuan Hu, Peng Cui, Bo Li, and Zheyan Shen. Heterogeneous Risk Minimization. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6804–6814. PMLR, 18–24 Jul 2021b. URL https://proceedings.mlr.press/v139/liu21h.html.

Madi Matymov, Ba-Hien Tran, Michael Kampffmeyer, Markus Heinonen, and Maurizio Filippone. Optimizing data augmentation through Bayesian model selection. In *International Conference on Learning Representations*, 2026.

Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. Learning from failure: Training debiased classifier from biased classifier. In *Advances in Neural Information Processing Systems*, 2020.

Elan Rosenfeld, Pradeep Ravikumar, and Andrej Risteski. The Risks of Invariant Risk Minimization. In *International Conference on Learning Representations*, 2021.

Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally Robust Neural Networks for Group Shifts: On the Importance of Regularization for Worst-Case Generalization. In *International Conference on Learning Representations*, 2020.

Baochen Sun and Kate Saenko. Deep CORAL: Correlation Alignment for Deep Domain Adaptation. In *European Conference on Computer Vision*, pp. 443–450. Springer, 2016.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

Huaxiu Yao, Yu Wang, Sai Li, Linjun Zhang, Weixin Liang, James Zou, and Chelsea Finn. Improving Out-of-Distribution Robustness via Selective Augmentation. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 25407–25437. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/yao22b.html.

Michael Zhang, Nimit S. Sohoni, Hongyang R. Zhang, Chelsea Finn, and Christopher Ré. Correct-N-Contrast: A contrastive approach for improving robustness to spurious correlations. In *Proceedings of the 39th International Conference on Machine Learning*, 2022.

A PROOFS

Proof of Proposition 1. Since $a_\theta \geq 0$, $0 < Z_\theta \leq 1$ and $q_\theta^\star$ is a probability measure equivalent to p. For $q \ll p$ with $\mathbb{E}_q[a_\theta] < \infty$, $\log \frac{dq}{dq_\theta^\star} = \log \frac{dq}{dp} + \frac{a_\theta}{\beta} + \log Z_\theta$, and integrating against q gives $D_{\mathrm{KL}}(q\|q_\theta^\star) = D_{\mathrm{KL}}(q\|p) + \mathbb{E}_q[a_\theta]/\beta + \log Z_\theta$. Multiplying by β yields the identity; if $\mathbb{E}_q[a_\theta] = \infty$ both sides are infinite. Since $D_{\mathrm{KL}}(q\|q_\theta^\star) \geq 0$ with equality if and only if $q = q_\theta^\star$, the minimizer is unique. □

Proof of Corollary 1. The value of the inner minimization over q is $J_\beta(\theta) = -\beta \log Z_\theta$ by Proposition 1. With $t = 1/\beta$ and $X = -a_\theta(\gamma)$, $J_\beta = -\frac{1}{t}\log \mathbb{E}_p e^{tX}$, and $t \mapsto \frac{1}{t}\log \mathbb{E}_p e^{tX}$ is nondecreasing on $(0, \infty)$ (it is the slope from the origin of the convex function $t \mapsto \log \mathbb{E}_p e^{tX}$, which vanishes at 0); hence J_β is nondecreasing in β. Jensen's inequality gives $\mathbb{E}_p e^{-a/\beta} \geq e^{-\mathbb{E}_p a/\beta}$, i.e. $J_\beta \leq \mathbb{E}_p[a]$, with equality if and only if a is p-a.s. constant (strict convexity of the exponential). As $t \to 0$, $\frac{1}{t}\log \mathbb{E}_p e^{tX} \to \mathbb{E}_p X$ whenever $\mathbb{E}_p a < \infty$; as $t \to \infty$ it tends to $\operatorname{ess\,sup} X = -\operatorname{ess\,inf} a$ (Laplace principle). Finally $\nabla_\theta J_\beta = -\beta \nabla_\theta Z_\theta / Z_\theta = \mathbb{E}_p[e^{-a/\beta}\nabla_\theta a]/Z_\theta = \mathbb{E}_{q_\theta^\star}[\nabla_\theta a]$. For the maximizing policy, $\max_q\{\mathbb{E}_q[a] - \beta D_{\mathrm{KL}}(q\|p)\} = \beta \log \mathbb{E}_p e^{a/\beta}$ by the same argument applied to $-a$; this is the entropic risk, whose dual is the KL-constrained worst case $\sup_{q: D_{\mathrm{KL}}(q\|p) \leq \rho} \mathbb{E}_q[a] = \inf_{\beta > 0}\{\beta\rho + \beta \log \mathbb{E}_p e^{a/\beta}\}$ (Ben-Tal et al., 2013). □

Proof of Corollary 2. By the Donsker–Varadhan variational formula, $\log \mathbb{E}_p e^{a/\beta} = \sup_{q \ll p}\{\mathbb{E}_q[a]/\beta - D_{\mathrm{KL}}(q\|p)\}$ whenever $\mathbb{E}_p e^{a/\beta} < \infty$, with the supremum attained by $dq^+/dp = e^{a/\beta}/\mathbb{E}_p e^{a/\beta}$ (the identity of Proposition 1 with the sign of a reversed). Multiplying by β gives the first claim, and for $D_{\mathrm{KL}}(q\|p) \leq \rho$ it gives $\mathbb{E}_q[a] \leq \beta D_{\mathrm{KL}}(q\|p) + J_\beta^+ \leq \beta\rho + J_\beta^+$. □

Proof of Proposition 2. Let $H(t) = \log \mathbb{E}_p e^{-ta}$ and q_t the tilted measure $dq_t/dp = e^{-ta - H(t)}$, so $q_\theta^\star = q_{1/\beta}$. Then $H'(t) = -\mathbb{E}_{q_t}[a]$, $H''(t) = \mathrm{Var}_{q_t}(a) \leq d^2/4$ (Popoviciu's inequality), and $D_{\mathrm{KL}}(q_t\|p) = \mathbb{E}_{q_t}[-ta - H(t)] = tH'(t) - H(t) = \int_0^t sH''(s)\,ds \leq t^2 d^2/8$, using $H(0) = 0$. Setting $t = 1/\beta$ gives the KL bound, and Pinsker's inequality $\mathrm{TV} \leq \sqrt{D_{\mathrm{KL}}/2}$ gives the total-variation bound. If $d = 0$ then $q_\theta^\star = p$. □

Proof of Proposition 3. For θ_{inv}, $\mathcal{J}(\theta_{\mathrm{inv}}, q) = a^\star + \beta D_{\mathrm{KL}}(q\|p) \geq a^\star$ with equality at $q = p \in \mathcal{Q}$. For θ_{s}, some $q \in \mathcal{Q}$ attains $\mathbb{E}_q[a(\theta_{\mathrm{s}})] + \beta D_{\mathrm{KL}}(q\|p) < a^\star$ if and only if $\beta D_{\mathrm{KL}}(q\|p) < a^\star - \mathbb{E}_q[a(\theta_{\mathrm{s}})]$ for some q; for $q = p$ this would require $\mathbb{E}_p[a(\theta_{\mathrm{s}})] < a^\star$, which is excluded by assumption, and for $q \neq p$ (so $D_{\mathrm{KL}}(q\|p) > 0$) it reads $\beta < (a^\star - \mathbb{E}_q[a(\theta_{\mathrm{s}})])/D_{\mathrm{KL}}(q\|p)$. Such a q exists if and only if β is below the supremum $\beta^\star(\mathcal{Q})$. For unrestricted policies take $q = p(\cdot \mid A)$: $D_{\mathrm{KL}}(q\|p) = \log(1/p(A))$ and $\mathbb{E}_q[a(\theta_{\mathrm{s}})] \leq \varepsilon$. □

Entropic partition score. For group risks $r \in \mathbb{R}^G$, apply Proposition 1 to the nonnegative scores $\max_h r_h - r_g$ with temperature τ and the uniform reference: $\max_{u \in \Delta_G}\{\sum_g u_g r_g - \tau D_{\mathrm{KL}}(u\|\mathrm{unif})\} = \max_h r_h - \min_u\{\mathbb{E}_u[\max_h r_h - r_g] + \tau D_{\mathrm{KL}}(u\|\mathrm{unif})\} = \tau \log\big(\frac{1}{G}\sum_g e^{r_g/\tau}\big)$. Bounding the sum between its largest term and G times it gives $\max_g r_g - \tau \log G \leq A \leq \max_g r_g$.

Latent KL. On Colored MNIST, $e = \sigma(\mu + e^s z)$ with $z \sim \mathcal{N}(0,1)$ independent of (μ, s) under both the policy and the reference. The induced distributions over e are the images of $q(\mu, s) \otimes \mathcal{N}(0,1)$ and $p(\mu, s) \otimes \mathcal{N}(0,1)$ under the same map, so by the data-processing inequality $D_{\mathrm{KL}}(q_e\|p_e) \leq D_{\mathrm{KL}}(q_{\mu,s} \otimes \mathcal{N}\|p_{\mu,s} \otimes \mathcal{N}) = D_{\mathrm{KL}}(q_{\mu,s}\|p_{\mu,s})$. The penalty we use is therefore at least as strong as the KL over e, and the optimal value of Eq. (1) over the latent family is at least $J_\beta(\theta)$.

B THE COLORED MNIST ESTIMATOR

For image i with label y_i let x_i^0 and x_i^1 be its versions whose color agrees and disagrees with y_i, and in environment e let $c_i \sim \mathrm{Bernoulli}(e)$ independently select the version. The minibatch risk $\frac{1}{|B|}\sum_i \ell(f(x_i^{c_i}), y_i)$ has expectation R_e as stated in Section 2. For the IRMv1 penalty, the minibatch scale gradient is $\bar{g} = \frac{1}{|B|}\sum_i g_i^{c_i}$ with $g_i^c = (\sigma(f(x_i^c)) - y_i) f(x_i^c)$ for the logistic loss. Its mean is $\frac{1}{|B|}\sum_i [(1-e)g_i^0 + e g_i^1]$ and, by independence, its variance is $\frac{1}{|B|^2}\sum_i e(1-e)(g_i^1 - g_i^0)^2$; $\mathbb{E}[\bar{g}^2]$ is their sum, which is Eq. (3). Both R_e and G_e are exact expectations over the color draws and are differentiable in e, so task gradients reach the policy through the reparameterized samples of e. We verified both formulas against enumeration over all $2^{|B|}$ color assignments for small batches, the scale gradient against automatic differentiation, and $\partial_e R_e, \partial_e G_e$ against finite differences. If f ignores color, $g_i^0 = g_i^1$ and $\ell_i^0 = \ell_i^1$, so neither term depends on e.

C SHORTCUT THRESHOLD ON COLORED MNIST

We instantiate Proposition 3 with two idealized predictors. The color-invariant predictor recovers the clean label from shape and outputs the Bayes logit $\pm \log 3$; its risk is the label-noise entropy $H(0.25)$, its two color versions coincide so the variance term of Eq. (3) vanishes, and the squared minibatch mean of g has expectation $\mathrm{Var}(g)/|B|$. This gives $a^\star = 0.563$ for $\lambda = 1$ (warm-up) and $a^\star = 0.584$ for $\lambda = 100$. The color-only predictor outputs $\pm c$ according to the color, so in environment e its risk is $(1-e)\log(1+e^{-c}) + e\log(1+e^{c})$ and its per-example scale gradients are $g^0 = -c\,\sigma(-c)$ and $g^1 = c\,\sigma(c)$. The reference is symmetric around $e = 0.5$, so under it the color-only risk averages to $\frac{1}{2}[\log(1+e^{-c}) + \log(1+e^{c})] \geq \log 2 > a^\star$ (equality only at $c = 0$), and the frozen policy never prefers it. For the latent Gaussian policy family we maximize $(a^\star - \mathbb{E}_q[a(\theta_s)])/D_{\mathrm{KL}}(q\|p)$ over the four policy parameters and c (Nelder–Mead from four starting points, 40,000 common random samples); any attained value v shows numerically that the color-only predictor attains the lower objective for all $\beta < v$; we obtain $v = 0.059$ ($\lambda = 1$) and $v = 0.038$ ($\lambda = 100$). For unrestricted policies we compute J_β by Monte Carlo under the reference and bisect on β, giving 0.19 and 0.15; the reference places mass ≈ 0.03 on $e < 0.01$. These values were computed after the β sweep had been run. The comparison concerns two idealized predictors and not the global optimum over all predictors.

D IMPLEMENTATION DETAILS

Colored MNIST. Labels are "digit < 5" flipped with probability 0.25; images are subsampled to 14×14 and placed in one of two channels according to the color. Data split, label noise and evaluation colors use data seed 42 and are shared by all methods and training seeds. MLP 392–256–256–1 with ReLU; Adam with learning rate 10^{-3} for the model and 0.1 for the policy; batch size 1024; 2001 updates; updates 1–1000 use IRM weight 1 and updates 1001–2001 use λ; for IRM methods both optimizers are reset when λ is activated; the whole objective is divided by $\max(1, \lambda_t)$, which preserves the relative weights of risk, penalty and KL; model and policy gradient norms are clipped separately at 10. Transformation methods draw five environments per update from the policy; restricted methods evaluate every minibatch in both environments $e \in \{0.1, 0.2\}$. The reference policy is $(m_\mu, l_\mu, m_s, l_s) = (0, 0, 0, 0)$ and UAED is initialized at $(-1.5, -5, -2, -5)$, a concentrated policy with mean $e \approx 0.18$; the policy is trained from the first update, $\beta = 1$. Validation checkpoints are evaluated every 100 updates and at the final update 2001; only post-warm-up checkpoints are eligible; ties are broken by lower mean validation negative log-likelihood and then by the earlier update. Checkpoint identities are written to disk before any test evaluation.

Waterbirds. Images are resized to 256×256 and center-cropped to 224×224; no augmentation. torchvision ImageNet-1k weights with a new two-class head; SGD with momentum 0.9, learning rate 10^{-3}, weight decay 10^{-4}; 30 epochs; batch size 128 (ResNet-50) or 64 (ViT-B/16); bfloat16 autocast. ERM uses uniform minibatches; class-balanced ERM and UAED (frozen and learned) use class-balanced minibatches (class labels only); GroupDRO uses the online algorithm of Sagawa et al. (2020) with step size 0.01, group-balanced minibatches and training group labels; JTT upweights by 100 the training examples misclassified by the reference model. The reference model is the ERM

run of the same seed and backbone after epoch 2 and is not selected on validation data. Partition methods train with the plain loss for two epochs with the policy fixed; UAED then learns softmax logits with Adam (learning rate 10^{-2}), $\beta = 0.05$, $\tau = 0.1$. Only post-warm-up epochs are eligible for selection.

E THE UAED UPDATE

Algorithm 1 summarizes one update of Eq. (1).

Algorithm 1 One UAED update (both parameters minimize the same objective).

Require: predictor θ, policy ϕ, reference Π_0, KL weight β, number of sampled constructions M

1: draw a minibatch B from the training set
2: draw $\gamma_1, \ldots, \gamma_M \sim \Pi_\phi$ by reparameterization (finite families: use all $\gamma \in \Gamma$ with weights $\Pi_\phi(\gamma)$)
3: compute the scores $a_{\gamma_m}(\theta; B)$, e.g. risk plus invariance penalty in environment γ_m
4: $\widehat{\mathcal{J}} \leftarrow \frac{1}{M} \sum_m a_{\gamma_m}(\theta; B) + \beta\, D_{\mathrm{KL}}(\Pi_\phi \| \Pi_0)$
5: take an optimizer step on (θ, ϕ) using $\nabla_{(\theta,\phi)} \widehat{\mathcal{J}}$ ▷ frozen policy: θ only

F ADDITIONAL RESULTS

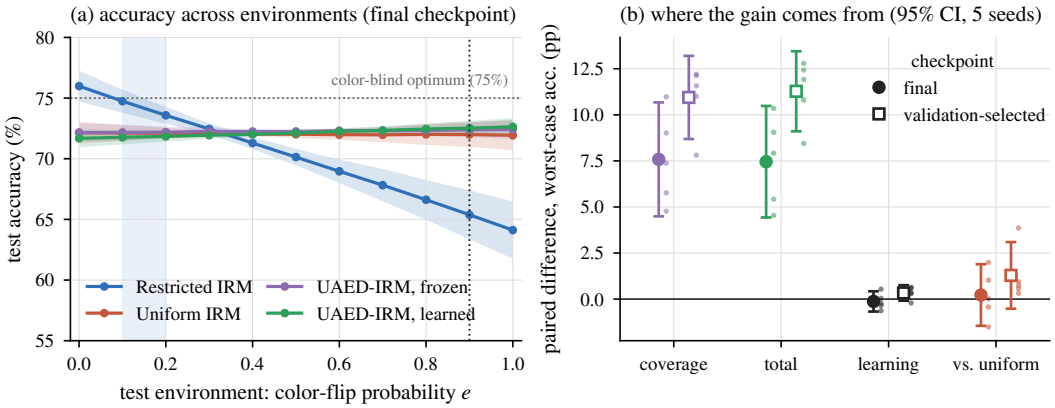

Figure 3: Colored MNIST, $\lambda = 100$, 5 seeds. **(a)** Test accuracy across environments; shaded: training environments of restricted methods; dotted: $e = 0.9$ and the 75% color-blind ceiling. **(b)** Paired differences in worst-case accuracy (mean, 95% interval, seeds).

Table 3: Information available to each method during training. All methods use the validation split for checkpoint selection; on Waterbirds this split carries group labels.

Method	Environments or groups used in training	Extra annotations
Colored MNIST		
Restricted ERM/IRM	two recolorings, $e \in \{0.1, 0.2\}$	which feature to recolor
Uniform ERM/IRM	all recolorings, $e \sim \mathcal{U}[0, 1]$	which feature to recolor
UAED-ERM/IRM, frozen	all recolorings, reference weights $p(e)$	which feature to recolor
UAED-ERM/IRM, learned	all recolorings, learned weights $q(e)$	which feature to recolor
Waterbirds		
ERM, ERM (class-balanced)	none	none
JTT (Liu et al., 2021a)	upweighted reference-model errors	none
UAED-GroupDRO, frozen	9 margin-based groupings, uniform weights	none
UAED-GroupDRO, cooperative	9 margin-based groupings, learned weights	none
GroupDRO (Sagawa et al., 2020)	true (label, background) groups	background labels

Standard fixed-color construction. In the standard construction each training image keeps one color draw, with $e = 0.1$ for one half of the training set and $e = 0.2$ for the other. Under our

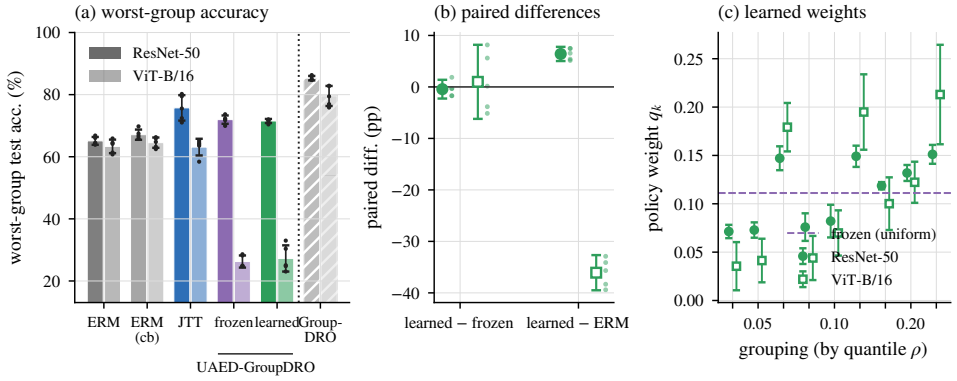

Figure 4: Waterbirds, main protocol, validation-selected, 5 seeds. **(a)** Worst-group test accuracy; GroupDRO (hatched) uses the true groups. **(b)** Paired differences with 95% intervals. **(c)** Learned partition weights against the uniform reference (dashed).

Table 4: Colored MNIST: mean accuracy over $e \in \{0, 0.1, \ldots, 1\}$ ($\lambda = 100$), and the standard fixed-color construction of Arjovsky et al. (2019) trained with our schedule.

	Final checkpoint		Validation-selected	
Method	$e = 0.9$	Mean over grid	$e = 0.9$	Mean over grid
Standard ERM (fixed colors)	29.7 ± 1.0	56.0 ± 0.4	27.2 ± 1.9	55.8 ± 0.5
Standard IRM (fixed colors)	29.8 ± 3.2	56.8 ± 1.1	27.9 ± 2.0	56.4 ± 1.0
Restricted IRM	65.4 ± 2.0	70.1 ± 0.7	62.0 ± 1.7	69.3 ± 0.5
UAED-IRM, frozen	72.4 ± 0.6	72.3 ± 0.1	71.3 ± 0.8	72.2 ± 0.3
UAED-IRM, learned	72.5 ± 0.6	72.1 ± 0.1	71.6 ± 0.7	72.2 ± 0.3

schedule (1000 warm-up updates without weight decay, about 40 passes over each half) the logged training risk at the end of warm-up is 0.13–0.19 across seeds and methods, below the Bayes risk of a predictor that uses both shape and color (about 0.35 averaged over the two environments), so the network fits individual training images, and IRM does not recover (Table 4). We attribute this to the overfitting failure of IRM discussed by Lin et al. (2022) but did not investigate it further. Arjovsky et al. (2019) train for 501 full-batch steps with weight decay; their published result is not comparable to our matched schedule, and we do not use the standard construction as a baseline. Integrating over the color draws, as all methods in Table 1 do, scores every image in both colors, so a fixed color per image cannot be memorized, although label noise can still be fit through shape.

Table 5: Colored MNIST, exploratory runs: UAED-IRM for several KL weights, and all IRM methods with $\lambda = 1000$. "Val. crit." is the checkpoint-selection criterion (worst validation accuracy at $e \in \{0.1, 0.2\}$) of the selected checkpoints.

		Worst over grid			
Setting	Method	Final	Validation	Val. crit.	Mean e (final)
$\lambda = 100,\ \beta = 0$	UAED-IRM	0.0 ± 0.0	0.0 ± 0.0	79.8	0.00
$\lambda = 100,\ \beta = 0.01$	UAED-IRM	0.0 ± 0.0	0.0 ± 0.0	79.8	0.01
$\lambda = 100,\ \beta = 0.1$	UAED-IRM	71.5 ± 0.5	70.9 ± 1.4	73.8	0.50
$\lambda = 100,\ \beta = 1$	UAED-IRM	71.6 ± 0.6	71.4 ± 0.7	73.6	0.50
$\lambda = 1000,\ \beta = 0$	UAED-IRM	0.0 ± 0.0	0.0 ± 0.0	79.8	0.00
$\lambda = 1000,\ \beta = 0.01$	UAED-IRM	0.0 ± 0.0	0.0 ± 0.0	79.8	0.00
$\lambda = 1000,\ \beta = 0.1$	UAED-IRM	51.7 ± 5.3	68.5 ± 2.6	71.6	0.50
$\lambda = 1000,\ \beta = 1$	UAED-IRM	51.7 ± 2.0	69.3 ± 1.4	72.5	0.50
$\lambda = 1000$	Standard IRM	40.7 ± 5.3	35.4 ± 3.5	75.9	–
	Restricted IRM	43.9 ± 9.5	52.9 ± 9.3	55.8	–
	Uniform IRM	50.8 ± 2.7	65.9 ± 3.3	68.9	–
	UAED-IRM, frozen	51.4 ± 6.7	67.8 ± 2.6	72.3	–

Larger IRM weight. With $\lambda = 1000$ (Table 5 and Fig. 5) and validation selection, the learned policy is ahead of the frozen one in every seed (+1.4 pp worst-case, CI [−0.1, +3.0]; +0.9 pp averaged over the grid, CI [+0.1, +1.7]) and ahead of uniform IRM by +3.2 pp averaged over the grid (CI [+0.7, +5.8]). By the final checkpoint, however, all three broad-family variants drift toward a nearly constant, low-confidence predictor with about 52% accuracy (the logged training risk rises toward $\log 2$ and test accuracy becomes flat in e), and their differences vanish. With the objective rescaled by $1/\lambda$ the risk carries little weight, and a constant output has zero IRM penalty. We therefore regard the advantage of the learned weights at $\lambda = 1000$ as suggestive rather than established. Restricted IRM reaches $77.7 \pm 1.6\%$ at $e = 0.9$, above the 75% ceiling of color-blind predictors, which is only possible by relying on *reversed* color; its worst-case accuracy is $43.9 \pm 9.5\%$. Accuracy at $e = 0.9$ alone can therefore be misleading, which is why we report the worst case over the grid.

Independent rerun. All Colored MNIST numbers in this paper come from one run of the released code on a single GPU. An earlier, independent run of the same code on CPU, without the uniform baseline, led to the same conclusions: learned-policy UAED-IRM improved worst-case accuracy over restricted IRM by 7.0 pp at the final checkpoint (95% CI [4.4, 9.5], all seeds), the learned and frozen policies were indistinguishable, and $\beta \leq 0.01$ collapsed onto the color shortcut in every seed. We report the GPU run only and do not pool the two.

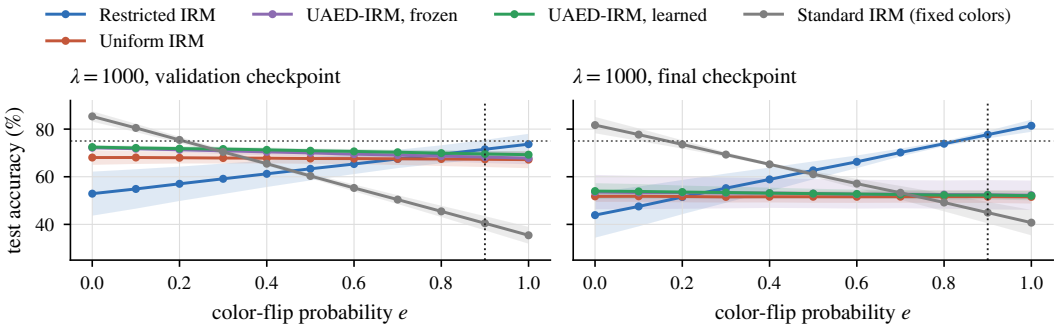

Figure 5: Colored MNIST, $\lambda = 1000$ (exploratory), 5 seeds: test accuracy across environments for both checkpoint rules. Dotted: $e = 0.9$ and the 75% color-blind ceiling.

Table 6: Waterbirds test accuracy (%) at the final checkpoint (epoch 30), 5 seeds, and the cooperative minus frozen difference in worst-group accuracy (95% CI). Only GroupDRO (grey) uses group labels.

	ResNet-50		ViT-B/16	
Method	Worst-group	Average	Worst-group	Average
Without training group labels				
ERM	61.6 ± 1.8	97.2 ± 0.1	61.9 ± 1.6	96.8 ± 0.1
ERM (class-balanced)	67.2 ± 1.4	97.1 ± 0.1	61.3 ± 2.1	96.9 ± 0.2
JTT	66.5 ± 4.1	96.9 ± 0.1	57.3 ± 1.4	96.9 ± 0.3
UAED-GroupDRO, frozen	70.0 ± 1.4	97.4 ± 0.2	9.1 ± 1.7	89.1 ± 1.1
UAED-GroupDRO, cooperative	69.7 ± 1.2	97.5 ± 0.2	10.6 ± 2.3	90.2 ± 0.5
Reference with training group labels				
GroupDRO	75.9 ± 1.4	97.3 ± 0.1	61.7 ± 8.5	96.8 ± 0.4
cooperative − frozen (pp)	−0.3 [−1.0, +0.5]		+1.5 [−3.0, +6.0]	

G Official GroupDRO code

Code and protocol. We use the official code of Sagawa et al. (2020) (commit `cbbc1c5`) with the official Waterbirds arguments, ResNet-50, learning rate 10^{-5}, weight decay 1, batch size 64 and 300 epochs, i.e. the strong-ℓ_2 setting of Sagawa et al. (2020), which is also the final-model setting of JTT (Liu et al., 2021a). ERM and GroupDRO run through the unmodified code path; with default arguments the patched code produces byte-identical training, validation and test logs to the original

on a synthetic check. The patch adds class-balanced sampling (labels only), the nine reference-margin groupings with the entropic score, the cooperative and robust policies, and optionally the annotated grouping. The reference model is our ERM run of the same seed after 60 epochs (JTT's identification length). Checkpoints are chosen from the official validation logs (best worst-group accuracy; warm-up epochs excluded for UAED), and test accuracy is read at that epoch. Rerunning ERM for one seed on a second node reproduced the training and validation logs and the saved model bit for bit.

Table 7: Waterbirds, official GroupDRO code (strong-ℓ_2, ResNet-50): test accuracy (%) over n seeds. Only GroupDRO (grey) uses group labels.

	Validation-selected		Final (epoch 300)		
Method	Worst	Average	Worst	Average	n
ERM	29.9 ± 3.3	95.9 ± 0.1	0.7 ± 0.1	85.1 ± 0.6	5
ERM (class-balanced)	49.9 ± 1.7	95.8 ± 0.0	11.3 ± 0.7	86.0 ± 0.4	5
UAED, frozen	81.8 ± 1.7	97.1 ± 0.3	33.8 ± 13.6	86.6 ± 3.4	5
UAED, cooperative	80.0 ± 1.4	97.1 ± 0.1	31.2 ± 13.1	86.4 ± 4.3	5
UAED, robust	81.9 ± 1.7	97.1 ± 0.2	34.6 ± 13.4	86.6 ± 3.4	5
GroupDRO	86.1 ± 1.1	92.7 ± 2.3	56.3 ± 5.8	82.8 ± 3.6	5

Published results. For orientation we list results reported by other authors; they are not reruns and differ from ours in model selection and other details. Sagawa et al. (2020, Table 1, strong ℓ_2): ERM 21.3 worst-group / 95.7 average, GroupDRO 84.6 / 96.6. Liu et al. (2021a, Table 1), with per-method hyperparameters tuned on worst-group validation accuracy (their final JTT model uses the strong-ℓ_2 setting, their ERM learning rate 10^{-3} and weight decay 10^{-4}): ERM 72.6 / 97.3, CVaR DRO 75.9 / 96.0, LfF 78.0 / 91.2, JTT 86.7 / 93.3, GroupDRO 91.4 / 93.5. Creager et al. (2021, Table 3), 300 epochs with a one-epoch reference model: EIIL 78.7 / 96.9.

H Robust policy and further follow-ups

Design. The robust policy changes only the sign of the policy's task term (one forward pass, two gradients); the predictor's gradient for a given policy is unchanged. Before any run we checked that both policies converge to the closed-form tilts of Proposition 1 and Corollary 2 for fixed scores, that the default (cooperative) setting reproduces the unmodified code bit for bit in the official code, that the predictor's gradient is identical under both signs, and, for Colored MNIST, the policy gradient against finite differences. The robust policy uses every setting of the cooperative one without tuning. These experiments, their predictions and the decision rules were written down before they ran, after the cooperative results had been seen; all fits are reported. Rows marked grey add the annotated (label, background) grouping to the family (K=10) and therefore use training group labels.

Table 8: Waterbirds, main protocol, ResNet-50: test accuracy (%), mean ± SD over 5 seeds. Frozen and cooperative rows repeat Table 2.

	Validation-selected		Final	
Method	Worst	Average	Worst	Average
UAED, frozen	72.0 ± 1.3	97.0 ± 0.3	70.0 ± 1.4	97.4 ± 0.2
UAED, cooperative	71.5 ± 0.7	97.0 ± 0.4	69.7 ± 1.2	97.5 ± 0.2
UAED, robust	71.5 ± 1.8	97.0 ± 0.4	70.3 ± 1.3	97.5 ± 0.1
UAED + true grouping, frozen	73.4 ± 1.1	97.0 ± 0.3	71.3 ± 1.3	97.4 ± 0.2
UAED + true grouping, cooperative	73.0 ± 1.3	97.0 ± 0.4	71.3 ± 1.1	97.5 ± 0.1
UAED + true grouping, robust	74.6 ± 2.9	96.8 ± 0.5	71.9 ± 1.2	97.4 ± 0.1

Policy weights. At its largest departure from the reference, the cooperative policy places 0.08–0.08 of its mass on the annotated grouping and the robust policy 0.10–0.11 (ranges over seeds; reference $1/10$). Without the annotated grouping, the robust policy places 0.41 of its mass on the tightest groupings and 0.22 on the softest at its largest departure (reference $1/3$ each). All policies return to the reference by the final epoch.

Table 9: Colored MNIST follow-ups, $\lambda = 100$, 5 seeds. "At reference": learned policy initialized at the reference; for $\beta \leq 0.01$ it ends at $e \approx 0$ in two seeds and $e \approx 1$ in three, hence the spread at $e = 0.9$.

Setting	$e = 0.9$ (final)	Worst (final)	Worst (validation)	Mean e
Robust UAED-IRM, $\beta = 1$	72.5 ± 0.7	71.5 ± 0.6	71.2 ± 0.7	0.50
Robust UAED-ERM, $\beta = 1$	69.4 ± 5.2	60.9 ± 6.7	62.8 ± 2.7	0.50
Cooperative, at reference, $\beta = 0$	58.0 ± 44.0	0.0 ± 0.0	0.0 ± 0.0	0.60
Cooperative, at reference, $\beta = 0.01$	58.0 ± 44.0	0.0 ± 0.0	0.0 ± 0.0	0.60
Cooperative, at reference, $\beta = 0.1$	72.4 ± 0.6	71.7 ± 0.5	70.8 ± 0.9	0.49
Cooperative, at reference, $\beta = 1$	72.3 ± 0.6	71.7 ± 0.7	71.5 ± 0.5	0.50

I DIAGNOSIS OF THE VIT-B/16 FAILURE

Checks for implementation errors. For every ViT-B/16 run we verified that the reference snapshot is the ERM model of the same seed after epoch 2, that the reference margins, the soft memberships and the training minibatches index the training set in the same order, that the reference margins are computed with the same preprocessing as training, and that the configurations of the ViT-B/16 and ResNet-50 partition runs differ only in the batch size. No run produced non-finite losses or policy weights. The two warm-up epochs of the partition methods are identical to class-balanced ERM, as intended.

Where the failure starts. Validation worst-group accuracy of both UAED variants peaks at the last warm-up epoch in every seed and falls as soon as the partition score is switched on; the training loss rises by a factor of 6.6 in that epoch, whereas the ratio of the losses of the first post-warm-up and the last warm-up epoch is below one for every other method and backbone. Validation accuracy falls in three of the four groups, most for waterbirds on land, while the largest group (landbirds on land) keeps it. JTT, which upweights the training examples misclassified by the same reference model, does not improve on ERM with ViT-B/16 either (Table 2), which suggests that this reference model separates the minority examples less sharply than the ResNet-50 reference. The learned ViT-B/16 policy departs further from the reference than with ResNet-50 (KL 0.45 at epoch 8, with mass 0.77 on the softest groupings) and returns to it as the training loss vanishes (KL 0.007 at epoch 30); since the frozen policy fails in the same way, this movement is not the cause of the failure.

Diagnostic runs. These runs were designed after the failure had been observed and are exploratory; none of them replaces a result in the main tables. The larger-batch, softer-score and combined arms were declared before any diagnostic result; the lower-learning-rate arm, the learned-policy arm and the matched control (class-balanced ERM at the lower learning rate) were added after the first diagnostic results, each declared before it ran. Each arm reuses the ViT-B/16 frozen-policy configuration of the main protocol and changes only the listed fields (verified programmatically against the stored configurations), reuses the ERM reference snapshot of each seed without retraining it, and runs the same five seeds on the same GPU type. Table 10 reports every arm.

Table 10: Exploratory ViT-B/16 diagnostics (frozen policy unless stated, 5 seeds): worst-group test accuracy (%). "Loss ratio": training loss in the first post-warm-up epoch over that in the last warm-up epoch.

Arm	Change from main protocol	Loss ratio	Validation-selected	Final
Main protocol	– (batch 64, $\tau = 0.1$, lr 10^{-3})	6.6	26.3 ± 2.0	9.1 ± 1.7
Larger batch	batch 128	3.2	39.6 ± 4.6	22.6 ± 5.8
Softer score	$\tau = 1$	4.0	37.2 ± 6.8	24.3 ± 7.4
Both	batch 128, $\tau = 1$	1.1	54.2 ± 12.8	49.7 ± 13.3
Both, learned policy	batch 128, $\tau = 1$	1.1	60.0 ± 6.4	48.6 ± 10.8
Lower learning rate	lr 10^{-4}	0.8	64.2 ± 1.3	64.2 ± 1.8
Class-balanced ERM, main protocol		0.5	64.5 ± 1.7	61.3 ± 2.1
Class-balanced ERM, lr 10^{-4} (matched control)		0.7	64.4 ± 1.4	64.2 ± 1.9

Recovery follows the loss ratio: the larger the increase of the training loss when the partition score is switched on, the lower the worst-group accuracy. A larger batch, which doubles the expected number of low-margin examples per minibatch, and a softer score each remove part of the increase; together

they remove most of it but leave a large variance across seeds. Lowering the learning rate alone removes it entirely, and the validation-selected and final checkpoints then agree. Our interpretation is that the near-maximum over small reference-defined groups makes the effective training signal much larger and noisier than the plain loss at the end of warm-up, which a learning rate tuned for the plain loss cannot absorb. This explanation is supported by the diagnostic runs but was not tested by a separate, predeclared experiment.