Algorithm 1 Pseudocode for environment inference (EI) with the invariance principle (realized via relaxed IRMv1 penalty) as the EI objective.

Input: Reference model Φ, dataset $\mathcal{D} = \{x_i, y_i\}$, loss ℓ, duration N_{steps}
Output: Worst case data splits $\mathcal{D}_1$, $\mathcal{D}_2$ for use with an invariant learner.

def $\tilde{R}^e(\Phi, \mathbf{q})$:
  return $\frac{1}{\sum_{i'} \mathbf{q}_{i'}(e)} \sum_i \mathbf{q}_i(e)\ell(\Phi(x_i), y_i)$ {Equation 4}

Randomly init. $\mathbf{q} \in [0,1]^N$ environment posterior ($\mathbf{q}_i(e) := q(e|x_i, y_i)$)
Randomly init. $\mathbf{q} \in [0,1]^N$ environment posterior ($\mathbf{q}_i(e) := q(e|x_i, y_i)$)
for $n \in 1 \dots N_{steps}$ **do**
  $SoftVari = \sum_{e \in \{1,2\}} ||\nabla_{\bar{w}} \tilde{R}^e(\bar{w} \circ \Phi, \mathbf{q})||$ {Aggregate reference model variances across soft envs}
  $Loss = -1 \cdot SoftVari$ {Maximize the EI objective by minimizing this loss}
  $\mathbf{q} \leftarrow OptimUpdate(\mathbf{q}, \nabla_{\mathbf{q}} Loss)$
end for
$\hat{\mathbf{q}} \sim Bernoulli(\mathbf{q})$ {sample splits}
$\mathcal{D}_1 \leftarrow \{x_i, y_i | \hat{\mathbf{q}}_i = 1\}$, $\mathcal{D}_2 \leftarrow \{x_i, y_i | \hat{\mathbf{q}}_i = 0\}$ {split data}
return $\mathcal{D}_1, \mathcal{D}_2$

A. Environment Inference Psuedocode

Algorithm 1 provides pseudocode for the environment inference procedure used in our experiments.

B. Proofs

B.1. Proof of Proposition 1

Consider a dataset with some feature(s) z which are spurious, and other(s) v which are valuable/causal w.r.t. the label y. This includes data generated by models where $v \to y \to z$, such that $P(y|v, z) = P(y|v)$. Assume further that the observations x are functions of both spurious and valuable features: $x := f(v, z)$. The aim of invariant learning is to form a classifier that predicts y from x that focuses solely on the causal features, i.e., is invariant to z and focuses solely on v.

Consider a classifier that produces a score $S(x)$ for example x. In the binary classification setting S is analogous to the model Φ, while the score $S(x)$ is analogous to the representation $\Phi(x)$. To quantify the degree to which the constraint in the Invariant Principle (EIC) holds, we introduce a measure called the *group sufficiency gap*[12]:

$$\Delta(S, e) = \mathbb{E}[\mathbb{E}[(y|S(x), e_1)] - \mathbb{E}[(y|S(x), e_2)]]$$

Now consider the notion of an environment: some setting in which the $x \to y$ relationship varies (based on spurious features). Assume a single binary spurious feature z. We restate Proposition 1 as follows:

Claim: If environments are defined based on the agreement of the spurious feature z and the label y, then a classifier that predicts based on z alone maximizes the group-sufficiency gap (and vice versa – if a classifier predicts y directly by predicting z, then defining two environments based on agreement of label and spurious feature—$e_1 = \{v, z, y | \mathbb{1}(y = z)\}$ and $e_2 = \{v, z, y | \mathbb{1}(y \neq z)\}$—maximizes the gap).

We can show this by first noting that if the environment is based on spurious feature-label agreement, then with $e \in \{0, 1\}$ we have $e = \mathbb{1}(y = z)$. If the classifier predicts z, i.e. $S(x) = z$, then we have

$$\Delta(S, e) = \mathbb{E}[\mathbb{E}[y|z(x), \mathbb{1}(y = z)] - \mathbb{E}[y|z(x), \mathbb{1}(y \neq z)]]$$

For each instance of x either $z = 0$ or $z = 1$. Now we note that when $z = 1$ we have $\mathbb{E}(y|z, \mathbb{1}(y = z)) = 1$ and $\mathbb{E}(y|z, \mathbb{1}(y \neq z)) = 0$, while when $z = 0$ $E(y|z, \mathbb{1}(y = z)]) = 0$ and $\mathbb{E}[y|z, \mathbb{1}(y \neq z)] = 1$. Therefore for each example $|E(y|z(x), \mathbb{1}(y = z)) - E(y|z(x), \mathbb{1}(y \neq z)| = 1$, contributing to an overall $\Delta(S, e) = 1$, which is the maximum value for the sufficiency gap.

[12] This was previously used in a fairness setting by Liu et al. (2019) to measure differing calibration curves across groups.

B.2. Heuristic for soft environment assignment based on binning violates the invariance principle

Here we analyze the heuristic discussed in Section 3.3. We want to show that finding environment assignments in this way both maximizes the violation of the softened version of the regularizer (Equation 3), and also also maximally violates the invariance principle (EIC).

Because the invariance principle $\mathbb{E}[Y|\Phi(X), e] = \mathbb{E}[Y|\Phi(X), e']\forall e, e'$ is difficult to quantify for continuous $\Phi(X)$, we consider a binned version of the representation, with b denoting the discrete index of the bin in representation space. Let $q_i \in [0, 1]$ denote the soft assignment of example i to environment 1, and $1 - q_i$ denote its converse, the assignment of example i to environment 2. Denote by $y_i \in \{0, 1\}$ the binary target for example i, and $\hat{y} \in [0, 1]$ as the model prediction on this example. Assume that ℓ represents a cross entropy or squared error loss so that $\nabla_w \ell(\hat{y}, y) = (\hat{y} - y)\Phi(x)$.

Consider the IRMv1 regularizer with soft assignment, expressed as

$$\begin{aligned}
D(q) &= \sum_e ||\nabla_{w|w=1.0} \frac{1}{N_e} \sum_i q_i(e)\ell(w \circ \Phi(x_i), y_i)||^2 \\
&= \sum_e ||\frac{1}{N_e} \sum_i q_i(e)(\hat{y}_i - y_i)\Phi(x_i)||^2 \\
&= ||\frac{1}{\sum_i' q_i'} \sum_i q_i(\hat{y}_i - y_i)\Phi(x_i)||^2 + ||\frac{1}{\sum_i' 1 - q_i'} \sum_i (1 - q_i)(\hat{y}_i - y_i)\Phi(x_i)||^2 \\
&= ||\frac{\sum_i q_i \hat{y}_i \Phi(x_i)}{\sum_{i'} q_{i'}} - \frac{\sum_i q_i y_i \Phi(x_i)}{\sum_{i'} q_{i'}}||^2 + ||\frac{\sum_i (1 - q_i)\hat{y}_i \Phi(x_i)}{\sum_{i'} 1 - q_{i'}} - \frac{\sum_i (1 - q_i) y_i \Phi(x_i)}{\sum_{i'} 1 - q_{i'}}||^2. \qquad (5)
\end{aligned}$$

Now consider that the space of $\Phi(X)$ is discretized into disjoint bins b over its support, using $z_{i,b} \in \{0, 1\}$ to indicate whether example i falls into bin b according to its mapping $\Phi(x_i)$. Thus we have

$$\begin{aligned}
D(q) = \sum_b (&||\frac{\sum_i z_{i,b} q_i \hat{y}_i \Phi(x_i)}{\sum_{i'} z_{i,b} q_{i'}} - \frac{\sum_i z_{i,b} q_i y_i \Phi(x_i)}{\sum_{i'} z_{i,b} q_{i'}}||^2 \\
&+ ||\frac{\sum_i z_{i,b}(1 - q_i)\hat{y}_i \Phi(x_i)}{\sum_{i'} z_{i,b}(1 - q_{i'})} - \frac{\sum_i z_{i,b}(1 - q_i) y_i \Phi(x_i)}{\sum_{i'} z_{i,b}(1 - q_{i'})}||^2) \qquad (6)
\end{aligned}$$

The important point is that within a bin, all examples have roughly the same $\Phi(x_i)$ value, and the same value for $\hat{y}_i$ as well. So denoting $K_b^{(1)} := \frac{\sum_i z_{i,b} q_i \hat{y}_i \Phi(x_i)}{\sum_{i'} z_{i,b} q_{i'}}$ and $K_b^{(2)} := \frac{\sum_i z_{i,b}(1-q_i)\hat{y}_i \Phi(x_i)}{\sum_{i'} z_{i,b}(1-q_{i'})}$ as the relevant constant within-bin summations, we have the following objective to be maximized by EIIL:

$$D(q) = \sum_b (||K_b^{(1)} - \frac{\sum_i z_{i,b} q_i y_i \Phi(x_i)}{\sum_{i'} z_{i,b} q_{i'}}||^2 + ||K_b^{(2)} - \frac{\sum_i z_{i,b}(1 - q_i) y_i \Phi(x_i)}{\sum_{i'} z_{i,b}(1 - q_{i'})}||^2.$$

One way to maximize this is to assign all $y_i = 1$ values to environment 1 ($q_i = 1$ for these examples) and all $y_i = 0$ to the other environment ($q_i = 0$). We can show this is maximized by considering all of the examples except the i-th one have been assigned this way, and then that the loss is maximized by assigning the i-th example according to this rule.

Now we want to show that the same assignment maximally violates the invariance principle (showing that this soft EIIL solution provides maximal non-invariance). Intuitively within each bin the difference between $\mathbb{E}[y|e = 1]$ and $\mathbb{E}[y|e = 2]$ is maximized (within the bin) if one of these expected label distributions is 1 while the other is 0. This can be achieved by assigning all the $y_i = 1$ values to the first environment and the $y_i = 0$ values to the second.

Thus a global optimum for the relaxed version of EIIL (using the IRMv1 regularizer) also maximally violates the invariance principle.

B.3. Given CMNIST environments are suboptimal w.r.t. sufficiency gap

The regularizer from IRMv1 encourages a representation for which sufficiency gap is minimized between the available environments. Therefore when faced with a new task it is natural to measure the natural sufficiency gap between these

environments, mediated through a naive or baseline method. Here we show that for CMNIST, when considering a naive color-based classifier as the reference model, the given environment splits are actually *suboptimal* w.r.t. sufficiency gap, which motivates the inference of environments via EIIL that have a higher sufficiency gap for the reference model.

We begin by computing $\Delta(S, e)$, the sufficiency gap for color-based classifier g over the given train environments $\{e_1, e_2\}$. We introduce an auxiliary color variable z, which is not observed but can be sampled from via the color based classifier g:

$$p(y|g(x) = x', e) = \mathbb{E}_{p(z|x')}\left[p(y|z, e, x').\right]$$

Denote by GREEN and RED the set of green and red images, respectively. I.e. we have $z \in G$ iff $z = 1$ and $x \in$ GREEN iff $z(x) = 1$. The the sufficiency gap is expressed as

$$\begin{aligned}
\Delta(S, e) &= \mathbb{E}_{p(x,e)}\left[\left|\mathbb{E}_{p(y|x,e_1)}[y|g(x), e_1] - \mathbb{E}_{p(y|x,e_2)}[y|g(x), e_2]\right|\right] \\
&= \mathbb{E}_{p(z,e)}\left[\left|\mathbb{E}_{p(y|z,e_1)}[y|z, e_1] - \mathbb{E}_{p(y|z,e_2)}[y|z, e_2]\right|\right] \\
&= \frac{1}{2}\sum_{z\in\{\text{GREEN},\text{RED}\}}\left[\left|\mathbb{E}_{p(y|z,e_1)}[y|z, e_1] - \mathbb{E}_{p(y|z,e_2)}[y|z, e_2]\right|\right] \\
&= \frac{1}{2}(|\mathbb{E}[y|z = \text{GREEN}, e_1] - \mathbb{E}[y|z = \text{GREEN}, e_2]| + |\mathbb{E}[y|z = \text{RED}, e_1] - \mathbb{E}[y|z = \text{RED}, e_2]|) \\
&= \frac{1}{2}(|0.1 - 0.2| - |0.9 - 0.8|) = \frac{1}{10}.
\end{aligned}$$

The regularizer in IRMv1 is trying to reduce the sufficiency gap, so in some sense we can think about this gap as a learning signal for the IRM learner. A natural question would be whether a different set of environment partition $\{e\}$ can be found such that this learning signal is stronger, i.e. the sufficiency gap is increased. We find the answer is yes. Consider an environment distribution $q(e|x, y, z)$ that assigns each data point to one of two environments. Any assignment suffices so far as its marginal matches the observed data: $\int_z \int_e q(x, y, z, e) = p^{\text{obs}}(x, y)$.

We can now express the sufficiency gap (given a color-based classifier g) as a function of the environment assignment q:

$$\begin{aligned}
\Delta(S, e \sim q) &= \mathbb{E}_{q(x,e)}[|\mathbb{E}_{q(y|x,e,x)}[y|g(x), e_1] - \mathbb{E}_{q(y|x,e,x)}[y|g(x), e_2]|] \\
&= \mathbb{E}_{q(x,e)}[|\mathbb{E}_{q(y|z,e,x)p(z|x)}[y|z, e_1] - \mathbb{E}_{q(y|z,e,x)p(z|x)}[y|z, e_2]|]
\end{aligned}$$

Where we use the same change of variables trick as above to replace $g(x)$ with samples from $p(z|x)$ (note that this is the color factor from the generative process p according with our assumption that g matches this distribution).

We want to show that there exists a q yielding a higher sufficiency gap than the given environments. Consider q that yields the conditional label distribution

$$q(y|x, e, z) := q(y|e, z) = \begin{cases} \mathbb{1}(y = z) \text{ if } e = e_1, \\ \mathbb{1}(y \neq z) \text{ if } e = e_2. \end{cases}$$

This can be realized by an encoder/auditor $q(e|x, y, z)$ that ignores image features in x and partitions the example based on whether or not the label y and color z agree. We also note that z is deterministically the color of the image in the generative process: $p(z|x) = \mathbb{1}(x = \text{RED})$

Now we can compute the sufficiency gap:

$$
\begin{aligned}
\Delta(S, e \sim q) &= \mathbb{E}_{q(x,e)}[|\mathbb{E}_{q(y|z,e,x)p(z|x)}[y|z,e_1] - \mathbb{E}_{q(y|z,e,x)p(z|x)}[y|z,e_2]|] \\
&= \frac{1}{2}\mathbb{E}_{x\in\text{RED}}|\mathbb{E}_{q(y|z,e,x)p(z|x)}[y|z,e_1] - \mathbb{E}_{q(y|z,e,x)p(z|x)}[y|z,e_2]| \\
&\quad + \frac{1}{2}\mathbb{E}_{x\in\text{GREEN}}|\mathbb{E}_{q(y|z,e,x)p(z|x)}[y|z,e_1] - \mathbb{E}_{q(y|z,e,x)p(z|x)}[y|z,e_2]| \\
&= \frac{1}{2}\mathbb{E}_{x\in\text{RED}}(|\sum_y\sum_z(y * \mathbb{1}(y=z) * \mathbb{1}(g(x)=z)) - \sum_y\sum_z(y * \mathbb{1}(y\neq z) * \mathbb{1}(g(x)=z))|) \\
&\quad + \mathbb{E}_{x\in\text{GREEN}}\frac{1}{2}(|\sum_y\sum_z(y * \mathbb{1}(y=z) * \mathbb{1}(g(x)=z)) - \sum_y\sum_z(y * \mathbb{1}(y\neq z) * \mathbb{1}(g(x)=z))|) \\
&= \frac{1}{2}\mathbb{E}_{x\in\text{RED}}(|\sum_y(y * \mathbb{1}(y=1) * \mathbb{1}(x\in\text{RED})) - \sum_y(y * \mathbb{1}(y\neq 1) * \mathbb{1}(x\in\text{RED}))|) \\
&\quad + \mathbb{E}_{x\in\text{GREEN}}\frac{1}{2}(|\sum_y\sum_z(y * \mathbb{1}(y=0) * \mathbb{1}(x\in\text{GREEN})) - \sum_y\sum_z(y * \mathbb{1}(y\neq 0) * \mathbb{1}(x\in\text{GREEN}))|) \\
&= \frac{1}{2}\mathbb{E}_{x\in\text{RED}}[|1-0|] + \mathbb{E}_{x\in\text{GREEN}}[\frac{1}{2}|0-1|] = \frac{1}{2} + \frac{1}{2} = 1.
\end{aligned}
$$

Note that 1 is the maximal sufficiency gap, meaning that the described environment partition maximizes the sufficiency gap w.r.t. the color-based classifier g.

C. Connections Between Invariant Learning and Algorithmic Fairness

Here we lay out some connections to algorithmic fairness, where demographic information, which is often considered "sensitive", is used to inform learning. Table 1 from the main paper provides a high-level comparison of the objectives and assumptions of several relevant methods. Loosely speaking, recent approaches from both areas share the goal of matching some chosen statistic across a conditioning variable e, representing sensitive group membership in algorithmic fairness or an environment/domain indicator in domain generalization. The statistic in question informs the *learning objective* for the resulting model, and is motivated differently in each case. In domain generalization, learning is informed by the properties of the test distribution where good generalization should be achieved. In algorithmic fairness the choice of statistic is motivated by a context-specific *fairness notion*, that likewise encourages a particular solution that achieves "fair" outcomes (Chouldechova & Roth, 2020).

Early approaches to learning fair representations (Zemel et al., 2013; Edwards & Storkey, 2016; Louizos et al., 2016; Zhang et al., 2018; Madras et al., 2018) leveraged statistical independence regularizers from domain adaptation[13] (Ben-David et al., 2010; Ganin et al., 2016; Tzeng et al., 2017; Long et al., 2018), noting that marginal or conditional independence from domain to prediction relates to the fairness notions of demographic parity $\hat{y} \perp e$ (Dwork et al., 2012) and equal opportunity $\hat{y} \perp e|y$ (Hardt et al., 2016).

Recall that (EIC) involves an environment-specific conditional label expectation given a data representation $\mathbb{E}[y|\Phi(x) = h, e]$. Objects of this type have been closely studied in the fair machine learning literature, where e now denotes a "sensitive" attribute indicating membership in a protected demographic group (age, race, gender, etc.), and the vector representation $\Phi(x)$ is typically replaced by a scalar score[14] $S(x) \in \mathbb{R}$. Noting that $\sigma(S(x))$ represents the probability of the model prediction, $\mathbb{E}[y|S(x), e]$ can now be interpreted as a *calibration curve* that must be regulated according to some fairness constraint. Chouldechova (2017) showed that equalizing this calibration curve across groups is often incompatible with a common fairness constraint, demographic parity, while Liu et al. (2019) studied "group sufficiency" of classifiers with strongly convex losses, concluding that ERM naturally finds group sufficient solutions without fairness constraints.

Because Liu et al. (2019) consider convex losses, their theoretical results do not hold for neural network representations. However, by noting the link between group sufficiency and the constraint from (EIC), we observe that the IRMv1 regularizer (applicable to neural nets) in fact minimizes the group sufficiency gap in the case of a scalar representation $\Phi(x) \subseteq \mathbb{R}$, and when e indicates sensitive group membership. It is worth noting that Arjovsky et al. (2019) briefly discuss using groups as

[13] Whereas domain generalization requires model predictions on entirely novel domains at test time, domain adaptation assumes a set of target domain examples are available at test time to guide model adaptation.

[14] For binary classification, score-based and representation-based approaches are closely related since scores are commonly implemented as (or can be interpreted as) as the linear mapping of a data representation: $S(x) = w \circ \Phi(x)$.

environments, but without specifying a particular fairness criterion. We leave an empirical study of these methods for future work.

Our approach in searching for worst-case data partitions in EIIL was inspired by recent work on fair prediction without sensitive labels (Kearns et al., 2018; Hébert-Johnson et al., 2018; Hashimoto et al., 2018; Lahoti et al., 2020). Reliance on sensitive demographic information is cumbersome since it often cannot be collected without legal or ethical repercussions. Hébert-Johnson et al. (2018) discussed the problem of mitigating subgroup unfairness when group labels are unknown, and proposed *Multicalibration* as a way of ensuring a classifier's calibration curve is invariant to efficiently computable environment splits. Since the proposed algorithm requires brute force enumeration over all possible environments/groups, Kim et al. (2019) suggested a more practical algorithm by relaxing the calibration constraint to an accuracy constraint, yielding a *Multiaccurate* classifier.[15] The goal here is to boost the predictions of a pre-trained classifier through multiple rounds of auditing (searching for worst-case subgroups using an auxiliary model) rather than learning an invariant representation.

A related line of work also leverages inferred subgroup information to improve worst-case model performance using the framework of DRO. Hashimoto et al. (2018) applied DRO to encourage long-term fairness in a dynamical setting where the average loss for a subpopulation influences their propensity to continue engaging with the model. Lahoti et al. (2020) proposed Adversarially Reweighted Learning (ARL), which extends DRO using an auxiliary model to compute the importance weights γ_i mentioned above. Amortizing this computation mitigates the tendency of DRO to overfit its reweighting strategy to noisy outliers.

Limitations of generalization-first fairness One exciting direction for future work is to apply methods developed in the domain generalization literature to tasks where distribution shift is related to some societal harm that should be mitigated. However, researchers should be wary of blind "solutionism", which can be ineffectual or harmful when the societal context surrounding the machine learning system is ignored (Selbst et al., 2019). Moreover, many aspects of algorithmic discrimination are not simply a matter of achieving few errors on unseen distributions. Unfairness due to task definition or dataset collection, as discussed in the study of target variable selection by Obermeyer et al. (2019), may not be reversible by novel algorithmic developments.

D. Dataset details

CMNIST This dataset was provided by Arjovsky et al. (2019)[16]. The two training environments comprise $25,000$ images each, with $Corr(color, label) = 0.8$ for the firs training environment and $Corr(color, label) = 0.8$ for the second. A held-out test set with $Corr(color, label) = 0.1$ is used for evaluation. Label noise is applied by flipping the binary target y with probability $\theta_y = 0.25$, with color correlation applied w.r.t. the noisy label. Given that only two color channels are used, we follow Arjovsky et al. (2019) in downsampling the digit images to 14×14 pixels and 2 channels.

Waterbirds We follow the procedure outlined by Sagawa et al. (2020) to reproduce the Waterbirds dataset. As noted by the authors, due to random seed differences our version of the dataset may differ slightly from the one originally used by the paper. The train/validation/test splits are of size $4,795/1,200/5,794$. As noted in the Appendix of (Sagawa et al., 2020), the validation and test distributions represent upweight the minority groups so that the number of examples coming from each habitat is equal (although there are still marginally more landbirds than waterbirds). For example on train set the subgroup sizes are $3,498/184/56/1,057$ while on the test set the sizes are $467/466/133/133$.

CivilComments-WILDS We use the train/validation/test splits from Koh et al. (2021); we refer the interested reader the Appendix of their paper for a detailed description of this version of the dataset, including how it differs from the original dataset (Borkan et al., 2019).

Constructing the Adult-Confounded dataset To create our semi-synthetic dataset, called Adult-Confounded, we start by observing that the conditional distribution over labels varies across the subgroups, and in some cases subgroup membership is very predictive of the target label. We construct a test set (a.k.a. the audit set) where this relationship between subgroups and target label is reversed.

The four sensitive subgroups are defined following the procedure of Lahoti et al. (2020), with sex (recorded as binary:

[15] Kearns et al. (2018) also proposed a boosting procedure to equalize subgroup errors without sensitive attributes.

[16] https://github.com/facebookresearch/InvariantRiskMinimization

Male/Female) and binarized race (Black/non-Black) attributes compose to make four possible subgroups: Non-Black Males (SG1), Non-Black Females (S2), Black Males (SG3), and Black Females (SG4).

We start with the observation that each subgroup has a different correlation strength with the target label, and in some cases subgroup membership alone can be used to achieve relatively low error rates in prediction. As these correlations should be considered "spurious" to mitigate unequal treatment across groups, we create a semi-synthetic variant of the UCI Adult dataset, which we call Adult-Confounded, where these spurious correlations are exaggerated. Table 6 shows various conditional label distributions for the original dataset and our proposed variant. The test set for Adult-Confounded reverses the correlation strengths, which can be thought of as a worst-case audit to ensure the model is not relying on subgroup membership alone in its predictions. We generate samples for Adult-Confounded using importance sampling, keeping the original train/test splits from UCI Adult as well as the subgroup sizes, but sampling individual examples under/over-sampled according to importance weights $\frac{p^{Adult-Confounded}}{p^{UCIAdult}}$.

Subgroup (SG)	$p(y=1\|SG)$			
	UCIAdult		Adult-Confounded	
	Train	Test	Train	Test
1	0.31	0.30	0.94	0.06
2	0.11	0.12	0.06	0.94
3	0.19	0.16	0.94	0.06
4	0.06	0.04	0.06	0.94

Table 6. Adult-Confounded is a variant of the UCI Adult dataset that emphasizes test-time distribution shift.

E. Experimental details

Model selection Krueger et al. (2021) discussed the pitfalls of achieving good test performance on CMNIST by using test data to tune hyperparameters. Because our primary interest is in the properties of the inferred environment rather than the final test performance, we sidestep this issue in the Synthetic Regression and CMNIST experiments by using the default parameters of IRM without further tuning. However for Adult-Confounded a specific strategy for model selection is needed.

We refer the interested reader to Gulrajani & Lopez-Paz (2021) for an extensive discussion of possible model selection strategies. They also provide a large empirical study showing that ERM is difficult baseline to beat when all methods are put on equal footing w.r.t. model selection.

In our case, we use the most relaxed model selection method proposed by Gulrajani & Lopez-Paz (2021), which amounts to allowing each method a 20 test evaluations using hyperparameter chosen at random from a reasonable range, with the best hyperparameter setting selected for each method. While none of the methods is given an unfair advantage in the search over hyperparameters, the basic model selection premise does not translate to real-world applications, since information about the test-time distribution is required to select hyperparameters. Thus these results can be understood as being overly optimistic for each method, although the relative ordering between the methods can still be compared.

Training times Because EIIL requires a pre-trained reference model and optimization of the EI objective, overall training time is longer than standard invariant learning. It depends primarily on the number of steps used to train the reference model and number of steps used in EI optimization. The extra training time incurred is manageable and varies from dataset to dataset.

In CMNIST, we train the ERM reference model for $1,000$ steps, which is the same duration as the downstream invariant learner that eventually uses the inferred environments. In this setting the $10,000$ steps required to optimize the EI objective is actually more than used for representation learning. The overall EIIL train time is 6.6 minutes to run 10 restarts on a NVIDIA Tesla P100, compared with 2.18 minutes for ERM and 2.20 minutes for IRM.

However, as the problem size scales, the relative overhead cost of EIIL becomes progressively discounted. On Waterbirds, training GroupDRO takes 4.716 hours on a NVIDIA Tesla P100. Our reference model trains for 1 epoch, so taking this into account along with the $20,000$ steps of EI optimization, EIIL runs at 4.737 hours. This is a relative increase of 0.4%.

Batch environment inference As mentioned in the main paper, we aggregate logits for the entire training set and optimize the EI objective using the entire training batches. This can be done by cycling through the train set once in minibatches, computing logits per minibatch, and aggregating the logits only (discarding network activations) prior to EI. We leave minibatched environment inference and amortization of the soft environment assignments to future work.

Experimental infrastructure Our experiments were run on a cluster of NVIDIA Tesla P100 machines.

CMNIST IRM is trained on the two training environments and tested on a holdout environment constructed from $10,000$ test images in the same way as the training environments, where colour is predictive of the noisy label 10% of the time. So using color as a feature to predict the label will lead to an accuracy of roughly 10% on the test environment, while it yields 80% and 90% accuracy respectively on the training environments.

To evaluate EIIL we remove the environment identifier from the training set and thus have one training set comprised of $50,000$ images from both original training environments. We then train an MLP with binary cross-entropy loss on the training environments, freeze its weights and use the obtained model to learn environment splits that maximally violate the IRM penalty. When optimizing the inner loop of EIIL, we use Adam with learning rate 0.001 for $10,000$ steps with full data batches used to computed gradients.

The obtained environment partitions are then used to train a new model from scratch with IRM. Following Arjovsky et al. (2019), we allow the representation to train for several hundred annealing steps before applying the IRMv1 penalty.

We used the default architecture—an MLP with two hidden layers of 390 neurons—and hyperparameter values[17]—-learning rate, weight decay, and penalty strength—from (Arjovsky et al., 2019). We do not use minibatches as the entire dataset fits into memory.

Waterbirds Following Sagawa et al. (2020), we use the default `torchvision` ResNet50 models, using the pre-trained weights as the initial model parameters, and train without any data augmentation using the For GroupDRO and ERM, we use hyperparameters reported by the authors[18], and note that the authors make use of the the validation set (whose distribution contains less group imbalance than the training data), to select hyperparameters in their experiments (all methods benefit equally from this strategy). We train for 300 epochs without any early stopping (to avoid any further influence from the validation data). For EIIL, we optimize the EI objective of EIIL with learning rate 0.01 for $20,000$ steps using the Adam optimizer, and use GroupDRO (using the same hyperparameters as the GroupDRO baseline) as the invariant learner. An ERM model trained for 1 epoch was used as the reference model. We also tried using reference modeled trained for longer, but found that EIIL did not perform as well in this case. We hypothesize that this is because the reference ERM model focuses on background features early in training, leading to stark performance discrepancies across subgroups, which in turn provides a strong learning signal for EIIL to infer effective environments. While subgroup disparities are present for more well-trained models, the learning signal in the EI phase will weaken.

Adult-Confounded Following Lahoti et al. (2020), we use a two-hidden-layer MLP architecture for all methods, with 64 and 32 hidden units respectively, and a linear adversary for ARL. We use IRM as the invariant learner in the final stage of EIIL. We optimize all methods using Adagrad; learning rates, number of steps, and batch sizes chosen by the model selection strategy described above (with 20 test evaluations per method), as are penalty weights for IRMv1 regularizer and standard weight decay. For the inner loop of EIIL (inferring the environments), we use the same settings as in CMNIST. We find that the performance of EIIL is somewhat sensitive to the number of steps taken with the IRMv1 penalty applied. To limit the number of test queries needed during model selection, we use an early stopping heuristic by enforcing the IRMv1 penalty only during the final 500 steps of training, with the previous steps serving as annealing period to learn a baseline representation to be regularized.

CivilComments-WILDS Following (Koh et al., 2021), we finetune DistilBERT embeddings (Sanh et al., 2019) using the default HuggingFace implementation and default weights (Wolf et al., 2019). EIIL uses an ERM reference classifier and its inferred environments are fed to a GroupDRO invariant learner. During prototyping the EI step, we noticed that the binning heuristic described in Section 3.3 consistently split the training examples into environments according to the error cases of

[17] https://github.com/facebookresearch/InvariantRiskMinimization/blob/master/code/colored_mnist/reproduce_paper_results.sh

[18] https://worksheets.codalab.org/worksheets/0x621811fe446b49bb818293bae2ef88c0

the reference classifier. Because error splitting is even simpler to implement than confidence binning, we used this heuristic for the EI step; we believe this is a promising approach for scaling EI to large datasets, and note its equivalence to the first stage of the method independently proposed by Liu et al. (2021), which is published concurrently to ours. We experimented with gradient-based EI on this dataset, but did not find any improvement over the (faster) heuristic EI.

On this dataset, we treat reference model selection as part of the overall model selection process, meaning that the hyperparameters of the ERM reference model are treated as a subset of the overall hyperparameters tuned during model selection. Specifically we used a grid search to tune the reference model learning rate (1e-5, 1e-4), optimizer type (Adam, SGD) and scheduler (linear, plateau), and gradient norm clamping (off, clamped at 1.0), as well as the invariant learner (GroupDRO) learning rate (1e-5, 1e-4). Moreover, we allow all methods to evaluate worst-group validation accuracy to tune these hyperparameters; such validation data will not be available in most settings, so this result can be seen as an optimistic view of the performance of all methods, including EIIL. We train all methods (including the reference model) for 5 epochs, with the best epoch chosen according to validation performance. Interestingly, the reference model chosen in this way was a constant classifier, so the overall EIIL solution is equivalent to GroupDRO using the class label as the environment label.

The oracle GroupDRO method trains on two environments, with one containing comments where *any* of the 8 sensitive groups was mentioned, and other environment containing the remaining comments. We experimented with allowing the oracle method access to more fine-grained environment labels by evaluating all 2^8 combinations of binary group labels, but did not find any significant performance boost (consistent with observations from Koh et al. (2021)).

F. Additional Empirical Results

F.1. Synthetic Data

	Causal MSE	Noncausal MSE
ERM	0.827 ± 0.185	0.824 ± 0.013
ICP	1.000 ± 0.000	0.756 ± 0.378
IRM	0.666 ± 0.073	0.644 ± 0.061
EIIL	$\mathbf{0.148 \pm 0.185}$	$\mathbf{0.145 \pm 0.177}$

Table 7. IRM using EIIL-discovered environments (e_{EIIL}) outperforms IRM in a synthetic regression setting without the need for hand-crafted environments (e_{HC}). This is because the reference representation $\tilde{\Phi} = \Phi_{\text{ERM}}$ uses the spurious feature for prediction. MSE + standard deviation across 5 runs reported.

We begin with a regression setting originally used as a toy dataset for evaluating IRM (Arjovsky et al., 2019). The features $\mathbf{x} \in \mathbb{R}^N$ comprise a "causal" feature $\mathbf{v} \in \mathbb{R}^{N/2}$ concatenated with a "non-causal" feature $\mathbf{z} \in \mathbb{R}^{N/2}$: $\mathbf{x} = [\mathbf{v}, \mathbf{z}]$. Noise varies across hand-crafted environments e:

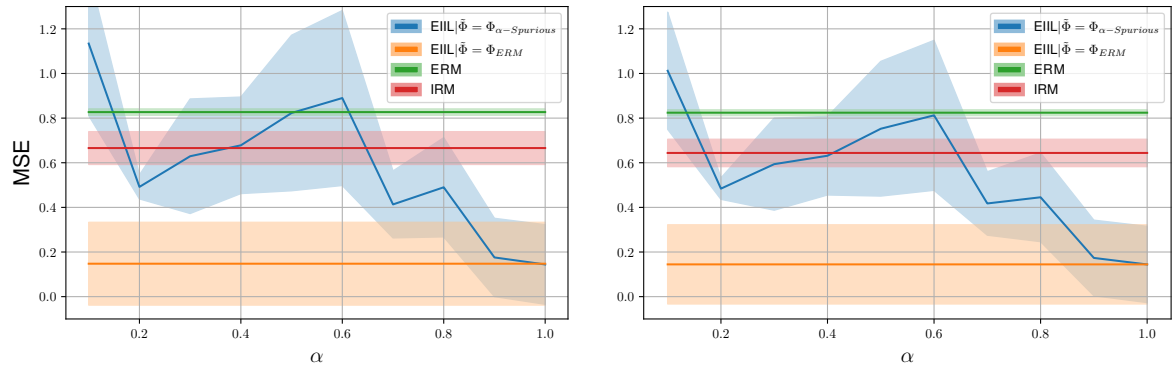
$$\begin{aligned} \mathbf{v} &= \epsilon_{\mathbf{v}} & \epsilon_{\mathbf{v}} &\sim \mathcal{N}(0, 25) \\ \mathbf{y} &= \mathbf{v} + \epsilon_{\mathbf{y}} & \epsilon_{\mathbf{y}} &\sim \mathcal{N}(0, e^2) \\ \mathbf{z} &= \mathbf{y} + \epsilon_{\mathbf{z}} & \epsilon_{\mathbf{z}} &\sim \mathcal{N}(0, 1). \end{aligned}$$

We evaluated the performance of the following methods:

- **ERM:** A naive regressor that does not make use of environment labels e, but instead optimizes the average loss on the aggregated environments;
- **IRM:** the method of Arjovsky et al. (2019) using hand-crafted environment labels;
- **ICP:** the method of Peters et al. (2016) using hand-crafted environment labels;
- **EIIL:** our proposed method (which does use hand-crafted environment labels) that infers useful environments based on the naive ERM, then applies IRM to the inferred environments.

The regression methods fit a scalar target $y = \mathbf{1}^T\mathbf{y}$ via a regression model $\hat{y} \approx \mathbf{w}^T\mathbf{x}$ to minimize $||y - \hat{y}||$ w.r.t. $\mathbf{w}$, plus an invariance penalty as needed. The optimal (causally correct) solution is $\mathbf{w}^* = [\mathbf{1}, \mathbf{0}]$ Given a solution $[\hat{\mathbf{w}}_v, \hat{\mathbf{w}}_z]$ from one of

the methods, we report the mean squared error for the causal and non-causal dimensions as $||\hat{\mathbf{w}}_v - \mathbf{1}||_2^2$ and $||\hat{\mathbf{w}}_z - \mathbf{0}||_2^2$ (Table 7). Because $\mathbf{v}$ is marginally noisier than $\mathbf{z}$, ERM focuses on the spurious $\mathbf{z}$. IRM using hand-crafted environments, denoted IRM, exploits variability in noise level in the non-causal feature (which depends on the variability of $\sigma_{\mathbf{y}}$) to achieve lower error. Using EIIL instead of hand crafted environments yields an improvement on the resulting IRM solution by learning worst-case environments for invariant training.



(a) Causal error (b) Non-causal error

Figure 5. MSE of the causal feature $\mathbf{v}$ and non-causal feature $\mathbf{z}$. EIIL applied to the ERM solution (Black) out-performs IRM based on the hand-crafted environment (Green vs. Blue). To examine the inductive bias of the reference model $\tilde{\Phi}$, we hard code a model $\tilde{\Phi}_{\alpha-\text{SPURIOUS}}$ where α controls the degree of spurious feature representation in the reference classifier; EIIL outperforms IRM when the reference $\tilde{\Phi}$ focuses on the spurious feature, e.g. with $\tilde{\Phi}$ as ERM or α-SPURIOUS for high α.

We show in a follow-up experiment that the EIIL solution is indeed sensitive to the choice of reference representation, and in fact, can only discover useful environments (environments that allow EIIL to learn the correct causal representation) when the reference representation encodes the *incorrect* inductive bias by focusing on the spurious feature. We can explore this dependence of EIIL on the mix of spurious and non-spurious features in the reference model by constructing a $\tilde{\Phi}$ that varies in the degree it focuses on the spurious feature, according to convex mixing parameter $\alpha \in [0, 1]$. $\alpha = 0$ indicates focusing entirely on the correct causal feature, while $\alpha = 1$ indicates focusing on the spurious feature. We refer to this variant as EIIL$|\tilde{\Phi} = \Phi_{\alpha-\text{SPURIOUS}}$, and measure its performance as a function of α (Figure 5). Environment inference only yields good test-time performance for high values of α, where the reference model captures the *incorrect* inductive bias.

F.2. ColorMNIST

	Train accs	Test accs
Grayscale (oracle)	75.3 ± 0.1	72.6 ± 0.6
IRM (oracle envs)	71.1 ± 0.8	65.5 ± 2.3
ERM	86.3 ± 0.1	13.8 ± 0.6
EIIL	73.7 ± 0.5	68.4 ± 2.7
Binned EI heuristic (Sec. 3.3)	73.9 ± 0.5	69.0 ± 1.5
Φ_{Color}	85.0 ± 0.1	10.1 ± 0.2
EIIL$\|\tilde{\Phi} = \Phi_{Color}$	75.9 ± 0.4	68.0 ± 1.2
ARL	88.9 ± 0.2	20.7 ± 0.9
GEORGE	84.6 ± 0.3	12.8 ± 2.0
LFF;$\mathcal{L}_{bias} = \text{GCE}_{q\to 0}$	96.6 ± 1.3	30.6 ± 1.0
LFF;$\mathcal{L}_{bias} = \text{GCE}_{q=0.7}$	15.0 ± 0.1	90.0 ± 0.3

Table 8. Additional baselines for the CMNIST experiment reported in Table 2. The mean and standard deviation of accuracy across ten runs ($\theta_y = 0.25$) are reported. See text for description of the baseline methods.

Table 8 expands on the results from Table 2 by adding the following baselines that do not require environment labels:

- Grayscale: a classifier that removes color via pre-processing, which represents an oracle solution
- EIIL$|\tilde{\Phi} = \Phi_{ERM}$ (reported as EIIL in Table 2
- Binned EI heuristic: the binning heuristic for environment inference described in Section 3.3.
- Φ_{Color}: a hard-coded classifier that predicts *only* based on the digit color
- EIIL$|\tilde{\Phi} = \Phi_{Color}$: EIIL using color-based classifier (rather than Φ_{ERM}) as reference.
- GEORGE (Sohoni et al., 2020): This two-stage method seeks to learn the "hidden subclasses" by fitting a latent cluster model to the (per-class) distribution of logits of a reference model. The inferred hidden subclasses are fed to a GroupDRO learner, so this approach can be seen as an instance of EIIL under particular choices of (unsupervised) EI and (robust optimization) IL objectives.
- ARL (Lahoti et al., 2020): A variant of DRO that uses an adversary/auxiliary model to learn worst-case per-example importance weights. Unlike with EIIL, the auxiliary model and main model are trained jointly.
- LFF (Nam et al., 2020) jointly trains a "biased" model f_B and "debiased" model f_D. f_B is similar to our ERM reference model, but is trained with $\text{GCE}_q(p(x;\theta), y) = \frac{1-p_y(x;\theta)^q}{q}$ with hyperparameter $q \in (0, 1]$,[19] and its per-example losses determine importance weights for f_D.

When expanding this study we find that, unlike EIIL, the new baselines fail to find an invariant classifier that predicts based on shape rather than color. Given that GEORGE does a type of unsupervised EI, it is perhaps surprising that it cannot uncover optimal environments for use with its GroupDRO learner. We hypothesize that this is due to assumption of the relevant latent environment labels being "hidden subclasses", meaning that all examples in an optimal environment must share the same class label value. In the CMNIST dataset, this assumption does not hold due to label noise.

We find that, on this dataset, LFF is very sensitive to the hyperparameter q, which shapes the GCE loss of f_B. Interestingly, using the default value of $q = 0.7$, LFF performs optimally on the test set, but this is *not* because the method has learned an invariant classifier based on the digit shape. The below-chance train set performance reveals that LFF has learned an *anti-color* classifier, exactly the opposite of what ERM does. When q approaches zero (GCE approaches standard cross entropy), LFF fails to generalizes to the OOD test distribution.

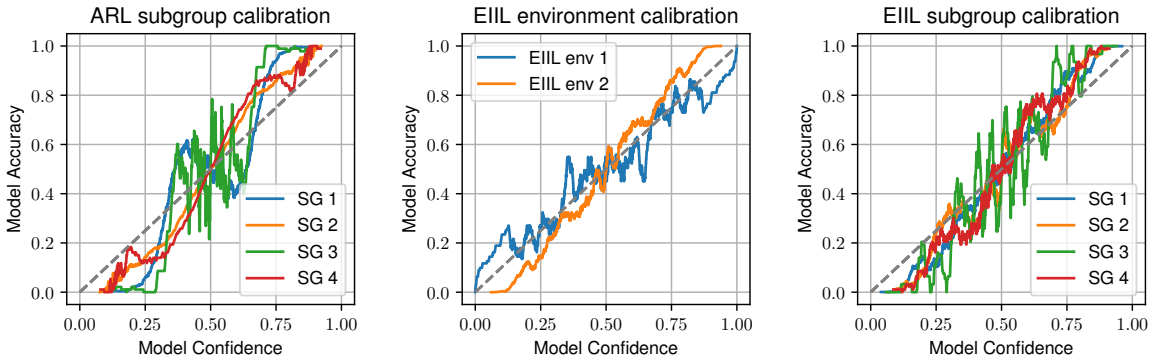
Finally, we found that because the reference classifier predicts with high confidence on the training set, there are only two populated bins in practice. Consequentially, the binned EI heuristic is equivalent to splitting errors into one environment and correct predictions into the other.

F.3. Adult-Confounded

Subgroup sufficiency In the main result we showed that EIIL improves test calibration and accuracy our variant of the UCIAdult dataset. Because the test set is subject to a drastic distribution shift where the correlation pattern between subgroup membership and label is reversed relative to the training set, we can say that this robustness in performance suggests that EIIL does not rely on subgroup membership to make its predictions.

Beyond the global calibration profile, we can also examine calibration curves for the various subgroups, noting again that subgroup labels were not used to train EIIL or the ARL baseline. Figure 6 shows the calibration profiles on the training data. We find that ARL contains noticeable discrepancies in the calibration curves across groups indicating that subgroup sufficiency has not been achieved. EIIL infers environments during the EI phase, which are then implicitly regularized to have roughly the same calibration profile during invariant learning. This can be seen by examining the calibration plots for the training data when it is stratified into the two inferred environments. Finally, looking at calibration curves for the subgroups themselves suggests that EIIL has improved on subgroup sufficiency relative to ARL by better matching the calibration curves across subgroups. These curves still exhibit some noise, indicating that further progress on subgroup sufficiency could be made by changing the invariant learner, possibly by using a different regularize (besides IRMv1) that better enforces the invariance principle.

[19] as $q \to 0$ GCE becomes standard cross entropy



(a) (b) (c)

Figure 6. We examine *subgroup sufficiency*—whether calibration curves match across demographic subgroups—on the Adult-Confounded dataset. Whereas ARL is not subgroup-sufficient (a), EIIL infers worst-case environments and regularizes their calibration to be similar (b), ultimately improving subgroup sufficiency (c). Note that neither method uses sensitive group information during learning.

Ablation Here we provide an ablation study extending Adult-Confounded experiments to demonstrate that both ingredients in the EIIL solution—finding worst-case environment splits and regularizing using the IRMv1 penalty—are necessary to achieve good test-time performance on the Adult-Confounded dataset.

	Train accs	Test accs
EIIL	68.7 ± 1.7	$\mathbf{79.8 \pm 1.1}$
EIIL (no regularizer)	78.6 ± 2.0	69.2 ± 2.8
IRM (random environments)	$\mathbf{94.7 \pm 0.1}$	17.6 ± 1.6

Table 9. Our ablation study shows that both ingredients of EIIL (finding worst-case environments and regularizing invariance across them) are required to achieve good test-time performance on the Adult-Confounded dataset.

From Lahoti et al. (2020) we see that ARL can perform favorably compared with DRO (Hashimoto et al., 2018) in adaptively computing how much each example should contribute to the overall loss, i.e. computing the per-example γ_i in $C = \mathbb{E}_{x_i,y_i \sim p}[\gamma_i \ell(\Phi(x_i), y_i)]$. Because all per-environment risks in IRM are weighted equally (see Equation IRMv1), and each per-environment risk comprises an average across per-example losses within the environment, each example contributes its loss to the overall objective in accordance with the size of its assigned environment. For example with two environments e_1 and e_2 of sizes $|e_1|$ and $|e_2|$, we implicitly have the per-example weights of $\gamma_i = \frac{1}{|e_1|}$ for $i \in e_1$ and $\gamma_i = \frac{1}{|e_2|}$ for $i \in e_2$, indicating that examples in the smaller environment count more towards the overall objective. Because EIIL can discover worst-case environments of unequal sizes, we measure the performance of EIIL using only this reweighting, without adding the gradient-norm penalty typically used in IRM (i.e. setting $\lambda = 0$). To determine the benefit of worst-case environment discovery, we also measure IRM with random assignment of environments. Table 9 confirms that both ingredients are required to attain good performance using EIIL.