# INTERPRETABLE RISK MINIMIZATION

**Anonymous authors**
Paper under double-blind review

## 1 INTRODUCTION

A few critical assumptions are often made by machine learning researchers and practitioners training neural networks. Specifically; an optimal neural network is robust to an unseen sample on the following primary assumptions: a) the chosen architecture is expressive enough to generalize to the data, and b) their dataset is a representative sample of the global distribution. We focus our discussion on this second assumption.

In most cases, especially in restricted or niche domains – it is incredibly rare to find and even harder to verify a truly representative dataset. Additionally, neural networks are fundamentally associative in nature. That, alongside the overparameterized nature of models with regularizers that promote using fewer parameters often results in model posteriors that either overfit to data or learn correlations within the dataset (Kanal & Randall, 1964).

Researchers today are generally aware of this. To ensure that models do not fit to spurious correlations, a significant body of research explores techniques for *machine interpretability* – the degree to which a human can understand the cause of a decision (Molnar, 2020).

Broadly, interpretability within machine learning is classified into two distinct types: post-hoc interpretability (Madsen et al., 2022) and intrinsic interepretability (Pintelas et al., 2020; Sarkar et al., 2022; Reddy et al., 2022). Post-hoc techniques for machine interpretability include approaches that are used as "sanity checks"; to ensure accurate convergence at the end of training, whereas intrinsic interpretability techniques include approaches that are used as part of the training regime, to guide model convergence.

The ideal objective is to ensure that the model backbone, $\Gamma(x; \mathbf{W}_{\mathbf{\Gamma}})$, learns an environment-invariant representation of the target, ensuring that the downstream MLP can perform the task without interference from correlating factors. For this work, we limit the scope of the overall objective to maximizing the worst-group performance. Group-split performance of the dataset refers to the accuracy of the model split by the specific environment in which the target is present. We discuss this in further detail under Section 2. Ensuring equal group-split performance within datasets is also a proposed definition of fairness. Therefore, an additional loosely defined-objective would be ensuring fairer representations of the dataset targets.

This work establishes a proof-of-concept that is able to leverage post-hoc interpretability techniques as part of the training process. By mathematically formalizing inductive loss functions that exploit implicit shared knowledge within the dataset, we aim to update the search space of the model to that of a convex optimization problem; following which backpropagration using elementary optimizers such as Stochastic Gradient Descent (Amari, 1993) are able to obtain the global cost minima, establishing an interpretable basis for training neural networks.

## 2 DATASET

For this task, we use leverage the **Waterbirds** (Sagawa et al., 2019) dataset to test the proposed approach. The dataset contains two classes: landbirds and waterbirds. This is a binary classification task. Crucially, these landbirds and waterbirds are placed in two distinct environments. There are, therefore four different "types" of images: **landbirds on land**, **landbirds on water**, **waterbirds on land**, and **waterbirds on water**. These groups are present in equal proportions during testing. During training, there is a group imbalance; with most samples of landbirds being on land, and most samples of waterbirds being on water, introducing out-of-distribution testing.

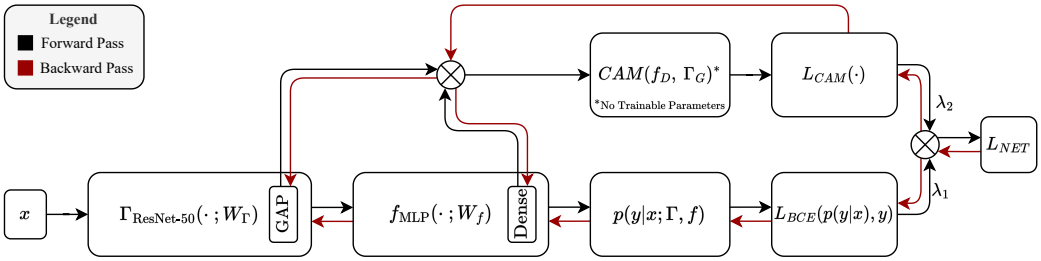



Figure 1: Model architecture with highlighted backward and forward passes.

Part of the research objective is to perform an ablation study using the proposed interpretable training techniques against the empirical risk minimization methods presented in (Liu et al., 2021), which is another reason for using the **Waterbirds** dataset.

The proposed approach has also been tested on a manually modified version of the **CALTECH-256** (Griffin et al., 2007) dataset. It must be noted that this was mostly to test the concept, and while it worked well to simulate the target situation, it is not a reliable benchmark. The **Waterbirds** dataset is a known benchmark for out-of-distribution generalization, and solves this.

## 3 METHODOLOGY

We establish the approach as a bi-leveled optimization problem. Similar to (Arjovsky et al., 2019), $D_E := \{(x_i^e, y_i^e)\}_{i=1}^{n_e}$ is the dataset consisting $E := \{e_i\}_{i=1}^{n_e}$ differing environments. Given $D_W$ s.t. $W \subset E$, our objective is to effectively predict $y_i^e \in D_U$ given $x_i^e$, where $U = E - W$.

$$\text{Let } \mathcal{X} \in \mathbb{R}^{n \times n \times c},\ \mathcal{H} \in \mathbb{R}^{l \times m}, \mathcal{Y} \subset \mathbb{R}$$
$$\Phi : \mathcal{X} \to \mathcal{H} \to \mathcal{Y}$$

Where $\Phi(\mathcal{X})$ is a multiheaded classifier outputting the hidden representation in addition to the classification. $\mathcal{X}$ is the input representation, $\mathcal{H}$ is an arbitrary latent representation and $\mathcal{Y}$ is the classification output. Our objective is to optimize $\Phi(x_i^e) := \{f(\Gamma(x; \mathbf{W_\Gamma}); \mathbf{W_f})^e, h_i^e\} \approx \{y_i^e, \mu\}; \mu \in \mathbb{R}^{\mathbb{N}}$ using dataset specific cost functions defined within Section 3.2.

### 3.1 ARCHITECTURE

The objective of the approach was to directly compare against similar approaches to risk minimization. While any architecture can be utilized for the approach provided that a class activation mapping is obtainable, we chose to use the architecture utilized by (Liu et al., 2021) for an ideal ablation study. We fine-tune the ResNet-50 backbone pre-trained on ImageNet, with a single MLP layer with softmax activation resulting in the final prediction. The architecture is as shown in Figure 1.

The model output is multiheaded, with the actual classification along with the class activation map as output. Each head has it's own loss, a linear combination of which returns the net resultant loss. Particulars regarding hyperparameters are discussed in Section 3.1.1.

#### 3.1.1 TRAINING DETAILS

**Learning Scheduler.** The learning rates as well as loss weights are updated by the learning scheduler, split into three phases: a) bootstrapping, b) training, and c) fine-tuning. The transitions occur at 20% and 80% of the total epochs on which the model was trained. Since the model was trained on 30 epochs, these transitions occurred at epochs 6 and 24 respectively. The learning rate (LR) and loss weight (LW) splits are as follows:

Epochs 1-5: LR: $10^{-3}$, LW: {BCE: $7 \times 10^{-1}$, CAM: 1}

Epochs 6-23: LR: $10^{-4}$, LW: $\{\text{BCE: } 8 \times 10^{-1}, \text{CAM: } 5 \times 10^{2}\}$

Epochs 24-30: LR: $10^{-6}$, LW: $\{\text{BCE: } 1, \text{CAM: } 0\}$

**Optimizer.** We use Stochastic Gradient Descent with momentum $\theta = 0.9$, with learning rate is updated based on the scheduler defined above.

**DataGenerator.** The waterbirds dataset contains a minor label shift between splits, with landbird : waterbird :: $\{3.3081, 3.5075, 3.5125\} : 1$, for training, validation and testing respectively. This split was not accounted for during the data generation process. However, sample stratification by class ratios for the training split was enforced, with random sampling applied for validation and testing sets.

**Batch Size.** We used a batch size of 24, due to the memory restrictions imposed by the available compute. Most other approaches (Liu et al., 2021) trained their models using a batch size of 64.

### 3.2 Class Activation Mappings

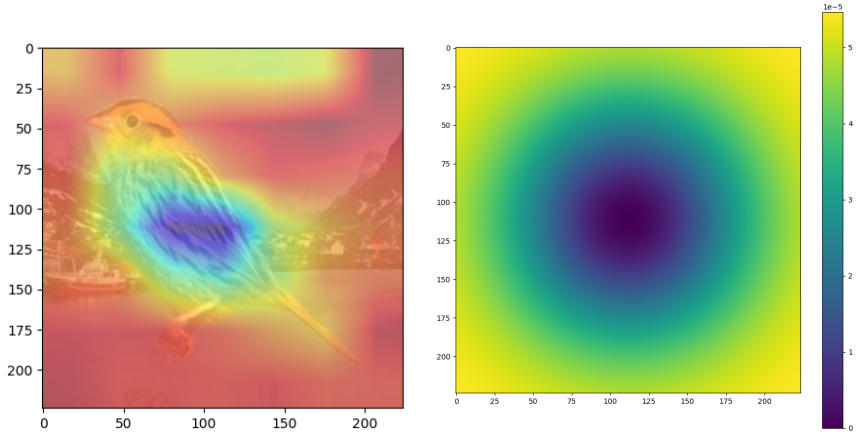
Class Activation Mappings (CAMs) (Zhou et al., 2016) is a method of target localization. By obtaining the projection of weights from the final convolutional layer onto the penultimate dense layer (under the assumption that the number of convolutional filters equals the number of dense neurons), and superimposing it over the original image, we are able to obtain a heatmap of the region used for classification.

We leverage this by utilizing the implicit knowledge within our provided dataset. As an example, both the **Waterbirds** and **CALTECH-256** datasets have centred every target within their images. We evaluate the mean-squared sum of an inverted 2D Gaussian projected against the Class Activation Mapping producing the additional, self-supervised loss function. Formally, we minimize the following objective function:

$$L(\Phi(x_i^e), y_i^e) = L_{BCE}(f(\Gamma(x; \mathbf{W_\Gamma}); \mathbf{W_f})_i^e, y_i^e) + L_{CAM}(h_i^e)$$

$$L(\Phi(x_i^e), y_i^e) = \frac{1}{N} \sum_{i=0}^{N} \sum_{j=0}^{1} y_i^e \log(f(\Gamma(x; \mathbf{W_\Gamma}); \mathbf{W_f})_i^e)) + \sum_{i=0}^{N} \sum_{j=0}^{l} \sum_{k=0}^{m} [(1 - e^{-j^2-k^2}) \cdot \sigma_{\text{relu}}(h_i^{j,k})]^2$$

A visualization of the obtained activation map for a sample spuriously correlating the background, along with the mask against which it is weakly supervised is as follows:




In this case, although the classification is accurate, the sample produces a high loss due to the fact that the background is used for basing the classification, and not the target itself. This establishes a curriculum-style approach to model optimizitation wherein intermediary layers are checkpointed

and backpropagated against. To ensure that our network is unable to overfit to the gaussian projection, we combine this with transformation and translation equivariant convolutional representations, following (Mouli & Ribeiro, 2021).

## 4 Related Work

Arjovsky et al. (2019) presents a causal approach towards out-of-distribution generalization. The paper discusses identifying environment-invariant predictors by deriving a penalty based on the structural causal model defined in their paper.

Liu et al. (2021) uses empirical methods to achieve the same objective. The authors train an identification model that associates sample with high training losses. These are then upweighted, enabling generalization across domains. However, the primary limitation of this work is that it requires the existence of a small set of samples within a dataset. These edge cases are then highlighted. While it allows for better generalization within the sampled distribution, it does not account for samples outside the training distribution.

Krueger et al. (2021) presents a form of robust optimization over a perturbation set of extrapolated domains in addition to a simpler penalty on the variance of training risks. It is able to recover the causal mechanisms of the targets and is also resilient to covariate shift.

## 5 Results

As stated within Section 1, the objective of the model was to learn a latent representation invariant to the background of the model. Since Class Activation Mappings are fundamentally a post-hoc tool, we can use them to compare the regions used for image identification. The following figure displays the class activation mappings for training run with standard ERM, and the updated CAM-based approach:

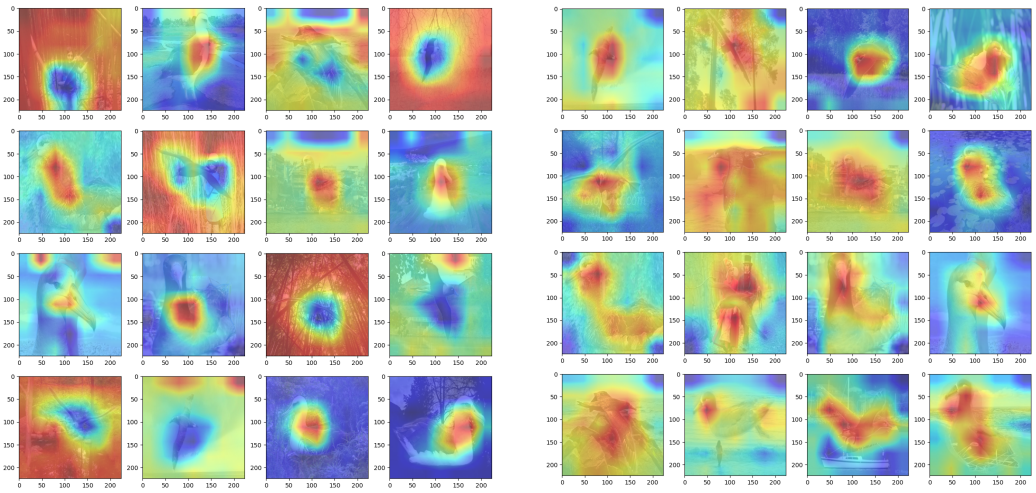


Figure 2: (a) CAMs of evaluations trained using ERM; (b) CAMs of evaluations trained using Interpret-RM.

We observe samples in the ERM-based approach where the model places a high weightage on the background of the image, implying that shortcut learning is employed to learn the data distribution. When the same model architecture is trained using Interpret-RM, we observe that this is mitigated, as there is a higher weightage placed on the actual targets within the image.

One of the concerns during the development of the approach was that by centering the image, we could overfit the model to the inverse gaussian, and the model would only observe the center, regardless of the location of the actual target. Crucially, when we tested a model trained with the updated approach against an image translationally migrated to have the target at the edge of the image, the

model was able to accurately identify the region where the target was present, and not the just focus on the center.

There are a couple of reasons for this. First, this approach was viable due to the inbuilt mechanism for translation equviariance within convolutional neural networks. Second, and likely more important, is that the loss is a reductionist value. The optimization algorithm does not know *why* it receives a high cost for a given inference. As long as the correlation between the metric for the interpretable loss devised by the practitioner is close enough that is can weakly identify when the model is performing optimally, cost function performs as though it is a background/foreground identifier, even the heuristic does not universally hold. The closest minima for the model is now updated to that of the target itself, and therefore the model focuses on the target regardless of it's location during inference. However, if we over-weight the devised mask, it will overfit to the central portion of the image and will not work optimally.

We evaluate our approach against a series of state-of-the-art approaches to group minimization. Updating the table from (Kim et al., 2022), we report the following metrics on the **Waterbirds** dataset:

| Method | Average | Group Average | Worst Group | LL | LW | WL | WW |
|---|---|---|---|---|---|---|---|
| ERM | 97.1 | 84.1 | 69.8 | $99.3 \pm 0.1$ | $71.5 \pm 0.5$ | $69.8 \pm 0.6$ | $95.8 \pm 0.2$ |
| SAM | **97.6** | 87.2 | 75.5 | $99.4 \pm 0.1$ | $77.5 \pm 1.8$ | $75.5 \pm 1.0$ | $\mathbf{96.4 \pm 0.3}$ |
| GDRO | 93.8 | 90.4 | 86.5 | $94.7 \pm 0.7$ | $86.5 \pm 0.7$ | $88.1 \pm 0.3$ | $92.3 \pm 0.3$ |
| SGDRO | 93.1 | **90.6** | **87.2** | $93.8 \pm 1.7$ | $\mathbf{87.2 \pm 0.5}$ | $\mathbf{89.3 \pm 0.3}$ | $92.0 \pm 0.8$ |
| **Interpret-RM (Ours)** | 86.3 | 80.6 | 48.8 | $\mathbf{99.6 \pm 0.1}$ | $83.7 \pm 0.1$ | $48.8 \pm 0.1$ | $90.3 \pm 0.8$ |

We observe that our approach does not perform optimally to the target dataset. We address this in Section 5.1.

### 5.1 Limitations

Regarding benchmark performance, we attribute it primarily to the low batch size used to train the model. We also did not perform any hyperparameter tuning, and assume improvements as a consequence. Finally, we also did not experiment with the various designs of the interpretable loss function, and do not believe the current solution is optimal for the task.

In addition, this approach fails to formally the objective function within the context of interpretability. One of the consequences for this is the naive design of the interpretable loss function. While it is able to converge towards solutions that do not use the background, it is not well balanced, and needs to often be manually tuned to ensure it is not over or under-powered.

Finally, the greatest limitation within the approach is it's lack of generalizability. It requires the dataset to contain a certain shared characteristics, that represents an inductive bias that can be mathematically applied to the dataset. For future work, the objective is to introduce generalizable metrics based on statistically observed behaviors within accurate and inaccurate activation mappings, rather than the use of hardcoded, task and dataset-specific masks to achieve a similar result.

**Reproducibility.** The code, data, models, experiments and execution pipelines are hosted at `https://dagshub.com/jinensetpal/lint`. The pipeline is structured as a DAG, and can be replicated end-to-end with a single command.

## References

Shun-ichi Amari. Backpropagation and stochastic gradient descent method. *Neurocomputing*, 5 (4-5):185–196, 1993.

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

Gregory Griffin, Alex Holub, and Pietro Perona. Caltech-256 object category dataset. 2007.

Laveen N Kanal and Neil C Randall. Recognition system design by statistical analysis. In *Proceedings of the 1964 19th ACM national conference*, pp. 42–501, 1964.

Taero Kim, Sungjun Lim, and Kyungwoo Song. Sharpness-aware minimization for worst case optimization, 2022.

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.

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 *International Conference on Machine Learning*, pp. 6781–6792. PMLR, 2021.

Andreas Madsen, Siva Reddy, and Sarath Chandar. Post-hoc interpretability for neural nlp: A survey. *ACM Computing Surveys*, 55(8):1–42, 2022.

Christoph Molnar. *Interpretable machine learning*. Lulu. com, 2020.

S Chandra Mouli and Bruno Ribeiro. Neural networks for learning counterfactual g-invariances from single environments. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=7t1FcJUWhi3.

Emmanuel Pintelas, Ioannis E Livieris, and Panagiotis Pintelas. A grey-box ensemble model exploiting black-box accuracy and white-box intrinsic interpretability. *Algorithms*, 13(1):17, 2020.

Abbavaram Gowtham Reddy, Vineeth N Balasubramanian, et al. On causally disentangled representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 8089–8097, 2022.

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. *arXiv preprint arXiv:1911.08731*, 2019.

Anirban Sarkar, Deepak Vijaykeerthy, Anindya Sarkar, and Vineeth N Balasubramanian. A framework for learning ante-hoc explainable models via concepts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10286–10295, 2022.

Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2921–2929, 2016.