Impact of Dissimilarity Loss on Out of Distribution Generalization

An introduction of a novel approach for mitigating shortcut learning

Alexandru Cristian Cazacu[1]

Supervisor: Wendelin Böhmer[1]

[1]EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
January 25, 2026

Name of the student: Alexandru Cristian Cazacu
Final project course: CSE3000 Research Project
Thesis committee: Wendelin Böhmer, David Tax

Abstract

Deep Learning has made neural networks ubiquitous in all kinds of applications. During training, models extract features that are predictive of labels, achieving high accuracy values when tested on in-distribution data. However, issues arise when these extracted features, while indicative in training, do not capture the actual underlying causal features of the data. This reliance on spurious correlations is known as "shortcut learning" and leads to failure to generalize on unseen data. In this paper, we introduce a novel regularizer, dissimilarity loss, which aims to penalize the excessive similarity between representations of samples that share the same spurious predictors. This encourages the model to move beyond shortcut features and learn more robust, task-relevant representations. We show that this additional regularization provides significant benefits to out-of-distribution accuracy compared to a baseline and discuss its drawbacks. Furthermore, we apply it without the spurious feature labels, a regime in which dissimilarity loss still remains effective under distribution shift, and explore other possible directions in which improvements can be made by future work.

1 Introduction

Deep Learning has revolutionized artificial intelligence, achieving remarkable success in a plethora of complex tasks, such as vision, language and speech processing. Modern neural networks achieve near perfect accuracy on large, curated benchmarks [1]. However, a major challenge remains, the ability to generalize reliably on unseen data. A well documented shortcoming, is the predilection to rely on spurious correlations, superficial features that are predictive in the training set, but do not capture the underlying causal structure of the data. These shortcut features result in good performance on the training set and on in-distribution test sets, however, when these features are absent, performance degrades significantly. This failure to generalize reliably on out-of-distribution (OOD) data has a wide-reaching impact, particularly in safety-critical domains such as medical imaging or autonomous driving [1].

This sensitivity to spurious correlations is a well-explored topic in literature. When presented with multiple explanations that describe the data well, the simplest ones are preferred even if they represent spurious patterns. Models are known to choose more available, high-variance, non-essential features (e.g. background, texture, secondary objects) that correlate to labels in the training set [1] . For example, consider a classifier trained to distinguish between cows and camels using images where cows typically appear in grassy fields and camels in desert environments. Due to this correlation, the model may learn to associate background features, e.g. grass or sand, with the respective animal labels. As a result, when shown an image of a cow on a beach, it may confidently misclassify it as a camel. Similarly, a camel inside a transport trailer might lead to a random or incorrect prediction, since the familiar shortcut is missing.

Several approaches have been proposed to address this issue, which can be separated based on the stages in the model pipeline they take place: Data-Centric methods, which directly affect the data before training; Representation Learning approaches, which change the training process or architecture to improve robustness; and Post-hoc methods, which are are put into effect after the training process has concluded [16].

One prominent Data-Centric method is Data Augmentation, in which training data is embellished with noise, rotation or cropping, in order to make shortcuts less reliable. Naturally, the perturbation of samples introduces new challenges, making training less stable and decreasing in-distribution (ID) accuracy. Recent research of Data Augmentation in Reinforcement Learning environments extends this method with SOft Data Augmentation (SODA), a technique which introduces a new regularizer, "consistency loss" which aims to "maximize the mutual information between latent representations of augmented and non-augmented data" [3]. This method adds a new objective that minimizes the Euclidean distance between the two embeddings.

Contrastive Loss, or Triplet Loss, is a well-known loss function that was first introduced for the FaceNet system which revolutionized facial recognition [13]. The relevant core concept of this system is that the representation of faces in embedding space is directly tied to their similarity: different images of a person's face have small distances between them, while faces of different people are distanced apart. Contrastive loss applications calculate the cosine distance between normalized vectors, which is more effective than the Euclidean distance as points tend to be further apart in multi-dimensional space.

In the presence of spurious correlations, the internal representations learned by neural networks tend to cluster around said spurious features. Additionally, if these shortcut features are more available than the core features of the data, the models will quickly latch on them (simplicity bias) [11] and the learned representations will end up mostly encoding the shortcut. Samples that share the same spurious feature(s), often become highly similar in embedding space, even if they meaningfully differ in the task-relevant features [15]. Once, this collapse happens, the model achieves very low training loss and is not incentivized to undergo further learning. Generalization still improves slightly after this point, a phenomenon called "grokking" [10], but not enough to be reliably correct. This early dominance of shortcuts and rapid collapse of training loss limits a model's capability to perform in the absence of the spurious features [1; 5; 16].

These concepts form the basis of the novel loss introduced in this paper: directly intervening in the structure of the embedding space with a repulsive force successfully counteracts shortcut learning, by penalizing the excessive similarity of samples that share the same spurious attributes. Dissimilarity loss (DL) pushes apart embeddings of samples that are correlated with the same shortcut features. This avoids the rapid clustering of embeddings, which leaves room for more learning of the underlying causal structure of the data, that

significantly improves out-of-distribution accuracy, at minimal performance overhead. Other techniques available in this space introduce comparatively more complexity [15] and performance costs [17].

Throughout the course of this paper, we create a dataset that makes shortcut features highly available and study the impact of dissimilarity loss on OOD accuracy compared to a baseline model. Our new loss function is weighted by a parameter, λ, and we explore its effects on stability and accuracy. We then theorize about other possible improvements and other topics and implications that require further study.

We highlight our main contributions as follows:

- We present dissimilarity loss, a lightweight regularizer that penalizes the excessive similarity of spuriously correlated representations.
- We study its effects on OOD accuracy and how it affects the stability of training.
- We comprehensively explore the impact of the weight of this novel loss function.
- We attempt to apply dissimilarity loss without the spurious feature labels and present our results.

2 Related Work

Spurious correlation appears in literature under many different names, such as shortcut learning, group robustness and simplicity bias. Geirhos et al. [1] popularized the term "shortcut learning" in modern deep learning by systematically analyzing the reason for the preference of deep neural networks for simple, non-causal decision rules and unifying many failure cases (e.g. texture bias, background bias) under one banner. Since then, notable advancements have been made in detecting and improving resilience to spurious correlations in many different fields. Ye et al. [16] provide an overview of the phenomenon and a comprehensive survey of state-of-the-art techniques for improving OOD robustness.

When provided with multiple redundant predictors of labels, models prefer to encode the most available ones in learned representations. Hermann et al. [4] analyze this pattern from multiple perspectives, providing evidence that availability shapes representation geometry, even in cases where more predictive, but harder to learn features exist. More recent work effectively argues that shortcut reliance increases with model capacity and just by introducing a single hidden layer a shortcut bias can appear, highlighting the importance of further study of shortcut learning [5]. Other work by Qiu et al. [11] supports this finding, asserting that low-complexity spurious features slow down the convergence of causal feature learning.

A large body of work introduces techniques that fall into the Representation Learning category. Based on the observation that shortcut features' contributions to model output increase with the strength of the spurious correlations, Yang et al [15], propose SPARE, identifying shortcut learning early in the training process and using importance sampling to mitigate this phenomenon. Correct-N-Contrast [17] identifies samples with spurious attributes by the representation clusters from a model trained under the standard Empirical Risk Minimization (ERM) regime and applies contrastive learning to improve robustness by contrasting correct and incorrect predictions. Chroma-VAE [14] uses generative classifiers to separate causal and spurious features, while Holstege et al. [6] introduce joint subspace estimation to identify and remove spurious concepts from learned representations. These approaches share the insight that explicitly shaping representation geometry can reduce shortcut reliance. However, many require additional supervision or complex training pipelines.

Other Post-hoc methods have been proposed, such as JTT [9] which improves robustness by collecting the samples that are misclassified by a standard model and increasing their weights when training the final model. NeuronTune [18] identifies and suppresses neurons that activate in the presence of shortcuts. When group or spurious labels are available, several approaches have been put forward which weaken shortcut reliance by reweighing or subsampling to artificially balance majority and minority groups [12].

An alternative line of work focuses on modifying inputs or training data to prevent shortcut exploitation. Masking and augmentation-based methods aim to remove or weaken spurious cues during training. For example, MaDi [2] learns to mask parts of images that are not constructive to the classification task in visual reinforcement learning, while SODA [3] has been shown to improve generalization and increase training stability under distribution shift, by minimizing the information lost between augmented and non-augmented data.

In contrast to prior methods, we investigate a simple dissimilarity-based regularization term that operates directly on the embedding space. Many of the aforementioned examples often require carefully constructed environments with task-specific design choices or additional computational overhead. By encouraging the separation of embeddings that share spurious attributes, the proposed approach aims to weaken shortcut learning while remaining easy to implement in standard machine learning systems.

3 Methodology

3.1 Dataset

To encourage shortcut learning, we have created our own synthetic dataset, that is composed of the combination of two well-known ML datasets, MNIST [8] digits superimposed on top of CIFAR-10 [7] images. Each class of numbers uses a disjunct set of background images. Every sample consists of a grayscale MNIST digit, zero-padded to 32×32 pixels and replicated across three channels, which is then overlaid on top of a color CIFAR background. Digits are rendered as black shapes via a binary mask, ensuring that the foreground digit remains clearly visible against the background content. A few samples from one dataset generation are displayed in Figure 1.

To avoid biases arising from class imbalance, all splits are first balanced to ensure that each label is represented by an equal number of samples. This is done by subsampling each class to the size of the smallest class in the dataset. All splits are derived from the balanced subsets, therefore there is no bias induced from classes being more numerous than others.

Figure 1: Three examples from a generated dataset.

The generation script takes as an argument a list of one or more parameters, N, that indicates how many unique background images are assigned to each digit class. For example, for $N = 1$, all digit samples of each class are overlaid on a single CIFAR image, while at $N = 64$, samples are equally overlaid on 64 randomly selected background images. The background sets are disjunct across classes, to vary how strong backgrounds are indicators of labels.

The script generates three dataset splits:

- **Training set:** Constructed with the class-specific background sets.
- **In-distribution test set:** Uses the same background sets as training, but with different digit samples.
- **Out-of-distribution test set:** Makes use of not-seen-before CIFAR backgrounds, excluded from the pool of available images for the previous splits, such that they are not predictive of labels.

This separation provides the opportunity to study performance on the OOD split that reflects reliance on spurious correlation rather than causal information.

3.2 Model

We use a convolutional neural network based on LeNet-5, which was originally designed for digit recognition tasks and evaluated on the MNIST dataset [8]. It remains a widely used baseline for experimental studies due to its simplicity and interpretability. In addition to the similar classification task and compactness, the model choice was also motivated by the goal of this experiment. A more complex, modern model would still be able to learn the "lookup table" of background to digit even in datasets with large N, making experiments more difficult and computationally intensive.

Training is performed with the Adam optimizer for 10 epochs using cross-entropy loss, optionally augmented by a dissimilarity loss weighted by a hyperparameter λ. When set $\lambda = 0$, training provides the baseline model reference scores. Performance is evaluated based on accuracy on the in-distribution and out-of-distribution test splits across the different values of N.

Additional architectural details and training hyperparameters are reported in Appendix B.

4 Proposed Dissimilarity Loss

To reduce models' reliance on spurious attributes, we introduce an additional optimization objective, dissimilarity loss, that directly impacts the latent space. The goal of this loss function is to discourage the encoding of shortcut features in learned representations, by penalizing similarity between samples that share the same spurious attributes. An illustration of this effect is presented in Figure 2.

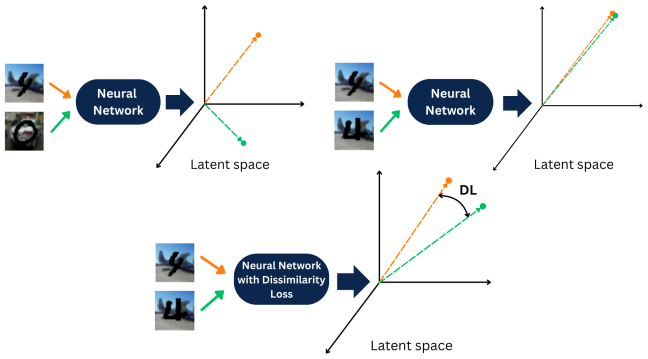

Figure 2: Latent space visualized, clockwise: Embeddings of different-label samples, same-label samples that share a spurious attribute from a typical ERM-trained model and from a model which implements dissimilarity loss.

Let $\mathbf{z}_i \in \mathbb{R}^d$ denote the embedding of sample i obtained from the penultimate layer of the network. They are normalized to unit length as such:

$$\hat{\mathbf{z}}_i = \frac{\mathbf{z}_i}{\|\mathbf{z}_i\|_2} \tag{1}$$

Afterwards, pairwise cosine similarity between embeddings is computed. The normalized dot product is chosen because it provides a scale-invariant metric of alignment between representations and is commonly used in representation and contrastive learning systems.

$$S_{ij} = \hat{\mathbf{z}}_i^\top \hat{\mathbf{z}}_j \tag{2}$$

The dissimilarity loss is defined over the set of pairs of samples that share both the same digit label (classification objective) and same background image (spurious attribute).

$$\mathcal{M} = \{(i,j) \mid i \neq j,\ label_i = label_j,\ bg_i = bg_j\} \tag{3}$$

For all found pairs, the dissimilarity loss is computed as the mean of the similarity matrix. This loss does not enforce a strict margin or minimum dissimilarity, it merely discourages excessive similarity that arises from the shortcut collapse.

$$\mathcal{L}_{\text{dissim}} = \frac{1}{|\mathcal{M}|} \sum_{(i,j)\in\mathcal{M}} S_{ij}. \tag{4}$$

The dissimilarity loss is combined with the standard cross entropy loss as follows:

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda\, \mathcal{L}_{\text{dissim}}, \tag{5}$$

where $\lambda \geq 0$ controls the strength of the regularization.

5 Results & Discussion

5.1 Impact of Dissimilarity Loss

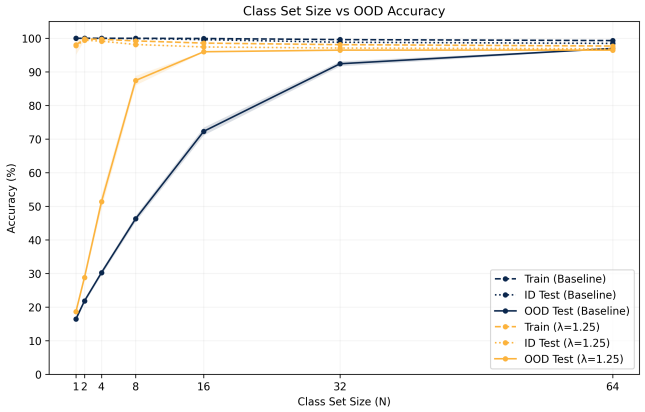

Figure 3: Accuracy across different N values for baseline and $\lambda = 1.25$. 10 models each trained on 10 random seeds with 7 N values. Line represents the mean and shaded region is 95% mean confidence intervals over 700 runs.

In Figure 3, the dark blue lines represent the results of the baseline and we can clearly observe the effects and consequences of shortcut learning. Across all values of N, the model performs great on the in-distribution test set. Cross-entropy loss rapidly collapses, reaching almost zero by the third epoch in almost all runs, achieving accuracy of more than 98%. When the shortcut is taken away, the models performance craters, scaling with N. At small N values, such as 1, 2 or 4, the disjunct sets of backgrounds form such great predictors of the labels, that close to no learning of the causal structure of data takes place. At $N = 1$, accuracy barely outperforms random guessing. As N increases, the backgrounds become a harder indicator of the class, as their number is too great for the complexity of the chosen model to learn. This makes the model focus more and more on the intrinsic features of the data that actually cause the label. As N reaches 32 and 64, the OOD accuracy almost reaches the same values as the ID results.

The addition of dissimilarity loss with weight $\lambda = 1.25$ improves generalization substantially. For small N, the benefits are small as the repulsive force of DL is too weak against the prominent shortcuts. As N gets larger, the model trained with dissimilarity loss shows a substantial improvement in OOD accuracy over the baseline. At this level, distancing the embeddings allows for more learning of the causal features of the digits and this shows in the increase in accuracy. However, at high N, background features no longer provide a reliable shortcut, and cross-entropy alone is sufficient to separate classes. In this regime, the dissimilarity loss introduces an additional optimization constraint that partially interferes with class separation, leading to a slight inversion of performance. For a clearer view of this scenario, we present Figure 4, where N has been increased past 64 and which shows the means crossing more clearly.

A secondary effect is the increased confidence intervals and decreased mean of the in-distribution scores. DL is a

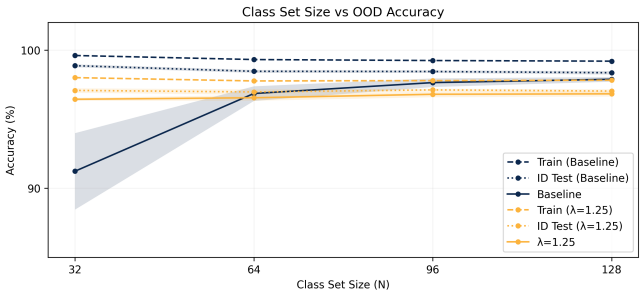

Figure 4: Accuracy across higher N values for baseline and $\lambda = 1.25$. One model trained on 10 random seeds with 4 N values. Line represents the mean and shaded region is 95% mean confidence intervals over 40 runs.

secondary optimization objective that gets added to cross-entropy loss to get the total loss value that the model is trying to minimize during training. By inspecting the training loss values, we have observed that in some training runs, DL pushes the model into local minima from which it is unable to recover. These results highlight a trade-off between improved OOD generalization and optimization stability, which must be considered when selecting the dissimilarity weight. Notably, increasing the weight of the dissimilarity loss also increases variability across random seeds, an effect we further analyze in the following section.

5.2 Effects of Lambda

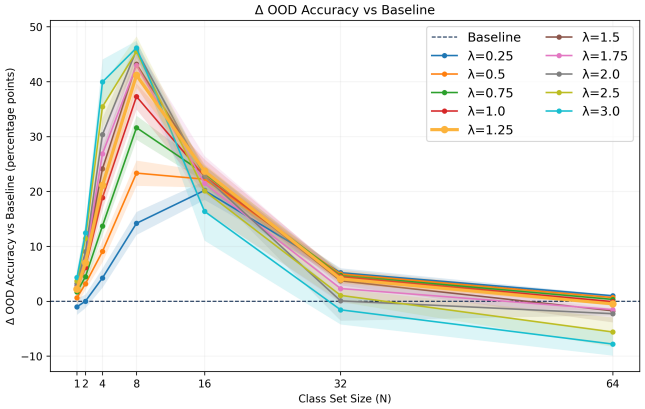

Figure 5: OOD accuracy improvement versus baseline. For each λ, 10 models were trained on 10 random seeds with 7 N values each. Line represents the mean and shaded region is 95% mean confidence intervals over 700 runs.

Figure 5 illustrates the effect of increasing the dissimilarity loss weight λ on out-of-distribution performance. For most values of λ, we observe a consistent improvement in OOD accuracy relative to the baseline at $N < 32$. Beyond $\lambda = 1.25$, further increasing λ yields diminishing returns and eventually degrades OOD accuracy. This suggests that moderate embedding repulsion encourages the model to rely less on spurious background correlations and to encode more task-relevant, causal features of the digits.

However, the relationship between λ and OOD performance is complex: monotonic increase for small to intermediate N and inverse afterwards. This indicates that overly strong dissimilarity regularization begins to interfere with the primary classification objective, making it harder for cross-entropy loss to form coherent class clusters in representation space. This very same previously-mentioned phenomenon emerges when considering larger values of N for all λ values. For $N > 32$, where the number of background variations becomes too large for background features to act as a reliable shortcut, all models trained with dissimilarity loss end up under-performing the baseline trained with cross-entropy alone. In this situation, shortcut learning is naturally suppressed by data diversity, and cross-entropy loss is sufficient to drive the model toward learning causal, digit-specific features. Introducing dissimilarity loss in this setting imposes an unnecessary force on the embedding space, which partially interferes with class separation and leads to a consistent decrease in OOD performance across all values of λ.

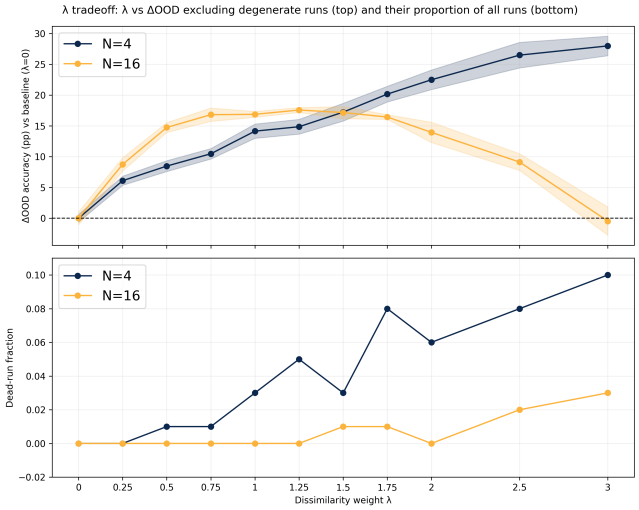

Figure 6: Top plot: OOD accuracy improvement over baseline for lambdas, line represents the mean and shaded region is 95% mean confidence intervals. Bottom plot: proportion of runs stuck in local minima across λ values. Results plotted for $N = 4$ and $N = 16$. Dead runs excluded from ΔOOD accuracy calculation. For each λ, 10 models were trained on 10 random seeds each, with a learning rate of 0.001.

This trade-off is further clarified in Figure 6, which jointly shows OOD accuracy improvement over baseline and the fraction of training runs that end up as degenerate solutions for $N = 4$ and $N = 16$. While OOD accuracy initially improves with increasing λ, the proportion of runs that become stuck in poor local minima rises at higher values. These "dead" runs are characterized by near-random prediction accuracy, indicating a failure of optimization rather than other shortcomings. For statistical purposes we consider a run as "dead" when the ID test accuracy is below 20%. Notice that even with the exclusion of runs counted as "dead", as λ increases, so do the confidence intervals of the OOD accuracy means, highlighting the decrease in stability with bigger λ.

Together, these results demonstrate that λ acts as a double-edged sword: small to moderate values improve robustness by discouraging shortcut learning, while large values destabilize training by overwhelming the classification signal.

5.3 Eliminating the Need for Spurious Feature Labeling

As it has been presented so far, dissimilarity loss has one major drawback from being applicable in real-world scenarios: the need for labeling of the spurious attribute, i.e. backgrounds. The results of the relaxation of this requirement are presented in Figure 7. The indiscriminate repulsion of all same label pairs of embeddings unsurprisingly interferes with the classification objective resulting in the severe degradation of OOD performance. A "tug-of-war" is created between cross-entropy which tries to cluster classes together while dissimilarity loss wants within-class dispersion.

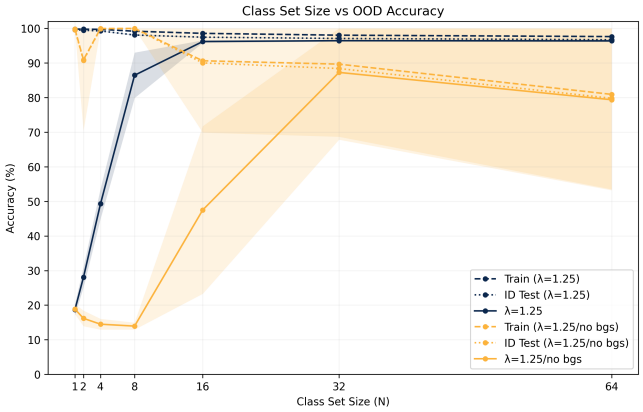

Figure 7: OOD accuracy for a model with the normal dissimilarity loss and one without the background constraint. Both have $\lambda = 1.25$, and were trained on 10 random seeds. Line represents the mean and shaded region is the 95% mean confidence intervals over 70 runs.

A more interesting approach comes from an observation presented in section 1 and exploited by other techniques in this space, the fact that the embeddings of samples that share a spurious correlation cluster together around these spurious attributes. Theoretically, by a careful choice of a threshold τ, one could achieve the same effect of dissimilarity loss by simply penalizing pairs that have a similarity $S_{ij} > 1 - \tau$. This proxy rule based on high similarity alone could possibly recover the set of pairs targeted by the original regularizer. The choice of this parameter τ, therefore, is highly sensitive, to avoid falling in the same predicament as in Figure 7, in which a lot of collateral pairs are affected, impacting generalization and stability.

To search for τ, we tracked similarity distributions separately for same-label pairs that share the same background and pairs with different backgrounds. Across seeds, we summarized these results by their 95-percentile statistics and their separation as a function of training epochs and then grouped them based on N. An important note is that the case $N = 1$ is absent as this difference is undefined as every label is cor-

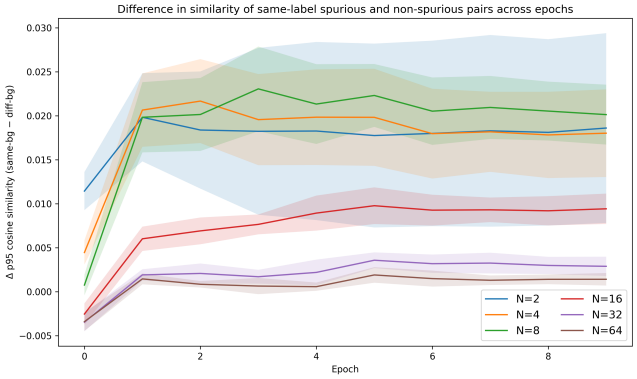

Figure 8: Difference between the 95-percentile similarity of same-label embedding pairs that share the same background and pairs that have different backgrounds across epochs and grouped by N. Statistics from training a baseline model, 10 random seeds. $N = 1$ is absent as the difference is undefined. Line represents the mean and shaded region is 95% mean confidence intervals.

related with one single background. The results are available in Figure 8. For every other N, the separation was too close to result in an viable threshold, and, moreover, the values are different across N. This is not optimal as by removing the need for background labels we create a dependence for labels indicating the N values. Therefore, we could not discern an global optimal threshold that would achieve the desired results. Another plausible direction would be to penalize the k most similar pairs in a batch, however, we leave the exploration of this possibility to future work.

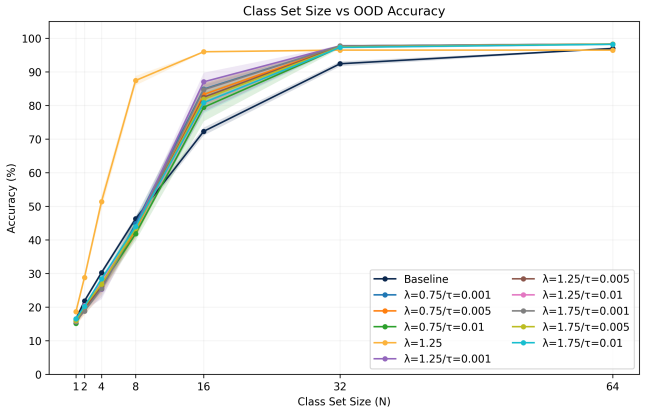

Figure 9: Accuracy across different N values for baseline, $\lambda = 1.25$ and different (λ, τ) pairs. 5 models each trained on 10 random seeds, except for the baseline and $\lambda = 1.25$ values, which are aggregated from 10 models. Line represents the mean and shaded region is 95% mean confidence intervals over 350 and 700 runs, respectively. ID and train test scores are omitted for clarity as they don't exhibit any unusual behavior.

Nevertheless, we experimented with a range of τ values; the results are shown in Figure 9. The hyperparameter pair (λ, τ) which yields the best results is $(1.25, 0.001)$. For small values of N, performance slightly degrades relative to the baseline. We ascertain this effect as a consequence of the previously mentioned interference with class structure: the loss unintentionally penalizes different-background pairs which causes the model to end up overfitting on background noise, which worsens generalization. However, for intermediate N, we observe an increase in OOD accuracy. While smaller than the one resulting from the dissimilarity loss using background labels, this signifies that this proxy rule successfully targets some of the spurious pairs penalized by the original loss. At the top end of the N range, where the shortcuts are naturally suppressed by data diversity and embedding pairs become less tightly clustered, this "semi-supervised" loss overtakes the original one. We hypothesize that this inversion is motivated by the same phenomenon shown in Figure 5, where lower lambda values perform better in high-N regimes.

6 Additional Considerations

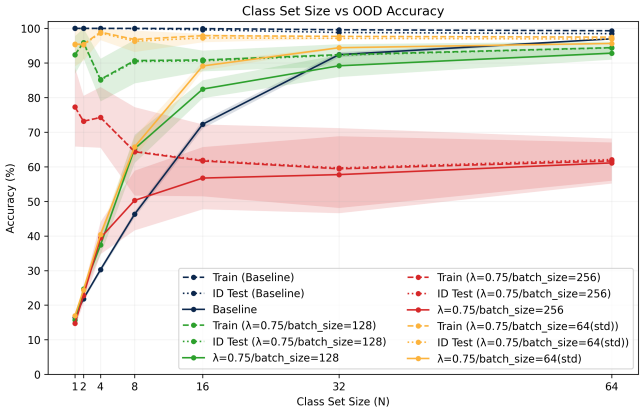

Figure 10: Accuracy across N values for baseline and different batch sizes. All DL models use $\lambda = 0.75$. 10 models trained on 10 random seeds each, with a learning rate of 0.001. Line represents the mean and shaded region is 95% mean confidence intervals over 700 runs. The label "std" refers to the standard batch size of 64 that was used in all other experiments in this paper.

Batch size has a direct and nontrivial impact on this method because the dissimilarity loss is computed within each mini-batch. As batch size increases, the composition of each batch changes: there are more pairwise terms, but also more variability in how many "valid" pairs exist (same label and same background), and the loss increasingly reflects a global average over many pair sets. This makes the strength of dissimilarity loss very inconsistent across training, which in turn, destabilizes training by interfering with cross-entropy which leads to the results in Figure 10. All other experiments in this paper were conducted with a batch size of 64.

Initially, all experiments were conducted using the standard learning rate for Adam applications, 0.001. Increasing it to 0.005 consistently improved performance and reduced the frequency of runs ending up in local minima. A plausible explanation is that with the addition of the dissimilarity term, the model must simultaneously satisfy cross-entropy class clustering and an additional geometric constraint on the embedding space. With a smaller learning rate, early updates

may be too conservative to escape undesirable local minima created by the interaction of these two objectives. A larger learning rate increases the effective "jump size" through this special loss landscape, making it more likely to escape degenerate solutions. Furthermore, the increased learning rate could possibly allow the model to act more on the information given by the dissimilarity loss before cross-entropy starts to dominate, which allows for better robustness. Unless specified otherwise, all experiments in this work were conducted with a learning rate of 0.005.

7 Responsible Research

7.1 Ethics

In this paper, we investigate shortcut learning and out-of-distribution generalization using a synthetic dataset generated from MNIST and CIFAR-10. MNIST consists of handwritten digits and CIFAR contains low-resolution natural images, both of which are widely used benchmarks with well-established usage in the research community. From an ethical viewpoint, neither contain personally identifiable information or sensitive attributes. Therefore, this work does not raise concerns related to privacy, consent or demographic bias in data collection.

However, this topic is highly relevant in safety critical applications such as medical imaging, autonomous driving, biometric identification and military. Models deployed in such domains can rely on spurious attributes rather than causal features and dissimilarity loss demonstrably improves on this fault, however, it is not a "silver bullet". Exploitation of shortcuts is not completely eliminated nor is reliability guaranteed for unseen data. Therefore, it should not be considered a complete solution for deployment in real-world systems. This is especially true for the aforementioned sensitive applications, where it must be accompanied by rigorous testing, validation and human oversight.

7.2 Reproducibility

All experiments in this work were designed to be fully deterministic and replicable. Dataset generation, data splits, and training procedures are controlled by fixed random seeds, all of which are recorded and available in Appendix A. The datasets used in the experiments are stored as pre-generated files indexed by their random seeds, ensuring that the exact same training and test sets can be reconstructed. Furthermore, all model hyperparameters, optimizer settings, and loss weights are logged and exported for every run using Weights & Biases[1], and the full experimental codebase is publicly available at `https://github.com/alexc-cazacu/ood-resilience-dissimilarity-loss`. This enables exact replication of both individual training runs and aggregate statistics reported in the paper.

[1] `https://wandb.ai/`

8 Future Work

8.1 Dissimilarity Loss Parametrization and Other Improvements

As demonstrated, the dissimilarity loss explored in this paper leads to statistically significant OOD improvements, but there are still design choices left unexplored.

A natural development is to implement ideas common in metric learning approaches. Instead of directly minimizing average similarity, a margin or offset value could be introduced, such that pairs are only distanced if their similarity exceeds a certain threshold. This would allow embeddings to remain close as long as they are sufficiently separated, preventing the perturbation of class structure observed at large N, as shown in Figure 4. Additionally, applying the common activation functions to the similarity term could also improve the same drawback, reducing interference with cross-entropy.

Furthermore, in this work the dissimilarity weight λ is treated as a fixed hyperparameter. However, as shown in Figure 5, its optimal value depends highly on how strong is the spurious correlation, the value for the best performing λ being inversely proportional to N. A promising idea is replace λ with a target dissimilarity level. From the best performing runs, through analysis, a desired dissimilarity value could be estimated, and an additional optimizer could minimize the difference between it and the observed dissimilarity. The model would then automatically adjust how strongly embeddings are repelled, removing a need for a fixed λ. This could dramatically improve training stability and increase accuracy across N.

8.2 Further Exploration of Hyperparameter Space

Due to time and computational constraints only a subset of the hyperparameter space was explored in this work. As presented in section 6, learning rate tuning yielded measurable improvements and reduced the number of runs stuck in local minima, but many dimensions remain open. Different learning rate schedulers or optimizer parameters could all interact with DL in nontrivial ways.

Initial experiments with stochastic gradient descent (SGD) yielded poor performance, but we are wary to discount its viability altogether. SGD requires different learning rates, momentum schedules or a warm-up strategy to function at its maximum potential. A more fine-grained exploration could reveal better results.

Another promising direction is the choice of layer where dissimilarity loss is calculated. We have explored the implementation for the penultimate layer, but earlier layers could lead to stronger resilience to shortcut learning.

Finally, it is important to test this approach beyond LeNet and our MNIST-on-CIFAR dataset. Applying dissimilarity-based regularization to modern architectures (e.g., ResNets, Vision Transformers), different datasets (e.g. FashionMNIST, Waterbird), different types of data (e.g., text, time series), or sequence models (e.g., RNNs) would clarify whether the observed benefits reflect a broader impact or are specific to the present experimental setup.

9 Conclusions

In this work, we investigated the use of a dissimilarity-based regularization term to mitigate shortcut learning and improve out-of-distribution generalization in image classification. Using a controlled synthetic dataset designed to induce spurious background correlations, we demonstrated that standard cross-entropy training leads to strong in-distribution performance while failing catastrophically under distribution shift. This confirms prior findings that neural networks readily exploit shortcuts when they provide an easier optimization path.

We showed that introducing dissimilarity loss can substantially improve OOD performance across a wide range of settings, particularly when spurious correlations are strong but not overwhelming. By explicitly discouraging excessive similarity between embeddings that share spurious attributes, the proposed method promotes reliance on more causal features of the data. Our results also highlight important trade-offs: excessive weighting of the dissimilarity objective can interfere with class separation, increase optimization instability, and lead to a higher fraction of training runs becoming trapped in poor local minima.

However, the presented regularizer is limited in its applicability due to the necessity of spurious feature labeling. We explore the relaxation of this constraint, which provides some moderate gains when applied with a similarity threshold, and show the limits of what the loss can achieve in its current state. Promising but fragile, we list several directions in which dissimilarity loss can be improved upon by future work.

Overall, this study contributes empirical evidence that representation-level constraints can meaningfully reduce shortcut reliance, while also underscoring the importance of careful hyperparameter tuning and stability analysis.

References

[1] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, November 2020.

[2] Bram Grooten, Tristan Tomilin, Gautham Vasan, Matthew E. Taylor, A. Rupam Mahmood, Meng Fang, Mykola Pechenizkiy, and Decebal Constantin Mocanu. Madi: Learning to mask distractions for generalization in visual deep reinforcement learning, 2023.

[3] Nicklas Hansen and Xiaolong Wang. Generalization in reinforcement learning by soft data augmentation, 2021.

[4] Katherine L. Hermann and Andrew K. Lampinen. What shapes feature representations? exploring datasets, architectures, and training, 2020.

[5] Katherine L. Hermann, Hossein Mobahi, Thomas Fel, and Michael C. Mozer. On the foundations of shortcut learning, 2024.

[6] Floris Holstege, Bram Wouters, Noud van Giersbergen, and Cees Diks. Removing spurious concepts from neural network representations via joint subspace estimation, 2024.

[7] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical Report 0, University of Toronto, 2009.

[8] Yann Lecun, Yere Yere, Patrick Haffner, Yoesoep Rachmad, and Leon Bottou. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86:2278 – 2324, 12 1998.

[9] Evan Zheran 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, 2021.

[10] Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. Grokking: Generalization beyond overfitting on small algorithmic datasets. *arXiv preprint arXiv:2201.02177*, 2022.

[11] GuanWen Qiu, Da Kuang, and Surbhi Goel. Complexity matters: Dynamics of feature learning in the presence of spurious correlations, 2024.

[12] 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, 2020.

[13] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, page 815–823. IEEE, June 2015.

[14] Wanqian Yang, Polina Kirichenko, Micah Goldblum, and Andrew Gordon Wilson. Chroma-vae: Mitigating shortcut learning with generative classifiers, 2022.

[15] Yu Yang, Eric Gan, Gintare Karolina Dziugaite, and Baharan Mirzasoleiman. Identifying spurious biases early in training through the lens of simplicity bias, 2024.

[16] Wenqian Ye, Luyang Jiang, Eric Xie, Guangtao Zheng, Yunsheng Ma, Xu Cao, Dongliang Guo, Daiqing Qi, Zeyu He, Yijun Tian, Megan Coffee, Zhe Zeng, Sheng Li, Ting-hao, Huang, Ziran Wang, James M. Rehg, Henry Kautz, and Aidong Zhang. The clever hans mirage: A comprehensive survey on spurious correlations in machine learning, 2025.

[17] 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, 2024.

[18] Guangtao Zheng, Wenqian Ye, and Aidong Zhang. Neurontune: Towards self-guided spurious bias mitigation, 2025.

A Seeds Used for Dataset Generation

Batch	Seeds
Batch 00	13 21 42 67 69 89 773 654 438 433
Batch 01	858 85 697 201 94 526 975 735 761 717
Batch 02	786 513 128 839 450 500 370 182 926 781
Batch 03	643 402 822 545 443 450 227 92 554 887
Batch 04	63 858 827 276 631 165 758 700 354 66
Batch 05	383329928 3324115917 2811363265 1884968545 1859786276 3687649986 369133709 2995172878 865305067 404488629
Batch 06	2261209996 4190266093 3160032369 3269070127 3081541440 3376120483 2204291347 550243862 3606691182 1934392873
Batch 07	2148995113 1592565387 784044719 3980425318 3356806662 2765379633 1728356534 3533734221 2342600227 1904449482
Batch 08	1934709050 975982878 395720738 2381923523 3813457839 274093027 3686333038 3554648816 1188673836 2712977936
Batch 09	709653493 3255962051 3008723231 1522677438 291714185 4169116269 1914213180 3835926007 2911639984 3343131663

Table 1: Random seeds used for dataset generation.

B Model

The network consists of two convolutional layers with ReLU activations followed by max-pooling operations, and three fully connected layers. The model contains 62k trainable parameters and takes three-channel 32×32 input images, uses 5×5 square convolution, mini-batching for the training and outputs class logits for the ten MNIST digit classes.

Unless stated otherwise, all experiments are ran using the Adam optimizer, a learning rate of 0.005 with a multi-step learning rate schedule with a parameter $\gamma = 0.1$ and mini-batch size of 64. Models are trained for 10 epochs and the primary criterion used is Cross-Entropy Loss in addition to the secondary criterion explored in this paper. Dissimilarity Loss is weighted by a hyperparameter λ, which when set $\lambda = 0$, provides the baseline model reference scores. Performance is evaluated based on accuracy on the in-distribution and out-of-distribution test splits across the different values of N.

C Statement on the Use of Large Language Models

Large Language Models (LLMs) were used in a limited capacity during the creation of this work. For the paper, LLMs were used to help with the usage of LaTeX and to improve grammar and readability of sentences, especially to synthesize long, drawn-out sentences that were unwieldy in the English language. For elaborating the codebase, LLMs aided mostly in the fixing and interpretation of programming errors, the interaction with the Weights & Biases API to download and aggregate statistics that were not saved locally and the plotting pipeline. All outputs were reviewed and verified.

Some representative samples of used prompts:

- *Please rephrase this sentence:* [...]
- *Please help me add this remark in this sentence:* [...]
- *I trained a model and logged the statistics using W&B, how do I download the logs locally*
- *I am training a model, please help me interpret this error:* [...]
- *I have a plotting script using Matplotlib, how do I increase the legend font size/display legend on two columns/create a top-bottom plot with the same x axis/etc.*