Evading the Simplicity Bias: Training a Diverse Set of Models Discovers Solutions with Superior OOD Generalization

Damien Teney[1,2] Ehsan Abbasnejad[2] Simon Lucey[2] Anton van den Hengel[2,3]
[1]Idiap Research Institute [2]Australian Institute for Machine Learning, University of Adelaide [3]Amazon
damien.teney@idiap.ch, {ehsan.abbasnejad, simon.lucey, anton.vandenhengel}@adelaide.edu.au

Abstract

Neural networks trained with SGD were recently shown to rely preferentially on linearly-predictive features and can ignore complex, equally-predictive ones. This simplicity bias can explain their lack of robustness out of distribution (OOD). The more complex the task to learn, the more likely it is that statistical artifacts (i.e. selection biases, spurious correlations) are simpler than the mechanisms to learn.

We demonstrate that the simplicity bias can be mitigated and OOD generalization improved. We train a set of similar models to fit the data in different ways using a penalty on the alignment of their input gradients. We show theoretically and empirically that this induces the learning of more complex predictive patterns.

OOD generalization fundamentally requires information beyond i.i.d. examples, such as multiple training environments, counterfactual examples, or other side information. Our approach shows that we can defer this requirement to an independent model selection stage. We obtain SOTA results in visual recognition on biased data and generalization across visual domains. The method – the first to evade the simplicity bias – highlights the need for a better understanding and control of inductive biases in deep learning.

1. Introduction

Inductive biases in deep learning. At the core of every learning algorithm are a set of inductive biases [45]. They define the learned function outside of training examples and they allow extrapolation[1] to novel test points. Deep neural networks are remarkably effective because their inductive biases happen to reflect properties of real-world data, although the reasons are still poorly understood [87]. In particular, the *simplicity bias* [27, 47, 51, 57, 70] has been proposed as a reason of their success. It makes networks trained with SGD[2] represent preferentially simple, approximately piecewise linear functions.[3] But the simplicity bias can also prevent the learning of complex patterns that are the actual mechanisms of the task of interest. This effect is problematic when the learned simple patterns correspond to spurious correlations a.k.a. statistical shortcuts [17]. In image recognition, an example of a shortcut is to use the

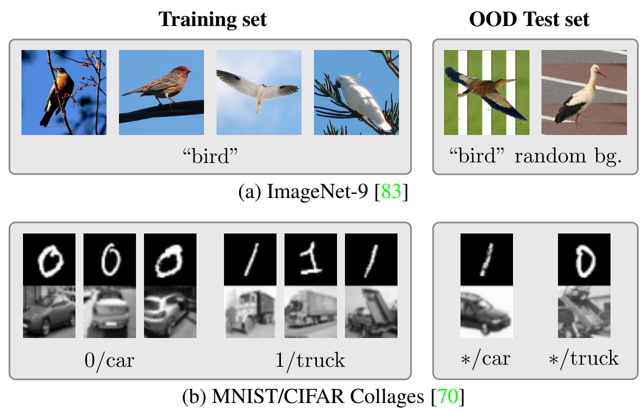


Figure 1. Training data often contains multiple predictive patterns. (a) In ImageNet-9, bird shapes and blue backgrounds are both predictive of the *bird* label. (b) In MNIST/CIFAR collages, both parts are equally predictive of training labels. Neural networks display a **simplicity bias**: they latch on the MNIST digit for example, and completely ignore the CIFAR part. We show that we can mitigate the simplicity bias by training a collection of models, allowing us to discover a diverse set of predictive patterns.

[1] Contrary to popular belief, deep neural networks rarely perform interpolation even in i.i.d. settings. In high dimensions, test points are extremely unlikely to lie within the convex hull of training points [24, 34].

[2] The simplicity bias is not a property of neural networks themselves, but also of their training with SGD, since it is possible to manually construct networks with arbitrarily poor generalization [87] *i.e.* no simplicity bias.

[3] We adopt the definition of *simplicity* of a feature from [70]: it is the minimum number of linear pieces in the decision boundary that achieves optimal classification accuracy using this feature. The definition naturally extends to the simplicity of a *model* implementing this decision boundary.

background rather than the shape of the object. In natural language understanding, an example is to use the presence of certain words rather than the overall meaning of a sentence. These shortcuts are inevitable byproducts the data collection process *e.g*. from selection biases. They are increasingly problematic as tasks tackled with deep learning grow in complexity. The mechanisms to learn are more and more likely to be overshadowed by simpler spurious patterns.

Role of inductive biases in OOD generalization. OOD Generalization or strong generalization is the capability of making accurate predictions under arbitrary covariate shifts. To achieve this, a model must learn and reflect the intrinsic (*i.e*. causal) mechanisms of the task of interest.[4] For example, recognizing objects in arbitrary scenes requires a model to learn about their shape and details. It cannot rely solely on the background or contextual cues (Figure 1a) . OOD Generalization fundamentally requires extra information beyond i.i.d. training examples [6, 68]. Existing methods use side information such as multiple training environments [1, 12, 56], counterfactual examples [25, 73], or non-stationary time series [23,30,58]. Importantly, OOD generalization is not achievable only through regularizers, network architectures, or unsupervised control of inductive biases [6]. To make this limitation intuitive, consider the task of image recognition in Figure 1. Should a *bird* label result from a bird shape or from a blue sky ? If shape and background are equally predictive of training labels, the data simply lacks the information to prefer one over the other (*i.e*. the task is underspecified: the same data could support a task where the labels relate to the background and not the objects). This is where a learning algorithm's inductive biases come into play, possibly detrimentally. The simplicity bias favors the most linearly-predictive patterns, but these may be spurious. While existing methods attempt to integrate extra information during training, we show this can be deferred to a model selection stage.

This study. We seek to control the simplicity bias of neural networks and investigate benefits in OOD generalization. Variations of architectures, hyperparameters, or random seeds have no effect on the simplicity bias. Instead, we train a collection of similar models to fit the training data in different ways. Each model is optimized for standard empirical risk minimization (ERM) while a regularizer encourages diversity across the collection. It pushes each model to rely on different patterns in the data, including complex ones that are otherwise ignore because of the simplicity bias. Identifying a model with good OOD performance is reduced to an independent model selection step that can use any type of side information such as those mentioned above.[5]

Applicability of our method. We use three image recognition datasets to demonstrate improvements in generalization relevant to computer vision. Issues with OOD generalization are also root causes of adversarial vulnerabilities [26], some model biases [49,66], and poor cross-domain/-dataset transfer [76]. Potential benefits in these areas remain to be investigated. Kariyappa *et al*. [32] already demonstrated improved adversarial robustness by increasing diversity in an ensemble with a method similar to ours.

Summary of contributions.

1. We review the fundamental requirements for OOD generalization and derive a rationale for addressing generalization during model selection rather than training.
2. We describe a method to overcome the simplicity bias by learning a collection of diverse predictors.
3. We demonstrate these benefits on existing benchmarks.
   (a) A new capability to learn multiple predictive patterns otherwise ignored because of the simplicity bias (multi-dataset collages [70]).
   (b) Improved activity recognition after training on visually-biased data (Biased Activity Recognition dataset [48]).
   (c) Improved object recognition across visual domains (PACS dataset [36]).

Far from a complete solution to OOD generalization, this paper highlights the need for a better understanding and control of inductive biases in deep learning.

2. Background

Simplicity bias. Deep learning is actively studied to understand reasons for its successes and failures. The simplicity bias [51, 70], gradient starvation [57], and the learning of functions of increasing complexity [57] help explain the lack of robustness of deep neural networks and why their performance degrade under minor distribution shifts and adversarial perturbations. Shah *et al*. [70] showed that neural networks trained with SGD are biased to learn the simplest predictive features in the data while ignoring others. Worryingly, approaches like ensembles and adversarial training – believed to improve generalization and robustness – are ineffective at mitigating the simplicity bias.

Shortcut learning [17, 35] is synonymous with poor OOD generalization. It happens when a model learns predictive

[4] OOD Generalization goes beyond the in-domain (ID) generalization of classical learning theory. Perfect ID generalization (*i.e*. reaching Bayes error rate on a test set from the same distribution as the training data) is achievable with infinite training data, but the predictions may rely entirely on spurious correlations (*e.g*. recognizing birds from blue skies).

[5] Model selection for OOD performance cannot be achieved with a standard (in-domain, ID) validation set [6]: high ID performance can be attained by relying on spurious patterns, which says nothing about the model's capabilities OOD. An OOD validation set is a valid option that makes this step similar to the cross-validation routinely used to select architectures and hyperparameters.

patterns that do not correspond to the task of interest. For example in object recognition (Figure 1), the model uses the background rather than the shape of an object [5, 18]. The model is accurate on in-domain (ID) data (*i.e.* from the same distribution as the training set) but is correct for the wrong reasons. Failures are apparent on OOD test data where the spurious patterns learned during training cannot be relied on. The more complex the task, the larger the space of possible spurious patterns that are simpler than the mechanisms of task, and the more likely is a case of shortcut learning. The simplicity bias exacerbates shortcut learning.

OOD Generalization (*i.e.* avoiding shortcut learning) is fundamentally not attainable solely with ERM [68]. ERM learns *any* pattern predictive of training labels. OOD generalization requires knowing which patterns correspond to causal mechanisms of the task (Figure 1). This information is lost by sampling i.i.d. training examples from the joint distribution produced by the data-generating process [6]. Current approaches to recover the missing information use multiple training environments [1, 12, 56], counterfactual examples [25, 73], or non-stationary time series [23, 30, 58]. Other options to improve OOD generalization rely on ad hoc task-specific knowledge [3, 11, 41, 48, 71, 77].

Ensembles. This paper is not about building ensembles. Ensembling means that multiple models are combined for inference. Rather, we train a collection of models and identify one for inference (experiments include ensembles for comparison). The goal of ensembling is to combine models with uncorrelated errors into one of lower variance. Our goal is to discover predictive patterns normally missed by a learning algorithm because of its inductive biases.

See Appendix D for an extended literature review.

3. Proposed method

Method overview. We train a collection of models in parallel (see Figure 2). A *diversity regularizer* encourages them to represent different functions. They share the same architecture and data. The regularizer is required because trivial options such as training models with different initial weights, hyperparameters, architectures, or shuffling of the data do not prevent converging to very similar solutions affected by the simplicity bias as demonstrated in [70].

Setup. We consider a supervised learning task, where one model is typically trained on a training set of examples $\boldsymbol{T}=\{(\boldsymbol{x}^k, \hat{\boldsymbol{y}}^k)\}_{k=1}^K$. The vectors $\boldsymbol{x}$ represent input data such as images and $\boldsymbol{y}$, in the case of a classification task, vectors of ground truth scores $[0,1]^C$ over C classes. The standard practice is to train a model (typically a neural network) for empirical risk minimization (ERM) on $\boldsymbol{T}$. A model implements a function $F : \mathrm{supp}(\boldsymbol{x}) \rightarrow \mathrm{supp}(\boldsymbol{y})$. We represent it as a composition $F = g \circ f$ of a feature extractor $f_{\boldsymbol{\theta}}(\cdot)$ and classifier $g_{\boldsymbol{\phi}}(\cdot)$ parametrized by weights $\boldsymbol{\theta}$ and

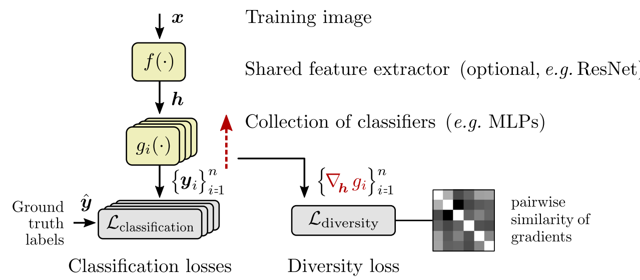


Figure 2. Our method trains a collection of classifiers in parallel to produce different predictions on OOD data. A diversity loss penalizes pairwise similarities between models, using each classifier's input gradient at training points. Combined with standard classification losses, we optimize models for both distinctness and predictive performance.

$\boldsymbol{\phi}$ respectively. We further define the hidden representation $\boldsymbol{h} = f_{\boldsymbol{\theta}}(\boldsymbol{x})$. The model F is typically optimized to minimize the risk $\mathcal{R}$ of a predictive loss $\mathcal{L}_{\text{classification}}$ on $\boldsymbol{T}$ by solving

$$\min_{\boldsymbol{\theta},\boldsymbol{\phi}} \mathcal{R}(F_{\boldsymbol{\theta},\boldsymbol{\phi}}) \tag{1}$$

$$\text{with the risk } \mathcal{R}(F_{\boldsymbol{\theta},\boldsymbol{\phi}}) = \Sigma_k^K\, \mathcal{L}_{\text{classification}}(\hat{\boldsymbol{y}}^k, \boldsymbol{y}^k) \tag{2}$$

$$\text{and predictions } \boldsymbol{y}^k = F_{\boldsymbol{\theta},\boldsymbol{\phi}}(\boldsymbol{x}^k) = g_{\boldsymbol{\phi}}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}^k)\big). \tag{3}$$

In the following, we call a *predictor* any function $F_{\boldsymbol{\theta}^\star,\boldsymbol{\phi}^\star}$ from the chosen hypothesis space (*e.g.* neural networks of a certain architecture) where $(\boldsymbol{\theta}^\star, \boldsymbol{\phi}^\star)$ is a solution to (1).

Why we sometimes need more complexity. The simplicity bias exposed in [70] implies that a neural network trained with SGD for (1) relies on the simplest features predictive of labels in $\boldsymbol{T}$. It ignores more complex ones even if equally predictive. The simplicity of a feature is defined in [70] as the minimum number of linear pieces in the decision boundary achieving optimal classification using this feature. We further assume that a predictor relying this feature implements its corresponding simple decision boundary (although not stated explicitly in [70] it seems supported by their experiments). If a simple spurious pattern exists in the data, the simplicity bias will prevent from learning any more complex mechanism of the task. In such cases, it is desirable to force learning a more complex predictor.

How diversity can induce complexity. By assumption of the simplicity bias, the default predictor learned by solving (1) with SGD is the simplest. In other words, the model learned by default lies at one end of the space of solutions. A diverse set of solutions departing from the default one will necessarily include more complex models, that represent more complex decision boundaries and rely on different features of the data.

How to quantify diversity. We compare the functions implemented classifiers using their input gradients *i.e.* the gradient of their output with respect to their input. Given any two classifiers g_{ϕ_1} and g_{ϕ_2} we quantify their similarity at a point $\boldsymbol{h} \in \text{preimage}(g)$ with

$$\delta_{g_{\phi_1}, g_{\phi_2}}(\boldsymbol{h}) \;=\; \nabla_{\boldsymbol{h}}\, g^\star_{\phi_1}(\boldsymbol{h}) \,.\, \nabla_{\boldsymbol{h}}\, g^\star_{\phi_2}(\boldsymbol{h}) \tag{4}$$

where the dot product measures the alignment of gradients. Since g is vector-valued, we denote with $\nabla g^\star$ the gradient of its largest component (top predicted score). We apply (4) below to encourage diversity over a collection of models.

Complete proposed method. Instead of training one model, we train a collection of models $\{F_i\}$ in parallel, where $F_i = g_{\phi_i} \circ f$. They share an optional feature extractor f (*e.g.* a ResNet) for computational reasons, whereas the model-specific classifiers g_{ϕ_i} are small multi-layer perceptrons (MLPs) in our experiments. We replace the training objective (1) with

$$\min_{(\boldsymbol{\theta},\{\phi_i\})} \; \Sigma_i^n \; \mathcal{R}(g_{\phi_i} \circ f_{\boldsymbol{\theta}}) \;+\; \lambda \; \Sigma_{i \neq j} \Sigma_k^K \; \delta_{g_{\phi i}, g_{\phi j}}(\boldsymbol{h}^k) \tag{5}$$

where the scalar λ controls the strength of the regularizer. The first term is the ERM objective. It ensures a low training error and usual asymptotic guarantees for in-domain data. The second term is the diversity regularizer. It minimizes the alignment of input gradients over pairs of models at all training points. We solve (5) by SGD, with $\{\boldsymbol{\theta}_i\}$ initialized differently to break the initial symmetry.

3.1. Additional considerations

Rationale for input gradients. Intuitively, we want each model to rely on different features in the data. And input gradients are indicative of the features used by the model [69]. Input gradients have the advantage of being implementation-invariant [72] (applicable to any differentiable model) and directly relevant to OOD predictions. The predictions of a classifiersg_{ϕ_i} at a test point of features $\boldsymbol{h}$ are denoted $g_{\phi_i}(\boldsymbol{h})$. Assuming g continuous and differentiable, the approximation with a first-order Taylor expansion about a nearby training point of features $\boldsymbol{h}^{\text{tr}}$ gives

$$g_{\phi_i}(\boldsymbol{h}) \;=\; g_{\phi_i}(\boldsymbol{h}^{\text{tr}}) \;+\; (\boldsymbol{h} - \boldsymbol{h}^{\text{tr}})\, \nabla_{\boldsymbol{h}} g^\star_{\phi_i}(\boldsymbol{h}^{\text{tr}})\,. \tag{6}$$

The ERM objective clamps the value at training points which makes the first term identical $\forall\, i$. But the diversity regularizer makes the second term different, causing predictions to diverge more and more across models as one moves away from training points. Simple alternatives in weight space (*e.g.* pushing parameters apart) would not guarantee learning different functions, since two networks can be equivalent under permutations and scalings of weights.

Evaluating the diversity of learned predictors. Our method produces a collection models, one of which has to be selected for inference. The selection is necessary, just like the cross-validation routinely performed to select hyperparameters and architectures, or the ubiquitous practice of early stopping. The choice of a selection method (see below) is orthogonal to our contribution of alleviating the simplicity bias. Therefore, the goal of our experiments is to demonstrate an increase in diversity (in terms of OOD performance) of the models that the selection can then operate on. Therefore we report the mean, ensemble, and maximum accuracy (oracle selection) of all models of a collection. We also perform a cross-dataset evaluation (Section 4.3) to verify that the maximum accuracy is meaningful and not merely an example of "overfitting to the test set".

Model selection for OOD performance. It is important to remember that selection for OOD performance cannot, by definition, be performed with a validation set from the same distribution as the training data [74]. OOD generalization fundamentally requires additional information beyond i.i.d. data [6, 68] (see Section 2). Our approach uses standard i.i.d. data during training, which means that extra information has to be brought in during model selection. It can be as simple as a small OOD validation set, and other options include any technique used to evaluate OOD performance: contrast sets, counterfactual examples [16], inspection through with explainability techniques and expert knowledge [62], etc. This flexibility of options is possible because we only require the extra information for model selection, compared to existing OOD methods that require extra information attached to every training example *e.g.* in multiple training environments [1, 12, 56], counterfactual examples [25, 73], or non-stationary time series [23, 30, 58].

Computational cost. Feeding all classifiers with the same mini-batches keeps the training cost small. We found no difference with feeding different mini-batches. Memory and compute scale linearly with the number of classifiers, but these are small MLPs with a tiny footprint compared to the shared feature extractor. With enough memory, the operations of all models can even be parallelized, hence no decrease in throughput. The computation of the diversity regularizer reuses the input gradients that are byproducts of the backpropagation necessary to train the classifiers. The only added cost is in the computation of second derivatives to optimize the regularizer. All our experiments were run on a single laptop (!) with a GeForce GTX 1050 Ti GPU.

We include an FAQ with past reviews in Appendix A. See Appendix A for FAQs from readers and reviewers.

4. Experiments

We designed a set of experiments to answer two questions.

1. Can we learn predictive patterns otherwise ignored by standard SGD and existing regularizers ? (Section 4.1).
2. Are these patterns relevant for OOD generalization in computer vision tasks ? (Sections 4.2 and 4.3).

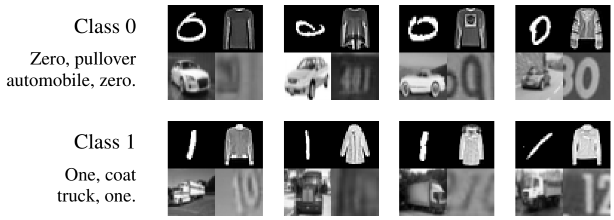


Figure 3. Training examples of collages. Each block features one of two pre-selected classes from MNIST, Fashion-MNIST, CIFAR-10, SVHN. All four blocks are predictive of training labels. Because of the simplicity bias, a standard classifier latches on MNIST and ignores others.

4.1. Multi-dataset collages

Dataset. We extend the collages of MNIST/CIFAR (see Figure 1a) used in previous investigations of the simplicity bias [70]. We also use Fashion-MNIST [82] and SVHN [50] to form four-block collages for a binary classification task (see Figure 3). Each block features one of two pre-selected classes from the corresponding dataset (details in Appendix E). In the training data, the contents of all four blocks are predictive of the labels. Because of the simplicity bias however, a standard model systematically relies on the MNIST digit and completely ignores other parts of the image. The dataset simulates the absence of prior preference for any image region, typical in vision tasks. Therefore **the goal is to learn predictive patterns from all four blocks**. This is evaluated with four test sets, in which the contents of all blocks but one are randomized to either of its two classes. The dataset available at https://github.com/dteney/collages-dataset.

Results of baselines. To test the simplest possible implementation of our method, we use a fully-connected 2-hidden layer MLP classifier (details in Appendix E). We obtained upper bounds on the accuracy attainable with this architecture with four training sets where all blocks but one are randomized (Table 1, top row). This provides a ranking of learning difficulty: MNIST, SVHN, Fashion-MNIST, CIFAR. We trained the baseline with popular regularizers. In all cases (32 models per experiment, repeated 5 times) the models use the MNIST digit exclusively and never perform above chance (50%) on the other test sets.

Results of our method. We then trained a collection of the same number of models (32) with our diversity regularizer. In every case (five runs) the collection contains models that use all four parts of the images. We determined that the models specialize but do not overlap: a model is typically good on one of the four test sets at a time (see Table 5 in the appendix). A larger number of models also seems beneficial (see discussion in Section 5). This is partly explained with the observation that, even with a large number of models, a larger fraction relies on the simpler MNIST and

Collages dataset (accuracy in %)	Best model on				
	MNIST	SVHN	Fashion-M.	CIFAR-10	Average
Upper bounds: one predictive block in tr.	99.7	89.7	77.4	68.7	83.9
Baseline, 32 models with different seeds	99.7 ±0.0	50.0 ±0.1	51.2 ±0.3	50.1 ±0.1	62.7
With dropout (best rate: 0.5)	98.7	54.8	52.9	54.9	65.3
With penalty on L1 norm of gradients	98.9	49.8	50.7	49.9	62.3
With Jacobian regularization [28]	98.8	49.8	50.7	49.9	62.3
With spectral decoupling [57]	99.1	49.8	50.7	49.9	62.4
Proposed, 8 models	**97.3** ±0.5	82.1 ±6.0	59.6 ±4.0	55.8 ±1.9	73.7
Proposed, 16 models	96.6 ±1.2	72.1 ±10.3	64.6 ±4.0	58.4 ±1.4	72.9
Proposed, 32 models	95.6 ±0.3	81.8 ±5.3	69.2 ±2.8	61.1 ±1.0	76.9
Proposed, 64 models	95.5 ±0.1	80.9 ±5.8	70.7 ±1.5	60.8 ±0.9	77.0
Proposed, 96 models	95.8 ±0.8	**84.7** ±4.0	**71.7** ±1.1	**61.7** ±1.2	**78.5**

Table 1. Results on collages. The upper bounds are obtained by training the baseline four times on data where all blocks but one are randomized. Other rows correspond to the training of 32 models, of which we report the best one on each test set. All existing methods fixate on the MNIST block. Ours discovers predictive signals from all four blocks (mean and std. dev. over five runs).

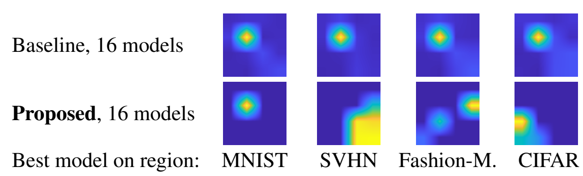


Figure 4. Visualization of input gradients $\nabla_{\boldsymbol{h}}\, g^{\star}(\boldsymbol{h})$ (abs. val. averaged over 10 test images; brighter means higher value). Each model specializes in different image regions.

Fashion-MNIST blocks than on the others. There is still room for improvement since our best models do not quite reach the upper bounds. Finally, a manual inspection of input gradients similar to some interpretability methods [69] is an easy way to assess which part of the image is used (see Figure 4). Combined with expert task knowledge, this could serve for model selection in some applications.

4.2. Biased activity recognition (BAR)

Figure 5. Training and test (in red) examples from BAR.

Dataset. The BAR dataset [48] was recently introduced

	Biased activity recognition (BAR) dataset		
Training collection of 64 models, reporting performance of:	**Single model** (average accuracy in the collection)	**Ensemble** (whole collection)	**Best single model** (oracle selection)
Baseline in [48]	51.9 ±5.92	N/A	N/A
Learning from failure [48]	63.0 ±2.76	N/A	N/A
Our strong baseline: frozen ResNet-50, 2-layer MLP	62.0 ±0.3	63.1 ±0.2	64.9 ±0.7
Penalty on sq. L2 norm of gradient (Jacobian reg. [28])	63.7 ±0.4	64.5 ±0.7	67.0 ±0.9
Penalty on sq. L2 norm of feature-gradient product (App. E)	62.8 ±0.1	63.9 ±0.6	65.9 ±0.5
Penalty on L1 norm of feature-gradient product (App. E)	63.9 ±0.3	64.6 ±0.4	66.1 ±0.3
Penalty on sq. L2 norm of logits (spectral decoupling [57])	64.3 ±0.2	65.2 ±0.5	67.0 ±0.4
Proposed, 8 models	**64.9** ±0.8	65.9 ±0.4	66.8 ±0.5
Proposed, 64 models	64.4 ±0.2	**66.1** ±0.3	**67.1** ±0.3

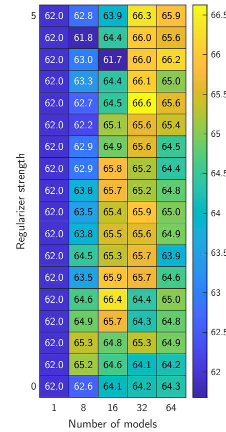


Table 2. Evaluation on BAR (mean accuracy and std. dev. over five runs). Each row corresponds to the training of 64 models, unless otherwise noted. The heat map (right; accuracy of best model) shows that the diversity regularizer clearly improves over a classical ensemble of independently-trained models (bottom row).

to evaluate debiasing methods for image recognition. The task is to classify photographs into six activities (see Figure 5). Training images were sampled from six pairs of actions/places in imSitu [85] *e.g.* climbing/rock. Test images were sampled from the same actions in different places *e.g.* climbing/ice. **The goal is to learn a model that relies more on a person's appearance than on the background** to recognize actions in arbitrary places. This is challenging because both are predictive of the action training labels. This represents an ubiquitous setting where training data is affected by selection biases but an image recognition model is still expected perform well on novel scenes.

Results. We follow [48] and implement a classifier on ResNet-50 features (details in Appendix E). We first tune a strong baseline classifier (almost equating the method in [48]). We then train a collection of these classifiers with our regularizer. The accuracy of the best model improves from 64.9 to 67.1 (Table 2). The *average* accuracy over the collection also increases, perhaps surprisingly. Indeed, an increase in diversity could induce as many worse models than better ones. But remember that the space of predictors ranked by complexity is one-sided (Section 3). The baseline is at the "simplest" end. Those learned with our method are more complex. With BAR, complex models happen to be better, as analyzed in [48]. Thanks to this, a simple ensemble (summing all predicted scores) improves over the same ensemble of models trained without our regularizer (63.1 → 66.1) with no model selection. We insist however that this is not a universal benefit of our method.

The BAR dataset contains no information to prefer backgrounds or persons' appearance. With the same training data and annotations, the task could as well be to recognize places rather than actions ! This reinforces our insistence that side information is necessary for OOD generalization. Debiasing methods like [48] rely on task-specific design choices. Our approach is of more general purpose.

The existing Jacobian regularizer [28] and spectral decoupling [57] produce models almost as good as our best one, although worse on average as seen with a lower *ensemble* accuracy. The advantage of our method is to produce a collection of diverse models, whereas any specific regularizer is either suited or not to the given task. If not, the practitioner has to manually find another one.

4.3. Domain generalization (PACS)

Dataset. PACS is a standard benchmark for visual domain generalization (DG) [36]. It contains images from seven classes and four visual domains: art paintings, cartoons, photographs, and sketches. It is used in a leave-one-out manner with three training domains and the remaining one for testing. The standard baseline uses images from all domains indistinctively whereas DG methods use domain labels to try and identify common features that should also be reliable in the test domain.

Results. We report an ablative study in Table 3 (see Appendix E for implementation details). We obtain a better model by training a collection with our regularizer, compared to the baseline where differences within the collection result only from different random seeds. The improvements are modest but they clearly result from our regularizer, as seen from a heatmap of accuracy as a function of the number of models and regularizer strength (also see Figure 6, left). A larger number of models also seem beneficial. We provide the training curves of all models in a collection in Figure 6. Looking at the OOD accuracy as training progresses, we see that the regularizer induces vastly more variance, producing both worse and better models as expected.

As observed with BAR, the average accuracy within a collection improves slightly with the regularizer – although not to the point that a naive ensemble would be beneficial.

Training set	PACS (cartoon, photo, sketch)			
Test set	PACS (art)			VLCS (horse/person AUC)
Model evaluated	Single	Ensemble	Best	Best model on PACS
Baseline, 64 models, no regularizer	84.48 $\pm$0.23	84.62	85.71	74.57
Penalty on sq. L2 norm of grad. (Jacobian reg. [28])	85.12 $\pm$0.33	**85.06**	85.84	74.10
Penalty on sq. L2 norm of ReLU of grad. (App. E)	84.62 $\pm$0.19	84.62	85.16	75.84
Penalty on sq. L2 norm of feature-grad. prod. (App. E)	84.61 $\pm$0.26	84.77	85.45	73.51
Penalty on L1 norm of grad. (App. E)	84.66 $\pm$0.45	84.67	86.13	76.29
Penalty on sq. L2 norm of logits (spectral dec. [57])	84.46 $\pm$0.32	84.81	85.16	74.60
Combination: proposed + spectral dec. [57]	84.31 $\pm$0.83	84.72	86.08	74.51
Proposed, 64 models	**85.14** $\pm$0.59	84.62	**86.80**	**79.66**

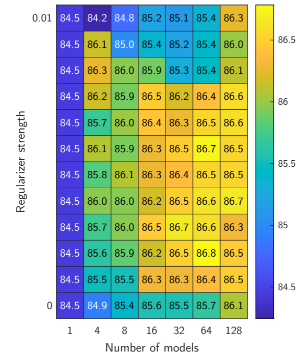


Table 3. (Left) Ablative evaluation on PACS (mean accuracy and std. dev. over five runs). We also evaluate models selected on PACS on test data from VLCS. This confirms that these models do indeed generalize better. (Right) This heatmap (accuracy on PACS, best model) clearly shows improvements over independently-trained models (bottom row).

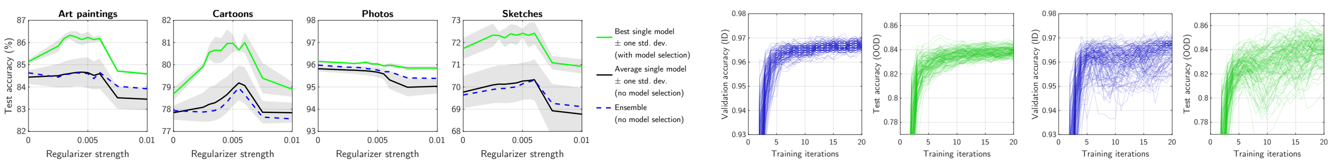


Figure 6. (Left) Accuracy on each test style of PACS as a function of the regularizer strength. (Right) Training curves of in-domain and OOD accuracy of 64 models, either trained independently (left two plots) or with our regularizer (right twos).

This corroborates the explanations given above for BAR.

We include an additional cross-dataset evaluation *i.e.* zero-shot transfer (Table 3, last column). We use the test images from VLCS [19] for a detection task of classes both in PACS and VLCS (*horse* and *person*; other VLCS classes serve as negatives). We report the area under curve (AUC) averaged over these two classes. We observe that the best model **selected on PACS** also brings substantial improvements on VLCS. This verifies that the accuracy of the best model on PACS is meaningful and not merely an example of "overfitting to the test set".

We evaluate regularizers previously proposed to improve generalization. We spent substantial effort tuning these alternatives to their best and trying multiple variants (see Appendix E for details). As seen in Table 3, a Jacobian regularizer provides small improvements. So does minimizing the squared L2 norm of the gradient. Our regularizer mainly minimizes the alignment between pairs of gradients but it can also reduce their absolute norm as a side effect, just like these two alternatives. The ablation shows this to be beneficial but also that it does not entirely explain the benefits of our method. Indeed on VLCS, only our regularizer provides a substantial improvement (74.57 $\rightarrow$ 79.66).

We provide a comparison with state-of-the-art DG methods in Table 4. Most methods rely on labels of training domains, which we do not use. Most were demonstrated on top of weak baselines, as pointed out in [21]. We use the same ResNet-18 feature extractor, we highly optimized it, and we still get substantial improvements. See Section 5 for more discussion about the comparison with DG methods.

5. Discussion

The above results show that (1) we now have a tool to expand the set of solutions learned by a neural network through SGD and (2) some solutions are relevant to computer vision as evidenced by improved OOD generalization.

Limitations of the method. The main hyperparameters are the regularizer strength and the number of models learned. Any number $>$1 obviously gives more options than the single model learned by default. Empirically, larger numbers ($>$64) seem beneficial, but we did not derive guarantees that a robust predictor will be found. The chosen number of models may not induce the optimal "granularity". Too small a number could produce a model that relies on multiple entangled features including robust and spurious ones. And too large a number could produce multiple models that each implement a different part of a robust solution.

An interesting direction for future work would be to encourage learning a "basis" of elemental predictors (suggested in [63]) for a given dataset by promoting notions of sparsity or complementarity. The model selection could thereafter search for their optimal combination as attempted in [53]. The approach would resemble the disentanglement methods of representation learning [39] but it would operate in the space of predictors optimized for ERM, rather than the space of features optimized for reconstruction.

The need for model selection. The utility of evading the simplicity bias is in improving OOD performance because this is where the simplicity bias shows its detrimen-

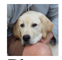
PACS Dataset Test style (leave-one-out)	 Art	 Cartoon	 Photo	 Sketch	Avg.
D-SAM baseline [15]	77.9	75.9	95.2	69.3	79.6
D-SAM*	77.3	72.4	95.3	77.8	80.7
Epi-FCR baseline [37]	77.6	73.9	94.4	74.3	79.1
Epi-FCR*	82.1	77.0	93.9	73.0	81.5
DMG baseline [9]	72.6	78.5	93.2	65.2	77.4
DMG*	76.9	80.4	93.4	75.2	81.5
DecAug baseline [4]	78.4	78.3	94.2	72.1	80.8
DecAug*	79.0	79.6	95.3	75.6	82.4
JiGen baseline [8]	77.9	74.9	95.7	67.7	79.1
JiGen	79.4	75.3	96.0	71.4	80.5
Latent domains baseline [44]	78.3	75.0	96.2	65.2	78.7
Latent domains	81.3	77.2	96.1	72.3	81.8
Our baseline, 64 independent models					
Random single model	84.4 ± 0.3	77.8 ± 0.3	95.8 ± 0.1	69.8 ± 0.7	82.0
Ensemble of all models	84.6 ± 0.1	77.9 ± 0.1	96.0 ± 0.1	69.6 ± 0.2	82.0
Best single model	85.1 ± 0.1	78.7 ± 0.3	**96.2** ± 0.1	71.7 ± 0.5	82.9
Proposed, 64 models					
Random single model	85.2 ± 0.6	79.6 ± 0.7	95.9 ± 0.1	70.8 ± 0.8	82.9
Ensemble of all models	84.8 ± 0.1	79.0 ± 0.1	96.0 ± 0.1	70.3 ± 0.1	82.5
Best single model	**86.5** ± 0.1	**81.1** ± 0.4	**96.2** ± 0.4	**72.8** ± 0.2	**84.2**

Table 4. Comparison with existing methods on PACS. For each method, we mention the accuracy of the baseline reported by its authors, and of the method itself. All methods are based on a ResNet-18. *Require labels of tr. domains. Our method discovers additional predictive features in the data. It returns a collection of models, among which some clearly have a better OOD performance (last row) than the baseline simply trained with different random seeds (82.9 $\rightarrow$ 84.2).

tal effects (high ID performance is achievable with less sophistication). The key novelty is to require no extra information **during training** contrary to existing methods that require upfront task-specific knowledge (*e.g.* debiasing methods [3, 11, 41, 48, 71, 77]) or additional annotations [1, 12, 56, 73]. The requirement for side information (besides a training set of i.i.d. examples) is fundamental for OOD performance (see [6, 68] and additional background in Appendix B) but we defer its use to an independent model selection step. This allows more flexibility in the type and quantity of side information used.

Limitations of the evaluation. In a review on domain generalization (*i.e.* OOD generalization across visual styles), Gulrajani and Lopez-Pas [21] note that model fitting and model selection are equally hard. They recommend that methods for datasets like PACS (Section 4.3) include a selection strategy. On the opposite, we show that the two steps can be completely decoupled. Our analysis focuses on the best learned model (denoted *oracle selection* in [21] and *best* in [53]). This optimistic choice is justified because it is the performance achievable with an optimal selection strategy. **This accounts for existing and future selection strategies** *e.g.* the calibration-based method of Wald *et al.* [80] that came out after the writing of this paper.

Is the comparison unfair with methods for PACS that train a single model ? We do not think so: existing methods were selected at the *paper* level. Methods with no improvement on the test set did not get published. It is unlikely that authors never peeked at test-set performance until getting published ! It was even showed in [21] that all tested methods lost their benefits when deprived of heavy hyperparameter tuning and early stopping based on OOD performance. Our approach makes the selection more explicit.

Heuristics about simplicity. Our improvements support previous claims [51, 70] that more complex models are sometimes preferable. Works in NLP have even argued that any simple correlation in a dataset is likely to be spurious [11, 78, 88]. We point out that such heuristics are necessarily task- or dataset-specific. However, we also remark that the tasks addressed with deep learning are increasingly complex (take visual question answering for example [75]). This implies that the space of potentially-spurious patterns that are simpler than the task in any given dataset is also growing. The above heuristics may therefore have a practical utility. It remains crucial to study their limits of applicability. They cannot be universal [45] and cannot obviate the need for extra information (or task-specific knowledge) in our model selection step.

Universality of inductive biases. The inductive biases of any learning algorithm cannot be universally superior to another's [81]. For example, weight decay, Jacobian regularization, or even data augmentation are only as good as they are tuned to a particular task. In comparison, our method does not affect inductive biases in a directed way. It only increases the variety of the learned models, so it could be seen as a "meta-regularizer". Our results also show that intuitive notions behind classical regularizers like smoothness (Jacobian regularization), sparsity (L1 norm), or simplicity (L2 norm) are sometimes detrimental.

Any quest for *universal* architectures, regularizers, data augmentations, or even dataless selection strategies [46, 88] is known to be futile. Benefits can only apply to a subset of learning tasks [81]. Questions remain: how small or vast is the subset of learning tasks that humans care about ? Which properties of naturally-produced data make some inductive biases *generally* useful ? Deep learning has proven surprisingly successful. Studying its inductive biases will help understanding its limits of applicability. And methods to control these biases will help expanding these limits.

References

[1] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019. 2, 3, 4, 8, 14, 17

[2] Alexander Bagnall, Razvan Bunescu, and Gordon Stewart. Training ensembles to detect adversarial examples. *arXiv preprint arXiv:1712.04006*, 2017. 18

[3] Hyojin Bahng, Sanghyuk Chun, Sangdoo Yun, Jaegul Choo, and Seong Joon Oh. Learning de-biased representations with biased representations. In *International Conference on Machine Learning*, pages 528–539. PMLR, 2020. 3, 8, 15, 17

[4] Haoyue Bai, Rui Sun, Lanqing Hong, Fengwei Zhou, Nanyang Ye, Han-Jia Ye, S-H Gary Chan, and Zhenguo Li. Decaug: Out-of-distribution generalization via decomposed feature representation and semantic augmentation. *arXiv preprint arXiv:2012.09382*, 2020. 8

[5] Nicholas Baker, Hongjing Lu, Gennady Erlikhman, and Philip J Kellman. Deep convolutional networks do not classify based on global object shape. *PLoS computational biology*, 14(12):e1006613, 2018. 3, 17

[6] Elias Bareinboim, JD Correa, Duligur Ibeling, and Thomas Icard. On pearl's hierarchy and the foundations of causal inference. *ACM Special Volume in Honor of Judea Pearl (provisional title)*, 2020. 2, 3, 4, 8

[7] Remi Cadene, Corentin Dancette, Hedi Ben-younes, Matthieu Cord, and Devi Parikh. Rubi: Reducing unimodal biases in visual question answering. *arXiv preprint arXiv:1906.10169*, 2019. 15

[8] Fabio M Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain generalization by solving jigsaw puzzles. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2019. 8

[9] Prithvijit Chattopadhyay, Yogesh Balaji, and Judy Hoffman. Learning to balance specificity and invariance for in and out of domain generalization. In *Proc. Eur. Conf. Comp. Vis.*, 2020. 8, 18

[10] Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. Don't take the easy way out: Ensemble based methods for avoiding known dataset biases. *arXiv preprint arXiv:1909.03683*, 2019. 15

[11] Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. Learning to model and ignore dataset bias with mixed capacity ensembles. *arXiv preprint arXiv:2011.03856*, 2020. 3, 8, 15, 17

[12] Anton van den Hengel Damien Teney, Ehsan Abbasnejad. Unshuffling data for improved generalization. *arXiv preprint arXiv:2002.11894*, 2020. 2, 3, 4, 8

[13] Harris Drucker and Yann LeCun. Improving generalization performance using double backpropagation. *IEEE Transactions on Neural Networks*, 1992. 18

[14] Yunshu Du, Wojciech M Czarnecki, Siddhant M Jayakumar, Mehrdad Farajtabar, Razvan Pascanu, and Balaji Lakshminarayanan. Adapting auxiliary losses using gradient similarity. *arXiv preprint arXiv:1812.02224*, 2018. 15, 17

[15] Antonio D'Innocente and Barbara Caputo. Domain generalization with domain-specific aggregation modules. In *German Conference on Pattern Recognition*. Springer, 2018. 8

[16] Matt Gardner, Yoav Artzi, Victoria Basmova, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, et al. Evaluating NLP models via contrast sets. *arXiv preprint arXiv:2004.02709*, 2020. 4, 15, 16

[17] 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, 2020. 1, 2

[18] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*, 2018. 3, 17

[19] Muhammad Ghifary, W Bastiaan Kleijn, Mengjie Zhang, and David Balduzzi. Domain generalization for object recognition with multi-task autoencoders. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2015. 7

[20] Kyle Gorman and Steven Bedrick. We need to talk about standard splits. In *Proc. Conf. Association for Computational Linguistics*, pages 2786–2791, 2019. 16

[21] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*, 2020. 7, 8, 16, 17, 19

[22] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182, 2003. 17

[23] Hermanni Hälvä and Aapo Hyvarinen. Hidden markov nonlinear ica: Unsupervised learning from nonstationary time series. In *Conference on Uncertainty in Artificial Intelligence*, pages 939–948. PMLR, 2020. 2, 3, 4, 17

[24] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data min-*

ing, inference, and prediction. Springer Science & Business Media, 2009. 1

[25] Christina Heinze-Deml and Nicolai Meinshausen. Conditional variance penalties and domain shift robustness. *arXiv preprint arXiv:1710.11469*, 2017. 2, 3, 4, 17

[26] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019. 2, 16

[27] Katherine L Hermann and Andrew K Lampinen. What shapes feature representations? exploring datasets, architectures, and training. *arXiv preprint arXiv:2006.12433*, 2020. 1

[28] Judy Hoffman, Daniel A Roberts, and Sho Yaida. Robust learning with jacobian regularization. *arXiv preprint arXiv:1908.02729*, 2019. 5, 6, 7, 18, 19

[29] Wei-Chih Hung, Varun Jampani, Sifei Liu, Pavlo Molchanov, Ming-Hsuan Yang, and Jan Kautz. Scops: Self-supervised co-part segmentation. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2019. 17

[30] Aapo Hyvarinen and Hiroshi Morioka. Nonlinear ica of temporally dependent stationary sources. In *Artificial Intelligence and Statistics*, pages 460–469. PMLR, 2017. 2, 3, 4, 17

[31] Chenchen Jing, Yuwei Wu, Xiaoxun Zhang, Yunde Jia, and Qi Wu. Overcoming language priors in VQA via decomposed linguistic representations. In *Proc. Conf. AAAI*, 2020. 15

[32] Sanjay Kariyappa and Moinuddin K Qureshi. Improving adversarial robustness of ensembles with diversity training. *arXiv preprint arXiv:1901.09981*, 2019. 2, 15, 17, 18

[33] David Klindt, Lukas Schott, Yash Sharma, Ivan Ustyuzhaninov, Wieland Brendel, Matthias Bethge, and Dylan Paiton. Towards nonlinear disentanglement in natural data with temporal sparse coding. *arXiv preprint arXiv:2007.10930*, 2020. 15

[34] 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). *arXiv preprint arXiv:2003.00688*, 2020. 1

[35] Sebastian Lapuschkin, Stephan Wäldchen, Alexander Binder, Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Unmasking clever hans predictors and assessing what machines really learn. *Nature communications*, 10(1):1–8, 2019. 2

[36] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pages 5542–5550, 2017. 2, 6

[37] Da Li, Jianshu Zhang, Yongxin Yang, Cong Liu, Yi-Zhe Song, and Timothy M Hospedales. Episodic training for domain generalization. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2019. 8

[38] Etai Littwin and Lior Wolf. The multiverse loss for robust transfer learning. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2016. 18

[39] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, pages 4114–4124. PMLR, 2019. 7

[40] David Lopez-Paz, Robert Nishihara, Soumith Chintala, Bernhard Scholkopf, and Léon Bottou. Discovering causal signals in images. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2017. 15

[41] Rabeeh Karimi Mahabadi and James Henderson. Simple but effective techniques to reduce biases. *arXiv preprint arXiv:1909.06321*, 2019. 3, 8, 15, 17

[42] Itzik Malkiel and Lior Wolf. Maximal multiverse learning for promoting cross-task generalization of fine-tuned language models. 2021. 18

[43] Zelda Mariet and Suvrit Sra. Diversity networks: Neural network compression using determinantal point processes. *arXiv preprint arXiv:1511.05077*, 2015. 17

[44] Toshihiko Matsuura and Tatsuya Harada. Domain generalization using a mixture of multiple latent domains. In *Proc. Conf. AAAI*, 2020. 8

[45] Tom M Mitchell. *The need for biases in learning generalizations*. Rutgers University, 1980. 1, 8

[46] Calvin Murdock and Simon Lucey. Dataless model selection with the deep frame potential. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pages 11257–11265, 2020. 8

[47] Preetum Nakkiran, Gal Kaplun, Dimitris Kalimeris, Tristan Yang, Benjamin L Edelman, Fred Zhang, and Boaz Barak. SGD on neural networks learns functions of increasing complexity. *arXiv preprint arXiv:1905.11604*, 2019. 1

[48] Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. Learning from failure: Training debiased classifier from biased classifier. In *Proc. Advances in Neural Inf. Process. Syst.*, 2020. 2, 3, 5, 6, 8, 17, 19, 20

[49] Vedant Nanda, Samuel Dooley, Sahil Singla, Soheil Feizi, and John P Dickerson. Fairness through robustness: Investigating robustness disparity in deep learning. *arXiv preprint arXiv:2006.12621*, 2020. 2, 16

[50] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011. 5

[51] Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. In search of the real inductive bias: On the role of implicit regularization in deep learning. *arXiv preprint arXiv:1412.6614*, 2014. 1, 2, 8

[52] Guillermo Ortiz-Jimenez, Itamar Franco Salazar-Reque, Apostolos Modas, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard. A neural anisotropic view of underspecification in deep learning. In *Proc. Int. Conf. Learn. Representations*, 2021. 15

[53] Daniel Pace, Alessandra Russo, and Murray Shanahan. Learning diverse representations for fast adaptation to distribution shift. *arXiv preprint arXiv:2006.07119*, 2020. 7, 8, 17

[54] Tianyu Pang, Kun Xu, Chao Du, Ning Chen, and Jun Zhu. Improving adversarial robustness via promoting ensemble diversity. In *International Conference on Machine Learning*, pages 4970–4979. PMLR, 2019. 18

[55] Giambattista Parascandolo, Alexander Neitz, Antonio Orvieto, Luigi Gresele, and Bernhard Schölkopf. Learning explanations that are hard to vary. *arXiv preprint arXiv:2009.00329*, 2020. 16

[56] Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2016. 2, 3, 4, 8, 17

[57] Mohammad Pezeshki, Sékou-Oumar Kaba, Yoshua Bengio, Aaron Courville, Doina Precup, and Guillaume Lajoie. Gradient starvation: A learning proclivity in neural networks. *arXiv preprint arXiv:2011.09468*, 2020. 1, 2, 5, 6, 7, 19

[58] Niklas Pfister, Peter Bühlmann, and Jonas Peters. Invariant causal prediction for sequential data. *Journal of the American Statistical Association*, 114(527):1264–1276, 2019. 2, 3, 4, 17

[59] Fengchun Qiao, Long Zhao, and Xi Peng. Learning to learn single domain generalization. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2020. 17

[60] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021. 16

[61] Neale Ratzlaff and Li Fuxin. HyperGAN: A generative model for diverse, performant neural networks. In *Proc. Int. Conf. Mach. Learn.*, 2019. 18

[62] Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond accuracy: Behavioral testing of NLP models with checklist. *arXiv preprint arXiv:2005.04118*, 2020. 4

[63] Andrew Ross, Weiwei Pan, Leo Celi, and Finale Doshi-Velez. Ensembles of locally independent prediction models. In *Proc. Conf. AAAI*, 2020. 7, 18

[64] Andrew Slavin Ross, Michael C Hughes, and Finale Doshi-Velez. Right for the right reasons: Training differentiable models by constraining their explanations. *arXiv preprint arXiv:1703.03717*, 2017. 18

[65] Andrew Slavin Ross, Weiwei Pan, and Finale Doshi-Velez. Learning qualitatively diverse and interpretable rules for classification. *arXiv preprint arXiv:1806.08716*, 2018. 14, 18

[66] Shiori Sagawa, Aditi Raghunathan, Pang Wei Koh, and Percy Liang. An investigation of why overparameterization exacerbates spurious correlations. In *Proc. Int. Conf. Mach. Learn.*, 2020. 2

[67] Victor Sanh, Thomas Wolf, Yonatan Belinkov, and Alexander M Rush. Learning from others' mistakes: Avoiding dataset biases without modeling them. *arXiv preprint arXiv:2012.01300*, 2020. 17

[68] Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 2021. 2, 3, 4, 8, 15

[69] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2017. 4, 5, 19

[70] Harshay Shah, Kaustav Tamuly, Aditi Raghunathan, Prateek Jain, and Praneeth Netrapalli. The pitfalls of simplicity bias in neural networks. *arXiv preprint arXiv:2006.07710*, 2020. 1, 2, 3, 5, 8, 15, 16, 18

[71] Joe Stacey, Pasquale Minervini, Haim Dubossarsky, Sebastian Riedel, and Tim Rocktäschel. Avoiding the hypothesis-only bias in natural language inference via ensemble adversarial training. In *Proc. Conf. Empirical Methods in Natural Language Processing*, volume 2020. 3, 8, 15, 17

[72] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proc. Int. Conf. Mach. Learn.*, 2017. 4

[73] Damien Teney, Ehsan Abbasnedjad, and Anton van den Hengel. Learning what makes a difference from counterfactual examples and gradient supervision. *arXiv preprint arXiv:2004.09034*, 2020. 2, 3, 4, 8, 17

[74] Damien Teney, Kushal Kafle, Robik Shrestha, Ehsan Abbasnejad, Christopher Kanan, and Anton van den Hengel. On the value of out-of-distribution testing: An example of goodhart's law. In *Proc. Advances in Neural Inf. Process. Syst.*, 2020. 4

[75] Damien Teney, Qi Wu, and Anton van den Hengel. Visual question answering: A tutorial. *IEEE Signal Processing Magazine*, 34:63–75, 2017. 8

[76] Antonio Torralba, Alexei A Efros, et al. Unbiased look at dataset bias. In *CVPR*, volume 1, page 7, 2011. 2, 16

[77] Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. Mind the trade-off: Debiasing nlu models without degrading the in-distribution performance. *arXiv preprint arXiv:2005.00315*, 2020. 3, 8, 15, 17

[78] Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. Towards debiasing nlu models from unknown biases. *arXiv preprint arXiv:2009.12303*, 2020. 8, 17

[79] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. In *Proc. Advances in Neural Inf. Process. Syst.*, 2018. 17

[80] Yoav Wald, Amir Feder, Daniel Greenfeld, and Uri Shalit. On calibration and out-of-domain generalization. *arXiv preprint arXiv:2102.10395*, 2021. 8, 15

[81] David H Wolpert. The supervised learning no-free-lunch theorems. *Soft computing and industry*, pages 25–42, 2002. 8

[82] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017. 5

[83] Kai Xiao, Logan Engstrom, Andrew Ilyas, and Aleksander Madry. Noise or signal: The role of image backgrounds in object recognition. *arXiv preprint arXiv:2006.09994*, 2020. 1

[84] Pengtao Xie, Yuntian Deng, and Eric Xing. On the generalization error bounds of neural networks under diversity-inducing mutual angular regularization. *arXiv preprint arXiv:1511.07110*, 2015. 18

[85] Mark Yatskar, Luke Zettlemoyer, and Ali Farhadi. Situation recognition: Visual semantic role labeling for image understanding. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2016. 6

[86] Yang Yu, Yu-Feng Li, and Zhi-Hua Zhou. Diversity regularized machine. In *Proc. Int. Joint Conf. Artificial Intell.*, 2011. 18

[87] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016. 1

[88] Jie M Zhang, Mark Harman, Benjamin Guedj, Earl T Barr, and John Shawe-Taylor. Perturbation validation: A new heuristic to validate machine learning models. *arXiv preprint arXiv:1905.10201*, 2019. 8