Interpretations are Useful: Penalizing Explanations to Align Neural Networks with Prior Knowledge

Laura Rieger [1] Chandan Singh [2] W. James Murdoch [3] Bin Yu [2 3]

Abstract

For an explanation of a deep learning model to be effective, it must both provide insight into a model and suggest a corresponding action in order to achieve an objective. Too often, the litany of proposed explainable deep learning methods stop at the first step, providing practitioners with insight into a model, but no way to act on it. In this paper we propose contextual decomposition explanation penalization (CDEP), a method that enables practitioners to leverage explanations to improve the performance of a deep learning model. In particular, CDEP enables inserting domain knowledge into a model to ignore spurious correlations, correct errors, and generalize to different types of dataset shifts. We demonstrate the ability of CDEP to increase performance on an array of toy and real datasets.

1. Introduction

In recent years, deep neural networks (DNNs) have demonstrated strong predictive performance across a wide variety of settings. However, in order to predict accurately, they sometimes latch onto spurious correlations caused by dataset bias or overfitting (Winkler et al., 2019). Moreover, DNNs are also known to exploit bias regarding gender, race, and other sensitive attributes present in training datasets (Garg et al., 2018; Obermeyer et al., 2019; Dressel & Farid, 2018). Recent work in explaining DNN predictions (Murdoch et al., 2019; Doshi-Velez & Kim, 2017) has demonstrated an ability to reveal the relationships learned by a model. Here, we extend this line of work to not only uncover learned relationships, but penalize them to improve a model.

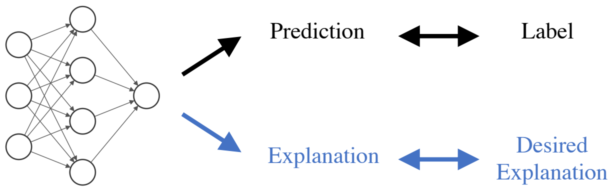

Figure 1. CDEP allows practitioners to penalize both a model's prediction and the corresponding explanation.

We introduce contextual decomposition explanation penalization (CDEP), a method which leverages a particular existing explanation technique for neural networks to enable the insertion of domain knowledge into a model. Given prior knowledge in the form of importance scores, CDEP works by allowing the user to directly penalize importances of certain features or feature interactions. This forces the neural network to not only produce the correct prediction, but also the correct explanation for that prediction.[1]

While we focus on the use of contextual decomposition, which allows the penalization of both feature importances and interactions (Murdoch et al., 2018; Singh et al., 2018), CDEP can be readily adapted for existing interpretation techniques, as long as they are differentiable. Moreover, CDEP is a general technique, which can be applied to arbitrary neural network architectures, and is often orders of magnitude faster and more memory efficient than recent gradient-based methods, allowing its use on meaningful datasets.

We demonstrate the effectiveness of CDEP via experiments across a wide array of tasks. In the prediction of skin cancer from images, CDEP improves the prediction of a classifier by teaching it to ignore spurious confounders present in the training data.

[1]DTU Compute, Technical University Denmark, 2800 Kgs. Lyngby, Denmark [2]EECS Department, UC Berkeley, Berkeley, California, USA [3]Department of Statistics, UC Berkeley, Berkeley, California, USA. Correspondence to: Laura Rieger <lauri@dtu.dk>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020. Copyright 2020 by the author(s).

[1] Code, notebooks, scripts, documentation, and models for reproducing experiments here and using CDEP on new models available at `https://github.com/laura-rieger/deep-explanation-penalization`.

In a variant of the MNIST digit-classification task where the digit's color is used as a misleading signal, CDEP regularizes a network to focus on a digit's shape rather than its color. Finally, simple examples show how CDEP can help mitigate fairness issues, both in text classification and risk prediction.

2. Background

Explanation methods Many methods have been developed to help explain the learned relationships contained in a DNN. For local or prediction-level explanation, most prior work has focused on assigning importance to individual features, such as pixels in an image or words in a document. There are several methods that give feature-level importance for different architectures. They can be categorized as gradient-based (Springenberg et al., 2014; Sundararajan et al., 2017; Selvaraju et al., 2016; Baehrens et al., 2010; Rieger & Hansen, 2019), decomposition-based (Murdoch & Szlam, 2017; Shrikumar et al., 2016; Bach et al., 2015) and others (Dabkowski & Gal, 2017; Fong & Vedaldi, 2017; Ribeiro et al., 2016; Zintgraf et al., 2017), with many similarities among the methods (Ancona et al., 2018; Lundberg & Lee, 2017). However, many of these methods have been poorly evaluated so far (Adebayo et al., 2018; Nie et al., 2018), casting doubt on their usefulness in practice. Another line of work, which we build upon, has focused on uncovering interactions between features (Murdoch et al., 2018), and using those interactions to create a hierarchy of features displaying the model's prediction process (Singh et al., 2019; 2020).

Uses of explanation methods While much work has been put into developing methods for explaining DNNs, relatively little work has explored the potential to use these explanations to help build a better model. Some recent work proposes forcing models to attend to regions of the input which are known to be important (Burns et al., 2018; Mitsuhara et al., 2019), although it is important to note that attention is often not the same as explanation (Jain & Wallace, 2019).

An alternative line of work proposes penalizing the gradients of a neural network to match human-provided binary annotations and shows the possibility to improve performance (Ross et al., 2017; Bao et al., 2018; Du et al., 2019) and adversarial robustness (Ross & Doshi-Velez, 2018). Two recent papers extend these ideas by penalizing gradient-based attributions for natural language models (Liu & Avci, 2019) and to produce smooth attributions (Erion et al., 2019). Du et al. (2019) applies a similar idea to improve image segmentation by incorporating attention maps into the training process.

Predating deep learning, Zaidan et al. (2007) consider the use of "annotator rationales" in sentiment analysis to train support vector machines. This work on annotator rationales was recently extended to show improved explanations (not accuracy) in particular types of CNNs (Strout et al., 2019).

Other ways to constrain DNNs While we focus on the use of explanations to constrain the relationships learned by neural networks, other approaches for constraining neural networks have also been proposed. A computationally intensive alternative is to augment the dataset in order to prevent the model from learning undesirable relationships, through domain knowledge (Bolukbasi et al., 2016), projecting out superficial statistics (Wang et al., 2019) or dramatically altering training images (Geirhos et al., 2018). However, these processes are often not feasible, either due to their computational cost or the difficulty of constructing such an augmented data set. Adversarial training has also been explored (Zhang & Zhu, 2019). These techniques are generally limited, as they are often tied to particular datasets, and do not provide a clear link between learning about a model's learned relationships through explanations, and subsequently correcting them.

3. Methods

In the following, we will first establish the general form of the augmented loss function. We then describe Contextual Decomposition (CD), the explanation method proposed by (Murdoch et al., 2018). Based on this, we introduce CDEP and point out its desirable computational properties for regularization. In Section 3.4 we describe how prior knowledge can be encoded into explanations and give examples of typical use cases. While we focus on CD scores, which allow the penalization of interactions between features in addition to features themselves, our approach readily generalizes to other interpretation techniques, as long as they are differentiable.

3.1. Augmenting the loss function

Given a particular classification task, we want to teach a model to not only produce the correct prediction but also to arrive at the prediction for the correct reasons. That is, we want the model to be right for the right reasons, where the right reasons are provided by the user and are dataset-dependent. Assuming a truthful explanation method, this implies that the explanation provided by the DNN for a particular decision should be aligned with a pre-supplied explanation encoding our knowledge of the underlying reasons.

To accomplish this, we augment the traditional objective function used to train a neural network, as displayed in Eq 1 with an additional component. In addition to the standard prediction loss $\mathcal{L}$, which teaches the model to produce the correct predictions by penalizing wrong predictions, we add an explanation error $\mathcal{L}_{\text{expl}}$, which teaches the model to produce the correct explanations for its predictions by penalizing wrong explanations.

In place of the prediction and labels $f_\theta(X), y$, used in the prediction error $\mathcal{L}$, the explanation error $\mathcal{L}_{\text{expl}}$ uses the explanations produced by an interpretation method $\text{expl}_\theta(X)$, along with targets provided by the user expl_X. As is common with penalization, the two losses are weighted by a hyperparameter $\lambda \in \mathbb{R}$:

$$\begin{aligned}\hat{\theta} = \underset{\theta}{\text{argmin}} & \overbrace{\mathcal{L}\left(f_\theta(X), y\right)}^{\text{Prediction error}} \\ & +\lambda \underbrace{\mathcal{L}_{\text{expl}}\left(\text{expl}_\theta(X), \text{expl}_X\right)}_{\text{Explanation error}}\end{aligned} \tag{1}$$

The precise meaning of expl_X depend on the context. For example, in the skin cancer image classification task described in Section 4, many of the benign skin images contain band-aids, while none of the malignant images do. To force the model to ignore the band-aids in making their prediction, in each image $\text{expl}_\theta(X)$ denotes the importance score of the band-aid and expl_X would be zero. These and more examples are further explored in Section 4.

3.2. Contextual decomposition (CD)

In this work, we use the CD score as the explanation function. In contrast to other interpretation methods, which focus on feature importances, CD also captures interactions between features, making it particularly suited to regularize the importance of complex features.

CD was originally designed for LSTMs (Murdoch et al., 2018) and subsequently extended to convolutional neural networks and arbitrary DNNs (Singh et al., 2018). For a given DNN $f(x)$, one can represent its output as a SoftMax operation applied to logits $g(x)$. These logits, in turn, are the composition of L layers g_i, such as convolutional operations or ReLU non-linearities.

$$\begin{aligned} f(x) &= \text{SoftMax}(g(x)) \qquad (2)\\ &= \text{SoftMax}(g_L(g_{L-1}(...(g_2(g_1(x)))))) \qquad (3)\end{aligned}$$

Given a group of features $\{x_j\}_{j \in S}$, the CD algorithm, $g^{CD}(x)$, decomposes the logits $g(x)$ into a sum of two terms, $\beta(x)$ and $\gamma(x)$. $\beta(x)$ is the importance score of the feature group $\{x_j\}_{j\in S}$, and $\gamma(x)$ captures contributions to $g(x)$ not included in $\beta(x)$. The decomposition is computed by iteratively applying decompositions $g_i^{CD}(x)$ for each of the layers $g_i(x)$.

$$\begin{aligned} g^{CD}(x) &= g_L^{CD}(g_{L-1}^{CD}(...(g_2^{CD}(g_1^{CD}(x))))) \qquad (4)\\ &= (\beta(x), \gamma(x)) \qquad (5)\\ &= g(x) \qquad (6)\end{aligned}$$

3.3. CDEP objective function

We now substitute the above CD scores into the generic equation in Eq 1 to arrive at CDEP as it is used in this paper. While we use CD for the explanation method $\text{expl}_\theta(X)$, other explanation methods could be readily substituted at this stage. In order to convert CD scores to probabilities, we apply a SoftMax operation to $g^{CD}(x)$, allowing for easier comparison with the user-provided labels expl_X. We collect from the user, for each input x_i, a collection of feature groups $x_{i,S}$, $x_i \in \mathbb{R}^d, S \subseteq \{1, ..., d\}$, along with explanation target values $\text{expl}_{x_{i,S}}$, and use the $\|\cdot\|_1$ loss for $\mathcal{L}_{\text{expl}}$.

This yields a vector $\beta(x_j)$ for any subset of features in an input x_j which we would like to penalize. We can then collect ground-truth label explanations for this subset of features, expl_{x_j} and use it to regularize the explanation. Using this we arrive at the equation for the weight parameters with CDEP loss:

$$\begin{aligned}\hat{\theta} = \underset{\theta}{\text{argmin}} & \overbrace{\sum_i \sum_c -y_{i,c} \log f_\theta(x_i)_c}^{\text{Prediction error}} \\ & +\lambda \underbrace{\sum_i \sum_S ||\beta(x_{i,S}) - \text{expl}_{x_{i,S}}||_1}_{\text{Explanation error}}\end{aligned} \tag{7}$$

In the above, i indexes each individual example in the dataset, S indexes a subset of the features for which we penalize their explanations, and c sums over each class.

Updating the model parameters in accordance with this formulation ensures that the model not only predicts the right output but also does so for the right (aligned with prior knowledge) reasons. It is important to note that the evaluation of what the right reasons are depends entirely on the practitioner deploying the model. As with the class labels, using wrong or biased explanations will yield a wrong and biased model.

3.4. Encoding domain knowledge as explanations

The choice of ground-truth explanations expl_X is dependent on the application and the existing domain knowledge. CDEP allows for penalizing arbitrary interactions between features, allowing the incorporation of a very broad set of domain knowledge.

In the simplest setting, practitioners may precisely provide groundtruth human explanations for each data point. This may be useful in a medical image classifications setting, where data is limited and practitioners can endow the model with knowledge of how a diagnosis should be made. However, collecting such groundtruth explanations can be very expensive.

To avoid assigning human labels, one may utilize programmatic rules to identify and assign groundtruth importance to regions, which are then used to help the model identify important/unimportant regions. For example, Sec 4.1 uses rules to identify spurious patches in images which should have zero importance and Sec 4.4 uses rules to identify and assign zero importance to words involving gender.

In a more general case, one may specify importances of different feature interactions. For example in Sec 4.2 we specify that the importance of pixels in isolation should be zero, so only interactions between pixels can be used to make predictions. This prevents a model from latching onto local cues such as color and texture when making its prediction.

3.5. Computational considerations

Previous work has proposed ideas similar to Eq 1, where the choice of explanation method is based on gradients (Ross et al., 2017; Erion et al., 2019). However, using such methods leads to three main complications which are solved by our approach.

The first complication is the optimization process. When optimizing over gradient-based attributions via gradient descent, the optimizer requires the gradient of the gradient, requiring that all network components be twice differentiable. This process is computationally expensive and optimizing it exactly involves optimizing over a differential equation, often making it intractable. In contrast, CD attributions are calculated along the forward pass of the network, and as a result, can be optimized plainly with backpropagation using the standard single forward-pass and backward-pass per batch.

A second advantage from the use of CD in Eq 7 is the ability to quickly finetune a pre-trained network. In many applications, particularly in transfer learning, it is common to finetune only the last few layers of a pre-trained neural network. Using CD, one can freeze early layers of the network and quickly finetune final layers, as the calculation of gradients of the frozen layers is not necessary.

Third, CDEP incurs much lower memory usage than competing gradient-based methods. With gradient-based methods the training requires the storage of activations and gradients for all layers of the network as well as the gradient with respect to the input (which can be omitted in normal training). Even for the simplest gradient-based methods, this more than doubles the required memory for a given batch and network size, sometimes becoming prohibitively large. In contrast, penalizing CD requires only a small constant amount of memory more than standard training.

4. Results

The results here demonstrate the efficacy of CDEP on a variety of datasets using diverse explanation types. Sec 4.1 shows results on ignoring spurious patches in the ISIC skin cancer dataset (Codella et al., 2019), Sec 4.2 details experiments on converting a DNN's preference for color to a preference for shape on a variant of the MNIST dataset (LeCun, 1998), Sec 4.3 showcases the use of CDEP to train a neural network that aligns better with a pre-defined fairness measure, and Sec 4.4 shows experiments on text data from the Stanford Sentiment Treebank (SST) (Socher et al., 2013).[2]

4.1. Ignoring spurious signals in skin cancer diagnosis

In recent years, deep learning has achieved impressive results in diagnosing skin cancer, with predictive accuracy sometimes comparable to human doctors (Esteva et al., 2017). However, the datasets used to train these models often include spurious features which make it possible to attain high test accuracy without learning the underlying phenomena (Winkler et al., 2019). In particular, a popular dataset from ISIC (International Skin Imaging Collaboration) has colorful patches present in approximately 50% of the non-cancerous images but not in the cancerous images as can be seen in Fig. 2 (Codella et al., 2019; Tschandl et al., 2018). An unpenalized DNN learns to look for these patches as an indicator for predicting that an image is benign as can be seen in Fig. 3. We use CDEP to remedy this problem by penalizing the DNN placing importance on the patches during training.

The task in this section is to classify whether an image of a skin lesion contains (1) benign lesions or (2) malignant lesions. In a real-life task, this would for example be done to determine whether a biopsy should be taken. The ISIC dataset consists of 21,654 images with a certain diagnosis (19,372 benign, 2,282 malignant), each diagnosed by histopathology or a consensus of experts. We excluded 2247 images since they had an unknown or not certain diagnosis.

To obtain the binary maps of the patches for the skin cancer task, we first segment the images using SLIC, a common image-segmentation algorithm (Achanta et al., 2012). Since the patches are a different color from the rest of the image, they are usually their own segment. Subsequently

[2] All models were trained in PyTorch (Paszke et al., 2017).

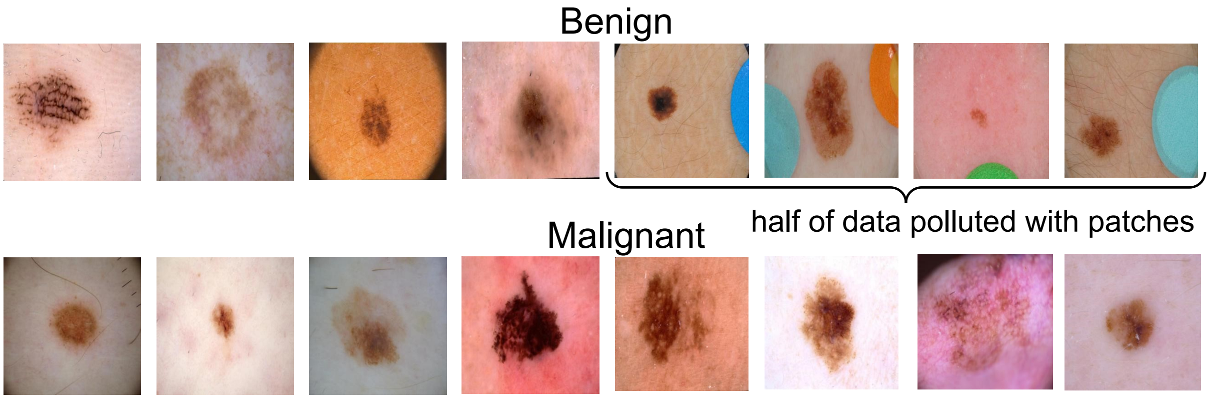

Figure 2. Example images from the ISIC dataset. Half of the benign lesion images include a patch in the image. Training on this data results in the neural network overly relying on the patches to classify images. We aim to avoid this with our method.

Table 1. Results from training a DNN on ISIC to recognize skin cancer (averaged over three runs). Results shown for the entire test set and for only the images the test set that do not include patches ("no patches"). The network trained with CDEP generalizes better, getting higher AUC and F1 on both. Std below 0.006 for all AUC and below 0.012 for all F1.

	AUC (NO PATCHES)	F1 (NO PATCHES)	AUC (ALL)	F1 (ALL)
VANILLA (EXCLUDING TRAINING DATA WITH PATCHES)	0.87	0.57	0.92	0.55
VANILLA	0.93	0.67	0.96	0.67
RRR	0.76	0.45	0.87	0.45
CDEP	**0.95**	**0.73**	**0.97**	**0.73**

we take the mean RGB and HSV values for all segments and filter for segments in which the mean was substantially different from the typical caucasian skin tone. Since different images were different from the typical skin color in different attributes, we filtered for those images recursively. As an example, in the image shown in the appendix in Fig. S3, the patch has a much higher saturation than the rest of the image.

After the spurious patches were identified, we penalized them with CDEP to have zero importance. For classification, we use a VGG16 architecture (Simonyan & Zisserman, 2014) pre-trained on the ImageNet Classification task(Deng et al., 2009)[3] and freeze the weights of early layers so that only the fully connected layers are trained. To account for the class imbalance present in the dataset, we weigh the classes to be equal in the loss function.

Table 1 shows results comparing the performance of a model trained with and without CDEP. We report results on two variants of the test set. The first, which we refer to as "no patches" only contains images of the test set that do not include patches. The second also includes images with those patches. Training with CDEP improves the AUC and F1-score for both test sets.

In the first row of Table 1, the model is trained using only the data without the spurious patches, and the second row shows the model trained on the full dataset. The network trained using CDEP achieves the best F1 score, surpassing both unpenalized versions.

Interestingly, the model trained with CDEP also improves when we consider the entire (biased) dataset, indicating that the model does in fact generalize better to all examples. We also compared our method against the method introduced in 2017 by Ross et al. (RRR). For this, we restricted the batch size to 16 (and consequently use a learning rate of 10^{-5}) due to memory constraints.[4]

Using RRR did not improve on the base AUC, implying that penalizing gradients is not helpful in penalizing higher-order features.[5] In fact, using RRR severely decreased per-

[3] Pre-trained model retrieved from torchvision.

[4] A higher learning rate yields NaN loss and a higher batch size requires too much GPU RAM, necessitating these settings. Due to this a wider sweep of hyperparameters was not possible.

[5] We were not able to compare against the method recently proposed in (Erion et al., 2019) due to its prohibitively slow training and large memory requirements.

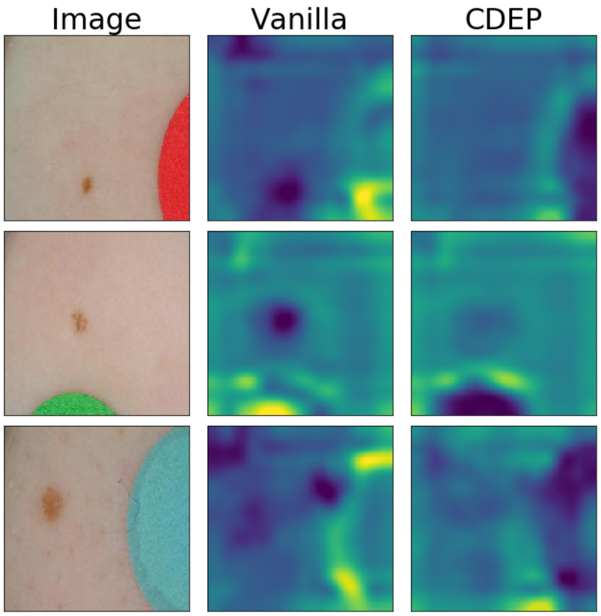

Figure 3. Visualizing heatmaps for correctly predicted exampes from the ISIC skin cancer test set. Lighter regions in the heatmap are attributed more importance. The DNN trained with CDEP correctly captures that the patch is not relevant for classification.

formance in all considered metrics, implying that penalizing gradients not only does not help but impedes the learning of relevant features.

Visualizing explanations To investigate how CDEP altered a DNN's explanations, we visualize GradCAM heatmaps (Ozbulak, 2019; Selvaraju et al., 2017) on the ISIC test dataset with a regularized and unregularized network in Fig. 3. As expected, after penalizing with CDEP, the DNN attributes less importance to the spurious patches, regardless of their position in the image. More examples are shown in the appendix. Anecdotally, patches receive less attribution when the patch color was far from a Caucasian human skin tone, perhaps because these patches are easier for the network to identify.

4.2. Combating inductive bias on variants of the MNIST dataset

In this section, we investigate CDEP's ability to alter which features a DNN uses to perform digit classification, using variants of the MNIST dataset (LeCun, 1998) and a standard CNN architecture for this dataset retrieved from PyTorch [6].

ColorMNIST Similar to one previous study (Li & Vasconcelos, 2019), we alter the MNIST dataset to include three color channels and assign each class a distinct color, as shown in Fig. 4. An unpenalized DNN trained on this biased data will completely misclassify a test set with inverted colors, dropping to 0% accuracy (see Table 2), suggesting that it learns to classify using the colors of the digits rather than their shape.

Here, we want to see if we can alter the DNN to focus on the shape of the digits rather than their color. We stress that this is a toy example where we artificially induced a bias; while the task could be easily solved by preprocessing the input to only have one color channel, this artificial bias allows us to measure the DNN's reliance on the confounding variable *color* in end-to-end training. By design, the task is intuitive and the bias is easily recognized and ignored by humans. However, for a neural network trained in a standard manner, ignoring the confounding variable presents a much greater challenge.

Interestingly, this task can be approached by minimizing the contribution of pixels in isolation (which only represent color) while maximizing the importance of groups of pixels (which can represent shapes). To do this, we penalize the CD contribution of sampled single pixel values, following Eq 7. By minimizing the contribution of single pixels we encourage the network to focus instead on groups of pixels. Since it would be computationally expensive and not necessary to apply this penalty to every pixel in every training input, we sample pixels to be penalized from the average distribution of nonzero pixels over the whole training set for each batch.

Table 2 shows that CDEP can partially divert the network's focus on color to also focus on digit shape. We compare CDEP to two previously introduced explanation penalization techniques: penalization of the squared gradients (RRR) (Ross et al., 2017) and Expected Gradients (EG) (Erion et al., 2019) on this task. For EG we additionally try penalizing the variance between attributions of the RGB channels (as recommended by the authors of EG in personal correspondence). None of the baselines are able to improve the test accuracy of the model on this task above the random baseline, while CDEP is able to significantly improve this accuracy to 31.0%. We show the increase of predictive accuracy with increasing penalization in the appendix. Increasing the regularizer rate for CDEP increases accuracy on the test set, implying that CDEP meaningfully captured and penalized the bias towards color.

DecoyMNIST For further comparison with previous work, we evaluate CDEP on an existing task: DecoyMNIST (Erion et al., 2019). DecoyMNIST adds a class-indicative gray patch to a random corner of the image. This

[6] Retrieved from github.com/pytorch/examples/blob/master/mnist.

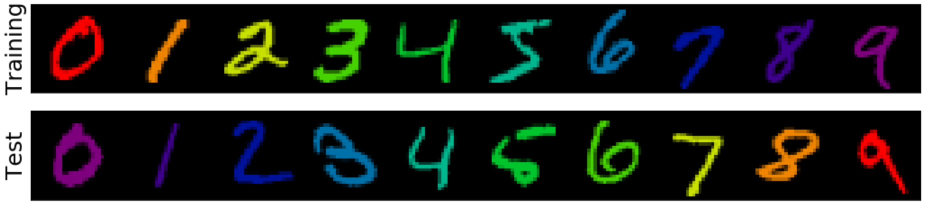

Figure 4. ColorMNIST: the shapes remain the same between the training set and the test set, but the colors are inverted.

Table 2. Test Accuracy on ColorMNIST and DecoyMNIST. CDEP is the only method that captures and removes color bias. All values averaged over thirty runs. Predicting at random yields a test accuracy of 10%.

	VANILLA	CDEP	RRR	EXPECTED GRADIENTS
COLORMNIST	0.2 ± 0.2	**31.0** ± **2.3**	0.2 ± 0.1	10.0 ± 0.1
DECOYMNIST	60.1 ± 5.1	**97.2** ± **0.8**	**99.0** ± **1.0**	**97.8** ± **0.2**

task is relatively simple, as the spurious features are not entangled with any other feature and are always at the same location (the corners). Table 2 shows that all methods perform roughly equally, recovering the base accuracy. Results are reported using the best penalization parameter λ, chosen via cross-validation on the validation set. We provide details on the computation time, and memory usage in Table S1, showing that CDEP is similar to existing approaches. However, when freezing early layers of a network and finetuning, CDEP very quickly becomes more efficient than other methods in both memory usage and training time.

4.3. Fixing bias in COMPAS

In all examples so far, the focus has been on improving generalization accuracy. Here, we turn to improving notions of fairness in models while preserving prediction accuracy instead.

We train and analyze DNNs on the COMPAS dataset (Larson et al., 2016), which contains data for predicting recidivism (i.e whether a person commits a crime / a violent crime within 2 years) from many attributes. Such models have been used for the purpose of informing whether defendants should be incarcerated and can have very serious implication. As a result, we examine and influence the model's treatment of race, restricting our analysis to the subset of people in the dataset whose race is identified as *black* or *white* (86% of the full dataset). All models were fully connected DNNs with two hidden layers of size 5, ReLU nonlinearity, and dropout rate of 0.1 (see appendix for details).

We analyze the effect of CDEP to alter models with respect to one particular notion of fairness: the wrongful conviction rate (defined as the fraction of defendants who are recommended for incarceration, but did not recommit a crime in the next two years). We aim to keep this rate low and relatively even across races, similar to the common "equalized odds" notion of fairness (Dieterich et al., 2016); note that a full investigation of fairness and its most appropriate definition is beyond the scope of the work here.

Table 3 shows results for different models trained on the COMPAS dataset. The first row shows a model trained with standard procedures and the second row shows a model trained with the race of the defendants hidden. The unregularized model in the first row has a stark difference in the rates of false positives between *black* and *white* defendants. Black defendants are more than twice as likely to be misclassified as high-risk for future crime. This is in-line with previous analysis of the COMPAS dataset (Larson et al., 2016).

Obscuring the sensitive attribute from the model does not remove this discrepancy. This is due to the fact that black and white people come from different distributions (e.g. black defendants have a different age distribution).

The third row shows the results for CDEP, where the model is regularized to place more importance on the race feature and its interactions, encouraging it to learn the dependence between race and the distribution of other features. By doing so, the model achieves a lower wrongful conviction rate for both black and white defendants, as well as bringing these rates noticeably closer together by disproportionally lowering the wrongful conviction rate for black defendants. Notably, the test accuracy of the model stays relatively fixed despite the drop in wrongful conviction rates.

Table 3. Fairness measures on the COMPAS dataset. WCR stands for wrongful conviction rate the fraction of innocent defendants who are recommended for incarceration). All values averaged over five runs.

	Test acc	WCR(Black)	WCR(White)
Vanilla	67.8±1.0	0.47± 0.03	0.22±0.03
Race hidden	68.5±0.3	0.44±0.02	0.23±0.01
CDEP	**68.8±0.3**	**0.39±0.04**	**0.20± 0.01**

4.4. Fixing bias in text data

To demonstrate CDEP's effectiveness on text, we use the Stanford Sentiment Treebank (SST) dataset (Socher et al., 2013), an NLP benchmark dataset consisting of movie reviews with a binary sentiment (positive/negative). We inject spurious signals into the training set and train a standard LSTM [7] to classify sentiment from the review.

Positive

pacino is the best **she**'s been in years and keener is marvelous

she showcases davies as a young woman of great charm, generosity and diplomacy

Negative

i'm sorry to say that this should seal the deal - arnold is not, nor will **he** be, back.

this is sandler running on empty, repeating what **he**'s already done way too often.

Figure 5. Example sentences from the SST dataset with artificially induced bias on gender.

We create three variants of the SST dataset, each with different spurious signals which we aim to ignore (examples in the appendix). In the first variant, we add indicator words for each class (positive: 'text', negative: 'video') at a random location in each sentence. An unpenalized DNN will focus only on those words, dropping to nearly random performance on the unbiased test set. In the second variant, we use two semantically similar words ('the', 'a') to indicate the class by using one word only in the positive and one only in the negative class. In the third case, we use 'he' and 'she' to indicate class (example in Fig 5). Since these gendered words are only present in a small proportion of the training dataset ($\sim 2\%$), for this variant, we report accuracy only on the sentences in the test set that do include the pronouns (performance on the test dataset not including the pronouns remains unchanged). Table 4 shows the test accuracy for all datasets with and without CDEP. In all scenarios, CDEP is successfully able to improve the test accuracy by ignoring the injected spurious signals.

Table 4. Results on SST. CDEP substantially improves predictive accuracy on the unbiased test set after training on biased data.

	Unpenalized	CDEP
Random words	56.6 ± 5.8	**75.4 ± 0.9**
Biased (articles)	57.8 ± 0.8	**68.2 ± 0.8**
Biased (gender)	64.2 ± 3.1	**78.0 ± 3.0**

5. Conclusion

In this work we introduce a novel method to penalize neural networks to align with prior knowledge. Compared to previous work, CDEP is the first of its kind that can penalize complex features and feature interactions. Furthermore, CDEP is more computationally efficient than previous work, enabling its use with more complex neural networks.

We show that CDEP can be used to remove bias and improve predictive accuracy on a variety of toy and real data. The experiments here demonstrate a variety of ways to use CDEP to improve models both on real and toy datasets. CDEP is quite versatile and can be used in many more areas to incorporate the structure of domain knowledge (e.g. biology or physics). The effectiveness of CDEP in these areas will depend upon the quality of the prior knowledge used to determine the explanation targets.

Future work includes extending CDEP to more complex settings and incorporating more fine-grained explanations and interaction penalizations. We hope the work here will help push the field towards a more rigorous way to use interpretability methods, a point which will become increasingly important as interpretable machine learning develops as a field (Doshi-Velez & Kim, 2017; Murdoch et al., 2019).

[7] Retrieved from github.com/clairett/pytorch-sentiment-classification.

References

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and Süsstrunk, S. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282, 2012.

Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., and Kim, B. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems*, pp. 9505–9515, 2018.

Ancona, M., Ceolini, E., Oztireli, C., and Gross, M. Towards better understanding of gradient-based attribution methods for deep neural networks. In *6th International Conference on Learning Representations (ICLR 2018)*, 2018.

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., and Samek, W. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015.

Baehrens, D., Schroeter, T., Harmeling, S., Kawanabe, M., Hansen, K., and Müller, K.-R. How to explain individual classification decisions. *Journal of Machine Learning Research*, 11(Jun):1803–1831, 2010.

Bao, Y., Chang, S., Yu, M., and Barzilay, R. Deriving machine attention from human rationales. *arXiv preprint arXiv:1808.09367*, 2018.

Bolukbasi, T., Chang, K.-W., Zou, J. Y., Saligrama, V., and Kalai, A. T. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in neural information processing systems*, pp. 4349–4357, 2016.

Burns, K., Hendricks, L. A., Saenko, K., Darrell, T., and Rohrbach, A. Women also snowboard: Overcoming bias in captioning models. *arXiv preprint arXiv:1803.09797*, 2018.

Codella, N., Rotemberg, V., Tschandl, P., Celebi, M. E., Dusza, S., Gutman, D., Helba, B., Kalloo, A., Liopyris, K., Marchetti, M., et al. Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic). *arXiv preprint arXiv:1902.03368*, 2019.

Dabkowski, P. and Gal, Y. Real time image saliency for black box classifiers. *arXiv preprint arXiv:1705.07857*, 2017.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

Dieterich, W., Mendoza, C., and Brennan, T. Compas risk scales: Demonstrating accuracy equity and predictive parity. *Northpointe Inc*, 2016.

Doshi-Velez, F. and Kim, B. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.

Dressel, J. and Farid, H. The accuracy, fairness, and limits of predicting recidivism. *Science advances*, 4(1): eaao5580, 2018.

Du, M., Liu, N., Yang, F., and Hu, X. Learning credible deep neural networks with rationale regularization. *arXiv preprint arXiv:1908.05601*, 2019.

Erion, G., Janizek, J. D., Sturmfels, P., Lundberg, S., and Lee, S.-I. Learning explainable models using attribution priors. *arXiv preprint arXiv:1906.10670*, 2019.

Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., and Thrun, S. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115, 2017.

Fong, R. C. and Vedaldi, A. Interpretable explanations of black boxes by meaningful perturbation. *arXiv preprint arXiv:1704.03296*, 2017.

Garg, N., Schiebinger, L., Jurafsky, D., and Zou, J. Word embeddings quantify 100 years of gender and ethnic stereotypes. *Proceedings of the National Academy of Sciences*, 115(16):E3635–E3644, 2018. ISSN 0027-8424. doi: 10.1073/pnas.1720347115. URL `https://www.pnas.org/content/115/16/E3635`.

Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*, 2018.

Jain, S. and Wallace, B. C. Attention is not explanation. *arXiv preprint arXiv:1902.10186*, 2019.

Larson, J., Mattu, S., Kirchner, L., and Angwin, J. How we analyzed the compas recidivism algorithm. *ProPublica (5 2016)*, 9, 2016.

LeCun, Y. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

Li, Y. and Vasconcelos, N. Repair: Removing representation bias by dataset resampling. *arXiv preprint arXiv:1904.07911*, 2019.

Liu, F. and Avci, B. Incorporating priors with feature attribution on text classification. *arXiv preprint arXiv:1906.08286*, 2019.

Lundberg, S. M. and Lee, S.-I. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, pp. 4768–4777, 2017.

Madras, D., Pitassi, T., and Zemel, R. Predict responsibly: improving fairness and accuracy by learning to defer. In *Advances in Neural Information Processing Systems*, pp. 6147–6157, 2018.

Mitsuhara, M., Fukui, H., Sakashita, Y., Ogata, T., Hirakawa, T., Yamashita, T., and Fujiyoshi, H. Embedding human knowledge in deep neural network via attention map. *arXiv preprint arXiv:1905.03540*, 2019.

Murdoch, W. J. and Szlam, A. Automatic rule extraction from long short term memory networks. *arXiv preprint arXiv:1702.02540*, 2017.

Murdoch, W. J., Liu, P. J., and Yu, B. Beyond word importance: Contextual decomposition to extract interactions from lstms. *arXiv preprint arXiv:1801.05453*, 2018.

Murdoch, W. J., Singh, C., Kumbier, K., Abbasi-Asl, R., and Yu, B. Interpretable machine learning: definitions, methods, and applications. *arXiv preprint arXiv:1901.04592*, 2019.

Nie, W., Zhang, Y., and Patel, A. A theoretical explanation for perplexing behaviors of backpropagation-based visualizations. *arXiv preprint arXiv:1805.07039*, 2018.

Obermeyer, Z., Powers, B., Vogeli, C., and Mullainathan, S. Dissecting racial bias in an algorithm used to manage the health of populations. *Science*, 366(6464):447–453, 2019.

Ozbulak, U. Pytorch cnn visualizations. `https://github.com/utkuozbulak/pytorch-cnn-visualizations`, 2019.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. Automatic differentiation in pytorch. 2017.

Ribeiro, M. T., Singh, S., and Guestrin, C. Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135–1144. ACM, 2016.

Rieger, L. and Hansen, L. K. Aggregating explainability methods for neural networks stabilizes explanations. *arXiv preprint arXiv:1903.00519*, 2019.

Ross, A. S. and Doshi-Velez, F. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In *Thirty-second AAAI conference on artificial intelligence*, 2018.

Ross, A. S., Hughes, M. C., and Doshi-Velez, F. Right for the right reasons: Training differentiable models by constraining their explanations. *arXiv preprint arXiv:1703.03717*, 2017.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. *See https://arxiv. org/abs/1610.02391 v3*, 7(8), 2016.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 618–626, 2017.

Shrikumar, A., Greenside, P., Shcherbina, A., and Kundaje, A. Not just a black box: Learning important features through propagating activation differences. *arXiv preprint arXiv:1605.01713*, 2016.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Singh, C., Murdoch, W. J., and Yu, B. Hierarchical interpretations for neural network predictions. *arXiv preprint arXiv:1806.05337*, 2018.

Singh, C., Murdoch, W. J., and Yu, B. Hierarchical interpretations for neural network predictions. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=SkEqro0ctQ`.

Singh, C., Ha, W., Lanusse, F., Boehm, V., Liu, J., and Yu, B. Transformation importance with applications to cosmology, 2020.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., and Potts, C. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 1631–1642, 2013.

Springenberg, J. T., Dosovitskiy, A., Brox, T., and Riedmiller, M. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

Strout, J., Zhang, Y., and Mooney, R. J. Do human rationales improve machine explanations? *arXiv preprint arXiv:1905.13714*, 2019.

Sundararajan, M., Taly, A., and Yan, Q. Axiomatic attribution for deep networks. *ICML*, 2017.

Tschandl, P., Rosendahl, C., and Kittler, H. The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific data*, 5:180161, 2018.

Wang, H., He, Z., Lipton, Z. C., and Xing, E. P. Learning robust representations by projecting superficial statistics out. *arXiv preprint arXiv:1903.06256*, 2019.

Winkler, J. K., Fink, C., Toberer, F., Enk, A., Deinlein, T., Hofmann-Wellenhof, R., Thomas, L., Lallas, A., Blum, A., Stolz, W., and Haenssle, H. A. Association Between Surgical Skin Markings in Dermoscopic Images and Diagnostic Performance of a Deep Learning Convolutional Neural Network for Melanoma RecognitionSurgical Skin Markings in Dermoscopic Images and Deep Learning Convolutional Neural Network Recognition of MelanomaSurgical Skin Markings in Dermoscopic Images and Deep Learning Convolutional Neural Network Recognition of Melanoma. *JAMA Dermatology*, 08 2019. ISSN 2168-6068. doi: 10.1001/jamadermatol.2019.1735. URL `https://doi.org/10.1001/jamadermatol.2019.1735`.

Zaidan, O., Eisner, J., and Piatko, C. Using annotator rationales to improve machine learning for text categorization. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pp. 260–267, 2007.

Zhang, T. and Zhu, Z. Interpreting adversarially trained convolutional neural networks. *arXiv preprint arXiv:1905.09797*, 2019.

Zintgraf, L. M., Cohen, T. S., Adel, T., and Welling, M. Visualizing deep neural network decisions: Prediction difference analysis. *arXiv preprint arXiv:1702.04595*, 2017.