Right for Better Reasons: Training Differentiable Models by Constraining their Influence Function

Xiaoting Shao, Arseny Skryagin, Wolfgang Stammer, Patrick Schramowski, Kristian Kersting
TU Darmstadt, Darmstadt, Germany
{xiaoting.shao, arseny.skryagin, wolfgang.stammer, schramowski, kersting}@cs.tu-darmstadt.de

Abstract

Explaining black-box models such as deep neural networks is becoming increasingly important as it helps to boost trust and debugging. Popular forms of explanations map the features to a vector indicating their individual importance to a decision on the instance-level. They can then be used to prevent the model from learning the wrong bias in data possibly due to ambiguity. For instance, Ross *et al.*'s "right for the right reasons" propagates user explanations backwards to the network by formulating differentiable constraints based on input gradients. Unfortunately, input gradients as well as many other widely used explanation methods form an approximation of the decision boundary and assume the underlying model to be fixed. Here, we demonstrate how to make use of influence functions—a well known robust statistic—in the constraints to correct the models behaviour more effectively. Our empirical evidence demonstrates that this "right for better reasons"(RBR) considerably reduces the time to correct the classifier at training time and boosts the quality of explanations at inference time compared to input gradients.

Introduction

Nowadays, with the success of deep neural networks, training a classifier to achieve high accuracy is relatively easy. Even then, gaining trust from human is still difficult for those models as the decision-making mechanism is often non-transparent. Just because a machine learning model is highly accurate on the given data does not mean it represents the correct mapping in that domain. Especially in high dimensional, real-world domains, these "Clever Hans"-like moments—making use of confounding factors within data sets—are observable due to spurious artifacts, which could be unwantedly learnt by the models.

More precisely, (Lapuschkin et al. 2019) reported that a deep neural network trained on the PASCAL VOC 2007 data set (Everingham et al. 2007) focuses on image source tags for classification, which only incidentally correlate with the class labels. This "Clever Hans"-like behavior (Sebeok and Rosenthal 1981) happens when the model has learnt the spurious artifact, also known as confounding factors. In this case, the model's underlying behavior is systematically wrong, and therefore may not generalize well to unseen data. Such systematic wrong behavior can be hard to spot and do real harm when applied in a real-world setting. For instance, Obermeyer et al. (2019) demonstrated that a widely-used commercial model for predicting medical needs exhibits significant racial bias—black patients are considerably sicker than white patients, at a given risk score. Obermeyer et al. attributed this to the fact that the model uses medical expenses to predict medical needs, however, black people have less access to medical care so that less medical expenses are given to them compared to white people in the same health condition. This racial bias in the model could pose a real danger to black patients.

These concerns and issues about machine learning (ML) models have motivated recent work in correcting the models based on user explanations such as training the models to be right for the right reasons (Ross, Hughes, and Doshi-Velez 2017). However, (Ross, Hughes, and Doshi-Velez 2017), as well as other work of its kind, rely on explanations showing how the prediction changes when the test point is perturbed (Simonyan, Vedaldi, and Zisserman 2013; Adler et al. 2018), assuming the model to be fixed. But how does the model actually change when perturbing the training data? This is answered well by the so-called influence function (Cook and Weisberg 1980; Koh and Liang 2017), which explains a model's predictions by tracing back to the training process. Built upon Ross, Hughes, and Doshi-Velez's "right for the right reasons", we therefore propose to improve its effectiveness by leveraging the higher-order and robust explanations due to influence functions.

Specifically, we make the following contributions:

- We propose the first interactive correction of (ML) models, called "right for better reasons" (RBR), based on influence functions.
- We improve the adversarial robustness of machine learning models by constraining the influence function.
- We demonstrate the effectiveness of RBR on both synthetic and real-world data and compare it to RRR across a number of data sets and model architectures.

To this end, we proceed as follows. We start off by reviewing related work on explainable ML. Afterwards, we introduce RBR. Before concluding, we present our empirical results.

Explainable Machine Learning

Explaining decisions of ML models has increasingly gained attention as the black-box models' opaqueness could undermine end-users' trust (Adler et al. 2018), complicate debugging the ML model (Shrikumar, Greenside, and Kundaje 2017), and potentially harm fairness or pose a safety hazard in real-world use. However, explanations are a very general concept and can take very different forms (Von Wright 2004). Essentially, explanations can be broken down into two categories: global explanations and local explanations. The former ones are conceived on the model level to extract some general understanding of the model (Buciluǎ, Caruana, and Niculescu-Mizil 2006; Bastani et al. 2017), while the latter ones often refer to instance-level explanations, which have arguably been extensively studied.

Consider e.g. LIME (Local Interpretable Model-agnostic Explanations) (Ribeiro, Singh, and Guestrin 2016). LIME takes samples at a local scope of the queried instance to approximate decision boundaries by a simplified linear model, which is interpretable. This provides an explanation for each instance. The main advantage of this explainer lies in its universal applicability. Its main drawback is that the process is very slow and unstable due to its sampling sub-procedure. Similar to LIME, input gradient (IG) (Baehrens et al. 2010; Simonyan, Vedaldi, and Zisserman 2013) also approximates decision boundaries with simplified local models. But this is computed in closed form as follows:

$$I_{\text{IG}} \coloneqq \frac{\partial}{\partial x_{nd}} \sum\nolimits_{k=1}^{K} \log(\hat{y}_{nk})$$

for inputs $X \in \mathbb{R}^{N \times D}$, labels $y \in \mathbb{R}^{N \times K}$ and outputs of a differentiable model $\hat{y} \in \mathbb{R}^{N \times K}$. This yields a relevance score for each feature on each instance, which we refer to as saliency maps. This statistic is much faster to compute compared to LIME. Moreover, it passes the sanity check due to Adebayo et al. (2018), who demonstrated that several explanation methods are simply independent of the model parameters and, in turn, are not capable of generating faithful explanations.

Inferring explanations takes one or multiple forward passes of the network. Backpropagating supervised explanations to the network has also been studied in recent works to train networks that align with user explanations. This is especially useful for preventing the models from inheriting accidental bias in the data so that it also performs robustly during test time when this bias is absent. Some research has been done towards this end. Ross, Hughes, and Doshi-Velez (2017) proposed to train models to be "right for the right reasons" (RRR) by formulating constraints based on supervised explanations using input gradient and user feedback. The constraints take a differentiable form and, hence, can be imposed on classifiers using standard gradient-based methods. Ross, Hughes, and Doshi-Velez showed empirically the effectiveness of RRR for training models to learn the right rules by explicitly penalizing the wrong rules, especially in the presence of bias in the data that confounds with prediction targets. Teso and Kersting (2019) recently extended the idea of RRR to a learner-agnostic setting. Rieger et al. (2019) followed a similar scheme to RRR but adapted its loss to contextual decompositions (CDs) (Murdoch, Liu, and Yu 2018). In a similar spirit, Erion et al. (2019) proposed to use expected gradients to encode explanations as prior knowledge such as the smoothness over adjacent pixels in the image domain. Related to this, Kim et al. (2019) unlearn the target bias in a data set by minimizing the mutual information between the transformed feature and the target bias. In order to do that, they add two auxiliary networks, f and h, for transforming features and predicting bias respectively, other than the label-predicting network g. The task is to optimize all networks jointly so that f extracts features containing no information of the target bias predicted by h, but the labels predicted by g still achieves satisfying accuracy.

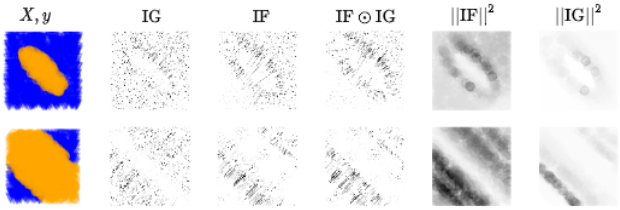

Figure 1: (Best viewed in color) Input Gradients (IG) versus Influence Function (IF) on two 2D data sets. From left to right: data, vector fields of IG and IF as well as their component-wise product, l^2-norm of IG and IF vectors. As one can see, IF integrates the different reasons for a decision into a better explanation.

Right for Better Reasons (RBR)

Our "right for better reasons" loss is triggered by Kim et al. (2019). However, they employ two auxiliary networks, next to the classifier network, for transforming features and predicting bias respectively. Doing so poses potentially a great overhead and may complicate training. The approaches of Ross, Hughes, and Doshi-Velez (2017), Rieger et al. (2019) and Erion et al. (2019) propagate only supervised explanations—coming from some external XAI module—as constraints directly back to the model. None of them explains the predictions of a model explicitly through its learning algorithm and back to the training data. But how can we then ensure that the explanation captures the model and the learning process?

To answer this, we therefore ask, what would the explanation look like if we did not have this training point, or if the values of this training point were changed slightly? To this end we adapt influence functions (Cook and Weisberg 1980; Koh and Liang 2017)—a classic technique from robust statistics—for explanatory interactive ML (Teso and Kersting 2019). They trace the models prediction through the learning algorithm and back to its training data, where the model parameters ultimately derive from, in a closed-form.

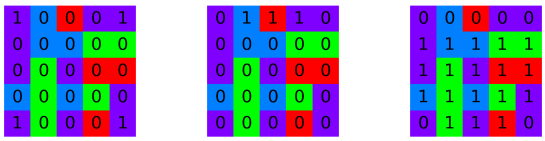

Figure 2: (Best viewed in color) Feedback masks A for the color data set penalizing rule 1 (Left) and 2 (Middle) respectively, and feedback on the decoyed color data set (Right).

Explanations via Influence functions. An influence function takes the following form:

$$I(z, z_{\text{test}})^{\text{T}}_{\text{IF}} := -\nabla_\theta L(z_{\text{test}}, \hat{\theta})^{\text{T}} H_{\hat{\theta}}^{-1} \nabla_x \nabla_\theta L(z, \hat{\theta})$$

where z and z_{test} are a training sample and a test sample respectively, L denotes the loss, x the input, θ the model parameters and $H := 1/n \sum_{i=1}^{n} \nabla^2_\theta L(z_i, \hat{\theta})$ the Hessian. $I(z, z_{\text{test}})^{\text{T}}_{\text{IF}}$ indicates the most influential direction of perturbing z for z_{test}, and the features of z in this direction explains why the prediction on z_{test} is made. Using just

$$I(z, \theta)^{\text{T}}_{\text{IF}} := H_{\hat{\theta}}^{-1} \nabla_x \nabla_\theta L(z, \hat{\theta})$$

computes the influence of z to θ based on the second-order approximation of the empirical loss around θ. Generally, $H_{\hat{\theta}}^{-1}$ provides the curvature information of the parameter space and offers a better local approximation of the loss compared to input gradient, and $\nabla_x \nabla_\theta L(z, \hat{\theta})$ points to the direction in which perturbing the training point z leads to most significant model update. Since we are mainly interested in the latter information, we replace $H_{\hat{\theta}}^{-1}$ by the identity matrix and, hence, propose the sum of $\nabla_x \nabla_\theta L(z, \hat{\theta})$ as a more robust statistics for explanatory interactive ML.

To illustrate this, consider Fig. 1. It gives some insights and intuitions on IG-generated explanations (I_{IG}), and IF $\odot$ IG-generated explanations ($I^{\text{T}}_{\text{IF}} \odot I_{\text{IG}}$) by visualizing their vector fields and l^2-norm generated by a three-layer MLP on some synthetic two-dimensional classification data sets. As Ross and Doshi-Velez (2018) noted, input gradients are sometimes noisy and not interpretable on their own. One can see that the vector field of IF $\odot$ IG is sharper around decision boundaries, while IGs yield quite blurry and noisy explanations over the whole domain. Since the decision boundary describes the model's behavior, having a less noisy and ambiguous decision boundary yields a better description of the model.

The RBR Loss Function. We now leverage the more robust statistics and formulate the constraints on the explanations to make the model right for better reasons (RBR). That is, we use information from the influence function (IF) to compute saliency maps of features and penalize features according to feedback using standard gradient-based methods. Note that the term RBR is used interchangeably with "IF-constrained" and "IF feedback" in the following, and the term RRR is used interchangeably with "IG-constrained" and "IG feedback".

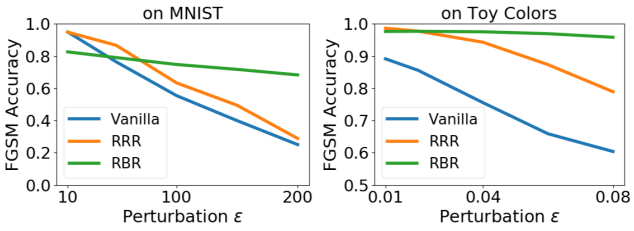

Figure 3: Accuracies of the vanilla model, RRR and RBR on adversarial examples with increasing perturbations ε.

To this end, we define the loss function as a weighted sum of the right answer loss (cross-entropy), the right reason loss (user feedback on saliency map) and L2 regularization:

$$\begin{aligned} L(\theta, X, y, A) = & \underbrace{\frac{1}{N} \sum_{n=1}^{N} \sum_{k=1}^{K} -y_{nk} \log(\hat{y}_{nk})}_{\text{right answers}} \\ & + \underbrace{\lambda \sum_{n=1}^{N} \sum_{d=1}^{D} (A_{nd} I(z, \theta)^{\text{T}}_{\text{IF}_{nd}} I_{\text{IG}_{nd}})^2}_{\text{right reasons}} \\ & + \underbrace{\sum_i \theta_i^2}_{\text{regularization}} \end{aligned} \tag{1}$$

where $A \in \{-1, 0, 1\}^{N \times D}$ encodes the user feedback and can be seen as a mask. See Fig. 2 for an illustration on the color data set described in the experimental section. For this data set we did not use -1 in the feedback, but only 0 and 1. λ controls the balance between the right reasons and the right answers. This loss poses bias towards the features annotated as -1s, against the features annotated as 1s and ignores the rest. We note that one should be mindful of the faithfulness of the saliency map when formulating right reason loss. This is because plugging in an unfaithful saliency map may lead to non-convergence. We use the influence of z on the model parameters, $I(z, \theta)^{\text{T}}_{\text{IF}}$, as a measure to approximate the relevance of each feature of z on the model, because we do not have access to the test set at training time.

Formulating constraints on the decision boundary this way is more efficient than using input gradients, cf. Fig. 1. And this indeed turns out to provide faster correction to the model compared to RRR as the user feedback is formulated in a more robust form, i.e. IF $\odot$ IG. Besides, we demonstrate that the model gains more generalization ability and adversarial robustness across multiple data sets in our empirical evaluation. At last, we showcase the effectiveness of RBR in high-dimensional domains.

Empirical Evaluations

Our intention here is to investigate the following questions empirically: **(Q1)** Does RBR regularization help to improve adversarial robustness? **(Q2)** Do classifiers learn the right rules when using RBR? **(Q3)** Does RBR help to converge faster than RRR? **(Q4)** How effective is RBR in high-dimensional domains? To this end, we ran experiments on

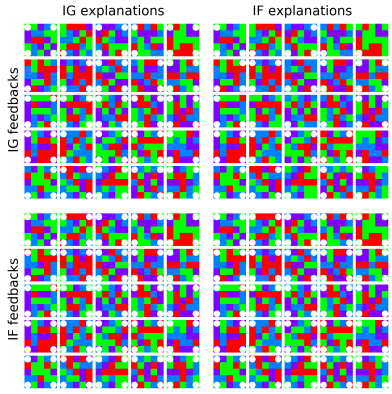

Figure 4: (Best viewed in color) Color data set—corner pixels: IF and IG explanations after penalizing the corner pixels with IF and IG respectively. White dots denote the salient features.

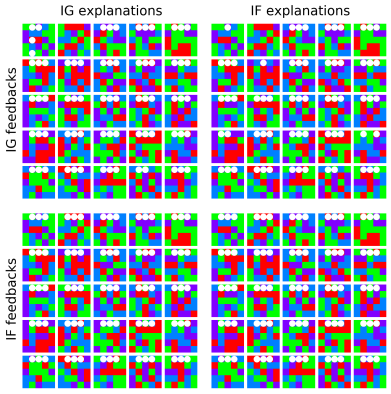

Figure 5: (Best viewed in color) Color data set—top middle pixels: IF and IG explanations after penalizing the top middle three pixels with IF and IG respectively. White dots denote the salient features.

a Linux machine with two Intel Xeon processors with 56 hyper-threaded cores, 4 NVDIA GeForce GTX 1080 under Ubuntu Linux 14.04. All the models were implemented in Python and Tensorflow. The experiments on the high dimensional domains (in **(Q4)**) were run on GPUs, the rest experiments were run on CPUs. The hyperparmeter for RBR and the baseline are both chosen based on the magnitude of the right reason loss and the right answer loss.

(Q1) Adversarial robustness. For a start, we investigated whether RBR can improve the adversarial robustness of a model. To this end, we trained an eight-layer MLP as the classifier on the toy color data set from (Ross, Hughes, and Doshi-Velez 2017) and MNIST (LeCun 1998) by directly constraining IFs. The toy color data set entails two independent rules: (1) four corner pixels are the same and (2) top middle three pixels are different. Samples satisfying both rules belong to class 1, and samples satisfying neither belong to class 2. These two class of samples constitute the whole data set. Each rule alone is sufficient to infer the class label. For this experiment, we set A in the loss function 1 to be an all-ones matrix instead of biased feedback. As baseline, a vanilla classifier trained without any form of constraint and a classifier trained with RRR were used. Grid search was done to select the best hyperparameters. Here, the hyperparameter λ for RBR and RRR is respectively 1e+8 and 1e+3 on MNIST, 1e-9 and 1e-4 on toy colors. To generate adversarial examples, we applied the scheme of the Fast Gradient Sign Method (FGSM) (Goodfellow, Shlens, and Szegedy 2015) but replaced the gradient with the influence function. As one can see in Fig. 3, the accuracy of RBR model drops only very slightly when perturbation on the input increases, while RRR and the vanilla model deteriorate significantly. This shows that RBR model is much more robust to adversarial perturbations on both data sets compared to the vanilla and the RRR model. This answers **(Q1)** affirmatively.

(Q2) The model learns the right rules. However, if we give biased feedback, does it yield models with the right rules? In order to investigate this, we again use the toy color data set. The user feedback for constraining each rule is shown in Fig. 2 (Left, Middle).

On the toy color data set we trained a MLP penalizing each rule where each feedback is formulated using IG, i.e., with RRR, and using IFs, i.e., with RBR. λ for RBR and RRR is respectively 10 and 0.1. Figs. 4 and 5 show the explanations across a few test set examples for each of these models. The salient features of each model on each example are induced by IG and IF respectively and are denoted as white dots on 25 randomly selected test samples. One can see that the model focuses on rule 2 when rule 1 is penalized, and vice versa. Also, RBR forces the model to learn better rules than RRR, as the explanations of the IG-constrained models on some samples either did not recognize the right rule or did not recognize the complete rule.

Additionally, we investigated the decoyed MNIST data set from (Ross, Hughes, and Doshi-Velez 2017). This data set adds grey patches to the baseline MNIST in randomly selected corners, whose shades are functions of the label in training set but random in test set. On this data set, the feedback annotates every feature confounded with a grey patch as one. And we run another experiment with similar setups. Fig. 6 illustrates the explanations of resp. IF and IG after training without constraint, with RRR and with RBR. One can see that the vanilla classifier mainly uses the confounding features to make predictions, but after training with user feedback the salient regions focus more on the digits. The effect of RBR is as appealing as RRR. Besides, IFs yield less noisy explanations compared to the explanations of IGs, as already mentioned above.

Hence, the question whether RBR makes the model learn the right and even better rules can be answered affirmatively. Thus, **(Q2)** can also be answered affirmatively.

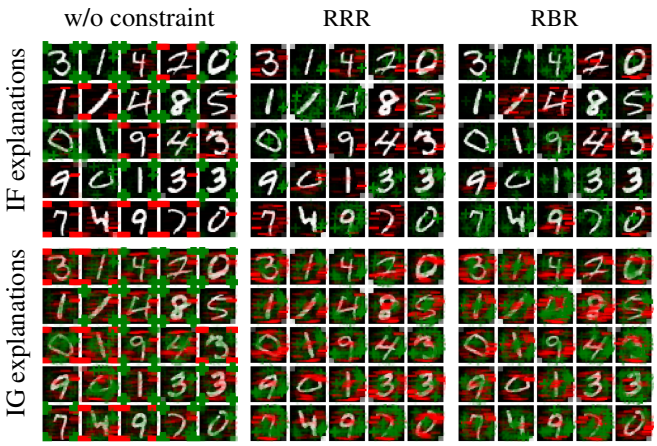

Figure 6: (Best viewed in color) IF and IG explanations on some randomly selected samples from models resp. trained without constraint, with RRR and RBR.

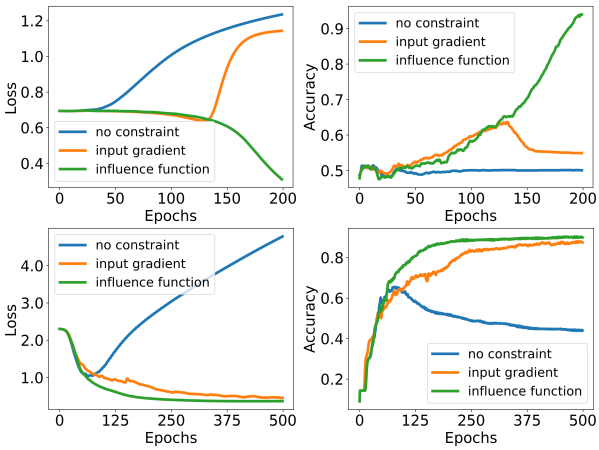

Figure 7: Loss (left column) and accuracy (right column) of the classifier on test set when training respectively on decoyed toy color data (top row) and decoyed MNIST (bottom row) with resp. no constraints, IG constraints and IF constraints.

(Q3) Convergence speed. We compared the convergence speed in presence of confounding factors. As confounded data sets, we used the decoyed MNIST data set and also constructed another data set by using the toy color data set as baseline and intentionally adding a patch at a random location to the training images, whose color is determined by the label. The feedback on the decoyed color data set provided annotation of ones on all the pixels that are not relevant for the two decision rules, cf. Fig. 2(Right). The feedback on the decoyed MNIST data set is the same as in **(Q2)**.

On each data set we trained three MLPs using no feedback, IG feedback (RRR) and IF feedback (RBR). λ for RBR and RRR is respectively 1e-3 and 10 on MNIST, 0.1 and 0.01 on toy colors. The cross-entropy and accuracy on the test set reflects how well the model generalizes to unseen data. They are shown over the training epochs in Fig. 7. Without any user feedback, we observed an accuracy of 100% on both training sets. But on the test set, the cross-entropy is surging and the accuracy dropping to random, suggesting that the model overfits to the confounding factor and does not generalize at all. Providing IF feedback prevents the classifier from learning the confounding rules since the decreasing cross-entropy and improved accuracy on the test set implies the model is able to generalize. Moreover, the convergence speed is much faster compared to RRR. Overall, the end2end running time was not affected much by the overhead of computing IF. On decoy MNIST, IG took 7 minutes, while IF took 9 minutes. This answers **(Q3)** affirmatively.

(Q4) Effectiveness in the high-dimensional domain. Finally, we illustrate the effectiveness of RBR in real-world tasks.

Explanatory Interactive PASCAL VOC 2007. Our first experiment used the PASCAL VOC 2007 benchmark (Everingham et al. 2007), consisting of labeled images from twenty object classes in realistic scenes. The goal is to correct the "Clever Hans"-like moments. Specifically, we considered only two object classes, horse and dog, and used VGG-16 (Simonyan and Zisserman 2015) as a classifier network. For optimization we fine tuned the initial weights pretrained on ImageNet. Performance is measured by the balanced accuracy score defined as the average of recall obtained on each class. Without user feedback on the explanations (vanilla model), our fine-tuned classifier reached a training accuracy of 99% and test accuracy of 87%. Applying input gradients across test set, we observed, as illustrated by the saliency maps in Fig. 8 (second row), that VGG-16 often unwantedly focuses on the source tags to make predictions, confirming (Lapuschkin et al. 2019).

To revise VGG-16, our feedback to VGG-16 was to avoid the source tags as salient features, as illustrated in Fig. 10 (Left). We train a RBR and a RRR respectively with λ equals 1e+7 and 1. Fig. 8 shows a few examples and their saliency maps from different models. As one can clearly see, VGG-16 decisions are based on the source tag without any revision (second row). But with RBR, the model does not focus on the features in the bottom left corner any more, instead the salient features overlap more with the horse or its rider. Although with RRR, the source tags are also not salient any more, but the salient regions are overlapping very few with the target object. One may argue that the rider could also be an confounding factor in this case and a dog image with a rider on it, imaginary, may therefore be categorized as a horse. We leave this out of consideration here and focus on removing one certain confounding factor which is the source tags.

Explanatory Interactive Deep Plant Phenotyping. To investigate RBR on a scientific task, we considered a phenotyping task on data acquired from plant physiologists. It consists of RGB images of leaf tissues on a nutrition so-

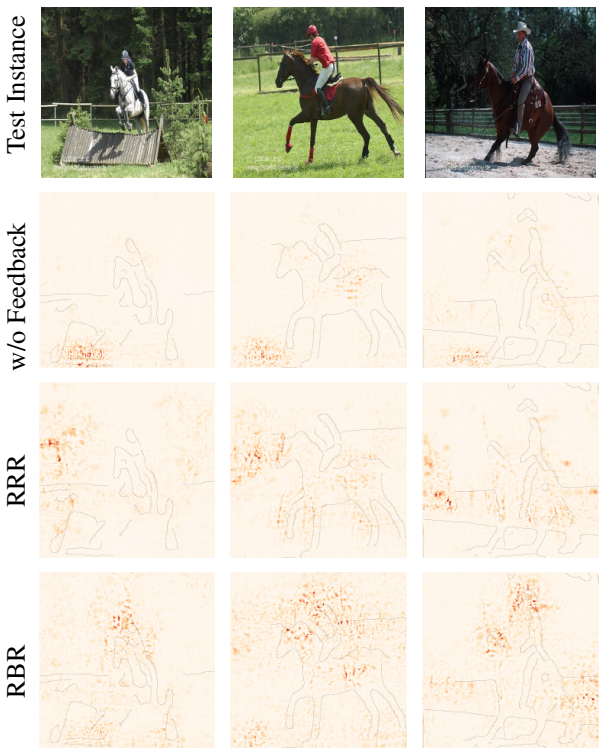

Figure 8: (Best viewed in color) Revising VGG-16 on PASCAL VOC 2007. Horse images (first row), their saliency maps on vanilla model (second row), and their saliency maps on RRR (third row) and RBR (last row).

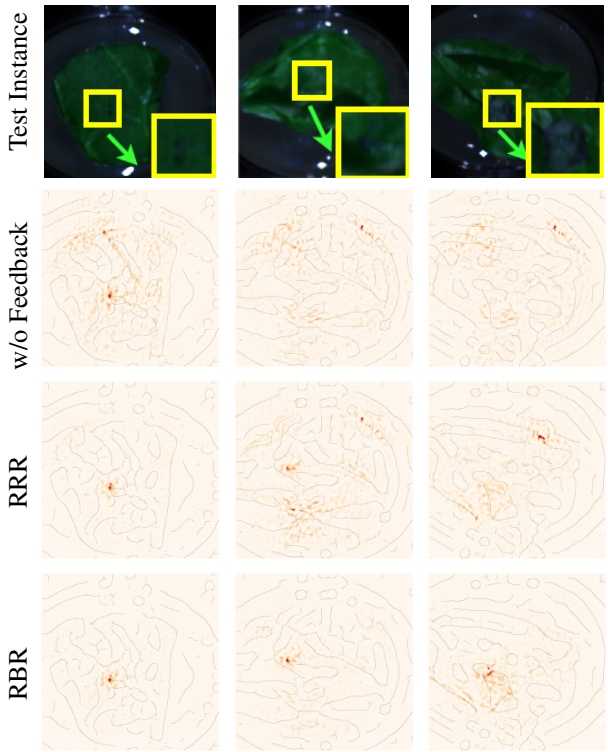

Figure 9: Plant examples (with zooms on biologically plausible reasons, yellow boxes) and their saliency maps before and after correction. As one case see, RBR revises the classifier to focus on the biologically plausible reasons, namely, on the leaves instead of the tissue.

lution labeled as healthy or Cercospora-inoculated. Using the same experimental setup as for PASCAL VOC 2007, the fine-tuned VGG-16 reached an accuracy of 82% on test set. Inspecting its predictions using saliency maps, we found that VGG-16 showed again a "Clever Hans"-like moment. It undesirably often looks at the border of the background, specifically on the nutrition solution, or irrelevant part of the leaf tissue to make inferences. These are biologically not plausible.

We acquired annotations of the background and ran another experiment with similar setups on PASCAL VOC 2007. The counter examples here were test set instances with randomized background, and the random examples were test set instances with randomized leaf tissue.

We then revised VGG-16 by providing constraints on the background, penalizing its relevance. Using the same constraints we trained RRR and RBR respectively with λ equals 1e-3 and 1e-4. Fig. 9 shows some examples from the test set and their saliency maps with no user correction (second row), RRR correction (third row) and RBR correction (last row). As one can see, (1) before the human interacts with VGG-16 through its explanations, VGG-16 often looked at the border of the background to make predictions, although a clearly visible sick spot is present. (2) After interacting through the explanations via RBR, VGG-16 learns to look at the sick spot to classify. (3) The explanations before RBR are complex and hardly make sense to a plant physiologist. In contrast, revising the model via RBR poses a constraint and makes the explanations more simple, concise, and biologically plausible. 4) Although RRR also induced more concise saliency maps, the salient features did not always land on the obvious sick spot.

Quantitative Comparison. To measure the effectiveness of this correction quantitatively, we randomized respectively the user-annotated unsalient features and all the rest features (possibly salient or unannotated) across the whole test set. We call the samples with randomized unsalient features "counter examples", and the samples with randomized rest features as "random examples". Intuitively, if a classifier is right for the right reasons, the accuracy on the counter examples should be high because the randomization only influenced the uninformative features. On the contrary, the accuracy on the random examples should be low because the randomization destroyed valuable information in the salient features. The empirical results are summarized in Tab. 1. It shows the (balanced) accuracy of counter examples and random examples on PASCAL VOC 2007 and the deep plant dataset, with and without user feedback. As one can see for PASCAL VOC 2007, without any correction, by only looking at the source tag, the classifier achieves higher ac-

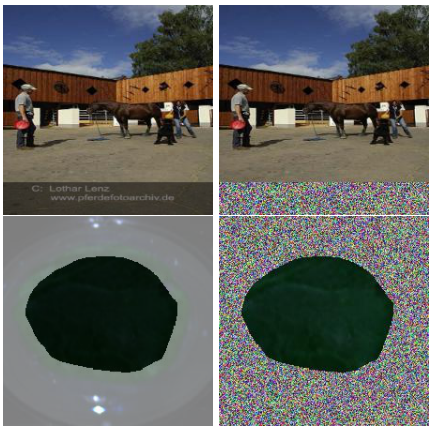

Figure 10: (Best viewed in color) User feedback on PASCAL VOC 2007 (Top) and the plant phenotyping data set (Bottom). Shown are the original images overlaid with user-annotated mask (dark overlay denote 1s in the mask and the rest are -1s) as well as randomized version.

curacy (74%) than by looking at the target object (71%). That is, the user-annotated unsalient features are more informative for the classifier than the user-annotated salient features. Fortunately, this "Clever Hans"-like moment can be revised by penalizing unsalient features based on human feedback: Via RBR, the model takes more information from the user-annotated salient features because the accuracy on the counter examples (84%) is higher than on the random examples. Although with RRR, similar pattern can be seen, but the accuracy for RRR is lower than the accuracy for RBR.

As for the deep plant dataset, this "Clever Hans"-like moment is more prominent: the accuracy is much higher by looking at the background (70%) than by looking at the leaf tissue (50%). That means, although VGG-16 converged to 87% accuracy on test set, it did not converge to a biologically plausible strategy. After correction, the performance dropped to almost prior distribution (51%) by only looking at the background, while the relevance of the leaf tissue increased considerably—the accuracy increased from 51% to 62% on the counter examples. With RRR, although the accuracy on the random examples dropped to 54%, the accuracy on the counter examples increased only to 54%. That implies the leaf tissue and the background are same informative to the classifier.

As a conclusion of this experiment, human feedback with RBR can help to considerably boost the performance of VGG-16. Both experiments on real-world datasets together demonstrate the practical use of user feedback via IFs in high-dimensional domains. This answers **(Q4)** affirmatively.

Conclusions and Future work

We proposed to use influence functions to encode user feedback in a robust form to avoid ML models from learning the wrong decision rule. We have shown that using robust user feedback can constrain classifiers better to learn the right rules in comparison to using only input gradients. Actually,

		Vanilla	RRR	RBR
PASCAL VOC 2007	counter examples	71%	81%	84%
	random examples	74%	65%	76%
deep plant	counter examples	50%	54%	62%
	random examples	70%	54%	51%

Table 1: Accuracies of three models on counter examples and random examples experimented respectively on PASCAL VOC 2007 and the deep plant dataset.

they improve the generalization ability of the model. Additionally, training to learn the right reasons required fewer epochs using RBR compared to RRR, while the overall run time stays on the same scale.

Our work provides many interesting avenues for future work. An interesting step would be to actually put humans into the training loop and let them correct the model online and interactively, rather than running simulated experiments as done here. Moreover, approximations to influence functions are known to still provide valuable information even on non-convex and non-differentiable models (Koh and Liang 2017). One should investigate RBR for this case. In general, not only learning but also model explanations should be interactive and not be just, say, a fixed heatmap, since there is great value in communication between the model, the explanations and the human. Additionally, our current work can not account for more complex user feedback, such as "color is not relevant for this classification". So one should also consider more complex feedback from the user's side. Finally, our "right for better reasons" approach may be of use in solving related problems, e.g., in maintaining robustness, fairness, accountability, transparency, and moral biases.

Acknowledgments

We thank Daniel Thuerck for helpful discussions. PS and KK acknowledge the German Federal Ministry of Food and Agriculture (BMEL) based on a decision of the Parliament of the Federal Republic of Germany via the Federal Office for Agriculture and Food (BLE) under the innovation support program, project DePhenSe (FKZ 2818204715). WS an KK were also supported by BMEL/BLE funds under the innovation support program, project AuDiSens (FKZ 28151NA187). XS and KK also acknowledge the support by the German Science Foundation project CAML (KE1686/3-1) as part of the SPP 1999 (RATIO). AS and KK acknowledge the support of the BMBF and the Hessian Ministry of Science and the Arts (HMWK) within the National Research Center for Applied Cybersecurity ATHENE.

References

Adebayo, J.; Gilmer, J.; Muelly, M.; Goodfellow, I.; Hardt, M.; and Kim, B. 2018. Sanity Checks for Saliency Maps. In *Proceedings of the International Conference on Neural Information Processing Systems, NeurIPS*, 9525–9536.

Adler, P.; Falk, C.; Friedler, S. A.; Nix, T.; Rybeck, G.; Scheidegger, C.; Smith, B.; and Venkatasubramanian, S. 2018. Auditing black-box models for indirect influence. *Knowledge and Information Systems* 54(1): 95–122.

Baehrens, D.; Schroeter, T.; Harmeling, S.; Kawanabe, M.; Hansen, K.; and Müller, K.-R. 2010. How to explain individual classification decisions. *Journal of Machine Learning Research* 11.

Bastani, O.; et al. 2017. Interpreting blackbox models via model extraction. *arXiv preprint arXiv:1705.08504* .

Buciluǎ, C.; Caruana, R.; and Niculescu-Mizil, A. 2006. Model Compression. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, SIGKDD*, KDD. ACM.

Cook, R.; and Weisberg, S. 1980. Characterizations of an empirical influence function for detecting influential cases in regression. *Technometrics* 22(4): 495–508.

Erion, G.; Janizek, J. D.; Sturmfels, P.; Lundberg, S.; and Lee, S.-I. 2019. Learning Explainable Models Using Attribution Priors. *arXiv preprint arXiv:1906.10670* .

Everingham, M.; Van Gool, L.; Williams, C.; Winn, J.; and Zisserman, A. 2007. The PASCAL Visual Object Classes Challenge 2007 (VOC) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.

Goodfellow, I.; Shlens, J.; and Szegedy, C. 2015. Explaining and Harnessing Adversarial Examples. In *International Conference on Learning Representations, ICLR*.

Kim, B.; Kim, H.; Kim, K.; Kim, S.; and Kim, J. 2019. Learning Not to Learn: Training Deep Neural Networks with Biased Data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR-19*, 9012–9020.

Koh, P. W.; and Liang, P. 2017. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70, ICML*. JMLR. org.

Lapuschkin, S.; Wäldchen, S.; Binder, A.; Montavon, G.; Samek, W.; and Müller, K.-R. 2019. Unmasking Clever Hans predictors and assessing what machines really learn. *Nature communications* 10.

LeCun, Y. 1998. The MNIST database of handwritten digits.

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

Obermeyer, Z.; Powers, B.; Vogeli, C.; and Mullainathan, S. 2019. Dissecting racial bias in an algorithm used to manage the health of populations. *Science* 366(6464): 447–453. ISSN 0036-8075. doi:10.1126/science.aax2342.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. 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, SIGKDD*. ACM.

Rieger, L.; Singh, C.; Murdoch, W. J.; and Yu, B. 2019. Interpretations are useful: penalizing explanations to align neural networks with prior knowledge. *arXiv preprint arXiv:1909.13584* .

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

Ross, A. S.; Hughes, M. C.; and Doshi-Velez, F. 2017. Right for the Right Reasons: Training Differentiable Models by Constraining their Explanations. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, 2662–2670. doi:10.24963/ijcai.2017/371.

Sebeok, T. A.; and Rosenthal, R. E. 1981. The Clever Hans phenomenon: Communication with horses, whales, apes, and people. *Annals of the New York Academy of Sciences* .

Shrikumar, A.; Greenside, P.; and Kundaje, A. 2017. Learning Important Features Through Propagating Activation Differences. In *Proceedings of the 34th International Conference on Machine Learning, ICML*, 3145–3153.

Simonyan, K.; Vedaldi, A.; and Zisserman, A. 2013. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034* .

Simonyan, K.; and Zisserman, A. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *Proceedings of the 3rd International Conference on Learning Representations, ICLR*.

Teso, S.; and Kersting, K. 2019. Explanatory Interactive Machine Learning. In *Proceedings of the 2nd AAAI/ACM Conference on AI, Ethics, and Society, AIES*.

Von Wright, G. H. 2004. *Explanation and understanding*. Cornell University Press.