MLVis International Workshop on Machine Learning in Visualisation for Big Data (2021)
D. Archambault, J. Peltonen, and I. Nabney (Editors)

Controllably Sparse Perturbations of Robust Classifiers for Explaining Predictions and Probing Learned Concepts

Jay Roberts[1] and Theodoros Tsiligkaridis[2]

MIT Lincoln Laboratory
[1] Homeland Sensors and Analytics Group
[2] Artificial Intelligence Technology Group

Abstract

Explaining the predictions of a deep neural network (DNN) in image classification is an active area of research. Many methods focus on localizing pixels, or groups of pixels, which maximize a relevance metric for the prediction. Others aim at creating local "proxy" explainers which aim to account for an individual prediction of a model. We aim to explore "why" a model made a prediction by perturbing inputs to robust classifiers and interpreting the semantically meaningful results. For such an explanation to be useful for humans it is desirable for it to be sparse; however, generating sparse perturbations can computationally expensive and infeasible on high resolution data. Here we introduce controllably sparse explanations that can be efficiently generated on higher resolution data to provide improved counter-factual explanations. Further we use these controllably sparse explanations to probe what the robust classifier has learned. These explanations could provide insight for model developers as well as assist in detecting dataset bias.

CCS Concepts

• ***Computing methodologies*** → *Machine learning; Artificial intelligence;*

1. Introduction

Deep Convolutional Neural Networks (CNNs) have revolutionized the field of computer vision [LBH15] and are increasingly being deployed across high stakes domains such as autonomous driving, medical diagnosis, and many others. Despite such proliferation, the high capacity complex nature of CNNs has made an encompassing theory of how they make their decisions elusive, with many end users treating CNNs as a "black box". This has led to thrusts in both academia and industry to establish frameworks of reliability and transparency for artificial intelligence as a whole [Pic18, Mic19, Lop20]. An additional concern for such high capacity models is their decisions may be unstable. Small perturbations of inputs can dramatically change a model's predictions [GSS15]. This work has been studied extensively in the field with many defenses proposed to make models robust to such attacks [KGB17, RDV18, MMS*18, MDUFF19].

Adversarial robustness conveys benefits beyond its original intent and may improve a wide class of explainability techniques. The improvement robustness provides to saliency maps has been studied before [ELMS19]. Figure 1 shows examples of common pixel attribution based methods, [STY17, STK*17] and how robustness leads improved visualizations over than their standard counterparts. Local linear proxy models have been used as explanation techniques [RSG16b, AMJ18, PASC*20]. There is ample work suggesting that the mechanism underlying adversarial robustness is the regularity (or local linearity) of the loss landscape for the model [LHL15, RDV18, MDUFF19, QMG*19], which can aid the search for these proxies. Finally, it has also been observed that robust models exhibit generative features that align with those a human would use to classify an image [STT*19, IST*19, EIS*19]. For these reasons we leverage adversarially robust models for xAI techniques that capture meaningful semantics.

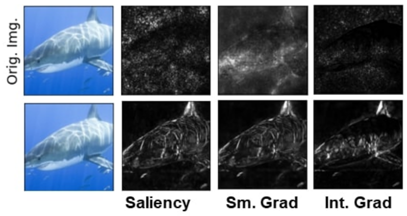


Figure 1: *A comparison of standard (top) and robust (bottom) models for various saliency explanation methods.*

1.1. Related Works

Many visual explanation techniques for image classification focus on pixel importance via gradient methods for prediction im-

DOI: 10.2312/mlvis.20211072

portance [STY17, STK*17] or class importance in deep features [BBM*15, SCD*19]. These methods attempt to answer the question "Where was the model focusing to make the prediction?". SUMMIT [HPRPC20] extends this work by providing attribution through the layers of activations in a network allowing users to visualize the representation hierarchy. Another interesting class of techniques is incorporating interpretable proxy models that mirror a model's prediction in a local neighborhood of an example [RSG16a, AMJ18, PASC*20, LAMJ18]. These methods translate questions about the model into questions about semantically meaningful features of the proxy models. A common issue with the above methods is the need to handle both instability of the predictions and explanations provided. This is ameliorated when using robust models as a starting point as we do here, or incorporating robustness into the explainability framework itself [AMJ18]. Our work is most similar to the image synthesis work in [STT*19] in that we are using a single robust classifier to generate images. However, our focus in this paper is on generating images for the purpose of explaining model predictions and probing for learned concepts.

1.2. Contributions

The primary goal of this paper is to demonstrate some explainability benefits of using robust image classifiers. We do this by giving examples of how to generate visual explanations using such models. Concretely we:

1. Demonstrate the ℓ_1-q sparse perturbations as a human interpretable efficient technique for generating visual counter-factual explanations
2. Provide a method for visualizing the concepts learned by a classifier using gradient-based optimization

2. Methods

Here we describe our algorithm for generating visual explanations of model predictions using the generative properties of robust classifiers [TSE*19, SIT*19]. Our explanations are perturbations of an input to the model which will make the model more or less confident in its prediction. Though optimizing the components of the logits or softmax layer of a network can be used for this we find that the cross entropy loss serves as a suitable surrogate and so in practice we aim to increase / decrease the loss in order to decrease / increase the model's confidence. Such perturbations are similar to, and in the case of increasing loss indeed are, adversarial attacks which for robust models generate perturbations and images that contain semantically meaningful features [WMR18, SHG20].

2.1. Perturbing for Visual Explanations

Given data (x, y_{gt}) from a dataset D, model f_θ, and model prediction y_p, our visulaizations are perturbations are defined as

$$\hat{\delta}(x, y_a; p, \varepsilon) = \underset{\delta \in B_p(\varepsilon)}{\operatorname{argmin}} \mathcal{L}(f_\theta(x+\delta),\ y_a) \tag{1}$$

where $B_p(\varepsilon)$ is the ℓ_p ball of radius ε. We say the perturbation is **label targeted** when $y_a = y_{gt}$ and say it is **prediction targeted** when $y_a = y_p$. In order to explain individual predictions we use small ε so that the original features remain present. In this case the perturbation will highlight existing features of an image or make small changes that result in a semantically meaningful difference in the original. To probe concepts we use a larger ε to allow the model to generate larger more visible features. Here, entire concepts may emerge, allowing us to see what features are most semantically meaningful to a model's representation of a class.

The constraint $p \in \{2, \infty\}$ are commonly used for adversarial attacks. Though both of these produce semantically meaningful features they result in dense perturbations which can be hard to interpret. In particular ℓ_∞ perturbations allow for equally sized perturbations of every pixel resulting in dense jagged images. The ℓ_2 perturbations are smoother but still dense. A natural solution to the problem of dense perturbations is to constrain the perturbations to the ℓ_1 ball; however, in practice such an optimization is inefficient and scales poorly with image resolution. We present an easy-to-implement modification of the ℓ_1 optimization that allows for efficient generation of controllably sparse perturbations.

For sake of brevity we focus on error explanation and concept probing, but note that explaining correct predictions can be done as well. To explain errors, we use prediction targeted perturbations to answer the questions

Why did the model make this mistake?

and label targeted perturbations to answer the question

What could have corrected this mistake?

2.2. The Optimization

In order to compute perturbations, we seek an approximate solution to the optimization in (1). This is done by an iterative gradient-based optimization algorithm. The common approach in adversarial robustness is to use Projected Gradient Decent [MMS*18]; however, this approach requires careful tuning of step size and projecting onto the ℓ_1 ball is non-trivial. For these reasons we use the Frank Wolfe (FW) approximation scheme [FW56, Jag13]. The algorithm is based on taking convex combinations of solutions to a sequence of linear approximations to the the original problem. It trivially handles the case of ℓ_1 constraints and is not sensitive to choice of step size. Though sparse attacks using FW have been done in the context of adversarial robustness [TB19, KSB*19, CZYG20], to our knowledge, this is the first work to use sparse FW perturbations as a means of explaining the predictions of robust classifiers.

A naive implementation of FW for the ℓ_1 constraint results in an approximation which modifies at most one pixel at every iteration, i.e., the pixel with the maximum gradient value is modified. While this guarantees sparsity of the perturbation, it becomes computationally prohibitive for high resolution images. Inspired by [KSB*19] we perform a modified FW ℓ_1 optimization wherein the pixels which are in the top-q percentile of gradient values are modified at each iteration. Our algorithm is simpler than that of [KSB*19] and so modifies the loss less aggressively, but it still efficiently produces successful adversarial examples and more importantly generates semantically meaningful visualizations. We call these ℓ_1-q perturbations and their generation is detailed in algorithm 1.

3. Results

In this section, we demonstrate how ℓ_1-q perturbations can be used in a counter-factual framework to help visually explain errors by showing (1) what features led to error and (2) what features could have corrected an error. We then demonstrate how the ℓ_1-q perturbations can be used to probe concepts the model has learned by having it perform basic generative tasks.

For all experiments we use a ResNet-50 architecture [HZRS16] trained with adversarial training to be robust to ℓ_2 adversaries on ImageNet [DDS*09]. We have experimented with training ℓ_1 and ℓ_1-q robust models on CIFAR-10 [KNH] but find that ℓ_2 robust models produce equally useful perturbations. The models used are hosted by MadryLab at [EIST19]. Though training such models can be computationally intensive, generating these visualizations takes less than 1 sec per image. Results were generated using 2 Nvidia Volta V100s.

Algorithm 1 ℓ_1-q Perturbation PseudoCode in PyTorch Style

```
Input: Network $f_\theta$, input image $x$, target label $y$, max number of steps $K$, maximum perturbation $\varepsilon$, constant $c \geq 1$, and percentile $q$.
Result: Perturbation $\delta$.
Initialize $\delta = 0$
for $0 \leq k < K$ do
    $\gamma = c/(c+k)$
    $\bar{\delta} = 0$
    $g = \nabla_\delta \mathcal{L}(f_\theta(x+\delta), y))$
    top_q = torch.topk($g$.abs(), $q * \dim(x)$)
    $\bar{\delta}$[top_q] = sgn($g$)[top_q]
    $\bar{\delta} = \bar{\delta}/||\bar{\delta}||_1$
    $\delta = \delta + (1-\gamma)\bar{\delta}$
end for
```

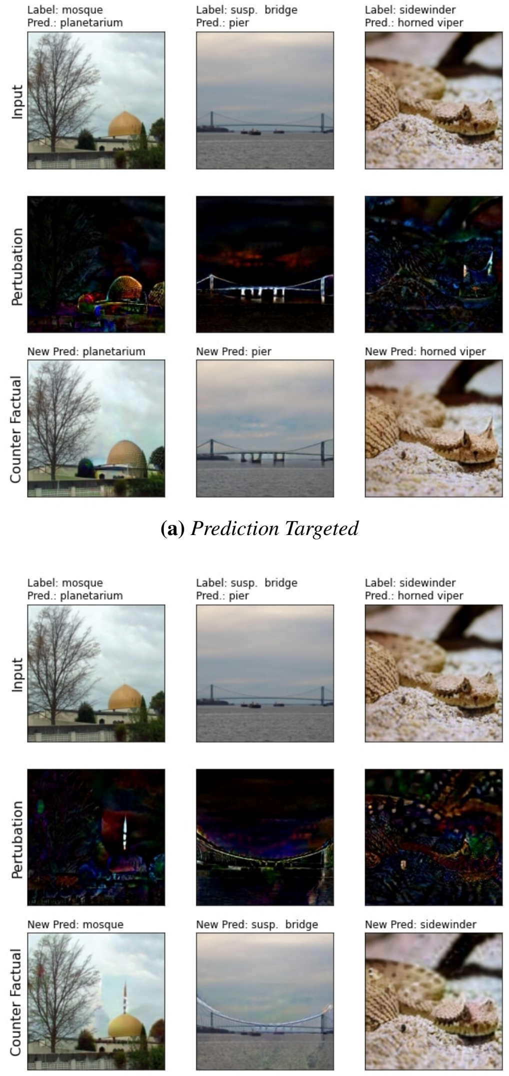
3.1. Error Counterfactuals

Figures 2a and 2b show counterfactual explanations for errors made by the model. The prediction targeted counterfactuals 2a answers the question: *Why did the model make this mistake?*

For example in the third column the model mistakes a sidewinder snake for a horned viper. To become more confident in this error the model makes the spikes on the eyes of the sidewinder more pronounced, making it appear more like the horned viper. This suggests that the model learned the characteristic that horned vipers have horn shaped eyes and that the pointy eyes common in sidewinders confused the model. The second column suggests that the ships under the bridge were mistaken as pier pilings.

Figure 2b shows label targeted counter-factuals, which aim to answer the question: *What could have corrected this mistake?*

We see the model corrects its prediction that the image is of a sidewinder if the image has more pronounced splotch patterns common in sidewinder snakes. However, the perturbations are rather diffuse. This indicates that the sidewinder-horned viper distinction may be a difficult one for the model. In the second column the suspension bridge prediction would have benefited from a more pronounced suspension cable. Of particular interest is the first column where the model misidentifies a mosque. It appears that the model would be more confident if there were a minaret emanating from the mosque dome. This suggests that mosques in the dataset may be biased towards having minarets.



(a) *Prediction Targeted*

(b) *Label Targeted*

Figure 2: *Prediction (a) and Label (b) targeted counterfactual explanations of errors using an ℓ_2 attack with, $\varepsilon = 12$, and 30 iterations. The Perturbations row has been magnified 5x for visibility.*

3.2. Visual Concept Probing

In order to probe what concepts a model has learned we ask the model to draw a picture of a given class on a "blank" image. A litany of more complex generative tasks have been demonstrated in [STT*19]; however, here we focus on a very general problem of visualizing the learned features of the model and further the ℓ_1-q perturbations constrain the perturbation to be localized which makes them easier to interpret than their ℓ_2 or ℓ_∞ counterparts. For each experiment we perform a label targeted ℓ_1-q perturbation of a "blank" input image. Since the ℓ_1-q algorithm is deterministic we choose input images who's components are drawn form a standard normal distribution and then perform a min-max scaling to force the pixel values into [0, 1].

For these experiments we also include the ℓ_2 perturbations as they have shown additional texture like features not present in the ℓ_1-q perturbations. All attacks are performed with a 30 steps of FW optimization. The ℓ_2 attacks use $\varepsilon_2 = 35$ and the ℓ_1-q attacks us $\varepsilon_1 = \varepsilon_2 * 224 * \sqrt{3}$. The scaling is chosen so that the corners of the ℓ_1 ball intersect with the ε_2 ℓ_2 ball.

We present the resulting images in Figure 3 for four classes (Jay, Rhinoceros Beetle, White Shark, and Mushroom). For each class we show an example input and the ℓ_1-q perturbations on the top row and the ℓ_2 perturbations on the bottom row. We note that these examples **were minimally curated** and due to the stability provided by robust models the phenomena is quite generic. Both the ℓ_1-q and ℓ_2 perturbations produce features that are semantically meaningful. The ℓ_1-q perturbations are more localized than those of the ℓ_2 which seem to capture more textures.

A few features are particularly noteworthy. Firstly, we can see that in the case of the rhinoceros beetle, both perturbations produced the distinctive horn feature. In the case of the jay, both perturbations contain the correct color (blue) and the correct wing stripe patterns that distinguish the jay from other common birds. This provides evidence that the model has learned these discriminating features. The white shark and mushroom perturbations are interesting because the ℓ_1-q and ℓ_2 perturbations are quite different. When forced to localize the perturbations (ℓ_1-q) we see for white sharks that the model produces a fin and the distinctive black-white patter of a white shark body, and for mushrooms we get fully formed fungi with caps and stems. However, the ℓ_2 perturbations of white sharks seem to add a dive-cage. This suggests that the white shark class may have a bias as to being viewed from behind a shark cage. In the case of the mushrooms the ℓ_2 perturbations are mostly the gill texture and the colors are diffuse, suggesting the model has learned not only the general shape of mushrooms but their structures such as gills.

4. Conclusion

In this work we present a method for generating sparse input perturbations with controllable degree of sparsity. We couple this technique with the generative properties of robust models to explore a counterfactual framework for explaining individual predictions, and a generative method for probing the concepts learned by the model. We believe these techniques can be used as a starting point to better understand what a model has learned, discover biases in a training dataset, identify common failure cases, and provide users with salient features used in a predictions. Future work may focus on aggregating the concepts across a set of examples to improve our understanding of deep learning models.

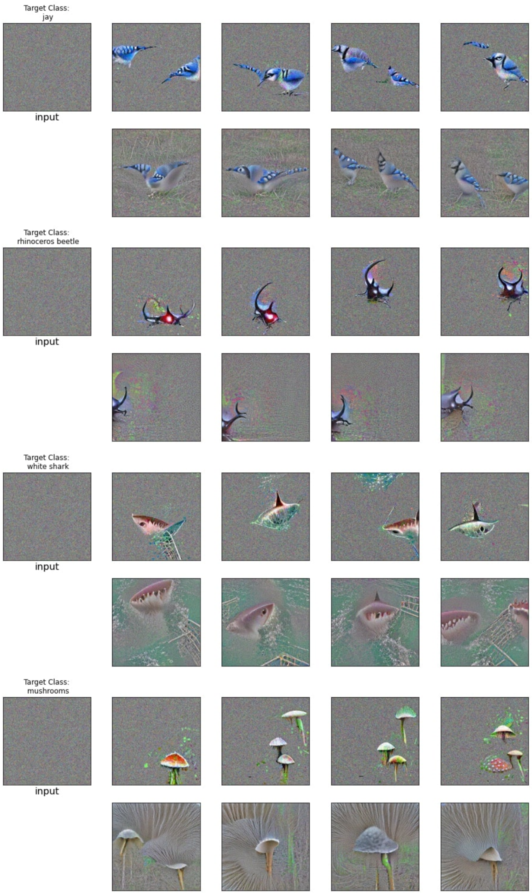


Figure 3: *Visualizing concepts for various ImageNet Classes. For each class we show label targeted ℓ_1-q (top row) and ℓ_2 (bottom row) perturbations. Each column is generated with a different random image with minimal curation.*

Acknowledgements

We thank the MLVis 2021 reviewers for their time and helpful comments. Further, the authors acknowledge the MIT SuperCloud and Lincoln Laboratory Supercomputing Center for providing HPC resources that have contributed to the research results reported within this paper.

Research was sponsored by the United States Air Force Research Laboratory and the United States Air Force Artificial Intelligence Accelerator and was accomplished under Cooperative Agreement Number FA8750-19-2-1000. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the United States Air Force or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

References

[AMJ18] Alvarez-Melis D., Jaakkola T. S.: Towards robust interpretability with self-explaining neural networks, 2018. arXiv:1806.07538. 1, 2

[BBM*15] Bach S., Binder A., Montavon G., Klauschen F., Muller K.-R., Samek W.: On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS one 10, 7 (2015). 2

[CZYG20] Chen J., Zhou D., Yi J., Gu Q.: A frank-wolfe framework for efficient and effective adversarial attacks. In Thirty-Fourth AAAI Conference on Artificial Intelligence (2020). 2

[DDS*09] Deng J., Dong W., Socher R., Li L.-J., Li K., Fei-Fei L.: Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (2009), Ieee, pp. 248–255. 3

[EIS*19] Engstrom L., Ilyas A., Santurkar S., Tsipras D., Tran B., Madry A.: Adversarial robustness as a prior for learned representations, 2019. arXiv:1906.00945. 1

[EIST19] Engstrom L., Ilyas A., Santurkar S., Tsipras D.: Robustness (python library), 2019. URL: https://github.com/MadryLab/robustness. 3

[ELMS19] Etmann C., Lunz S., Maass P., Schonlieb C.-B.: On the connection between adversarial robustness and saliency map interpretability. In ICML (2019). 1

[FW56] Frank M., Wolfe P.: An algorithm for quadratic programming. Naval research logistics quarterly 3 (1956), 95–110. 2

[GSS15] Goodfellow I. J., Shlens J., Szegedy C.: Explaining and harnessing adversarial examples. In International Conference on Learning Representations (2015). 1

[HPRPC20] Hohman F., Park H., Robinson C., Polo Chau D. H.: Summit: Scaling deep learning interpretability by visualizing activation and attribution summarizations. IEEE Transactions on Visualization and Computer Graphics 26, 1 (2020), 1096–1106. doi:10.1109/TVCG.2019.2934659. 2

[HZRS16] He K., Zhang X., Ren S., Sun J.: Deep residual learning for image recognition. In CVPR (2016). 3

[IST*19] Ilyas A., Santurkar S., Tsipras D., Engstrom L., Tran B., Madry A.: Adversarial examples are not bugs, they are features, 2019. arXiv:1905.02175. 1

[Jag13] Jaggi M.: Revisiting frank-wolfe: Projection-free sparse convex optimization. In ICML (2013), pp. 427–435. 2

[KGB17] Kurakin A., Goodfellow I. J., Bengio S.: Adversarial machine learning at scale. In International Conference on Learning Representations (2017). 1

[KNH] Krizhevsky A., Nair V., Hinton G.: Cifar-10 (canadian institute for advanced research). URL: http://www.cs.toronto.edu/~kriz/cifar.html. 3

[KSB*19] Kang D., Sun Y., Brown T., Hendrycks D., Steinhardt J.: Transfer of adversarial robustness between perturbation types. arXiv preprint arXiv:1905.01034 (2019). 2

[LAMJ18] Lee G.-H., Alvarez-Melis D., Jaakkola T. S.: Game-theoretic interpretability for temporal modeling, 2018. arXiv:1807.00130. 2

[LBH15] LeCun Y., Bengio Y., Hinton G.: Deep Learning. Nature 521, 7533 (2015), 436–444. 1

[LHL15] Lyu C., Huang K., Liang H.-N.: A unified gradient regularization family for adversarial examples. In IEEE International Conference on Data Mining (ICDM) (2015). 1

[Lop20] Lopez C. T.: DOD Adopts 5 Principles of Artificial Intelligence Ethics, 2020. URL: https://www.defense.gov/Explore/News/Article/Article/2094085/dod-adopts-5-principles-of-artificial-intelligence-ethics/. 1

[MDUFF19] Moosavi-Dezfooli S.-M., Uesato J., Fawzi A., Frossard P.: Robustness via curvature regularization, and vice versa. In IEEE Conference on Computer Vision and Pattern Recognition (2019). URL: https://openaccess.thecvf.com/content_CVPR_2019/papers/Moosavi-Dezfooli_Robustness_via_Curvature_Regularization_and_Vice_Versa_CVPR_2019_paper.pdf. 1

[Mic19] Microsoft: Microsoft AI principles, 2019. URL: https://www.microsoft.com/en-us/ai/responsible-ai. 1

[MMS*18] Madry A., Makelov A., Schmidt L., Tsipras D., Vladu A.: Towards deep learning models resistant to adversarial attacks. In International Conference on Learning Representations (2018). URL: https://openreview.net/forum?id=rJzIBfZAb. 1, 2

[PASC*20] Plumb G., Al-Shedivat M., Cabrera A. A., Perer A., Xing E., Talwalkar A.: Regularizing black-box models for improved interpretability, 2020. arXiv:1902.06787. 1, 2

[Pic18] Pichai S.: AI at Google: our principles, 2018. URL: https://www.blog.google/technology/ai/ai-principles/. 1

[QMG*19] Qin C., Martens J., Gowal S., Krishnan D., Dvijotham K., Fawzi A., De S., Stanforth R., Kohli P.: Adversarial robustness through local linearization. In NeurIPS (2019). 1

[RDV18] Ros A. S., Doshi-Velez F.: Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In AAAI Conference on Artificial Intelligence (2018). 1

[RSG16a] Ribeiro M., Singh S., Guestrin C.: why should i trust you? explaining the predictions of any classifier. In ACM SIGKDD Intl. Conference on Knowledge Discovery and Data Mining (2016). 2

[RSG16b] Ribeiro M. T., Singh S., Guestrin C.: "why should i trust you?": Explaining the predictions of any classifier, 2016. arXiv:1602.04938. 1

[SCD*19] Selvaraju R. R., Cogswell M., Das A., Vedantam R., Parikh D., Batra D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. International Journal of Computer Vision 128, 2 (Oct 2019), 336âĂŞ359. URL: http://dx.doi.org/10.1007/s11263-019-01228-7, doi:10.1007/s11263-019-01228-7. 2

[SHG20] Sharma S., Henderson J., Ghosh J.: Certifai: A common framework to provide explanations and analyse the fairness and robustness of black-box models. In AIS (2020). 2

[SIT*19] Santurkar S., Ilyas A., Tsipras D., Engstrom L., Tran B., Madry A.: Image synthesis with a single (robust) classifier. In NeurIPS (2019). 2

[STK*17] Smilkov D., Thorat N., Kim B., ViÃľgas F., Wattenberg M.: Smoothgrad: removing noise by adding noise, 2017. arXiv:1706.03825. 2

[STT*19] Santurkar S., Tsipras D., Tran B., Ilyas A., Engstrom L., Madry A.: Image synthesis with a single (robust) classifier, 2019. arXiv:1906.09453. 1, 2, 4

[STY17] Sundararajan M., Taly A., Yan Q.: Axiomatic attribution for deep networks, 2017. arXiv:1703.01365. 1, 2

[TB19] TramÃĺr F., Boneh D.: Adversarial training and robustness for multiple perturbations, 2019. arXiv:1904.13000. 2

[TSE*19] Tsipras D., Santurkar S., Engstrom L., Turner A., Madry A.: Robustness may be at odds with accuracy. In International Conference on Learning Representations (2019). URL: https://openreview.net/forum?id=SyxAb30cY7. 2

[WMR18] Wachter S., Mittelstadt B., Russell C.: Counterfactual explanations without opening the black box: Automated decisions and the gdpr. Harvard Journal of Law & Technolog 31, 2 (2018). 2