A Review of Self-Explaining Neural Networks

Rico Mossinkoff
12805157
ricokoff@hotmail.com

Yke Rusticus
11306386
yke.rusticus@student.uva.nl

Roberto Schiavone
12883980
r.schiavone@student.vu.nl

Ewoud Vermeij
11348860
ewoudvermeij@gmail.com

ABSTRACT

Nowadays, machine learning models are getting more and more complex. This causes them to be less transparent and understandable as they behave as a 'black box'. To improve the transparency, most researches focus on a posteriori explanations of such models. In the research of Alvarez-Melis and Jaakkola [1] a new approach is used, where human-understandable explanations are generated during training of the model, using learned concepts for high-dimensional data. In this work, we have attempted to reproduce this research and extend it with a more sophisticated method of visualizing the concepts. While we were partly successful in replicating the research, we found that the concepts are less interpretable than was claimed by the authors. Nevertheless, the proposed self-explaining neural network (SENN) is a step forwards toward more transparent AI, and more research into obtaining interpretable basis concepts is therefore recommended.

1 INTRODUCTION

In the field of machine learning, the tasks to solve are getting more and more complex. Systems needed for this are obviously getting more complex too. This causes them to behave as a "black box". The output of such systems can be biased and hard to explain. This biased output has implications on the fairness of such a system. In the field of fairness and transparency, systems are analyzed to find out how to make systems fair and easier to understand. In most research, this is accomplished by a posteriori explanation of an already trained model. However, there are some limitations with this approach [4] [10]. In this work we partially reproduced the research of Alvarez-Melis and Jaakkola [1]. The authors come up with a new approach to obtain explainable models. They propose a bottom up way of designing models such that they allow for human-understandable explanations. For example, they argue that this can be achieved by reducing high dimensional input data to interpretable basis concepts, such that each of them ensures fidelity, diversity and grounding. Predictions can then be explained directly by the relevance score of each concept. In theory, this approach should lead to interpretable models if the concepts themselves are interpretable. In this work, we have extended the research by generating synthetic images that visualize the concepts. In order to do this, the authors of the original work proposed to maximize the activation of one concept, while reducing the activation of the others. We have developed several criteria that the visualizations aim to fulfill, by using the proposed criterion from Alvarez-Melis and Jaakkola with several extensions and adjustments. These extensions were necessary steps towards more interpretable synthetic images, as we will show in the next sections.

2 METHODS

The authors propose a self-explaining neural network (SENN). A generalized form of the model can be written as follows:

$$f(x) = \theta(x)^T h(x) \tag{1}$$

This formula stems from a simple linear model of the form: $f(x) = \theta^T x$. The authors describe a linear model to be the basis for a self-explaining model for three reasons: i) input features are in line with the observations, ii) each parameter θ_i provides a positive or negative contribution of input x_i to the final prediction, iii) the aggregation of feature specific terms θ_i is additive without conflating feature-by-feature interpretation of impact. In SENN, $\theta(x)$ is typically a complex model configuration, while still maintaining (locally) linear properties. $\theta(x)$ is refered to as the "parametrizer". $h(x)$ represents the "conceptizer", a function able to transform high dimensional input into low dimensional, easy digestible output. Depending on the objective, $h(x)$ can be defined as pre-grounded feature extractors, an aggregation of the input, learned representations or simply $h(x) = x$. For further generalization, the authors consider a more general function for the aggregation of all elements $z_i = \theta(x)_i h(x)_i$ such that different types aggregations can be applied:

$$f(x) = g(z_1, ..., z_k) \tag{2}$$

To preserve the desired interpretation of $\theta(x)$ and $h(x)$, the aggregation should (i) be permutation invariant, (ii) isolate the effect of individual $h(x)_i$'s and (iii) preserve the sign and relative magnitude of the impact of the relevance values $\theta(x)_i$. For the purposes of this work we have taken $g(\cdot) = \text{sum}(\cdot)$, as to reproduce the results presented in the original paper.

In order to let the elements of $\theta(x)$ act as linear coefficients on $h(x)$ and preserve interpretability, special properties are needed. We must ensure that close inputs x and x_0 should not result in significant difference in $\theta(x)$ and $\theta(x_0)$ and therefore must act locally linear. We quote the following definition from the original paper:

Definition 2.1. *$f : \mathcal{X} \subseteq \mathbb{R}^n \to \mathbb{R}^m$ is **locally difference bounded** by $h : \mathcal{X} \subseteq \mathbb{R}^n \to \mathbb{R}^k$ if for every x_0 there exists $\delta > 0$ and $L \in \mathbb{R}$ such that $\|x - x_0\| < \delta$ implies $\|f(x) - f(x_0)\| \leq L\|h(x) - h(x_0)\|$*

Then we can call (2) a self-explaining model where:

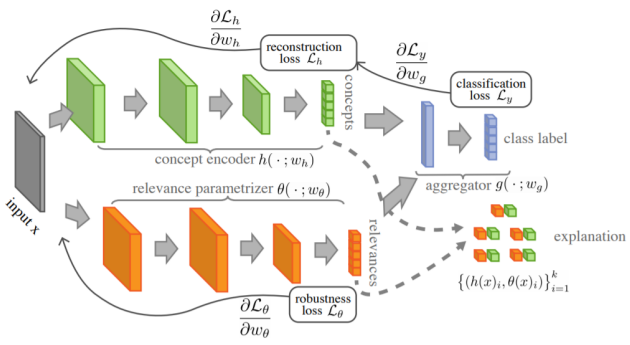

Figure 1: Structure of the SENN system, taken from the original paper [1].

P1) g is monotone and completely additively separable
P2) For every z_i, g satisfies $\frac{\partial g}{\partial z_i} / ge 0$
P3) θ is locally difference bounded by h
P4) $h(x)_i$ is an interpretable representation of x
P5) k is small

For (high-dimensional) images, $h(x)$ is realized using an autoencoder that learns (low-dimensional) basis concepts. Each image class activates a combination of concepts differently. The authors propose three desiderata for interpretable concepts:

(1) **Fidelity:** *The representation of x in terms of concepts should preserve relevant information.* Enforced by training h as an autoencoder.
(2) **Diversity:** *Inputs should be representable with few non-overlapping concepts.* Enforced by using a sparse autoencoder.
(3) **Grounding:** *Concepts should have an immediate human understandable interpretation.* Not enforced, but tested subjectively using visualizations of the concepts.

Finally, the SENN model is trained by optimizing for the loss:

$$\mathcal{L} = \mathcal{L}_y(f(x), y) + \lambda \mathcal{L}_\theta(f) + \xi \mathcal{L}_h(x, \hat{x}). \tag{3}$$

The first term is the (cross-entropy) prediction loss. The second term is the gradient regularizer for parametrizer θ, weighted by the hyperparameter λ, which effectively makes the model more stable, i.e. more (locally) linear. The third and final term is the reconstruction loss of the conceptizer h, given input image x and reconstruction $\hat{x}$, and weighted by ξ. For more details on the individual loss terms we refer to the original paper [1].

As an extension to the original paper, we have implemented several different methods for visualizing the concepts in the case that h is learnt. In the paper, a concept i is visualized by the samples in the data that maximize its activation, i.e. by the set $X^i = \arg\max_{\hat{X} \subseteq X} \sum_{x \in \hat{X}} h(x)_i$. However, there is one thing to be noted. $\theta(x)$ and $h(x)$ can both be negative and since they are multiplied, the minimum of an activation can be as expressive as the maximum, and possibly weigh even more into the final prediction of the model. So first of all, we have re-implemented this method such that concepts are visualized by both the samples in the data that maximize and minimize their activations.

Secondly, we have developed a way to generate synthetic images that visualize the learned concepts. For concept i, the optimal synthetic image is given by $\hat{x}^i = \arg\min_x \mathcal{L}^i(x)$, for a given loss function $\mathcal{L}^i(x)$. To optimize for $\mathcal{L}^i(x)$, we used gradient descent over the input image x while keeping all pre-trained model parameters fixed. In the simple case where we just want to max- or minimize the activation of a given concept i, the loss function is:

$$\mathcal{L}^i(x) = -s \cdot h(x)_i, \tag{4}$$

where $s \in \{-1, +1\}$ specifies whether we want to minimize or maximize activation $h(x)_i$, respectively. To extend this loss function, we may add a term that penalizes activation of the other concepts, where we introduce parameter α to allow for adjusting the weight of the penalty:

$$\mathcal{L}^i = -s \cdot h(x)_i + \alpha \mathcal{L}^i_h(x) \tag{5}$$

The authors of the paper propose a form equivalent to the case where $\mathcal{L}^i_h = s \cdot \sum_{j \neq i} h(x)_j$. We will refer to this as "difference forcing" between the activations of concept i and all other concepts. This form is flawed, however, which becomes clear if we recall that negative and positive concept activations should be considered as equal in importance. In this case that means that by "difference forcing" we do not only enforce the generated image to resemble one concept, but also all other concepts' negative (or positive) counterpart. This can be fixed by instead using $\mathcal{L}^i_h = \frac{1}{2} \sum_{j \neq i} h(x)_j{}^2$, an L2-norm penalty often used in machine learning. This penalty encourages all other concept activations to be close to zero, rather than driving them all to the opposite extreme of the activation of concept i. We therefore refer to the usage of this form as "zero forcing". In this case the generated image should only resemble one concept (for either high activation or low activation).

Finally, we add one more regularization term to the loss function. Motivated by the fact that all input of the model is normalized in preprocessing, we add the Kullback-Leibler (KL) divergence between two Gaussian distributions to the loss. For two distributions $x \sim \mathcal{N}(\mu, \sigma^2)$ and $x^* \sim \mathcal{N}(0, 1)$, the KL-divergence is given by[1]:

$$D_{\mathrm{KL}}(x \| x^*) = -\log \sigma + \frac{\sigma^2 + \mu^2 - 1}{2}. \tag{6}$$

This term, which we will write $D_{\mathrm{KL}}(x)$ for short, penalizes any deviation of x from a normal distribution. The loss becomes:

$$\mathcal{L}^i = -s \cdot h(x)_i + \alpha \mathcal{L}^i_h(x) + \beta D_{\mathrm{KL}}(x) \tag{7}$$

where we have introduced β to allow for adjusting the weight of the penalty. A benefit that this penalty might bring is the potential of "flattening out" or de-noising the generated image, such that only the most important parts of the image remain present. This could be used to enhance the interpretability of the generated images. An visualization of the whole SENN system can be found in Figure 1

3 EXPERIMENTAL SETUP

In our effort to reproduce the paper's original results, we focused only on the results obtained with SENN on the COMPAS[2] and

[1] https://stats.stackexchange.com/questions/7440/kl-divergence-between-two-univariate-gaussians
[2] https://github.com/adebayoj/fairml/blob/master/doc/example_notebooks/propublica_data_for_fairml.csv

MNIST[3] datasets. The COMPAS dataset consists of samples labeled with criminal recidivism (or relapse) risk scores. These samples are represented using demographic features, so this dataset is well suited for testing the fairness of a model. The MNIST dataset consists of hand-written digits in the form of 28×28 images.

Specifically, we have attempted to reproduce: clear and understandable explanations for the predictions for samples in the datasets, and visualizations of the concepts; removable features that capture the relevance of each concept; the evidence for the tradeoff between stability and prediction accuracy; and several visualizations of the models' robustness for different values of λ (see equation (3)). For the extended part we attempted to visualize the concepts with out sythetic images produced with different parameter settings of α and β from equation (7). We compared the synthetic images to the prototypes of each concept and evaluated their explicitness and intelligibility.

As described in Section 2, the authors state a number of desiderata for interpretable concepts.

The authors also state three criteria to evaluate the interpretability of the whole approach:

- **Explicitness/Intelligibility**: *Are the explanations immediate and understandable?*
- **Faithfulness**: *Are relevance scores indicative of "true" importance?*
- **Stability**: *How consistent are the explanations for similar/ neighboring examples?*

Finally, it is important to evaluate the predictive performances of all models and determine whether the approaches suffer from performance losses. We have determined whether our models are compliant with all desiderata the authors apply to their models. The results are be discussed in Section 4.

We enriched the available open source[4] code provided by the authors and integrated the extension for our further research. Six different model configurations have been trained on COMPAS and eighteen different configurations on MNIST.

All our models are trained with different configurations to evaluate difference in interpretability and predictive performances. For each configuration, we used the Adam optimizer [3] for training with initial learning rate $l = 2 \times 10^{-4}$ and we set $\xi = 2 \times 10^{-5}$, in case h is learnt. The architectures are used exactly as specified in the paper [1]. For COMPAS, this means for the conceptizer that $h(x) = x$. For MNIST we have trained models for which: $h(x)$ maps to input x; $h(x)$ maps input x to 5 concepts; and $h(x)$ maps input x to 20 concepts. Each model is trained for various regularization strengths $\lambda \in \{0, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$. An overview for the MNIST model configurations is given in the Appendix, Table 1.

4 RESULTS

4.1 Predictive preformance

In general we can conclude that the reproduction of the models trained on COMPAS and MNIST are successful in terms of performance. Predictive performances of our COMPAS models show similar accuracies compared to the scores achieved by the authors,

Figure 2: Sample image, relevance scores $\theta(x)_i$, and prototypes per concept, taken from the dataset.

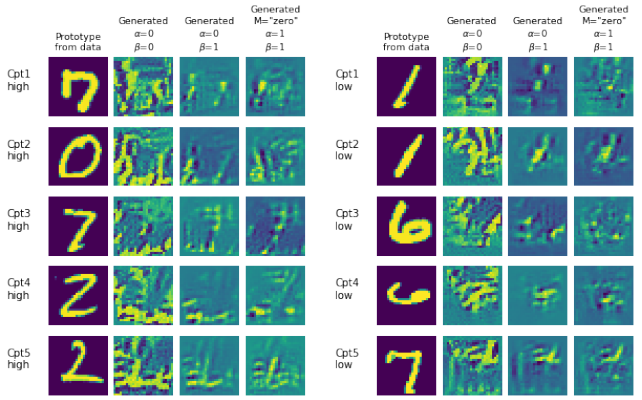

Figure 3: Synthetic visualizations of all concepts. (Left) positively activated, and (right) negatively activated. $\lambda = 10^{-1}$ was used.

which are around 80%. Exception to this is the model trained using a regularization strength $\lambda = 1$, giving a poor predictive performance of only 67.54%. For the models trained on the MNIST dataset we see a similar performance deficit when using $\lambda = 1$. All other MNIST models perform similar to the models of the authors, around 98.5%. Unfortunately, we could not find the issue among models trained with regularization strength $\lambda = 1$. Accuracies of all model configurations are given in the Appendix, Figure 8.

4.2 Explicitness/Intelligibility

The authors show how an example image is explained by SENN through the relevance scores provided by the parametrizer $\theta(x)$, together with the concepts prototypes in order to clarify the final predictions. In comparison, Figure 2 shows the explanation of a sample image representing a 4, using the same method as described in the paper. In this figure, it seems that concepts 4 and 5 capture highly similar shapes, as the best prototypes are all 2's. Even though this is the case, they have opposite relevance scores. Additionally, concept 1, which actually has some 4's present as prototypes, has a relevance score that is similar to concepts 3 and 5.

This makes the explanation not very understandable if we compare the explainable behaviour to the original paper.

Our synthetic images activating each concept most positively and negatively with their top 1 prototype for comparison are shown in Figure 3. Unfortunately, the synthetic images do not give a more clear, human interpretable, view on the explanations of SENN. It is shown that we can reduce the noise by setting β to 1, effectively

[3] http://yann.lecun.com/exdb/mnist/
[4] https://github.com/dmelis/SENN

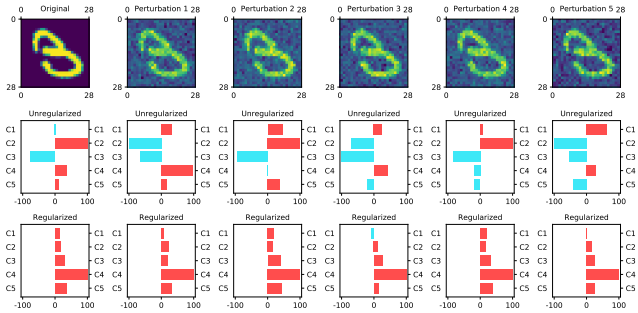

Figure 4: Stability of an unregularized MNIST model versus a gradient-regularized model using $\lambda = 10^{-1}$ for varying levels of gaussian perturbation.

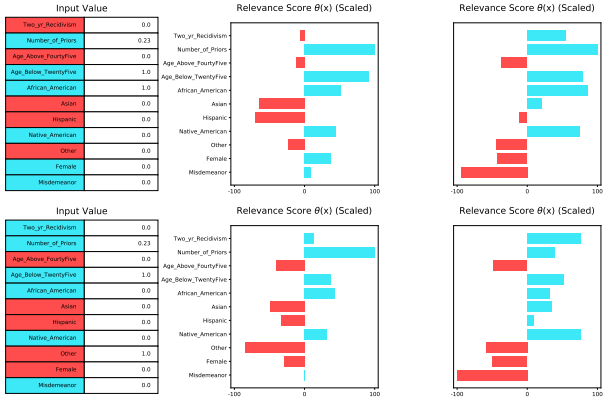

Figure 5: Prediction explanation for two individuals differing in only on the protected variables (`African_American` and `other`) in the COMPAS dataset. The second column shows the method trained with no regularization $\lambda = 0$, while the third column shows the gradient-regularized SENN ($\lambda = 10^{-1}$).

encouraging a gaussian distribution over the generated image and making the important shapes more clear. Some strokes of the prototype seem to match the strokes in the synthetic images. But the synthetic images are mainly vague and not interpretable at all. Even while the performance are not inferior to the performances of the authors.

4.3 Stability

Overall, we were able to partially reproduce the findings concerning stability. As Figure 4 shows, a difference in stability can be noted between an unregularized and a regularized model in the prediction explanations for a sample digit undergoing increasing levels of perturbation. The regularized model shows in this case more stability than the unregularized model.

For the stability on the COMPAS dataset we get less similar results compared to those of the original paper. As shown in Figure 5, a regularized and an unregularized SENN are comparibly stable for perturbations in two variables. This shows that besides showing transparency, maintaining stability can contribute to a more fair predicting model.

Figure 6 shows the local Lipschitz constant L estimates (see definition 2.1), and prediction accuracies for varying regularization strengths. This figure suggest, as was proposed in the original paper, that stability (and therefore interpretability) comes at the price of performance.

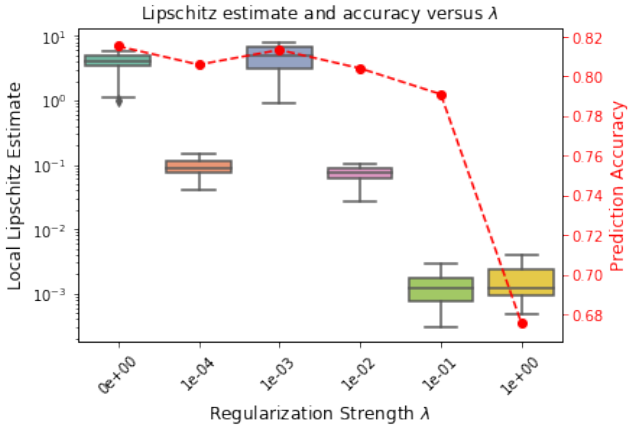

Figure 6: Effect of regularization on SENN's performance for the COMPAS models.

4.4 Faithfulness

Our approach to estimate feature relevance mirrors the one suggested by the paper, hence the *faithfulness* of relevance scores with respect to the model they are explaining relies on the fact that we can measure the importance of the features by observing the effect of removing them on the model's prediction.

Figure 7 shows an example of the relationship between the feature relevance for the prediction, and the probability drop when that feature is removed. As the image shows, a relationship between probability drop and relevance removal is hardly present. Whereas in the original paper, removal of high relevance features seems to give a relatively large drop, this image shows that removing a feature with low relevance, has similar or sometimes more effect on the probability drop. This is again due to the fact that concept activations can be negative as well. Given that the final prediction is highly dependent on the size and sign of the concept activations, we cannot simply explain and measure faithfulness by removing relevance scores.

5 DISCUSSION

In our attempt to reproduce the research of Alvarez-Melis and Jaakkola [1], we encountered a few issues making the reproduction less successful than we expected. Most accuracies, stability and fairness evaluations were on par with the original paper. However, the explainability of our SENN models were not as interpretable as was claimed in the origininal paper, including our own synthtic representations of the basis concepts. Whereas the authors visualize concepts by the samples in the data that activate them the most, we visualized the concepts by generating images that directly activate

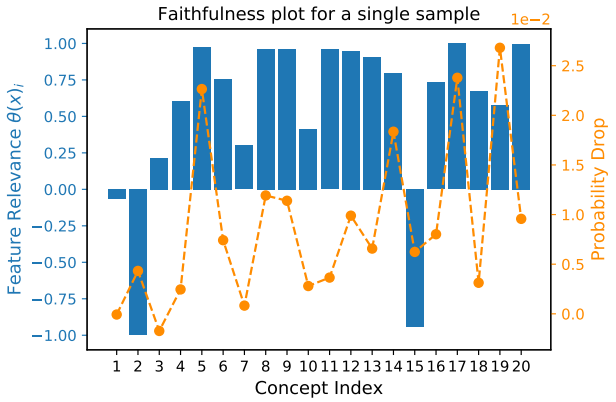

Figure 7: Faithfulness evaluation SENN on MNIST with learnt concepts for a single digit.

a desired concept. Our results suggest that the learned concepts lack human interpretability.

Furthermore, we found that an important note was missing from the original paper, namely that both the relevance scores $\theta(x)_i$ and the concept activations $h(x)_i$ can be negative. This means that a negative relevance score is in fact as important as a positive relevance score of equal strength, which has several implications on the presented results. And which can partly explain why some of the original results were not fully reproducible.

When the conceptizer is taken to be $h(x) = x$, SENN shows stability and accuracy results comparible with the paper. This demonstrates that SENN is a good candidate for a transpanrent models that allows for direct fairness testing by manipulating features. However, more research must be done towards a more interpretable conceptizer h.

Similarly to as proposed by the authors, further research could explore the (convolutional) conceptizer $h(x)$, by visualizing directly the filters in the network.

6 BROADER IMPLICATIONS

While this paper focuses on the transparency and interpretability of a models' decision, transparency can help to determine how fair a models' decision is. This is especially interesting for the models trained on the COMPAS dataset, where proper transparency could reveal an unfair bias in the recidivism risk score among, for example, different ethnicities. When present, one could take action to reduce an unfair bias. In earlier research in this field a method is proposes a method to apply decision and fairness criteria to a decisive model [5]. One could for example, enforce equal selection rates among groups or apply an equal opportunity policy making the recidivism risk independent of a certain group an individual is part of. Further research must show how the transparent models react on these kinds of fairness enforcements.

Another important domain in AI is accountability. Because the complexity of AI solutions increases and the internals of such solutions are unknown, accountability is often a challenge [8]. An organization using a specific system can only really be held accountable for it if it knows why the system makes decisions a certain way. As this paper focuses on explaining decisions of a system, it helps improving the transparency, and therefore the accountability. Organizations will know where to alter the workflow to realize a fairer output and can be held accountable if they do not change the system.

In the field of confidentiality, SENN has thus far no application. For example, differential privacy concerns the protection of user personal data in a models' final output [2, 7]. Since SENN uses input data directly for explaining its predictions, the input data is not protected (it can be derived from the output). The conflict between applying both transparency and confidentiality lies at the heart of the implementation of SENN.

SENN stands on the shoulders of giants like LIME [9] and SHAP [6] and provides a novel approach to tackle the transparency problem in machine learning. Compared to the previous explainability methods, SENN is the right step forward in embedding the explanations of the predictions inside the model, no longer inferred *a posteriori*.

7 CONCLUSIONS

In summary, we have been able to fully or partly reproduce to following:

- A model able to directly explain predictions by relevance scores $\theta(x)_i$.
- Increasing stability in explanations for increasing regularization strengths.
- Evidence for the tradeoff between stability and model accuracy.

We have not been able to reproduce:

- A direct and clear faithfulness measure of the relevance scores.
- Clear and understandable concepts, as illustrated by synthetically generated images as extension to the original paper.

As a useful aid, openly available source code was used to come to these results. However, as the code initially contained major bugs and unorganised comments, we suspect this not to be the final version of the code that was used for the presented paper. Therefore we assign the badge "Artifact Available" from the ACM badging system, though only "Functional". The work presented here is only partly eligible for the badge "Results Replicated", as the subsequent interpretations of the authors of the results are inappropriate and different from ours, for the case that concepts are used instead of raw input. Nevertheless, SENN shows great potential of bringing more transparency in artificial intelligence, if further research is focused on the interpretability of the proposed concepts.

REFERENCES

[1] David Alvarez-Melis and Tommi S. Jaakkola. 2018. Towards Robust Interpretability with Self-Explaining Neural Networks. *32nd Conference on Neural Information Processing Systems (NeurIPS 2018)* 2, 1 (Dec. 2018), 1–16. https://doi.org/10.5555/3327757.3327875

[2] Matthew Jagielski, Michael Kearns, Jieming Mao, Alina Oprea, Aaron Roth, Saeed Sharifi-Malvajerdi, and Jonathan Ullman. 2018. Differentially private fair learning. *arXiv preprint arXiv:1812.02696* (2018).

https://www.acm.org/publications/policies/artifact-review-badging

[3] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[4] Oscar Li, Hao Liu, Chaofan Chen, and Cynthia Rudin. 2017. Deep Learning for Case-Based Reasoning through Prototypes: A Neural Network that Explains Its Predictions. *AAAI Conference on Artificial Intelligence* 2 (Nov. 2017), 8.

[5] Lydia T. Liu, Sarah Dean, Esther Rolf, Max Simchowitz, and Moritz Hardt. 2018. Delayed Impact of Fair Machine Learning. *Proceedings of the 35th International Conference on Machine Learning* 4 (April 2018), 1–37.

[6] Scott M Lundberg and Su-In Lee. 2017. A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.). Curran Associates, Inc., 4765–4774. http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf

[7] H Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. 2017. Learning differentially private recurrent language models. *arXiv preprint arXiv:1710.06963* (2017).

[8] José Mena, Oriol Pujol, and Jordi Vitrià. 2020. Dirichlet uncertainty wrappers for actionable algorithm accuracy accountability and auditability. *20: Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency.* 5 (Jan. 2020), 1–16. https://doi.org/10.1145/3351095.3372825

[9] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. Why Should I Trust You? Explaining the Predictions of Any Classifier. *Publication: KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 22 (August 2016), 1135–1144. https://doi.org/10.1145/2939672.29397785

[10] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. 2015. Understanding neural networks through deep visualization. *Deep Learning Workshop, 31 st International Conference on Machine Learning* 3 (June 2015), 1–12.

A APPENDIX

A.1 Model Parameters and Performance

MNIST				
Nr.	$h(x)$	Nr. of concepts	θ reg strength	Sparsity parameter
1	CNN	5	0e+00	2e-05
2	CNN	5	1e-01	2e-05
3	CNN	5	1e-02	2e-05
4	CNN	5	1e-03	2e-05
5	CNN	5	1e-04	2e-05
6	CNN	5	1e+00	2e-05
7	CNN	20	0e+00	2e-05
8	CNN	20	1e-01	2e-05
9	CNN	20	1e-02	2e-05
10	CNN	20	1e-03	2e-05
11	CNN	20	1e-04	2e-05
12	CNN	20	1e+00	2e-05
13	Input	N/A	0e+00	N/A
14	Input	N/A	1e-01	N/A
15	Input	N/A	1e-02	N/A
16	Input	N/A	1e-03	N/A
17	Input	N/A	1e-04	N/A
18	Input	N/A	1e+00	N/A

Table 1: Configurations and performance of MNIST models.

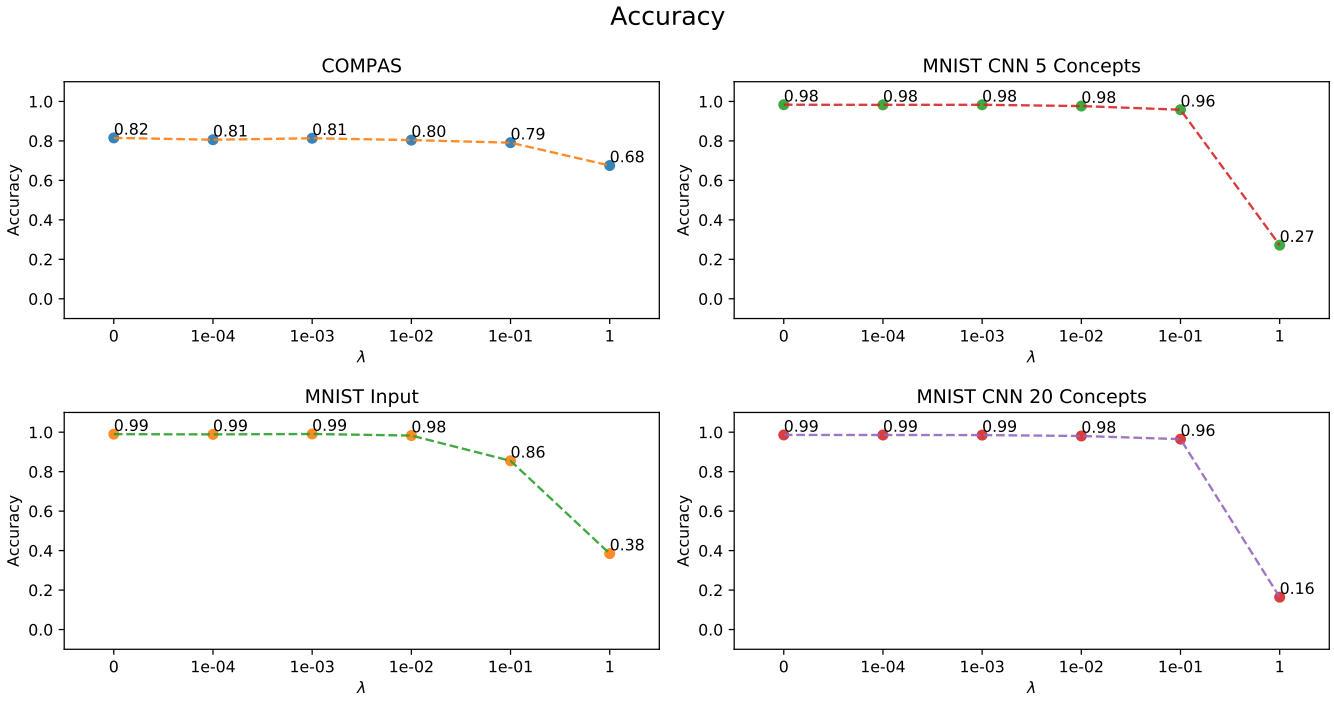

Figure 8: Accuracies of all model configurations.

A.2 Contribution Statement

Every group member contributed fairly to this report. In detail:

- **Ewoud Henri Johannes Vermeij**: Update open source repository and made compatible with newest versions of Pytorch and dependencies. Some contributions to visualizations. Paper contributions in methods, experimental setup and results.
- **Yke Rusticus**: Responsible for: synthetic image generation (the expanded part), and combining the math involved; analysing the tradeoff Lipschitz constant versus accuracy and faithfulness of relevance scores; and critically reviewing and editing the overall paper.
- **Rico Mossinkoff**: Focused mainly on writing the report and creating/presenting the presentation.
- **Roberto Schiavone**: worked on the visualization API, the Jupyter Notebook, and the report results with a focus on stability and faithfulness, and the transparency implications