Scrutinizing and De-Biasing Intuitive Physics with Neural Stethoscopes

Fabian B. Fuchs[1]
fabian@robots.ox.ac.uk

Oliver Groth[12]
ogroth@robots.ox.ac.uk

Adam R. Kosiorek[1]
adamk@robots.ox.ac.uk

Alex Bewley*[13]
bewley@google.com

Markus Wulfmeier*[14]
mwulfmeier@google.com

Andrea Vedaldi[2]
vedaldi@robots.ox.ac.uk

Ingmar Posner[1]
ingmar@robots.ox.ac.uk

[1] Applied AI Lab,
University of Oxford,
UK

[2] Visual Geometry Group,
University of Oxford,
UK

[3] Google Research,
Zürich, Switzerland

[4] Google Deepmind,
London, UK

Abstract

Visually predicting the stability of block towers is a popular task in the domain of intuitive physics. While previous work focusses on prediction accuracy, a one-dimensional performance measure, we provide a broader analysis of the learned physical understanding of the final model and how the learning process can be guided. To this end, we introduce *neural stethoscopes* as a general purpose framework for quantifying the degree of importance of specific factors of influence in deep neural networks as well as for actively promoting and suppressing information as appropriate. In doing so, we unify concepts from multitask learning as well as training with auxiliary and adversarial losses. We apply neural stethoscopes to analyse the state-of-the-art neural network for stability prediction. We show that the baseline model is susceptible to being misled by incorrect visual cues. This leads to a performance breakdown to the level of random guessing when training on scenarios where visual cues are inversely correlated with stability. Using stethoscopes to promote meaningful feature extraction increases performance from 51% to 90% prediction accuracy. Conversely, training on an easy dataset where visual cues are positively correlated with stability, the baseline model learns a bias leading to poor performance on a harder dataset. Using an adversarial stethoscope, the network is successfully de-biased, leading to a performance increase from 66% to 88%.

1 Introduction

In the deep learning community, intuitive physics describes the concept of training neural networks to solve physics-related tasks in a data-driven as opposed to rule-based manner. However, what kind of function the trained network represents highly depends on the types of scenarios it is confronted with and the task it is trying to solve. Furthermore, it depends on the network architecture, on regularisation techniques, on the training procedure, *etc*. As a result, in contrast to a rule-based approach, it is often hard to assess what form of physical understanding a neural network has developed. We are specifically interested in whether the network uses visual cues as shortcuts which reflect correlations in the dataset but are incommensurate with the underlying laws of physics the network was intended to learn.
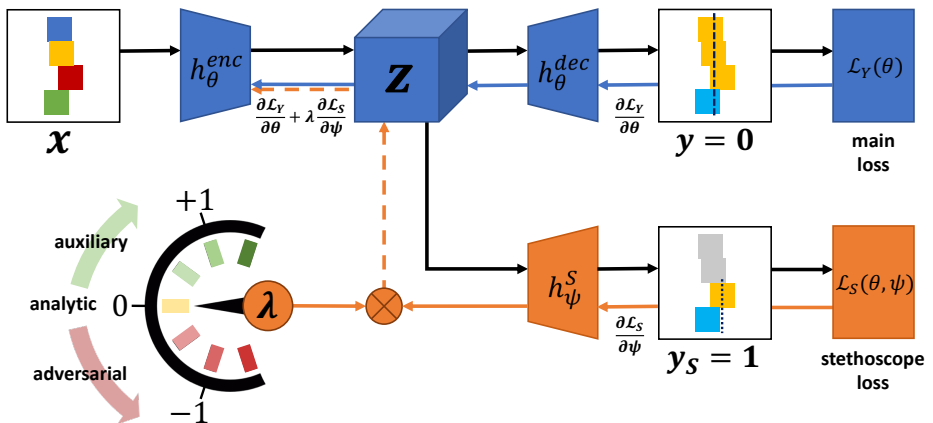

Figure 1: The stethoscope framework. The main network (blue), comprised of an encoder and a decoder, is trained for global stability prediction of block towers. The stethoscope (orange), a two layered perceptron, is trained to predict a nuisance parameter (local stability) where the input is Z, an arbitrary layer of the main network. The stethoscope loss is back-propagated with weighting factor λ to the main network. The value of λ determines whether the stethoscope operates in analytic ($\lambda = 0$), auxiliary ($\lambda > 0$) or adversarial manner ($\lambda < 0$).

In this paper, we specifically focus on stability prediction of block towers, a task which has gained interest in both the deep learning [9, 11, 24] and the robotics community in recent years [12, 13]. Images of towers of blocks stacked on top of each other are shown to a neural network. Its task is to predict whether the tower will fall over or not resulting in a binary classification problem. End-to-end learning approaches as well as simulation-based approaches achieve super-human performance on a real dataset [9, 11, 24]. However, with investigation of trained deep learning models limited to occlusion-based attention analyses [9, 11], it is not clear to what extent neural networks trained on this task take into account physical principles such as centre-of-mass or whether they follow visual cues instead. To this end, we introduce a variation of the ShapeStacks dataset presented by [9] which facilitates the analysis of the effects of visual cues on the learning process.

Motivated by the need for an effective tool to understand and guide the physical reasoning of the neural network and inspired by prior research in interpretability, multi-task learning and adversarial training, we present *neural stethoscopes* as a unified framework for the interrogation and perturbation of task-specific information at any layer. A stethoscope can be

deployed in a purely *analytic* fashion whereby a question is posed via a stethoscope loss which is not propagated back into the main network. It can also be used to promote or suppress specific information by deploying either an auxiliary or an adversarial training mechanism. The concept is illustrated in Figure 1. We demonstrate that deploying an auxiliary stethoscope can be used to promote information conducive to the main task improving overall network performance. Conversely, we show that an adversarial stethoscope can mitigate a specific bias by effectively suppressing information. Moreover, the main network does not need to be changed in order to apply a neural stethoscope.

In this work, we present two contributions: (1) An in-depth analysis of the state-of-the-art approach for intuitive stability prediction. To that end, we also introduce an extension to the existing ShapeStacks dataset which will be made publicly available.[1] (2) A framework for interpreting, suppressing or promoting extraction of features specific to a secondary task unifying existing approaches from interpretability, auxiliary and adversarial learning. While we frame this work in the context of intuitive physics, questions regarding model interpretability and, consequently, systematic, targeted model adaptation find applicability in all domains of deep learning. For a study of two MNIST toy problems with neural stethoscopes, please see Appendix C.

2   Related Work

This work touches on two fields within deep learning: intuitive physics, specifically stability prediction, and targeted model adaptation/interpretation.

Stability Prediction Vision-based stability prediction of block towers has become a popular task for teaching and showcasing physical understanding of algorithms. Approaches include end-to-end deep learning algorithms [9, 11] as well as pipelines using computer vision to create an abstract state representation which can then be used by a physics simulator [6, 24]. [6] achieve impressive results with a pipeline approach to robotic stone stacking. Interestingly, on a publicly available real-world dataset of block tower images [11], the state-of-the-art is shared between an end-to-end learning approach [9] and a pipeline method using a physics simulator [24]. While being much easier to implement, end-to-end approaches have the downside of being significantly harder to interpret. Interpretability brings at least two advantages: (1) Trust by understanding the model's reasoning and therefore also its potential failure cases. (2) Identification of potentials to improve the model. Both Lerer et al. [11] and Groth et al. [9] conduct occlusion-based attention analyses. Groth et al. [9] find that the focus of the algorithm's attention lies within a bounding box around the stability violation in 80% of the cases. While encouraging, conclusions which can be drawn regarding the network's understanding of physical principles are limited. Moreover, Selvaraju et al. [18] shows that attention analyses can be misleading: The Grad-CAM visualisation does not change even for artificially crafted adversarial examples which maximally confuse the classifier.

Neural Stethoscopes The notion of passively interrogating and actively influencing feature representations in hidden layers of neural networks connects disparate fields including interpretability, auxiliary losses and multitask learning as well as adversarial training. We note that much of the required machinery for neural stethoscopes already exists in a number of sub-domains of deep learning. Mirowski et al. [17] use a small probing network to predict

---

[1]Dataset available at https://shapestacks.robots.ox.ac.uk/

an agent's position from a latent space for analytical purposes without backpropagating to the main network. Work on auxiliary objectives [10, 17] as well as multi-task learning (*e.g.* Caruana [4, 5]) commonly utilises dedicated modules with losses targeted towards a variety of tasks in order to optimise a representation shared across tasks. Based on this notion both *deep supervision* [23] and *linear classifier probes* [2] reinforce the original loss signal at various levels throughout a network stack. Although their work is restricted to reinforcing the learning signal via the same loss applied to the global network [2] in particular demonstrate that the accessibility of information at a given layer can be determined - and promoted - by formulating a loss applied locally to that layer.

Conversely, in order to encourage representations invariant to components of the input signal, regularisation techniques are commonly utilised (*e.g.* Srivastava et al. [19]). To obtain invariance with respect to known systematic changes between training and deployment data, [7, 25] propose methods for domain adaptation. In order to prevent a model fitting to a specific nuisance factor, Louizos et al. [14] minimise the Maximum Mean Discrepancy between conditional distributions with different values of a binary nuisance variable in the conditioning set. This method is limited to discrete nuisance variables and its computational cost scales exponentially with the number of states. Xie et al. [26] address both issues via adversarial training, optimising an encoding to confuse an additional discriminator, which aims to determine the values of nuisance parameters. This approach assumes that the nuisance variable is known and is an input to the main model during training and deployment. Louppe et al. [15] follow a similar approach, applying the discriminator to the output of the main model instead of its intermediate representations.

3   Neural Stethoscopes

We use stethoscopes to analyse and influence the learning process for stability prediction, but present it in the following as a general framework which can be applied to any set of tasks.

In supervised deep learning, we typically look for a function $f_\theta : X \to Y$ with parameters θ that maps an input $\boldsymbol{x} \in X$ to its target $\boldsymbol{y} \in Y$. Often the function internally computes one or more intermediate representations $\boldsymbol{z} \in Z$ of the data. In this case, we rewrite f_θ as the composition of the encoder $h_\theta^{\text{enc}} : X \to Z$, which maps the input to the corresponding features $\boldsymbol{z} \in Z$, and the decoder $h_\theta^{\text{dec}} : Z \to Y$, which maps features to the output.

Let the stethoscope be defined as an arbitrary function $h_\psi^{\text{s}} : Z \to S$ with parameters ψ. h_ψ^{s} is trained on a supplemental task with targets $s \in S$ and not for the main objective. The loss function for the stethoscope is defined as $\mathcal{L}_s(\theta, \psi)$, which measures the performance on the supplemental task. The weights of the stethoscope are updated as

$$-\Delta\psi \propto \nabla_\psi \mathcal{L}_s(\theta, \psi) \tag{1}$$

to minimise $\mathcal{L}_s(\theta, \psi)$.

The loss function measuring the performance on the main task, i.e., the discrepancy between predictions f_θ and the true task y, is defined as $\mathcal{L}_y(\theta)$. Crucially, the weights of the main network are updated not just as a function of $\mathcal{L}_y$, but also of $\mathcal{L}_s$:

$$\mathcal{L}_{y,s}(\theta, \psi) = \mathcal{L}_y(\theta) + \lambda \cdot \mathcal{L}_s(\theta, \psi). \tag{2}$$

$$-\Delta\theta \propto \nabla_\theta \mathcal{L}_{y,s}(\theta, \psi) \tag{3}$$

By choosing different values for the constant λ we obtain three very different use cases:

Analytic Stethoscope ($\lambda = 0$)   Here, the gradients of the stethoscope, which acts as a passive observer, are not used to alter the main model. This setup can be used to interrogate learned feature representations: if the stethoscope predictions are accurate, the features can be used to solve the task.

Auxiliary Stethoscope ($\lambda > 0$)   The encoder is trained with respect to the stethoscope objective, hence enforcing correlation between main network and supplemental task. This setup is related to learning with auxiliary tasks, and helpful if we expect the two tasks to be beneficially related.

Adversarial Stethoscope ($\lambda < 0$)   By setting $\lambda < 0$, we train the encoder to maximise the stethoscope loss (which the stethoscope *still* tries to minimise), thus encouraging independence between main network and supplemental tasks. This is effectively an adversarial training framework and is useful if features required to solve the stethoscope task are a detrimental nuisance factor.

For the analytic stethoscope, to fairly compare the accessibility of information with respect to a certain task in different feature representations, we set two criteria: (1) The capacity of the stethoscope architecture has to be constant regardless of the dimensions of its input. (2) The stethoscope has to be able to access each neuron of the input separately. We guarantee this by fully connecting the input with the first layer of the stethoscope using a sparse matrix. This matrix has a constant number of non-zero entries (criterion 1) and connects every input unit as well as every output unit at least once (criterion 2). For more details, see Appendix A.

In auxiliary and adversarial mode, we attach the stethoscope to the main network's last layer before the logits in a fully connected manner. This setup proved to have the highest impact on the main network. The stethoscope itself is implemented as a two-layer perceptron with ReLU activation and trained with sigmoid or softmax cross-entropy loss on its task $\mathcal{S}$.

For numerical stability, the loss of the encoder in the adversarial setting is rewritten as

$$\tilde{\mathcal{L}}_{y,s}(\theta, \psi) = \mathcal{L}_y(\theta) + |\lambda| \cdot \mathcal{L}_{\bar{s}}(\theta, \psi) \tag{4}$$

where $\mathcal{L}_{\bar{s}}(\theta, \psi)$ is the stethoscope loss with flipped labels. The objective is similar to the confusion loss formulation utilised in GANs to avoid vanishing gradients when the discriminator's performance is high [8].

4   Vision-Based Stability Prediction of Block Towers

Previous work has shown that neural networks are highly capable of learning physical tasks such as stability prediction. However, unlike approaches using physics simulators [6, 24], with pure-learning based approaches, it is hard to assess what reasoning they follow and whether they gain a sound understanding of the physical principles or whether they learn to take short-cuts following visual cues based on correlations in the training data.

In this section, we follow the state-of-art approach on visual stability prediction of block towers and examine as well as influence its learning behaviour. We introduce a variation of the ShapeStacks dataset from [9] which is particularly suited to study the dependence of network predictions on visual cues. We examine how suppressing or promoting the extraction of certain features influences the performance of the network using neural stethoscopes.

Dataset   As shown in [9], a single-stranded tower of blocks is stable if, and only if, at every interface between two blocks the centre of mass of the entire tower above is supported by the convex hull of the contact area. If a tower satisfies this criterion, *i.e.*, it does not collapse, we
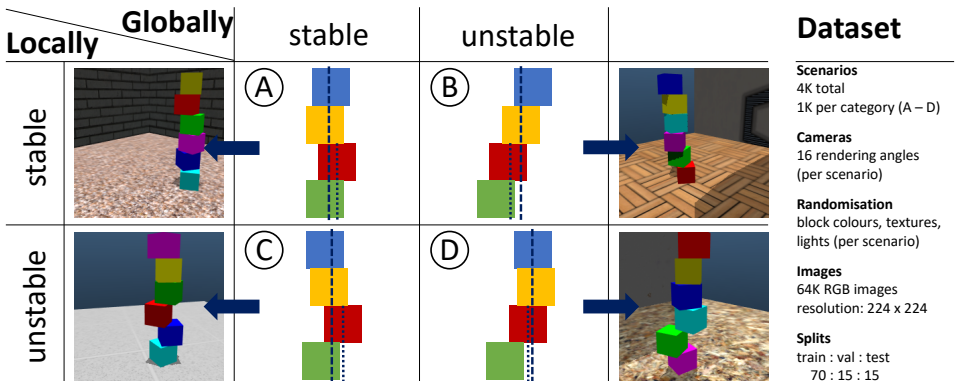
Figure 2: We have four qualitative scenarios: (A) Globally stable towers which also do not exhibit any local stability violation. (B) Towers without any local instability which are globally unstable because of the skew. (C) Towers in which one local stability violation is counterbalanced by blocks above. (D) Globally unstable towers where the site of local and global violation is perfectly correlated. The dashed line shows the projection of the cumulative centre of mass of the upper tower (red, yellow and blue block) whereas the dotted line depicts the projection of the local centre of mass of the red block. A tower is globally stable, if and only if the global centre of mass is always supported whereas the individual local centre of masses are not indicative of global structure stability. Global and local centre of masses for the green, yellow and blue block have been omitted for clarity of presentation.

call it *globally stable*. To be able to quantitatively assess how much the algorithm follows visual cues, we introduce a second label: We call a tower *locally stable* if, and only if, at every interface between two blocks, the centre of mass of the block immediately above is supported by the convex hull of the contact area. Intuitively, this measure describes, if taken on its own without any blocks above, each block would be stable. We associate binary prediction tasks y_G and y_L to respective global and local stability where label $\mathbf{y} = 0$ indicates *stability* and $\mathbf{y} = 1$ *instability*. Global and local instability are neither mutually necessary nor sufficient, but can easily be confused visually which is demonstrated by our experimental results. Based on the two factors of local and global stability, we create a dataset[2] with 4,000 block tower scenarios divided into four qualitative categories (cf. Figure 2). The dataset is divided into an *easy* subset, where local and global stability are always positively correlated, and a *hard* subset, where this correlation is always negative. The dataset will be made available online.[3]

Model   We choose the Inception-v4 network [20] as it yields state-of-the-art performance on stability prediction [9]. The model is trained in a supervised setting using example tuples $(\mathbf{x}, \mathbf{y}_G, \mathbf{y}_L)$ consisting of an image $\mathbf{x}$ and its global and local stability labels $\mathbf{y}_G$ and $\mathbf{y}_L$. Classification losses use sigmoid cross entropy. We use the RMSProp optimiser [21] throughout all our experiments.[4]

Local Stability as a Visual Cue   Based on the four categories of scenarios described in Figure 2, we conduct an initial set of experiments to gauge the influence of local stability on the network predictions. If local stability had, as it would be physically correct, no influence on the network's prediction of global stability, the performance of the network should be

---
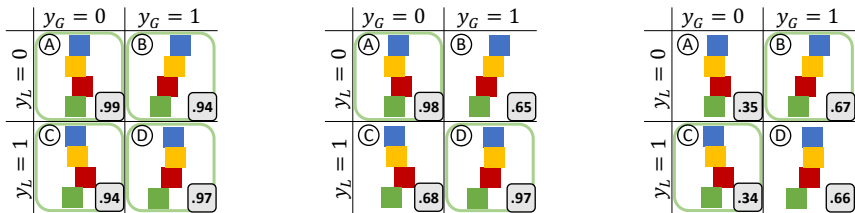
(a) Trained on All: $\varnothing_{acc} = 0.96$ (b) Trained on Easy: $\varnothing_{acc} = 0.82$ (c) Trained on Hard: $\varnothing_{acc} = 0.51$

Figure 3: The influence of local instability on global stability prediction. In setup (a) we train on all 4 tower categories (indicated by green frames). Global stability prediction accuracies on per-category test splits are reported in the bottom right grey boxes. In (b) we train solely on easy scenarios (A & D) where global and local stability are positively correlated. In (c) we only present hard scenarios during training featuring a negative correlation between global and local stability. The performance differences clearly show that local stability influences the network's prediction for global stability.

equal for *easy* and *hard* scenarios, regardless of the training split. However, Figure 3 shows a strong influence of local stability on the prediction performance. When trained on the entire, balanced data set, the error rate is three times higher for hard than for easy scenarios (6% vs. 2%). When trained on easy scenarios only, the error rate even differs by a factor of 13. Trained on hard scenarios only, the average performance across all four categories is on the level of random chance (51%), indicating that negatively correlated local and global stability imposes a much harder challenge on the network.

5   Using Neural Stethoscopes to Guide the Learning Process

After demonstrating the influence of local stability on global stability prediction we turn our attention to the use of neural stethoscopes to quantify and actively mitigate this influence.
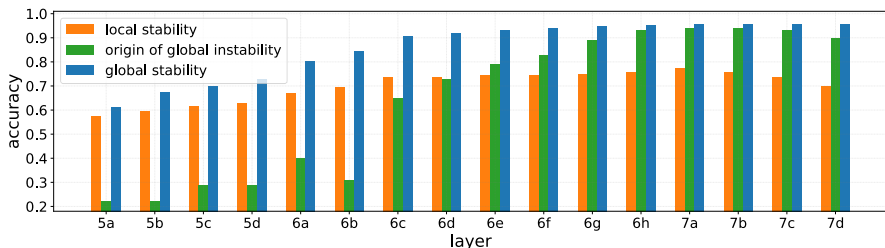

Figure 4: Analysis of the prediction performance throughout the Inception-v4 network trained on predicting global stability. We report average performances on balanced test data after 50 epochs of stethoscope training. All stethoscopes have been attached to the respective activation tensors[5] with sparse connection matrices as described in Section 3.

Task Relationship Analysis   We seek to quantify the correlation of features extracted for global stability prediction with the task of local instability detection. We train the main

network on global stability prediction while attaching stethoscopes to multiple layers. The stethoscope is trained on three tasks: global stability prediction (binary), local stability (binary) and origin of global instability, particularly the interface at which this occurs (n-way one-hot). Figure 4 reveals that the network gradually builds up features which are conducive to both global and local stability prediction. However, the performance of local stability prediction peaks at the activation tensor after layer 7a whereas the performance for global stability prediction (both binary and n-way) keeps improving. This is in line with the governing physical principles of the scenarios and yields the conjecture that the information about local stability can be considered a *nuisance factor* whereas information about the global site of stability violation can serve as a *complementary factor* for the main task.
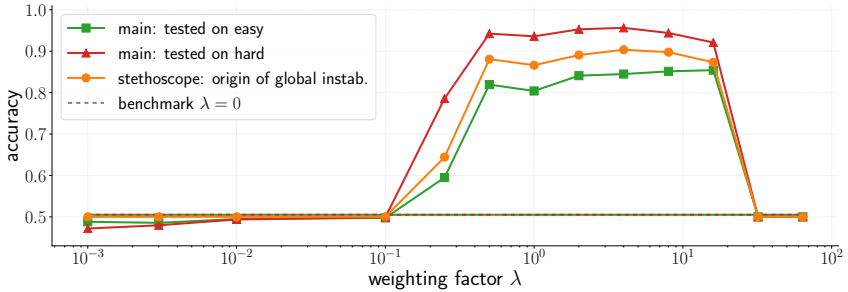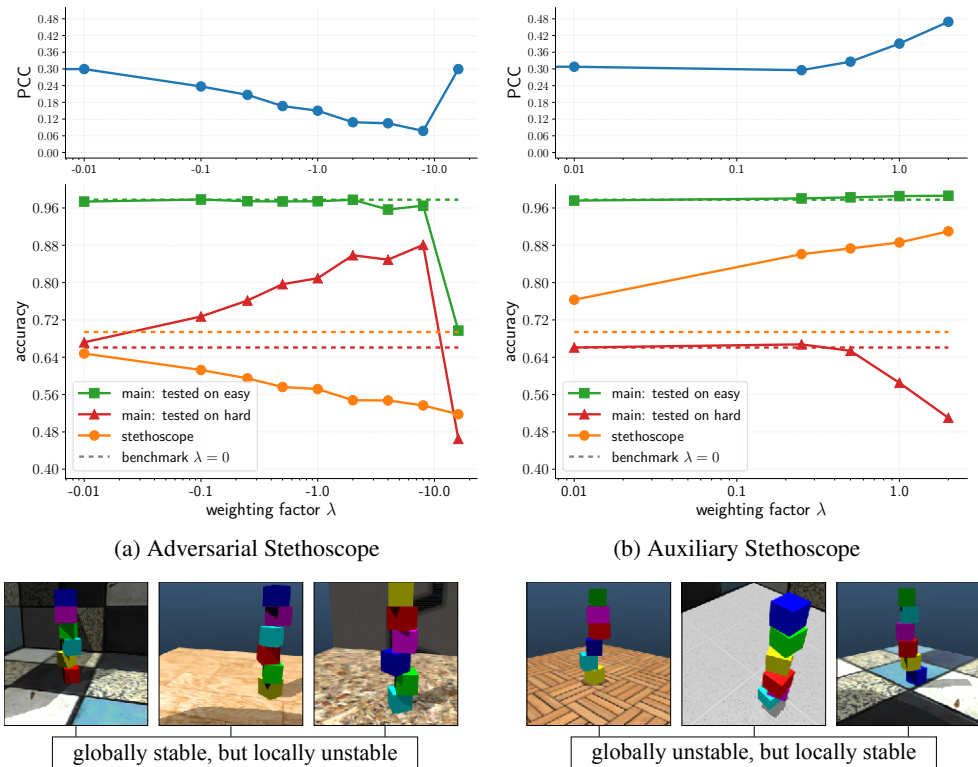

Figure 5: Performance gains by promoting complementary feature extraction with auxiliary stethoscopes. The main network is trained on binary global stability labels while the stethoscope is trained on more fine grained labels, origin of global stability (n-way). The network was trained on *hard* scenarios only but evaluated on all. Dashed lines are baselines ($\lambda = 0$).

Promotion of Complementary Information   We now test the hypothesis that fine-grained labels of instability locations help the main network to grasp the correct physical concepts. To that end, we consider the setup from Figure 3c where the training data only consists of *hard* scenarios with a baseline performance of 51%. The main network is trained on global stability while the stethoscope is trained on predicting the origin of global instability, namely the interface at which the instability occurs. Figure 5 shows that auxiliary training substantially improves the performance for weighting parameters $\lambda \in [0.5, 16]$. However, for very small values of λ, the contribution of the additional loss term is too small while for large values, performance deteriorates to the level of random chance as a result of the primary task being far out-weighted by the auxiliary task. Note that it is to be expected that performance on *hard* exceeds performance on *easy* as the latter type is not encountered during training.
Suppression of Nuisance Information   Results from Figure 3 indicate that the network might use local stability as a visual cue to make biased assumptions about global stability. We now investigate whether it is possible to debias the network by forcing it to pay less attention to local stability. To that end, we focus on the scenario shown in Figure 3b, where we only train the network on global stability labels for *easy* scenarios. As shown in Figure 3b, the performance of the network suffers significantly when tested on *hard* scenarios where local and global stability labels are inversely correlated.

The hypothesis is that forcing the network not to focus on local stability weakens this bias.

---

[5]The stethoscopes are connected to outputs of inception and reduction modules of the Inception v4 network, cf. [20]. Note that this analysis is specific to this architecture and could yield very different results, *e.g.* for networks with residual connections.

(a) Adversarial Stethoscope

(b) Auxiliary Stethoscope

globally stable, but locally unstable

globally unstable, but locally stable

(c) Success cases where algorithm only predicted correctly after de-biasing ($\lambda = -8.0$).

Figure 6: Successful debiasing by suppressing a nuisance factor with adversarial training while auxiliary training worsens the bias. The main network is trained on global stability while the supplementary task is to predict presence of local instabilities. Training data for global stability only comprises *easy* scenarios while training data for the supplementary task comprises both *easy* and *hard* scenarios. **(a)** & **(b)** Green squares and red triangles show accuracies of the main network when predicting global stability of block towers for *easy* scenarios and *hard* scenarios, respectively. Orange circles depict the performance of the stethoscope on the task of local stability. Blue circles represent the Pearson Correlation Coefficient of predicted global stability and ground truth local stability which gives an indication of how much the network follows visual cues. **(c)** Images show *hard* scenarios which the algorithm classified correctly only when adversarial training was used.

In Figure 6, we use active stethoscopes ($\lambda \neq 0$) to test this hypothesis. We train a stethoscope on local stability on labels of all categories (in a hypothetical scenario where local labels are easier to obtain than global labels) and use both the adversarial and the auxiliary setup in order to test the influence of suppressing and promoting accessibility of information relevant for local stability in the encoded representation, respectively. In Figure 6, the results of both adversarial and auxiliary training are compared to the baseline of $\lambda = 0$.

Figure 6a shows that adversarial training does indeed partly remove the bias and significantly increases the performance of the main network on *hard* scenarios while maintaining its high accuracy on *easy* scenarios. The bias is removed more and more with increasing magnitude of the weighting factor λ up to a point where further increasing λ jeopardises

the performance on the main task as the encoder puts more focus on confusing the stethoscope than on the main task (in our experiments this happens at $\lambda \approx 10^1$). Interestingly, the correlation of local stability and global stability prediction rises at this point as for pushing the stethoscope below 50% (random guessing), the main network has to extract information about local stability. Naturally, the performance of the stethoscope continuously decreases with increasing λ as the encoder puts more and more focus on confusing the stethoscope.

This scenario could also be seen from the perspective of feeding additional information into the network, which could profit from more diverse training data. However, Figure 6b shows that naively using an auxiliary setup to train the network on local stability worsens the bias. With increasing λ and increasing performance of the stethoscope, the network slightly improves its performance on *easy* scenarios while accuracy deteriorates on *hard* scenarios. These observations are readily explained: local and global stability are perfectly correlated in the *easy* scenarios, *i.e.*, the (global) training data. The network is essentially free to chose whether to select local or global features when predicting global stability. Auxiliary training on local stability further shifts the focus to local features. When tested on *hard* scenarios, where local and global stability are inversely correlated, the network will therefore perform worse when it has learned to rely on local features.

6   Conclusion

We study the state-of-the-art approach for stability prediction of block towers and test its physical understanding. We create a new dataset and introduce the framework of *neural stethoscopes* unifying multiple threads of work in machine learning related to analytic, auxiliary and adversarial probing of neural networks. The analytic application of stethoscopes allows measuring relationships between different prediction tasks. We show that the network trained on stability prediction also obtains a more fine-grained physical understanding of the scene (origin of instability) but at the same time is susceptible to potentially misleading visual cues (*i.e.*, local stability). In addition to the analysis, the auxiliary and adversarial modes of the stethoscopes are used to support beneficial complementary information (origin of instability) and suppress harmful nuisance information (visual cues) without changing the network architecture of the main predictor. This yields substantial performance gains in unfavourable training conditions where data is biased or labels are partially unavailable. We encourage the use of neural stethoscopes for other application scenarios in the future as a general tool to analyse task relationships and suppress or promote extraction of specific features. This can be done by collecting additional labels or using existing multi-modal data.

Acknowledgements

References

[1] Martin Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. Tensorflow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, pages 265–283, 2016.

[2] Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. In *International Conference on Learning Representations (ICLR) Workshop*, 2016.

[3] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. *International Conference on Machine Learning*, 2017.

[4] Rich Caruana. Learning many related tasks at the same time with backpropagation. In *Advances in neural information processing systems*, 1995.

[5] Rich Caruana. Multitask learning. In *Learning to learn*, pages 95–133. Springer, 1998.

[6] Fadri Furrer, Martin Wermelinger, Hironori Yoshida, Fabio Gramazio, Matthias Kohler, Roland Siegwart, and Marco Hutter. Autonomous robotic stone stacking with online next best object target pose planning. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 2350–2356, 2017.

[7] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

[8] Ian Goodfellow. Tutorial: GANs. *arXiv*, 2016. ISSN 0253-0465. doi: 10.1001/jamainternmed.2016.8245.

[9] Oliver Groth, Fabian Fuchs, Ingmar Posner, and Andrea Vedaldi. Shapestacks: Learning vision-based physical intuition for generalised object stacking. *ECCV*, 2018.

[10] Max Jaderberg, Volodymyr Mnih, Wojciech Czarnecki, Tom Schaul, Joel Z. Leibo, David Silver, and Koray Kavukcuoglu. Reinforcement learning with unsupervised auxiliary tasks. *CoRR*, abs/1611.05397, 2016.

[11] Adam Lerer, Sam Gross, and Rob Fergus. Learning physical intuition of block towers by example. In *International Conference on Machine Learning - Volume 48*, ICML, 2016.

[12] Ruihao Li, Sen Wang, Zhiqiang Long, and Dongbing Gu. UnDeepVO: Monocular Visual Odometry through Unsupervised Deep Learning. *IEEE International Conference on Robotics and Automation (ICRA)*, 2017.

[13] Wenbin Li, Jeannette Bohg, and Mario Fritz. Acquiring Target Stacking Skills by Goal-Parameterized Deep Reinforcement Learning. pages 1–10, 2017.

[14] Christos Louizos, Kevin Swersky, Yujia Li, Max Welling, and Richard S. Zemel. The variational fair autoencoder. *ICLR*, 2015.

[15] Gilles Louppe, Michael Kagan, and Kyle Cranmer. Learning to pivot with adversarial networks. In *NIPS*, 2017.

[16] Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. The numerics of gans. In *Advances in Neural Information Processing Systems 30*. 2017.

[17] Piotr W. Mirowski, Razvan Pascanu, Fabio Viola, Hubert Soyer, Andrew J. Ballard, Andrea Banino, Misha Denil, Ross Goroshin, Laurent Sifre, Koray Kavukcuoglu, Dharshan Kumaran, and Raia Hadsell. Learning to navigate in complex environments. *ICLR*, 2017.

[18] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[19] Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 2014.

[20] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*, 2017.

[21] T. Tieleman and G. Hinton. Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning, 2012.

[22] Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. *IEEE International Conference on Intelligent Robots and Systems*, 2012.

[23] Liwei Wang, Chen-Yu Lee, Zhuowen Tu, and Svetlana Lazebnik. Training deeper convolutional networks with deep supervision. *arXiv*, 1505.02496, 2015.

[24] Jiajun Wu, Erika Lu, Pushmeet Kohli, Bill Freeman, and Josh Tenenbaum. Learning to see physics via visual de-animation. In *Advances in Neural Information Processing Systems 30*. 2017.

[25] Markus Wulfmeier, Alex Bewley, and Ingmar Posner. Addressing appearance change in outdoor robotics with adversarial domain adaptation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, October 2017.

[26] Qizhe Xie, Zihang Dai, Yulun Du, Eduard H. Hovy, and Graham Neubig. Controllable invariance through adversarial feature learning. In *NIPS*, 2017.