Differentiable Logics for Learning with Logical Constraints

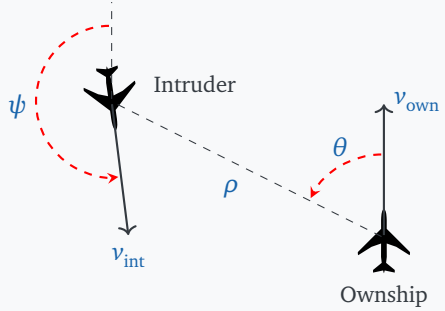
Thomas Flinkow Barak A. Pearlmutter Rosemary Monahan

Department of Computer Science
Maynooth University

VTSA 2024

Issue:
Neural networks fail to learn (safety) properties from data alone!

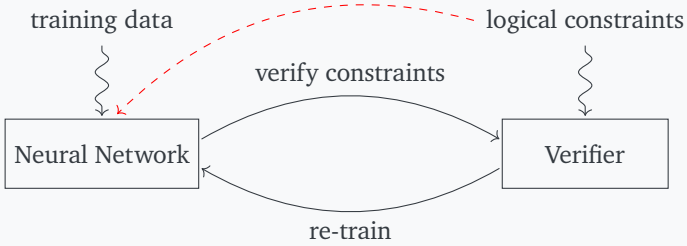
Example: Reluplex (Katz et al., 2017)
'If an intruder is *near* and *approaching from the left*, network should advise *strong right*'.



Task: train a neural network $\mathcal{N}$ to satisfy constraint ϕ.

Train: given data, true labels, and loss function, iteratively update network weights.

Verify: afterwards, using α, β-CROWN, Marabou, NNV, ERAN, …



Note

Training with constraints does not guarantee their satisfaction!

Given data $\boldsymbol{x}_0$ and label $\boldsymbol{y}$, and constraint ϕ,
obtain optimal network weights $\boldsymbol{\theta}^+$ by

$$\boldsymbol{\theta}^+ = \arg\min_{\boldsymbol{\theta}} \mathcal{L}_{\text{CE}}(\boldsymbol{x}_0, \boldsymbol{y}) + \lambda \mathcal{L}_{\text{C}}(\boldsymbol{x}_0, \boldsymbol{y}, \phi).$$

DL2 (Fischer et al., 2019) mapping $[\![\cdot]\!]_{\text{DL2}} : \Phi \to [0, \infty)$

$$[\![\phi \wedge \psi]\!]_{\text{DL2}} := [\![\phi]\!]_{\text{DL2}} + [\![\psi]\!]_{\text{DL2}} \qquad [\![\phi \vee \psi]\!]_{\text{DL2}} := [\![\phi]\!]_{\text{DL2}} \cdot [\![\psi]\!]_{\text{DL2}}$$

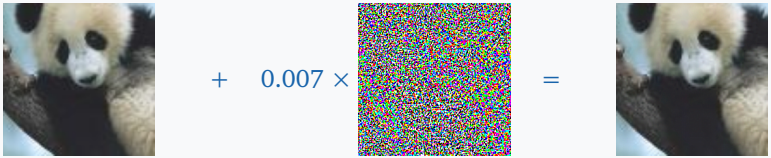
Fuzzy Logic (van Krieken et al., 2022) mapping $[\![\cdot]\!]_L : \Phi \to [0, 1]$

$$[\![\phi \wedge \psi]\!]_{\text{G}} := \min\{\phi, \psi\} \qquad [\![\phi \vee \psi]\!]_{\text{G}} := \max\{\phi, \psi\}$$
$$[\![\phi \wedge \psi]\!]_{\text{ŁK}} := \max\{0, \phi + \psi - 1\} \qquad [\![\phi \vee \psi]\!]_{\text{ŁK}} := \min\{1, \phi + \psi\}$$
$$[\![\phi \wedge \psi]\!]_{\text{P}} := \phi \cdot \psi \qquad [\![\phi \vee \psi]\!]_{\text{PS}} := \phi + \psi - \phi \cdot \psi$$

Definition

A neural network is locally robust in input x_0, if

$$\underbrace{\forall x. \|x - x_0\|_\infty \leq \varepsilon}_{\text{all elements in the input space close to } x_0} \quad \text{implies} \quad \underbrace{\|\mathcal{N}(x) - \mathcal{N}(x_0)\|_\infty \leq \delta}_{\text{the classification is roughly the same}}$$



‘panda’
57.7 % confidence

adversarial
noise

‘gibbon’
99.3 % confidence

Figure 1: Adversarial attack (Goodfellow et al., 2015).

Insight from DL2 (Fischer et al., 2019)

Learning to satisfy $\forall x. x \vDash \phi$ by finding x^* such that $x^* \nvDash \phi$.

1. Approximate counterexample *outside* of the training data using PGD:

$$\boldsymbol{x}^* = \underset{\boldsymbol{x} \in \|\boldsymbol{x} - \boldsymbol{x}_0\|_\infty \leq \epsilon}{\arg\max} \mathcal{L}_{\mathrm{C}}(\boldsymbol{x}_0, \boldsymbol{x}, \boldsymbol{y}, \phi)$$

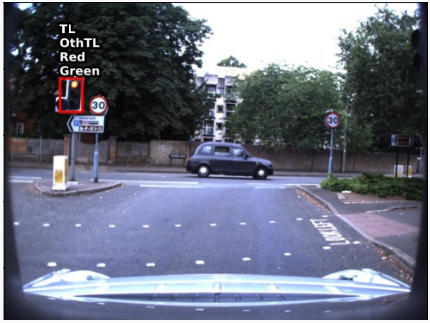
2. Use this counterexample in training:

$$\boldsymbol{\theta}^+ = \underset{\boldsymbol{\theta}}{\arg\min}\ \mathcal{L}_{\mathrm{CE}}(\boldsymbol{x}_0, \boldsymbol{y}) + \lambda \mathcal{L}_{\mathrm{C}}(\boldsymbol{x}_0, \boldsymbol{x}^*, \boldsymbol{y}, \phi).$$

Example: ROAD-R Data Set (Giunchiglia et al., 2023)

Videos annotated with background knowledge (propositional logic).

$$\{\neg \text{Ped}, \neg \text{Cyc}\} \cup \{\neg \text{Red}, \neg \text{Green}\} \cup \{\neg \text{Green}, \neg \text{Mov}\} \cup \ldots$$



What about more expressive logics? (e.g. temporal or probabilistic)

RQ1: What properties to specify and verify?

RQ2: How expressive should the logics be? (e.g. temporal or probabilistic)

RQ3: How to utilise (and ideally, *enforce*) these properties during training to obtain correct-by-construction models?

REFERENCES

Fischer, M., Balunovic, M., Drachsler-Cohen, D., Gehr, T., Zhang, C., & Vechev, M. (2019).DL2: Training and Querying Neural Networks with Logic. *Proceedings of the 36th International Conference on Machine Learning*, 1931–1941. Retrieved April 13, 2023, from https://proceedings.mlr.press/v97/fischer19a.html

Giunchiglia, E., Stoian, M. C., Khan, S., Cuzzolin, F., & Lukasiewicz, T. (2023).ROAD-R: The autonomous driving dataset with logical requirements. *Machine Learning*, *112*(9), 3261–3291. https://doi.org/10.1007/s10994-023-06322-z

Goodfellow, I. J., Shlens, J., & Szegedy, C. (2015, March). Explaining and Harnessing Adversarial Examples. https://doi.org/10.48550/arXiv.1412.6572

Katz, G., Barrett, C., Dill, D. L., Julian, K., & Kochenderfer, M. J. (2017). Reluplex: An Efficient SMT Solver for Verifying Deep Neural Networks. In R. Majumdar & V. Kunčak (Eds.), *Computer Aided Verification* (pp. 97–117). Springer International Publishing. https://doi.org/10.1007/978-3-319-63387-9_5

van Krieken, E., Acar, E., & van Harmelen, F. (2022).Analyzing Differentiable Fuzzy Logic Operators. *Artificial Intelligence*, *302*, 103602. https://doi.org/10.1016/j.artint.2021.103602