Reliable Learning by Subsuming a Trusted Model: Safe Exploration of the Space of Complex Models

Till Speicher [1] **Muhammad Bilal Zafar** [1] **Krishna P. Gummadi** [1] **Adish Singla** [2] **Adrian Weller** [3 4]

Abstract

Designing machine learning algorithms that are reliable, safe, and trustworthy is an important factor when using predictions to make critical decisions in real-world applications including healthcare, law, and self-driving cars. A fundamental challenge faced by a practitioner is how to trade-off higher accuracy of a complex model with more reliability of a simpler, trusted model. In this paper, we propose a novel learning framework to tackle this challenge—our key idea is to safely explore the space of complex models by subsuming a base model which is already trusted. We achieve this via enforcing a regularization constraint in the learning process of the complex model based on the predictions of a trusted model. Our approach is generic, allowing us to consider different trusted models and different ways to enforce the regularization constraint. We demonstrate these ideas via experiments using synthetic and real-world datasets.

1. Introduction

State-of-the-art machine learning methods achieve very good performance across a wide range of real world tasks. However, these methods often rely on training complex models, such as deep neural networks or kernel SVMs, that are hard for humans (even domain experts) to understand. While these models may provide high prediction accuracy in the general case, they may be vulnerable to egregious errors, particularly when presented with data points that are not well-represented in the training set (Nguyen et al., 2014). Consequently, in safety-critical domains (e.g., medical diagnosis or recidivism risk predictions), experts worry about the difficulty of verifying that the behavior of a trained complex model conforms to important domain specific knowledge and requirements (Caruana et al., 2015).

1.1. The simple vs. complex models debate

Today, ML practitioners working in safety-critical domains face a dilemma: they can either use complex (hard to understand / interpret) models that offer high performance, but come with the risk of making catastrophic errors. Or, they could choose to use simple (easy to understand / interpret) models that sacrifice performance (lower prediction accuracy in the average case) for guarantees on not making egregious errors (worst case prediction accuracy).

This dilemma is reflected in recent works on training interpretable models that try to build interpretability into the models itself. These models based on rule lists or decision trees (*e.g.*, (Lakkaraju et al., 2016)), suggest a direct trade-off between interpretability and accuracy. In many cases, interpretability comes at a high cost in prediction accuracy, as restricting learning to finding a set of decision rules excludes the potential for leveraging more nuanced patterns in the data.

In this paper, we propose a way out of this bind. Specifically, we propose a way to establish reliability (i.e., confidence in avoiding egregious errors) of complex ML models by ensuring that in a particular way, they must perform at least as well as simple and easy-to-understand models. Our approach may be considered a form of *safe exploration* of the space of possible models, starting from a base model which is already *trusted*.

1.2. Our proposal: Trusted model subsumption

Our proposal relies on the notion of a *trusted model*, which is (a potentially simple) model that is trusted by domain experts (i.e., domain experts have high confidence in its behavior) to not make egregious errors. In practice, domain experts might train such a model themselves as a first step.

We then modify the training process of a complex model such that, with high probability, it will make correct pre-

[1]MPI-SWS [2]ETH Zurich [3]University of Cambridge [4]Alan Turing Institute. Correspondence to: Till Speicher <tspeicher@mpi-sws.org>, Muhammad Bilal Zafar <mzafar@mpi-sws.org>, Krishna P. Gummadi <gummadi@mpi-sws.org>, Adish Singla <adish.singla@inf.ethz.ch>, Adrian Weller <aw665@cam.ac.uk>.

Proceedings of the 34^{th} International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017.

dictions whenever the (simpler and base) trusted model is correct. Our insight is that the higher complexity of the model enables better performance on inputs which the simple model misclassifies, while matching the correct predictions of the simple model ensures that the chances of catastrophic errors are quite low. This enables people concerned about the reliability of a complex model to train and verify the behavior of a simple, trusted model and use it to establish a lower bound on the performance of more complex models, which may be hard to verify, while still gaining the benefit of better accuracy enabled by higher complexity.

2. Formalizing safe exploration

Let $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$ denote a dataset of N labeled examples, where $\boldsymbol{x} \in \mathcal{X} = \mathbb{R}^d$ denotes the feature vector and $y \in \mathcal{Y} = \mathbb{Z}_{>0}$ denotes the class label. We define H_{trust} as a class of trusted models, satisfying some desirable properties such as being interpretable or simple (*e.g.*, linear models); each element of H_{trust} is a function $h : \mathcal{X} \rightarrow \mathcal{Y}$. Let $\hat{h} \in H_{trust}$ denote a model trained on dataset $\mathcal{D}$ (for instance, via empirical risk minimization) or selected by a domain expert as a trustworthy model. Let $\mathcal{D}_{trust}$ denote a subset of examples on which $\hat{h}$ makes correct predictions, *i.e.*, $\mathcal{D}_{trust} = \{(\boldsymbol{x}, y) \in \mathcal{D} \mid \hat{h}(\boldsymbol{x}) = y)\}$.

Our goal is to train a model from a complex hypothesis class (*e.g.*, deep neural network, or kernel SVMs) using dataset $\mathcal{D}$, while ensuring that the complex model performs "well" on $\mathcal{D}_{trust}$. In other words, we can view $\mathcal{D}_{trust}$ as a regularization constraint enforcing us to explore the space of complex hypothesis class in a safe and trustworthy manner.

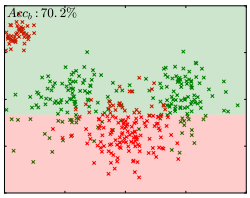
In particular, in this paper, we consider a fully connected neural network consisting of M layers as our complex hypothesis class. Let $\boldsymbol{\theta}_j$ be the parameters for the layer j, then the function computed by the neural network can be written as:

$$\boldsymbol{F} = f_M(\boldsymbol{\theta}_M, ...f_3(\boldsymbol{\theta}_3, f_2(\boldsymbol{\theta}_2, f_1(\boldsymbol{\theta}_1, \boldsymbol{x})))). \quad (1)$$

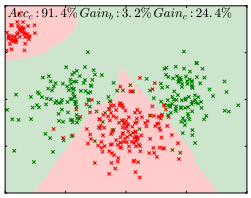
where $\boldsymbol{F} \in \mathbb{R}^{|\mathcal{Y}|}$ usually consists of normalized (softmax) probabilities for each class, that is, $\boldsymbol{F}_k = p(\hat{y} = k|\boldsymbol{x}, \boldsymbol{\Theta})$ and $(\sum_{p \in \boldsymbol{F}} p) = 1$. One then predicts the class with the highest probability, that is, $\hat{y} = \underset{k \in \mathcal{Y}}{\operatorname{argmax}} \boldsymbol{F}_k$. Given $\mathcal{D}$ (and in the absence of $\mathcal{D}_{trust}$), one learns the parameters $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_j\}_{j=0}^{M}$ by minimizing the cross entropy loss:

$$\ell(\boldsymbol{\Theta}, \mathcal{D}) = \sum_{(\boldsymbol{x}, y) \in \mathcal{D}} -\log(p(\hat{y} = y|\boldsymbol{x}, \boldsymbol{\Theta})) \quad (2)$$

While ℓ is a non-convex loss function, one can empirically find "good enough" local minima (Goodfellow et al., 2016) using backpropagation. Next, we enforce the regularization constraint of $\mathcal{D}_{trust}$ by altering the loss function as follows:



(a) Trusted model



(b) Complex models

Figure 1. The decision boundary of a trusted baseline model (that works by thresholding on a single feature) and a relatively complex neural network model (that works by using a non-linear combination of the two features). The accuracy of the baseline model (Acc_b) is not very high, but the outcomes are highly interpretable. On the other hand, the accuracy of the complex neural network (Acc_c) is much higher but outcomes are not easily interpretable. For 24.4% of the dataset (denoted by $Gain_c$), the complex model provides gains over the trusted baseline model, that is, these are the points that are incorrectly classified by the trusted model, but correctly classified by the complex model. However, for 3.2% of the dataset (denoted by $Gain_b$), the complex model gives the wrong predictions even while the trusted model classifies them correctly (*i.e.*, even when these points are in $\mathcal{D}_{trust}$).

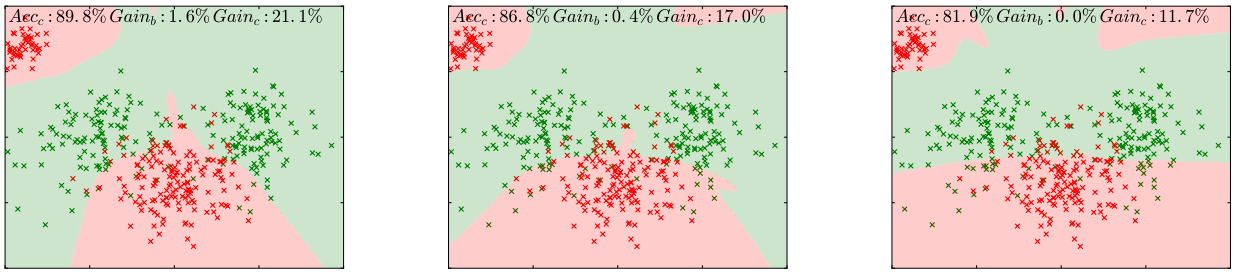
$$\ell(\boldsymbol{\Theta}, \mathcal{D}) = \sum_{(\boldsymbol{x}, y) \in \mathcal{D}} -\log(p(\hat{y} = y|\boldsymbol{x}, \boldsymbol{\Theta})) + \alpha \sum_{(\boldsymbol{x}, y) \in \mathcal{D}_{trust}} \max(\boldsymbol{F}(\boldsymbol{x})) - \boldsymbol{F}_y(\boldsymbol{x}) \quad (3)$$

The second term in the loss function will be positive if an example from $\mathcal{D}_{trust}$ is not classified correctly by the complex model. By increasing the strength of α, one can control how compliant to the constraints the outcomes of the complex model would be.

Our methodology is partly inspired by similar techniques used in the area of fairness-aware classification (Goh et al., 2016; Kamishima et al., 2011; Zafar et al., 2017).

3. Evaluation

In this section, we evaluate the effectiveness of our methodology for training fully connected neural networks that adhere to the given reliability constraints (Section 2). For illustrative purposes, we take our base trusted model to be a simple threshold function for the one most informative feature. We first visually show on a synthetic dataset how the neural network adjusts its decision boundary to correctly classify a trusted set. We then demonstrate the effectiveness of our scheme on real world datasets by conducting empirical experiments on each of the ProPublica COMPAS dataset (Larson et al., 2016) and the Adult dataset (Adult,



(a) $\alpha = 5$ (b) $\alpha = 25$ (c) $\alpha = 500$

Figure 2. Effect of the penalty parameter α on the accuracy (Acc_c) and misclassification in the trusted set ($Gain_b$) by the complex neural network model. The hypothesis space of the network model constricts as we increase the value of α. As a result, the misclassifications in $\mathcal{D}_{trust}$ go down monotonically as the neural network boundary morphs into a boundary very close to that of the trusted baseline model.

1996). The ProPublica COMPAS dataset consists of 7,214 examples and 9 features (number of prior offenses, gender, etc). The task is to predict whether a criminal defendant would recidivate within two years or not. The Adult dataset, also called the Census income dataset, consists of 45,222 examples and 14 features (educational level, race, etc). The task is to predict whether a person earns more than 50K USD per year (positive class) or not (negative class).

3.1. Synthetic dataset

To generate a synthetic dataset, we sample 1600 2-D feature vectors from $p(\boldsymbol{x}|y=-1) = \mathcal{N}([0,-3],[4,0;0,4])$ and 400 from $p(\boldsymbol{x}|y=-1) = \mathcal{N}([-9,7],[0.5,0;0,0.5])$. Additionally we sample 1000 feature vectors from each of $p(\boldsymbol{x}|y=1) = \mathcal{N}([5,0],[4,0;0,4])$ and $p(\boldsymbol{x}|y=1) = \mathcal{N}([-5,0],[4,0;0,4])$.

Trusted model vs. complex model. We train a trusted baseline model which picks an optimal decision threshold (maximizing accuracy) for each of the individual features in the dataset, and selects the feature (along with its threshold) that maximizes accuracy (*i.e.*, the most informative feature). We show the decision boundary of such a trusted model in Fig. 1(a). In this case, this method picks a threshold on feature f_2 (y-axis) and achieves an accuracy of 70.2%. We chose the single feature thresholding model to be our trusted baseline since it represents the simplest sensible such baseline model which is highly interpretable. We denote the data points correctly classified by the trusted model as $\mathcal{D}_{trust}$.

Next, we train a neural network with one hidden layer consisting of 15 neurons. The decision boundary of the neural network is shown in Fig. 1(b). We can see that the neural network clearly outperforms the baseline in terms of accuracy. In fact, for 24.4% (which we call $Gain_c$) of the total data points, the neural network classifies them correctly whereas the trusted model does not. However, for 3.2% (which we call $Gain_b$) of the total data points, the neural network incurs misclassifications while the trusted baseline model classifiers them correctly – *i.e.*, these points are in $\mathcal{D}_{trust}$.

Constraining the complex model to better match the trusted model on $\mathcal{D}_{trust}$. Next, we enforce the constraint that all the points in $\mathcal{D}_{trust}$ be classified correctly by the complex neural network model. Notice that the complex model already correctly classifies a majority of the members of $\mathcal{D}_{trust}$, however, our goal is to correctly classify all of them.

To this end, we retrain the neural network by using the regularized loss in Eq. 3 with increasing value of α. A higher value of α limits the complex model's hypothesis class further to the space of models where an increasing number of the members of the set $\mathcal{D}_{trust}$ are correctly classified.

The results in Fig. 2 show that with an increasing value of α, the neural network boundary moves in order to correctly classify the remaining misclassified points in $\mathcal{D}_{trust}$. With a value of $\alpha = 25$, there are only 0.4% of points left in the dataset that are in $\mathcal{D}_{trust}$ yet the neural network misclassifies them. Finally, increasing the value of α to 500 reduces this number of 0. By this point the resulting decision boundary for the lower three clusters is close to the decision boundary of the trusted model. However, the complex model can still classify the upper left cluster correctly because it is misclassified by the baseline model and therefore not in $\mathcal{D}_{trust}$. That allows it to outperform in overall accuracy by 11.7%.

This shows that in general, even for very high values of α, a more complex model need not be close to replicating the simple trusted model – it might be able to achieve much higher overall accuracy while still being correct on almost

α	$\mathbf{Acc_c}$	$\mathbf{Gain_b}$	$\mathbf{Gain_c}$
0.0	67.9%	8.0%	11.7%
0.5	67.7%	6.2%	10.1%
1.0	68.0%	5.6%	8.6%
5.0	66.9%	2.5%	4.8%
10.0	66.4%	2.2%	4.3%
20.0	65.2%	1.4%	2.3%

Table 1. [COMPAS dataset] Accuracy achieved by a neural network with an increasing value of α. $\mathbf{Gain_b}$ ($\mathbf{Gain_c}$) denotes the percet of the data points, out of all the data, that the truested baseline model (complex neural network model) classifies correctly but the neural network (trusted baseline model) does not. The trusted baseline model achieves an accuracy of 64.5%.

α	$\mathbf{Acc_c}$	$\mathbf{Gain_b}$	$\mathbf{Gain_c}$
0.0	84.4%	6.0%	10.8%
0.5	84.2%	5.4%	10.0%
1.0	84.1%	5.2%	9.6%
5.0	83.3%	4.7%	8.4%
10.0	80.9%	1.6%	2.9%
20.0	80.0%	1.2%	1.6%

Table 2. [Adult dataset] Accuracy achieved by a neural network with an increasing value of α. $\mathbf{Gain_b}$ ($\mathbf{Gain_c}$) denotes the percet of the data points, out of all the data, that the truested baseline model (complex neural network model) classifies correctly but the neural network (trusted baseline model) does not. The trusted baseline model achieves an accuracy of 79.6%.

all points within $\mathcal{D}_{trust}$.

3.2. Real-world dataset

We now conduct experiments on ProPublica COMPAS dataset (Larson et al., 2016). For building the trusted baseline model, we use the threshold on the single feature "number of prior offenses" (the most informative feature in the dataset). This classifier leads to an accuracy of 64.5%. We consider all the data points correctly classified by this trusted classifier as $\mathcal{D}_{trust}$, that is, any complex model should aim to classify these points correctly.

We then train a neural network with one hidden layer having 150 neurons with $tanh$ activation function and a dropout probability of 0.5. This network leads to an accuracy of 67.9%, however, the outcomes of the neural network results in a $\mathbf{Gain_b}$ of 8%. That is, 8% of all data points are in $\mathcal{D}_{trust}$ but are not classified correctly by the neural network model.

Finally, we retrain the neural network while introducing the regularization penalty from Eq. 3. The results in Table 1 show that an increasing value of α reduces the misclassification in the set $\mathcal{D}_{trust}$, however, it comes at a cost of deteriorating accuracy.

We also conduct experiments on the Adult dataset (Adult, 1996). The trusted model in this case uses the optimal threshold on the (most informative) feature "capital gain" and achieves an accuracy of 79.6%. The results for the unconstrained and constrained neural network model yield insights that are very similar to the ProPublica COMPAS dataset (Table 2).

4. Conclusion and future work

There are many situations where high prediction accuracy of an algorithm is of great importance, yet we must also be able to trust its output. This is very challenging as models become more complex and hard to understand. Here we have presented preliminary work investigating how we might safely explore the space of more complex models, while leveraging our trust in a simple model. A key insight is that we should like a complex model to achieve correct predictions at least on all those data points where a trusted model was already correct. We look forward to developing this framework further, and extending it to a wider range of complex models (*e.g.*, kernel SVMs). Other interesting future directions would be formalizing the tradeoffs between the safe exploration and accuracy of the complex model and exploring the effects of data drift on the trustworthiness of the complex model.

Acknowledgments

AW acknowledges support by the Alan Turing Institute under EPSRC grant EP/N510129/1, and by the Leverhulme Trust via the CFI.

References

Adult. http://tinyurl.com/UCI-Adult, 1996.

Caruana, Rich, Lou, Yin, Gehrke, Johannes, Koch, Paul, Sturm, Marc, and Elhadad, Noemie. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1721–1730. ACM, 2015.

Goh, Gabriel, Cotter, Andrew, Gupta, Maya, and Friedlander, Michael. Satisfying Real-world Goals with Dataset Constraints. In *NIPS*, 2016.

Goodfellow, Ian, Bengio, Yoshua, and Courville, Aaron. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

Kamishima, Toshihiro, Akaho, Shotaro, Asoh, Hideki, and Sakuma, Jun. Fairness-aware Classifier with Prejudice Remover Regularizer. In *PADM*, 2011.

Lakkaraju, Himabindu, Bach, Stephen H, and Leskovec, Jure. Interpretable decision sets: A joint framework for description and prediction. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2016.

Larson, Jeff, Mattu, Surya, Kirchner, Lauren, and Angwin, Julia. https://github.com/propublica/compas-analysis, 2016.

Nguyen, Anh Mai, Yosinski, Jason, and Clune, Jeff. Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images. *CoRR*, 2014.

Zafar, Muhammad Bilal, Valera, Isabel, Rodriguez, Manuel Gomez, and Gummadi, Krishna P. Fairness Constraints: Mechanisms for Fair Classification. In *AISTATS*, 2017.