Resistance of accumulated local effects to data poisoning attacks

Mateusz Błajda MB406098@STUDENTS.MIMUW.EDU.PL
Michał Krutul MK405353@STUDENTS.MIMUW.EDU.PL
Maciej Nadolski MN394491@STUDENTS.MIMUW.EDU.PL
University of Warsaw, Poland

Abstract

As shown in the past Partial Dependency (PD) explanations can be easily fooled with the data poisoning method. In this work, we show, that another explanation method, namely: Accumulated Local Effects (ALE) can be more resistant to this kind of attack. Moreover, we implemented the ALE++ framework, for finetuning models to match the ALE plot on the generated poisoned data while maintaining decent performance.

1. Introduction

Data poisoning is a kind of attack targeting explanations. It fools them by maliciously changing the dataset used to generate an explanation. The assumption is, that one can freely modify that dataset.

PD (Friedman, 2001) is an explanation method measuring how a particular feature affects prediction. It is an expected value of Ceteris Paribus explanations over all instances in the dataset. (Where Ceteris Paribus measures how the value of one feature changes the prediction of one instance, assuming other features stays the same)

ALE (Daniel W. Apley, 2020) is a newer explanation method, created to fix the PD problem with correlated features. It takes the correlation between features into account while distilling the individual contribution of a given variable. ALE tracks the local curvature of a model (captured by the model derivative), shifts and averages it according to the conditional distribution $X^{-j}|X^j = z$, and then accumulates those effects with an integral to reproduce the effect of the feature x_j PD is shown (Baniecki et al., 2022) to be very prone to this kind of manipulation.

The ease at which PD can be manipulated is a cause for concern, given its widespread usage as an explanation method. Therefore, there is a need for an explanation of methods immune to such kinds of attacks. In this work we show, that another similar explanation method - Accumulated Local Effects (ALE) (Daniel W. Apley, 2020) is more immune to data poisoning attacks than PD, therefore it can be a better choice, especially in the most critical areas.

Our approach begins by training the model on original data. Next, we conduct a data poisoning attack ALE and check the plots. Finally, we finetune the model using regularization in an operation called 'ALE++' to make it produce similar explainations on both the original and poisoned dataset, while maintaining acceptable accuracy.

2. Related work

The work (Tang et al., 2021) introduces a training framework which protects against introducing data perturbations as a way of manipulating explanations. It focuses, however, on image data and local explanations while this work attempts to provide attack immunity against global explanations on tabular data.

3. Methodology

3.1 Data Poisoning

In the work (Baniecki et al., 2022) data poisoning attacks are performed in two fashions - via gradient descent and via a genetic algorithm. We opted to focus on the former, although the finetuning is agnostic to what kind of attack was performed and could be performed in either setup.

The attack is performed as follows: we train the parameters $\mathcal{Z} \in \mathbb{R}_{n \times m}$ to minimize:

$$\mathcal{L}(\mathcal{Z}) = MSE(-ALE_{\mathcal{M}}(X), ALE_{\mathcal{M}}(\mathcal{Z}))$$

$ALE_{\mathcal{M}}(\cdot) \in \mathbb{R}^k$ represents an ALE explanation of model $\mathcal{M}$ approximated for k equally-spaced points. Our objective is $-ALE_{\mathcal{M}}(X)$ which is the true ALE explanations. The training is performed based on the gradient of the loss function using the Adam optimizer. Note that this optimization objective differs from (Baniecki et al., 2022) work on fooling Partial Dependence, where the target plot was equal to the original mirrored at the average. This change was made with the intuition about the properties of ALE - its value at z_0 is always equal to 0 and it's not constrained to the range of the model.

3.2 ALE++

ALE++ is a framework designed to immunize a model for data poisoning attacks on ALE explanations. It achieves it by fine-tuning the model using the ALE++ Loss:

$$Loss_{ALE++}(\mathcal{M}, y, X, X') = Loss(\mathcal{M}(X), y) + \alpha_{reg} MSE(ALE_{\mathcal{M}}(X), ALE_{\mathcal{M}}(X'))$$

where $\mathcal{M}$ is the model, X, X' are original, and poisoned datasets, α_{reg} is regularization factor and y is ground truth target.

By joining the standard model loss with the mean squared error between the ALE explanations, we can achieve a model with a similar plot on both original and poisoned data while maintaining decent performance.

4. Experiments

4.1 Datasets

For conducting our experiments we used the Bike Sharing Dataset(Fanaee-T and Gama, 2013). The dataset consists of 2 years' worth of data about the number of bikes rented aggregated on an hourly basis. Besides the date-time information, this dataset also contains weather information. We choose this dataset because it is a real-world example, plus it was used in an original ALE paper (Daniel W. Apley, 2020) We performed a poisoning attack

variable	pre-finetuning	post-finetuning
hr	240.36	25.88
atemp	41.21	0.19
mnth	7.14	0.06
windspeed	13.03	0.57
weathersit	4.45	0.39

Table 1: Mean absolute difference between the original ALE explanation and the ALE explanation on a poisoned dataset. The left column provides the metric before ALE++ is applied, the right column - after. The poisoned dataset used to compute the latter is generated by performing a separate attack after ALE++ procedure.

on the same variables, namely: month, hour, weather situation, wind speed, and feeling temperature.

4.2 Model

In this work, we train a network model using the Keras framework from the Tensorflow library. The model includes a normalization layer, followed by two fully connected layers with ReLU activation, and a final output layer with one neuron and ReLU activation.

4.3 Results

Part of the results from performing a data poisoning attack on ALE is presented in Figures 1, 2. The remaining plots are located in the appendix.

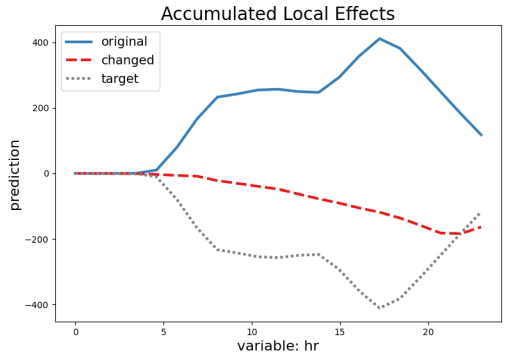

Figure 1: ALE plots for original and poisoned data for hour variable.

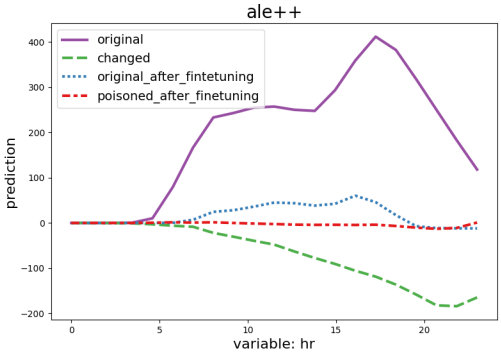

Figure 2: ALE plots for original, poisoned data for hour variable, but also using finetuned model .

In Fig. 1 we observe that ALE plot on poisoned data got closer to target, but we cannot say it is similar. But on the other hand in Fig 2 we can see how model weights were updated

during ALE++, so original and poisoned ALE plots met somewhere between and are now much similar. Table 1 reinforces that observation - we can see that the mean absolute difference between honest and poisoned ALE explanations is much smaller after performing the ALE++ finetuning.

We build our codebase on top of code available as a supplement to (Baniecki et al., 2022). Our fork with adjustments for performing experiments described in here is publicly available on GitHub: `https://github.com/MaciejNadolski98/fooling-partial-dependence`.

5. Limitations and future work

The current poisoning approach produced data points whose values were well out of feature distributions. Therefore, in future work we plan to make an adjustment to the attack algorithm by imposing a restriction on how much the poisoned dataset can differ from the original dataset, although we decided against including it in the presentation. It was implemented by introducing a non-linear relationship between $\mathcal{Z}$ (starting from random) and X':

$$X' = X + \alpha \tanh(\mathcal{Z})$$

This way the absolute difference between original and poisoned datasets is bounded by a constant $\alpha \in \mathbb{R}$.

6. Conclusion

We presented the results from performing a data poisoning attack on ALE explanation and then finetuning it with the ALE++ framework. The results are slightly better than the results from poisoning PD plots - ALE plots are much harder to bend, but there is still space for improvement.

References

Hubert Baniecki, Wojciech Kretowicz, and Przemysław Biecek. Fooling partial dependence via data poisoning. *ECML PKDD*, 2022.

Jingyu Zhu Daniel W. Apley. Visualizing the effects of predictor variables in black box supervised learning models. *Journal of the Royal Statistical Society Series B*, 82(4): 1059–1086, September 2020.

Hadi Fanaee-T and João Gama. Bike-sharing dataset, 12 2013.

Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232, 2001. ISSN 00905364. URL `http://www.jstor.org/stable/2699986`.

Ruixiang Tang, Ninghao Liu, Fan Yang, Na Zou, and Xia Hu. Defense against explanation manipulation, 2021. URL `https://arxiv.org/abs/2111.04303`.

Appendix A.

Plots from PD poisoning attacks

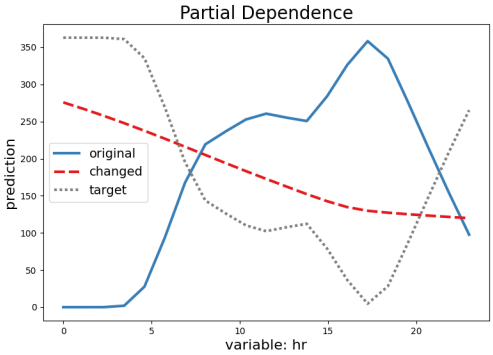

Figure 3: PD hour.

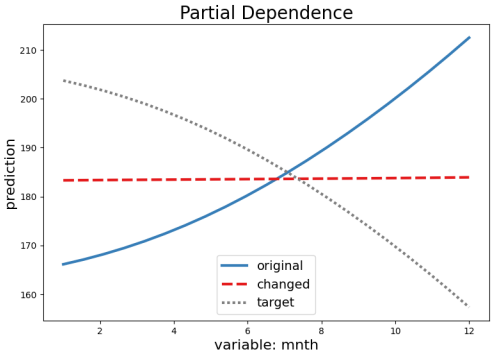

Figure 4: PD month.

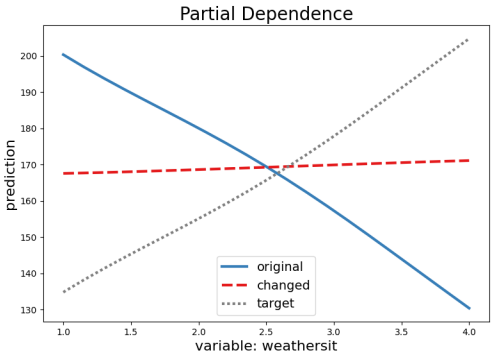

Figure 5: PD weather situation.

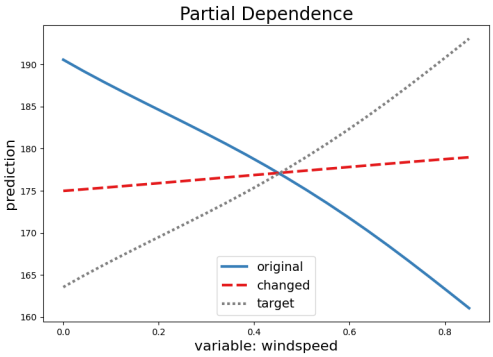

Figure 6: PD windspeed.

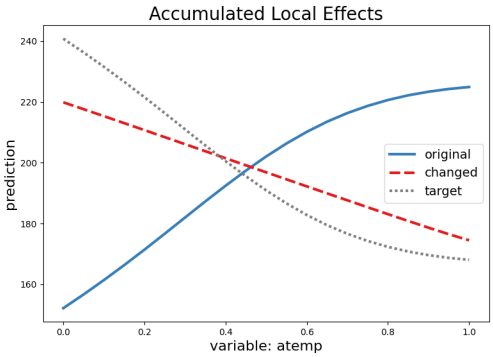

Figure 7: PD feeling temperature.

Appendix B. ALE and ALE++ on remaining variables

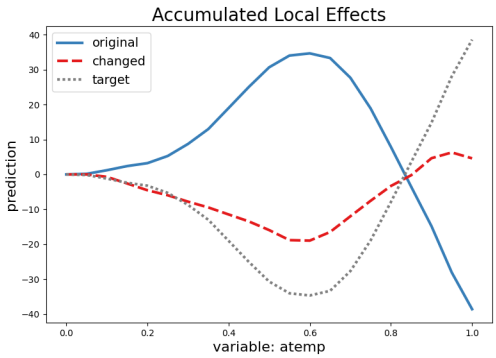

Figure 8: ALE plots for original and poisoned data for hour variable.

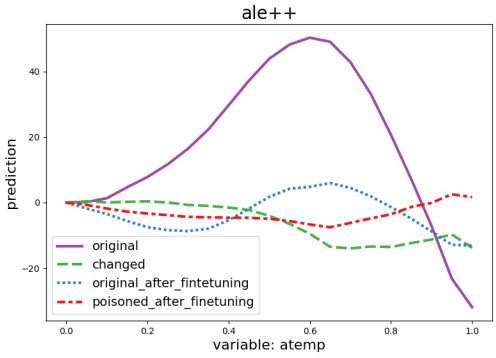

Figure 9: ALE plots for original, poisoned data for hour variable, but also using finetuned model .

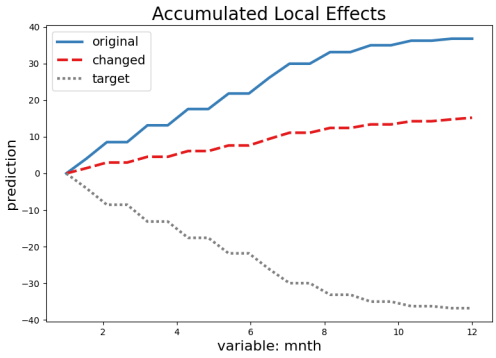

Figure 10: ALE plots for original and poisoned data for hour variable.

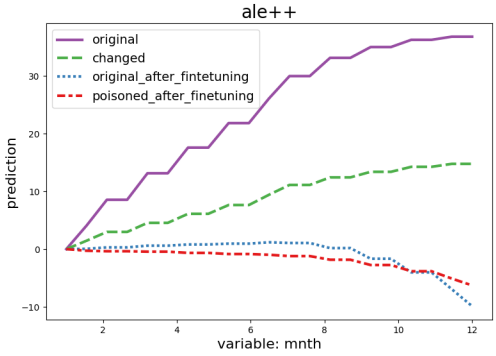

Figure 11: ALE plots for original, poisoned data for hour variable, but also using the finetuned model .

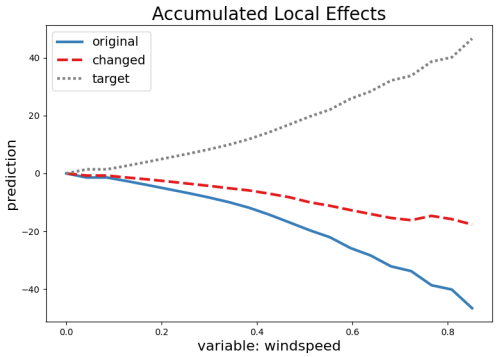

Figure 12: ALE plots for original and poisoned data for hour variable.

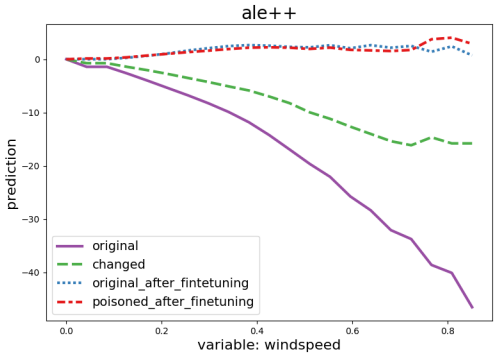

Figure 13: ALE plots for original, poisoned data for hour variable, but also using finetuned model .

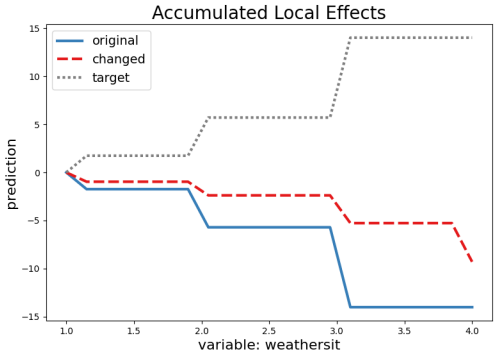

Figure 14: ALE plots for original and poisoned data for hour variable.

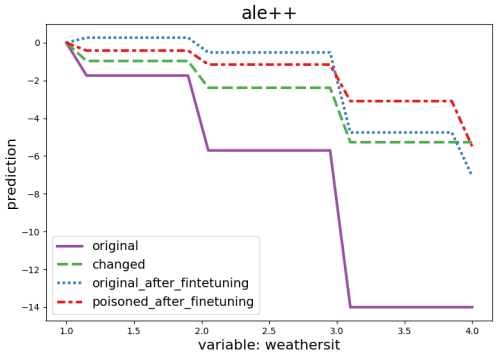

Figure 15: ALE plots for original, poisoned data for hour variable, but also using the finetuned model .