# Data poisoning on ALE and PDP - checkpoint 2

Julia Chylak, Aleksandra Mysiak, Krzysztof Tomala

Winter semester 2022/23, December 2022

**Introduction** The goal of our project is to compare data poisoning for two explanation types – Partial Dependence Profiles (PDP) [4] and Accumulated Local Effects (ALE) [2]. Here, data poisoning means modifying a dataset on which we are calculating an explanation so that, with the model left unchanged, the explanation changes as much as possible. Our work extends the work of [3] to feature an additional explanation type.

**Methodology** The data poisoning strategy we are using follows the gradient strategy described in [3]. It uses simple gradient descent with L2 loss between the inverted original explanation (the *target* explanation) and the manipulated explanation. We decided not to manipulate the variable for which we are analysing the explanations, as for PDP, it does not impact the results, and for ALE it would allow manipulation by moving data between buckets. We are not interested in a solution that, for example, simply moves all of the data into one bucket.

**Datasets** We have carried out experiments on 3 datasets: 2 artifically generated, and 1 real-life dataset. All of them are adjusted to be classification datasets for consistency, following examples from [3].

1. **XOR** – a very simple dataset used in code examples from [3]. It contains 3 features drawn from $\mathcal{N}(0,1)$. The response variable indicates if the product of the three features is greater than zero.
2. **Friedman** – the first Friedman regression task, adapted for classification. We are using a variable number of features up to 5, all drawn from $\mathcal{U}((0,1))$. Traditionally (see `make_friedman1`), the response variable is $y(X) = 10 * \sin(\pi * X_1 * X_2) + 20 * (X_3 - 0.5)^2 + 10 * X_4 + 5 * X_5$. Here, the response variable has been set to indicate if $y$ is greater than its expected value. For dimensionality of features $n$, we have replaced features with indices greater than $n$ with their expected values.
3. **Heart** – a real-life heart-disease dataset with 14 features containing patient data, and a response variable indicating if the patient has heart disease [1].

**Evaluation methods** We use 4 measures of difference between explanations:

- L2 metric – standard L2 metric between the target and manipulated explanations, same as loss function.
- L1 metric – standard L1 metric between the target and manipulated explanations.
- L$\infty$ – maximum of absolute values of differences between explanations.
- Spearman's $\rho$ – a standard measure of rank correlation, used between the original and manipulated explanations. A negative value suggests at least partial reversal of monotonicity.

Overall, the $\rho$ metric seems to be most useful when comparing explanations, since the other metrics are scale-dependent and, as such, are hard to compare between PDP and ALE.

### Results

- ALE has predominantly better metrics compared with PDP, especially the Spearman's $\rho$, which means that it differs less from the original explanation. Figure 2 shows that PDP usually tends to the target shape, while ALE is more robust to the attack.
- ALE and PDP tend to follow similar patterns – which is also true on poisoned data. As such, they are both prone to attacks to a certain degree, which can be seen on both Figure 1 and 2.

### Limitations and next steps

- Optimising on ALE sometimes leads to poor results. A working hypothesis is that when poisoning PDP, each poisoned sample has an impact on the whole curve, which is not true when poisoning for ALE, since data points are constrained to their initial bins, and ALE is not differentiable on bin borders. A planned solution is implementation of weight smoothing for ALE.
- The gradient algorithm has no constraints on the change of data distribution, therefore we are planning to normalize the data for prediction using sigmoid function, so that they stay in the same interval as original data.
- Testing on the bike-sharing dataset referenced in the ALE paper [2].
- Implementing ALE++ framework.

Figure 1: XOR dataset, model size = 16, learning rate = 0.1, max_iter = 50

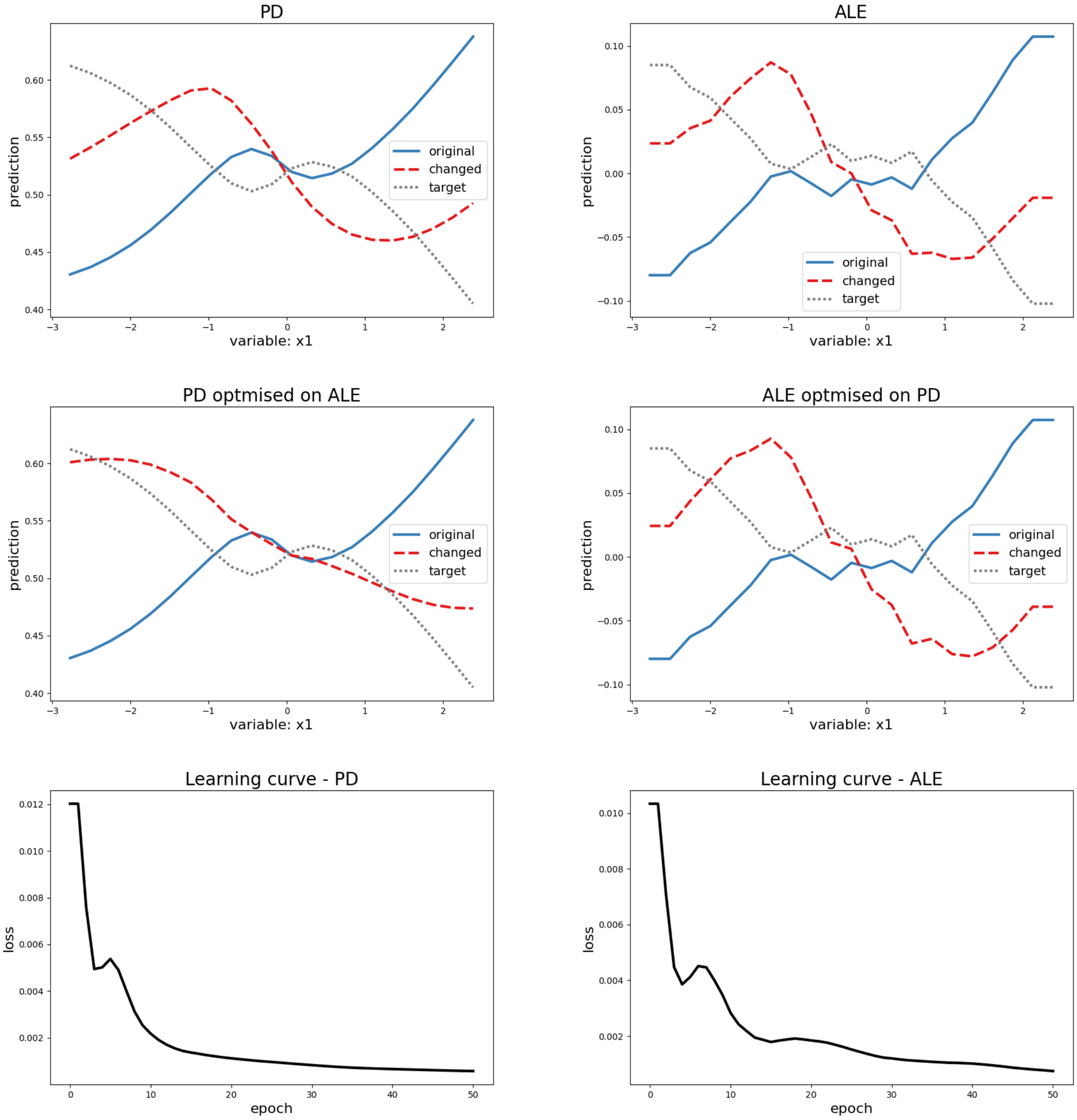


Figure 2: Heart dataset, model size = 64, learning rate = 0.1, max_iter = 50

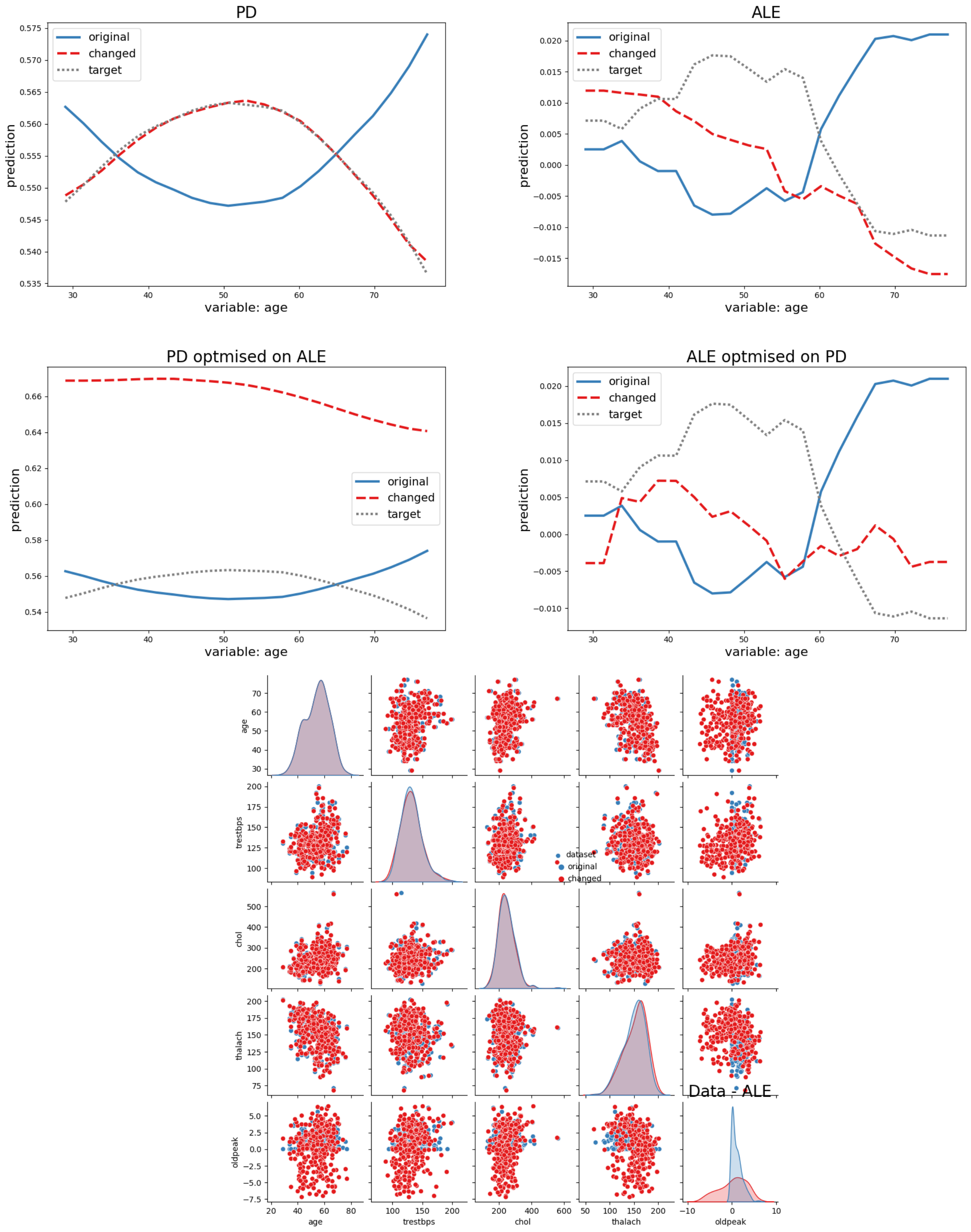
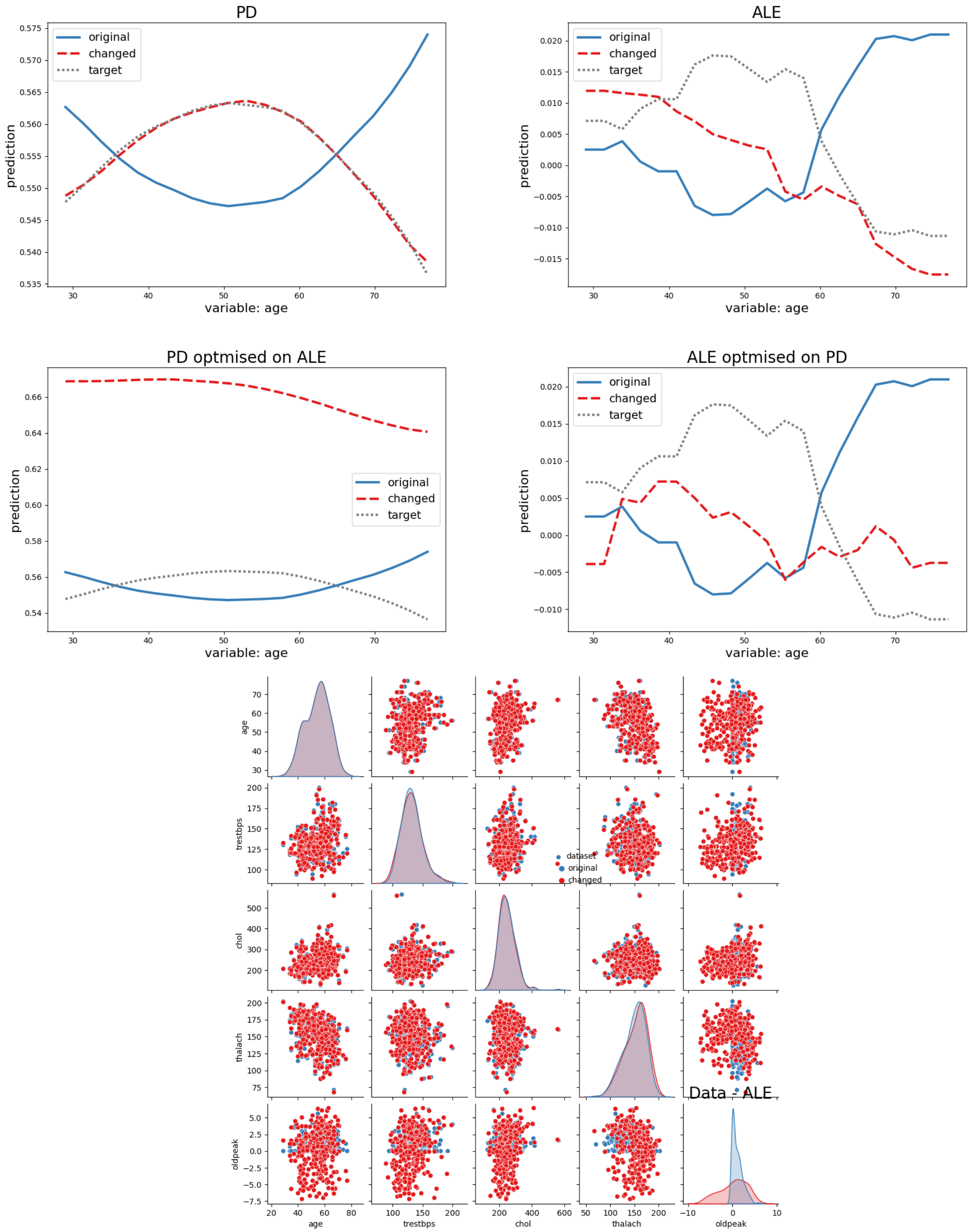


Figure 3: Heart dataset, model size = 64, learning rate = 0.1, max_iter = 50



### Deliverables

- Result metrics are available here. The name of each experiment is in the form of {dataset}/{model-size}_{number-of-observations}_{random-seed}_{type-of-algorithm}_{learning-rate}_{maximum-number-of-iterations}.
- Project repository.

## References

[1] UCI Heart Disease Data. https://www.kaggle.com/datasets/redwankarimsony/heart-disease-data.

[2] Daniel W. Apley and Jingyu Zhu. Visualizing the effects of predictor variables in black box supervised learning models, 2016.

[3] Hubert Baniecki, Wojciech Kretowicz, and Przemyslaw Biecek. Fooling partial dependence via data poisoning. *CoRR*, abs/2105.12837, 2021.

[4] Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189 – 1232, 2001.