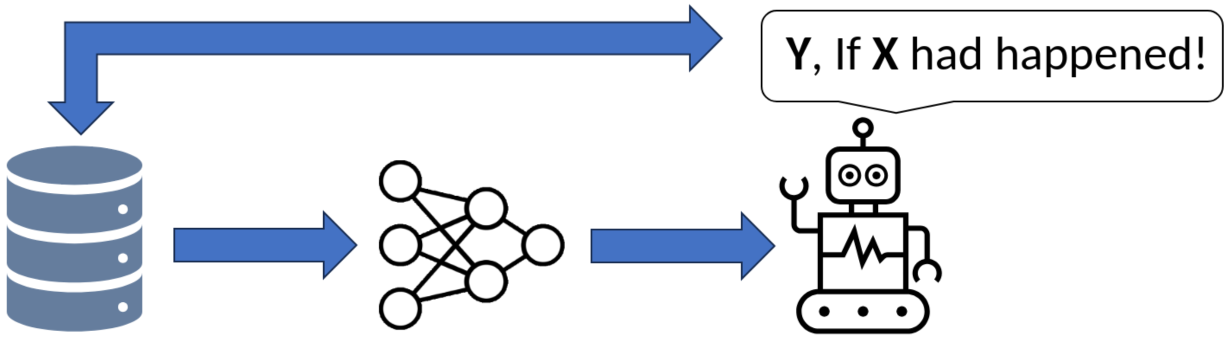
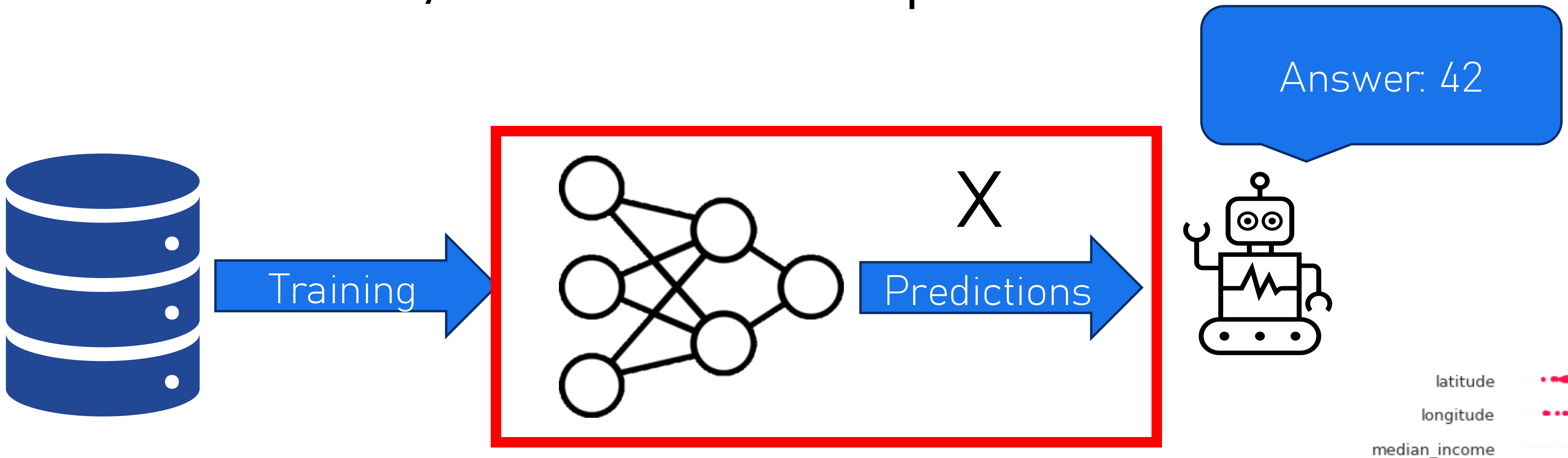
"Who Dunit?" -- Tracing Explanations back to Training Samples

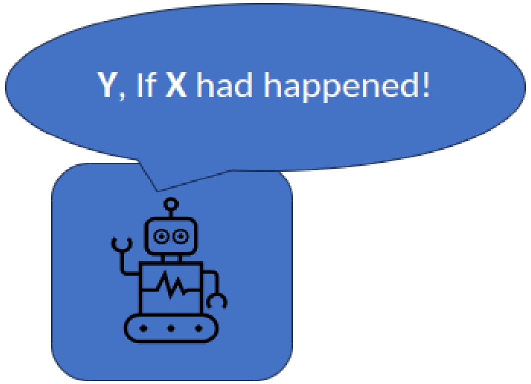


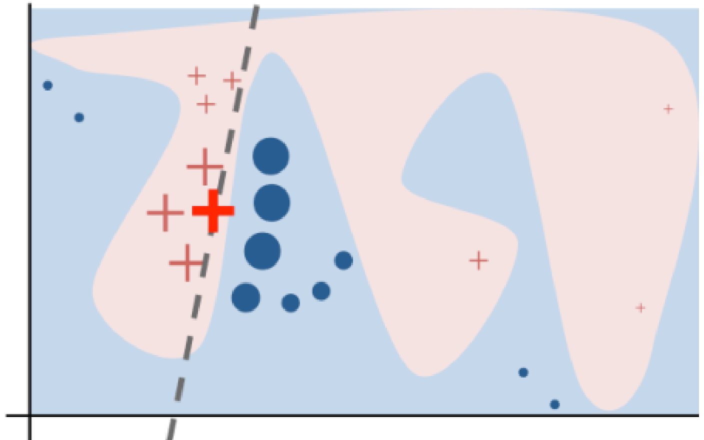
DR. ANDRÉ ARTELT, BIELEFELD UNIVERSITY, GERMANY

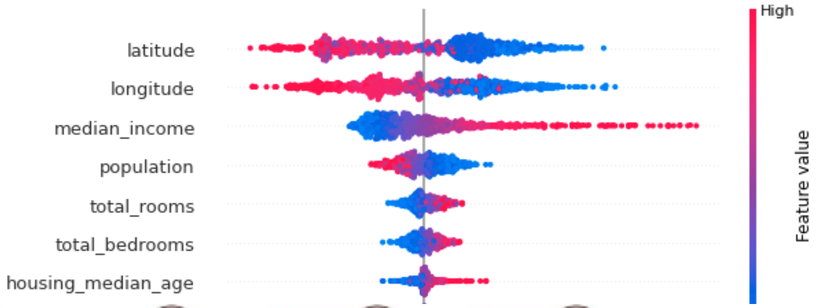
AARTELT@TECHFAK.UNI-BIELEFELD.DE

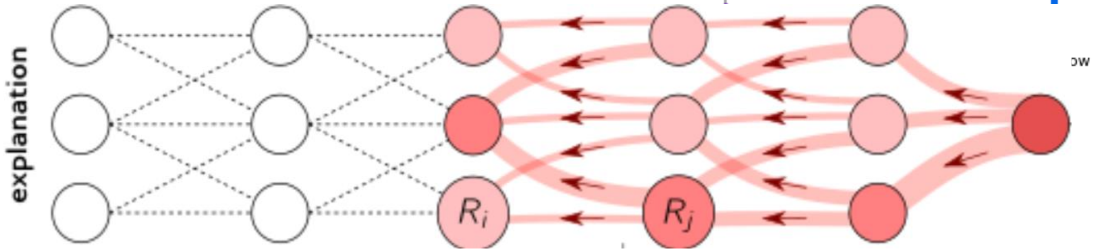
XAI – How/What to Explain











What Causes an Explanation?

or *"How to explain an Explanation"*

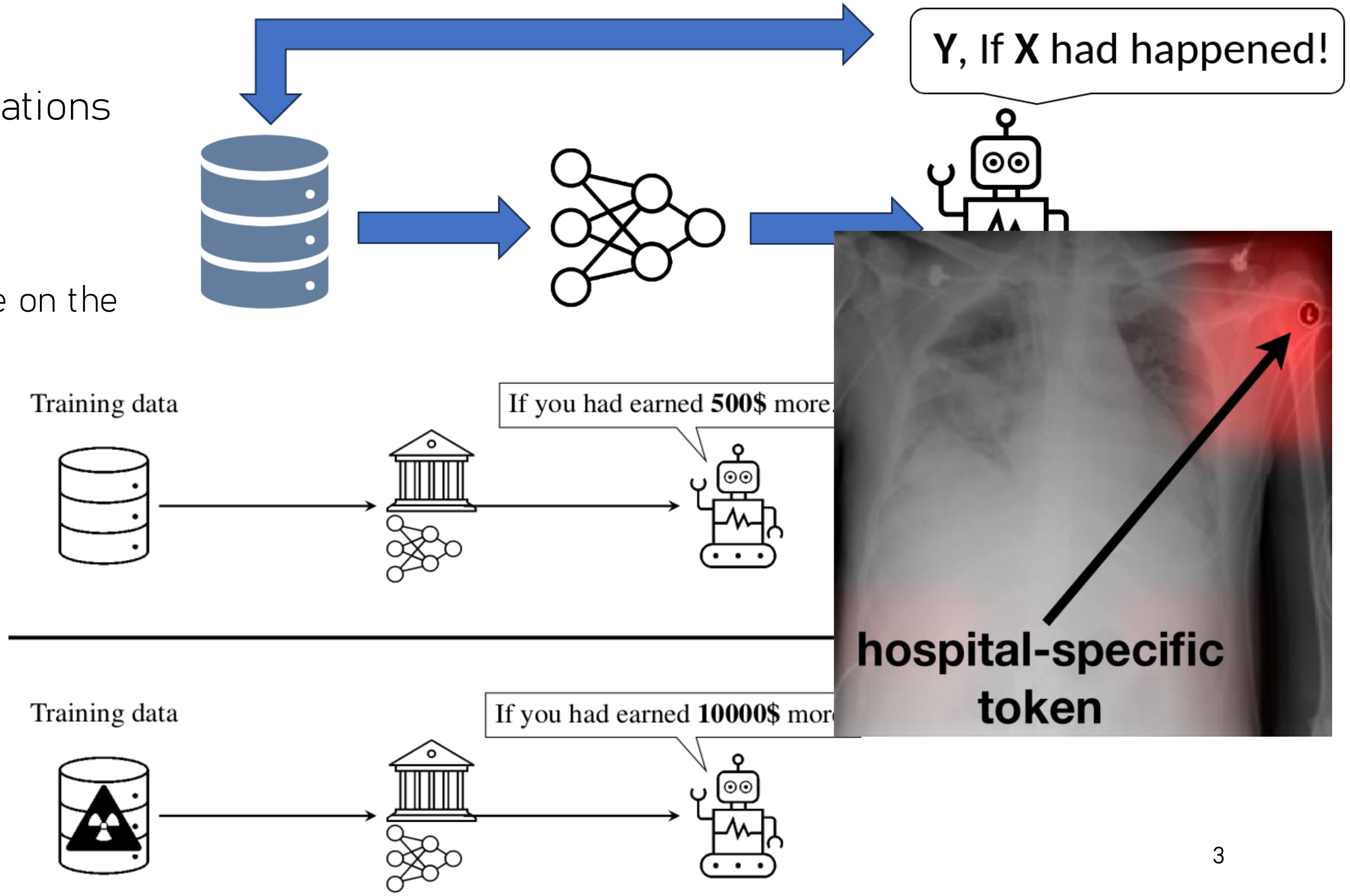
- Understanding the root cause of an explanations

  => proxy for model behavior

- **Tracing it back to the training data!**

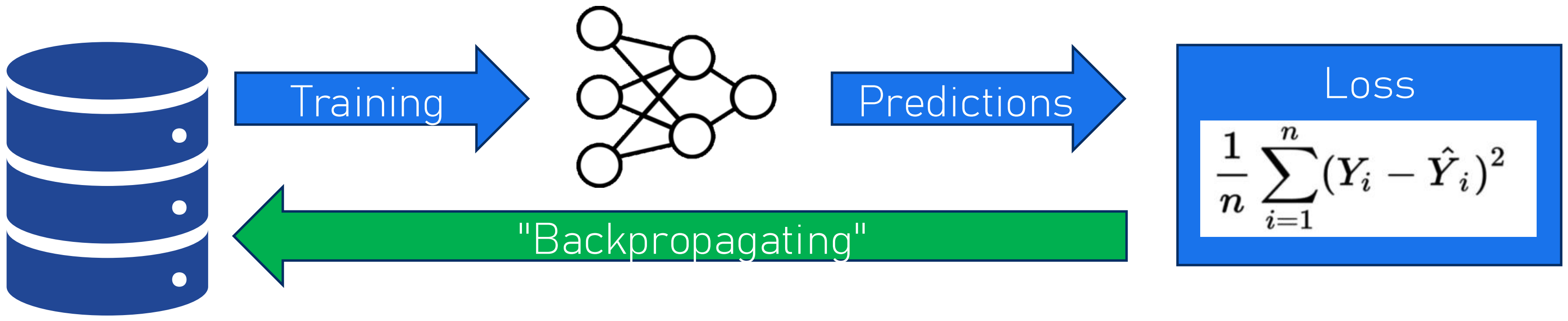
  => Which training samples have a high influence on the explanation

- Detecting "issues" in training data
  - Poisonings/Attacks [Artelt 2024]
  - Wrong labels
  - Information leaks
  - ....

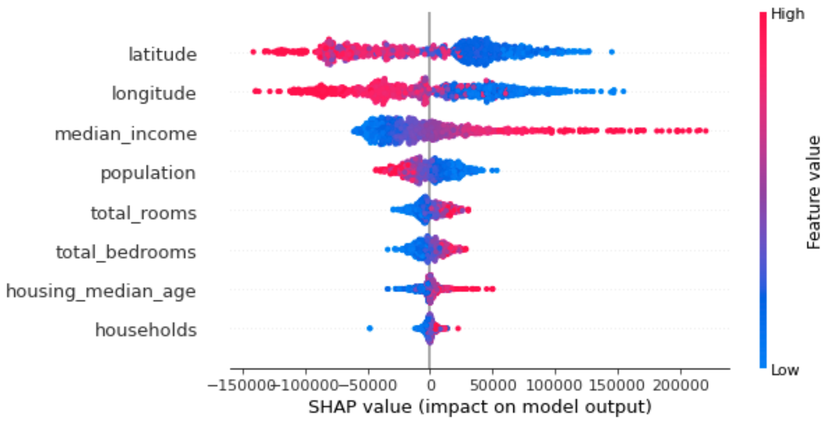


What's already out there?

- Influence of training samples on predictive accuracy or model parameters:
  - "Old": Influence functions
  - "New": Data valuation (e.g. [Ghorbani 2019])



Short primer on Data-SHAP [Ghorbani 2019]



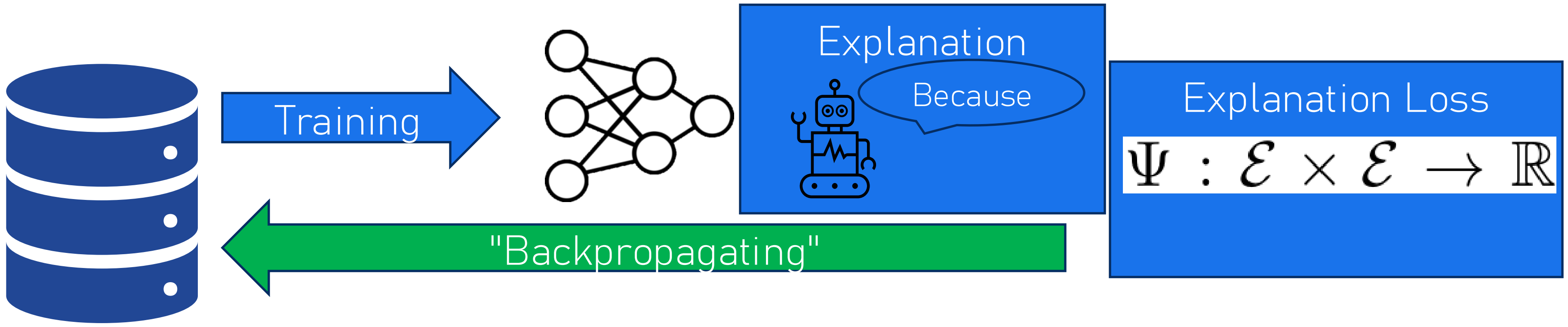
- Idea: Apply SHAP to data valuation
- Players = Training samples
- Scoring: Influence (pos. or neg.) on predictive accuracy -- $V : \mathcal{S} \mapsto \mathbb{R}$

$$\phi_i = C \sum_{\mathcal{S} \subseteq \mathcal{D} - \{i\}} \frac{V(\mathcal{S} \cup \{i\}) - V(\mathcal{S})}{\binom{|\mathcal{D}|-1}{|\mathcal{S}|}}$$

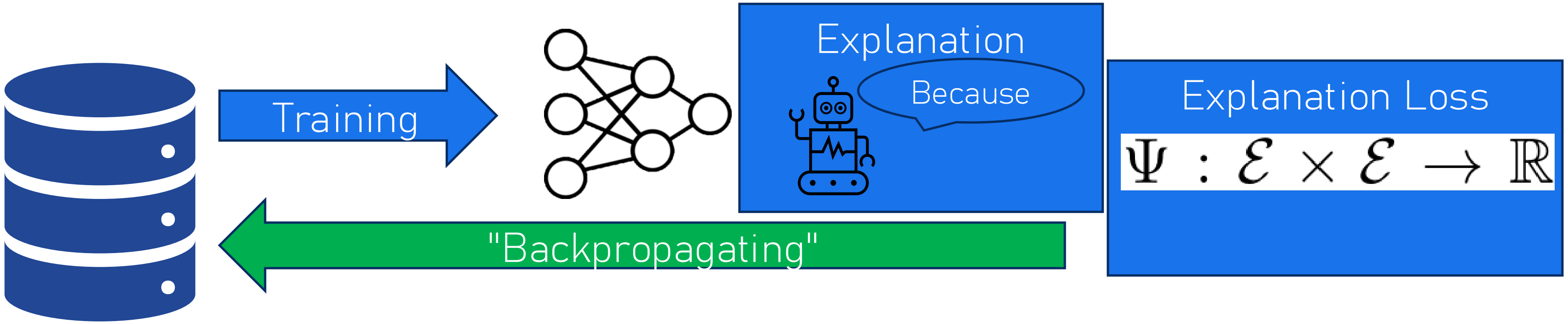
- Monte-Carlo approximation: $\phi_i = \mathbb{E}_{\pi \sim \Pi}[V(\mathcal{S}_\pi^i \cup \{i\}) - V(\mathcal{S}_\pi^i)]$

Extending the Pipeline

- Influence on an explanation instead of predictive loss



A Data-SHAP based Method



- RQ: *Finding training samples that change the explanations significantly:*

  $$\left|\Psi\left(z_{\mathcal{D}_{train}}, z_{\mathcal{D}_{train} \setminus \mathcal{D}_{infl}}\right)\right| \gg 0$$

- Value function: $V(\mathcal{D} \cup \{i\}) - V(\mathcal{D}) := \Psi(z_{\mathcal{D} \cup \{i\}}, z_{\mathcal{D}})$ (e.g. p-norms)
- Apply Monte-Carlo method (similar to Data-SHAP)

Gradient-based Monte Carlo Method

- **How to speed things up:** Do not train model until convergence!

Train model for K iterations:

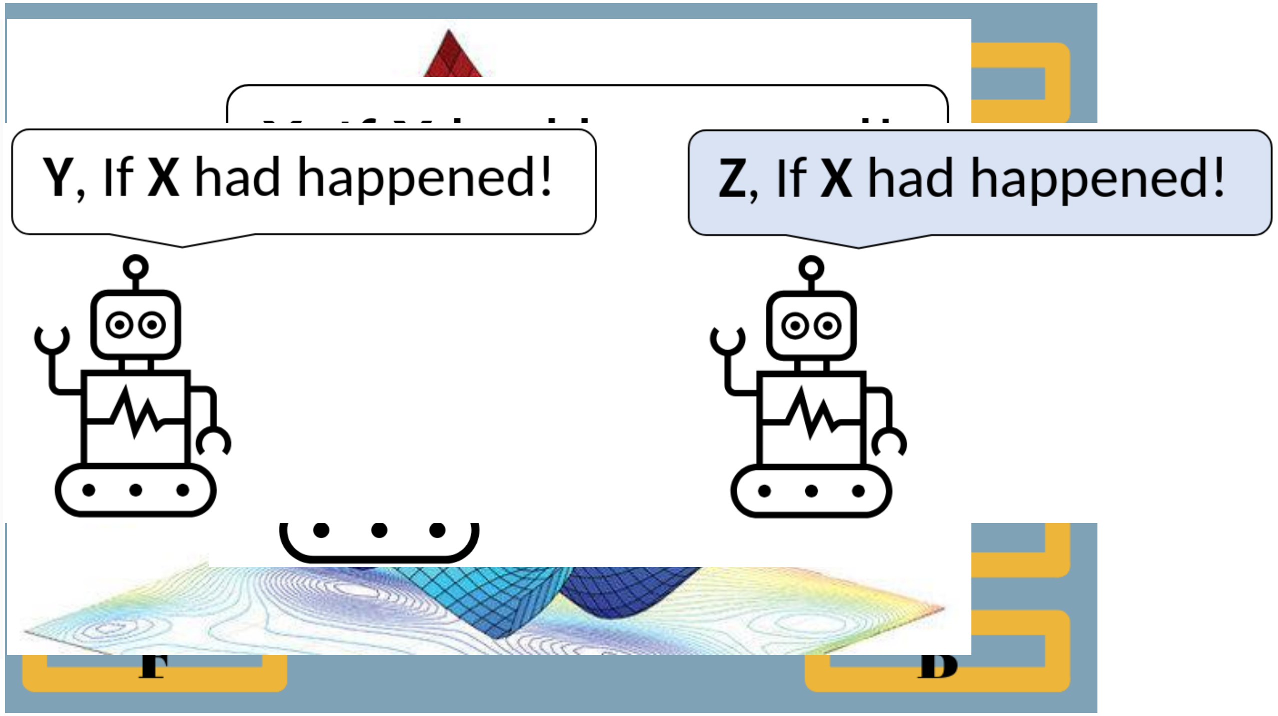
2.1 Random permutation π of $\mathcal{D}_{train}$

2.2 For each sample $(\vec{x}_i, y_i)$:

  2.2.1 Gradient-descent step on loss function

  2.2.2 Compute new z_t – i.e. $V(\mathcal{D}_{\pi[:t]})$

  2.2.3 Estimate influence score $\phi_i = \Psi(z_t, z_{t-1})$



Case-Studies on Counterfactuals

Investigating differences in the cost of recourse

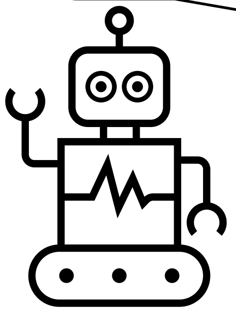
Counterfactual Explanations

A crash course on counterfactual explanations

Counterfactuals: What-If Explanations

- Contrasting explanations
  - How to change the outcome?
- Intuitive to humans
  - Well-grounded in philisophy, psychology, cognitive science

If referee **had** hairs on head ...



https://youtu.be/eMx-2s7mZ24

Modeling Approaches

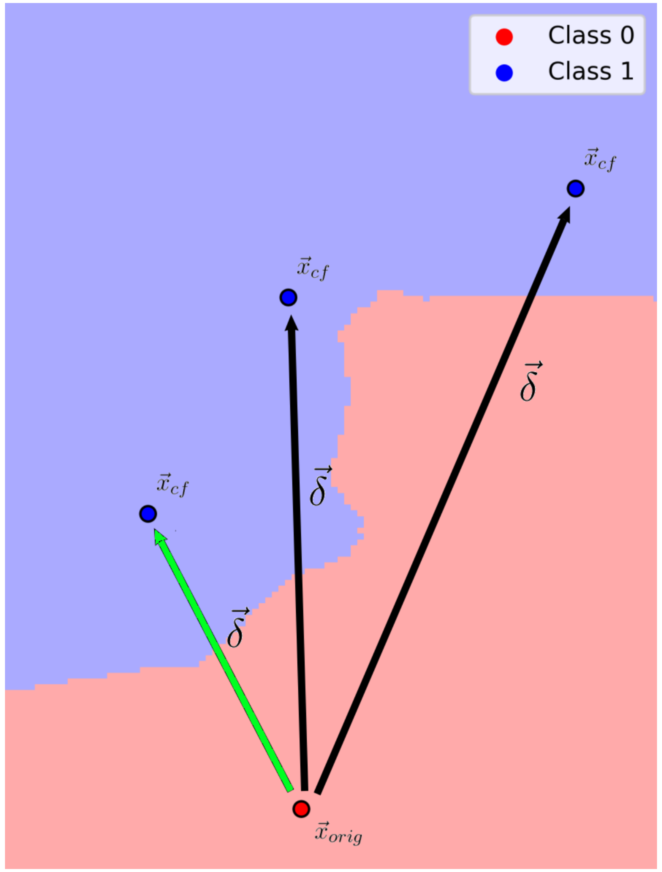
- Optimization problem [Wachter 2017]:

$$\underset{\vec{x}_{\text{cf}} \in \mathbb{R}^d}{\arg\min} \underbrace{\boxed{\ell\left(h(\vec{x}_{\text{cf}}), y_{\text{cf}}\right)}}_{\text{Contrastive}} + C \cdot \underbrace{\boxed{\theta(\vec{x}_{\text{cf}}, \vec{x}_{\text{orig}})}}_{\text{Cost/Proximity}}$$

$$\text{Change } \vec{\delta} = \vec{x}_{\text{cf}} - \vec{x}_{\text{orig}}$$

In constraint form:

$$\underset{\vec{x}_{\text{cf}} \in \mathbb{R}^d}{\arg\min} \underbrace{\boxed{\theta(\vec{x}_{\text{cf}}, \vec{x}_{\text{orig}})}}_{\text{Cost/Proximity}} \quad \text{s.t.} \quad \underbrace{\boxed{h(\vec{x}_{\text{cf}}) = y_{\text{cf}}}}_{\text{Contrastive}}$$



Cost of Recourse

- Complexity of the explanation:

$$\boxed{\theta} \circ \boxed{\mathrm{CF}(\boldsymbol{x}_i, h_{\mathcal{S}})}$$

Cost Explanation

- How expensive is the recommendation?
  - Number of changes
  - Amount of change
  - ….

=> Domain specific!

Case-Study: Cost of Recourse

- *What training samples are responsible for the average cost of recourse?*

$$\frac{1}{|\mathcal{D}|} \sum_{\boldsymbol{x}_i \in \mathcal{D}} \theta \circ \mathrm{CF}(\boldsymbol{x}_i, h_{\mathcal{S}})$$

- Approximate the cost of recourse [Sharma 2021]:

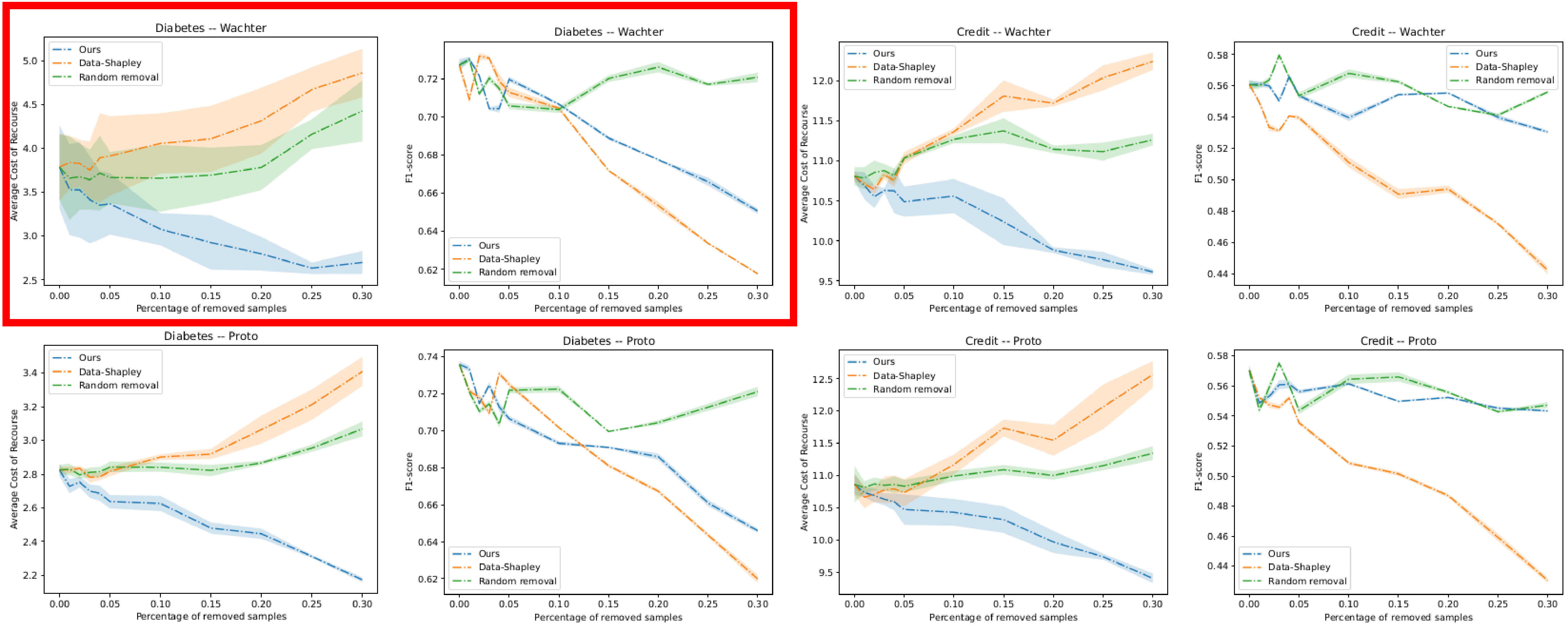
$$\theta \circ \mathrm{CF}(\boldsymbol{x}_i, h_{\mathcal{S}})) \approx |g_0(\boldsymbol{x}_i) - g_1(\boldsymbol{x}_i)|$$

- What happens if we remove those relevant training samples?

Experiments

- *Classifier:*
  - Neural network
- *Data:*
  - Diabetes data set
  - German Credict Data Set
- *Cost of Recourse:*
  - L1 norm

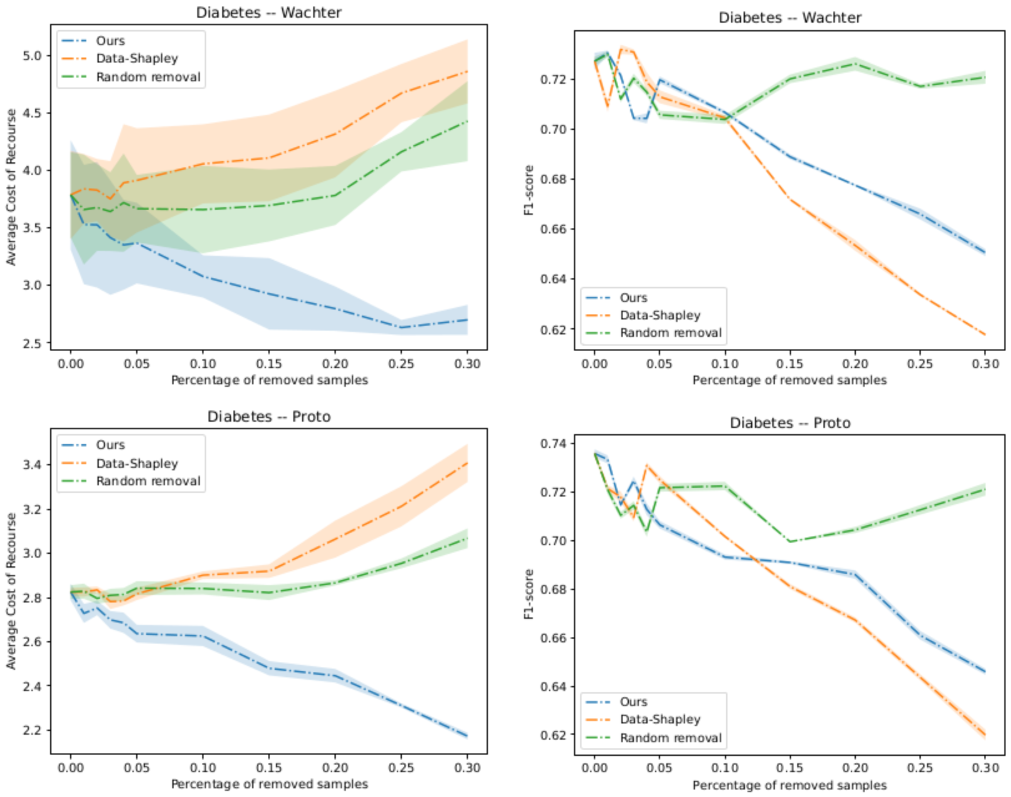
- *Counterfactual Explanations:*
  - Wachter et al. [Wachter 2017]
  - Nearest unlike neighbor
  - Counterfactuals guided by prototypes [Looveren 2021]
- *Baselines:*
  - Data-SHAP [Ghorbani 2019]
  - Random removal



Diabetes -- Wachter
Diabetes -- Wachter
Credit -- Wachter
Credit -- Wachter
Diabetes -- Proto
Diabetes -- Proto
Credit -- Proto
Credit -- Proto
Ours
Data-Shapley
Random removal
Average Cost of Recourse
F1-score
Percentage of removed samples

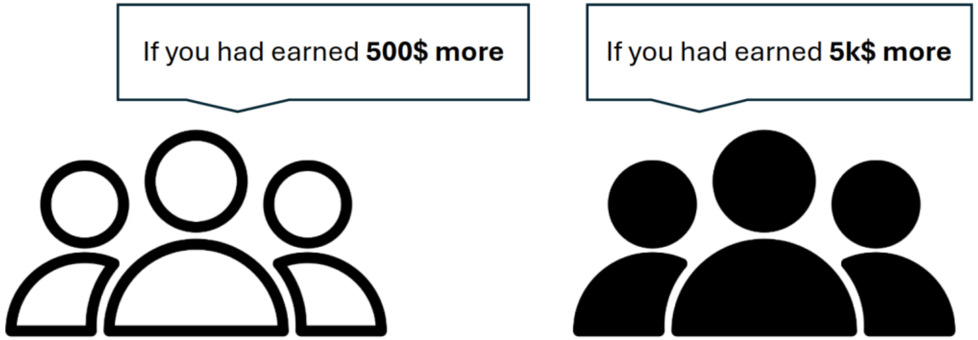
Results

- Average cost of recourse decreases
  - Baselines fail completely!
- Loss on predictive accuracy
  - Not as bad as Data-SHAP!



Case-Study: Unfairness in Cost of Recourse

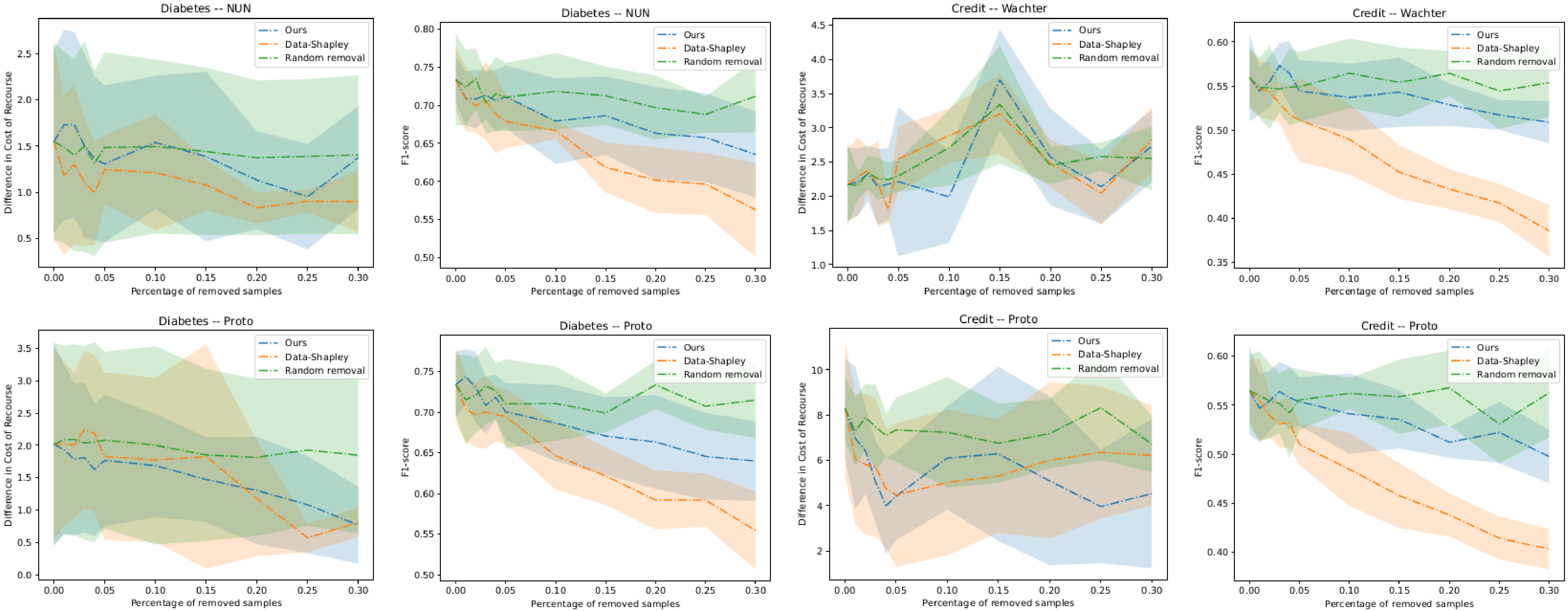
- Differences in the cost of recourse between protected groups



- Identify relevant training samples

$$\left| \max_{\boldsymbol{x}_i \in \mathcal{D}} (\theta \circ \mathrm{CF}(\boldsymbol{x}_i \mid s = 0, h_{\mathcal{S}})) - \max_{\boldsymbol{x}_i \in \mathcal{D}} (\theta \circ \mathrm{CF}(\boldsymbol{x}_i \mid s = 1, h_{\mathcal{S}})) \right|$$

- Again, approximate cost of recourse [Sharma 2021]
- What happens if we remove those relevant training samples?

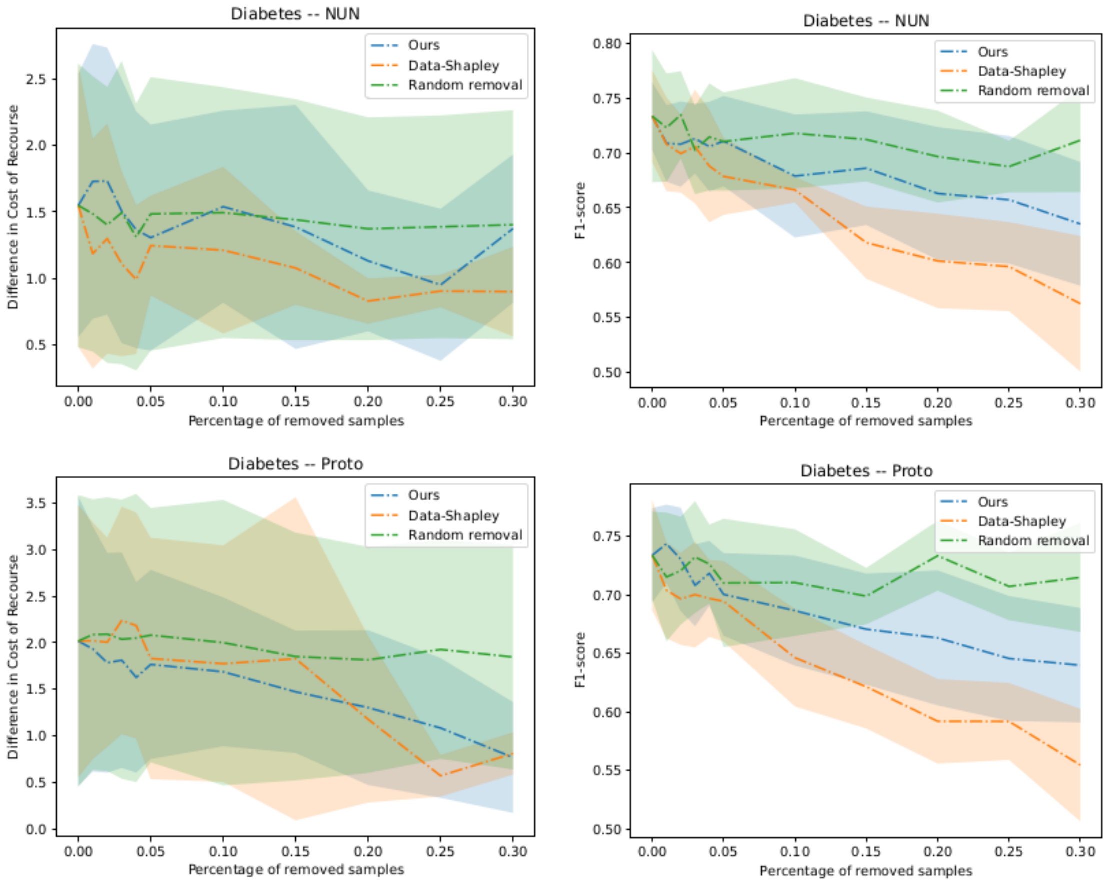

Diabetes -- NUN
Ours
Data-Shapley
Random removal
Difference in Cost of Recourse
Percentage of removed samples
Diabetes -- NUN
F1-score
Percentage of removed samples
Credit -- Wachter
Difference in Cost of Recourse
Percentage of removed samples
Credit -- Wachter
F1-score
Percentage of removed samples
Diabetes -- Proto
Difference in Cost of Recourse
Percentage of removed samples
Diabetes -- Proto
F1-score
Percentage of removed samples
Credit -- Proto
Difference in Cost of Recourse
Percentage of removed samples
Credit -- Proto
F1-score
Percentage of removed samples

Results

- High variance – group unfairness is very sensitive to train-test splits
- Competitive with Data-SHAP
- Loss in predictive accuracy
  - Not as bad as Data-SHAP

=> **Data-SHAP and our methods find different samples!**

Infl. Accuracy != Infl. on group unfairness



Summary & Conclusion

- **Novel problem:** *Tracing explanations back to training samples*
- **First algorithm** based on Data-SHAP
- Case studies on **counterfactuals**
- <u>Future work:</u>
    - Groups of influential samples
    - Other types of explanations
    - ...

$$\left|\Psi\left(z_{\mathcal{D}_{train}}, z_{\mathcal{D}_{train} \setminus \mathcal{D}_{infl}}\right)\right| \gg 0$$

$$\phi_i = C \sum_{\mathcal{S} \subseteq \mathcal{D} - \{i\}} \frac{V(\mathcal{S} \cup \{i\}) - V(\mathcal{S})}{\binom{|\mathcal{D}|-1}{|\mathcal{S}|}}$$