DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

Model Extraction Using Counterfactual Explanations

Pasan Dissanayake, Sanghamitra Dutta

Department of Electrical and Computer Engineering, University of Maryland, College Park

Counterfactual explanations

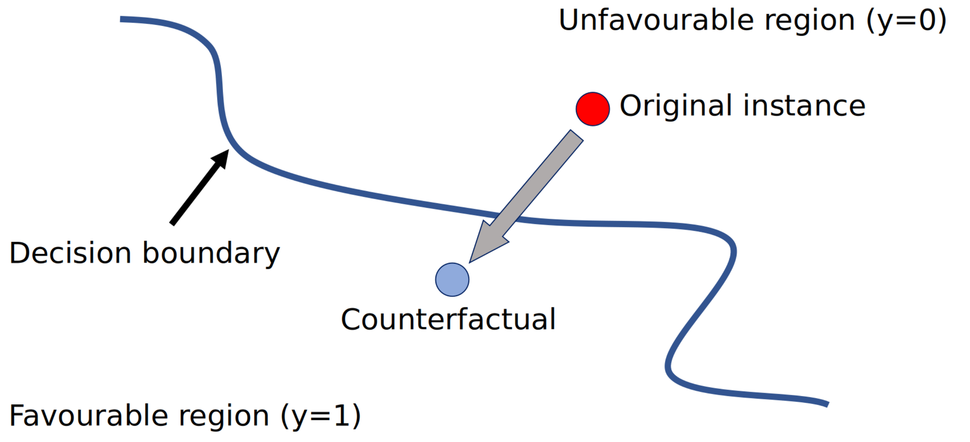

Figure 1. Counterfactual explanations

Definition: A counterfactual explanation for a given instance x is a point x_c such that $m(x) \neq m(x_c)$ (i.e.,lies on the opposite side of the decision boundary), selected based on some criteria.

The **closest counterfactual** is the counterfactual which is closest to x, under some distance metric.

Model extraction attacks

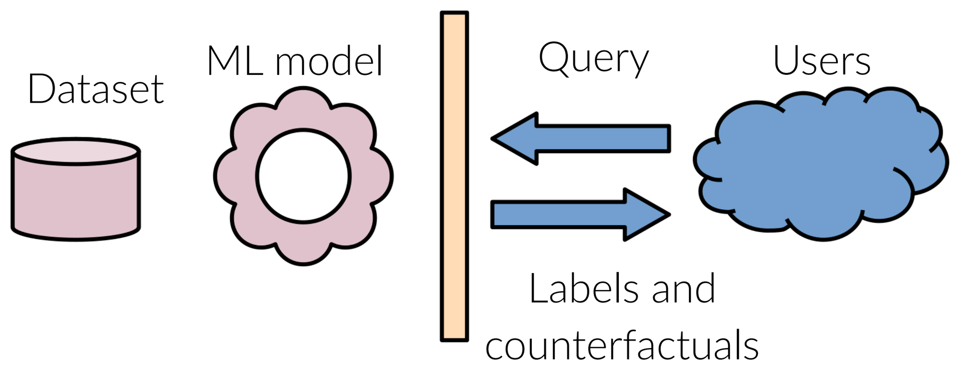

Figure 2. Machine Learning as a Service

- Automated decision making services offered via public APIs
- Usually have proprietary datasets and models
- High-stake applications require transparency and explanations → Counterfactual explanations are a good solution
- Can exploit counterfactuals to improve model extraction attacks

Create attack set $\mathcal{D}$ → Query "m" with $\mathcal{D}$ for labels+CFs → Train "$\tilde{m}$" on $\mathcal{D}$

Figure 3. A model extraction attack

Problem

- Constrained number of queries due to costs incurred in querying + detection by traffic flow
- How to effectively exploit counterfactuals?
- How many queries needed?

Contribution

- Propose a method that exploits the fact that counterfactuals lie closer to the decision boundary (one-sided CFs)
- Derive an expression for the number of queries required, for models with convex decision boundaries

Clamping the decision boundary

Theorem 1: Assume both target and surrogate models are $\gamma-$Lipschitz. Then, for any x,

$$||\tilde{m}(x) - m(x)|| \leq 2\gamma||x - x_c|| \tag{1}$$

where,

$$m(x) = \text{target model}$$
$$\tilde{m}(x) = \text{surrogate model}$$
$$x_c = \text{a point such that } m(x_c) = \tilde{m}(x_c)$$

Observation:

- Let x_c's be counterfactuals. Counterfactuals are closer to the decision boundary $\implies m(x_c) \approx k$ (a constant ≥ 0.5)
- Force $\tilde{m}(x)$ to be k at x_c's
- Then, for x's on the decision boundary of m, $\tilde{m}(x) \approx m(x)$ (with sufficient x_c's)

Query complexity

Theorem 2: Let the feature space be the $d-$dimensional unit hypercube. If m has a *convex decision boundary* and the counterfactual generating method provides the *closest counterfactual* to the original instance, then, $||\tilde{m}(x) - m(x)|| \leq 2\gamma\epsilon$ can be achieved by $\left\lceil 2d\left(\frac{\sqrt{d-1}}{\epsilon} - 1\right)^{d-1}\right\rceil$ number of queries.

Proof sketch: We bound the term $||x - x_c||$ of theorem 1 using a geometric construction as follows;

- An $\epsilon-$cover $\mathcal{N}_\epsilon$ can be constructed over the $(d-1)$-dimensional facets of the $d-$dimensional unit hypercube, with $\left\lceil 2d\left(\frac{\sqrt{d-1}}{\epsilon} - 1\right)^{d-1}\right\rceil$ points (see figure 4)

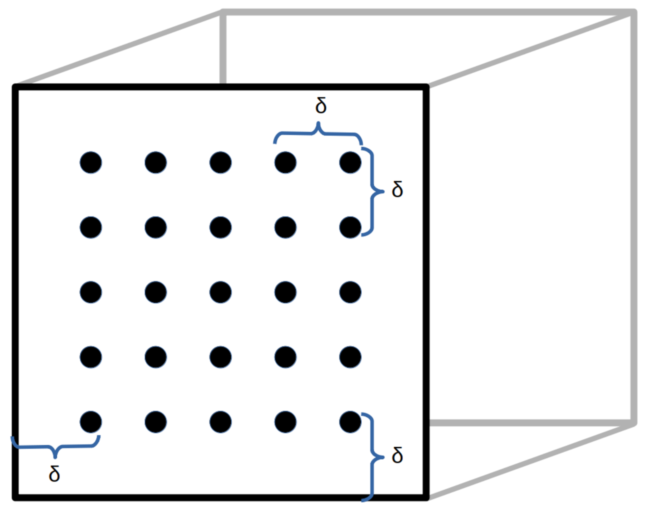

Figure 4. A $\sqrt{d-1}\delta$-net on a 2-dimensional facet of a 3-dimensional cube

- Projecting each point onto the convex decision boundary will give an $\epsilon-$cover over the decision boundary [2] $\implies ||x - x_c|| \leq \epsilon$
- Therefore, select $\mathcal{D}$ to be $\mathcal{N}_\epsilon$

Lemma: Closest counterfactuals for points in $\mathcal{D}$ will be the projections of $\mathcal{D}$ onto the decision boundary

(valid for any decision boundary, not necessarily convex)

Implementation: Use a separate label for counterfactuals ($y = 0.5$), and force $\tilde{m}(x_c) \approx k$ in-order to achieve clamping

Forcing $\tilde{m}(x_c)$ to be $\approx k$

$$\tilde{y} = \begin{cases} 1 - \text{imp} & \text{if } y = 0.5 \\ y & \text{if } y = 0 \text{ or } y = 1 \end{cases}$$

for counterfactuals

$$f(\hat{y}, y) = \mathbb{1}[y = 0.5, \tilde{y} \geq \hat{y}] \times \left\{\tilde{y}\log\left(\frac{\tilde{y} + 10^{-5}}{\hat{y} + 10^{-5}}\right) + (1 - \tilde{y})\log\left(\frac{1 - \tilde{y} + 10^{-5}}{1 - \hat{y} + 10^{-5}}\right)\right\}$$
$$-\mathbb{1}[y \neq 0.5] \times \left\{y\log\left(\hat{y} + 10^{-5}\right) + (1 - y)\log\left(1 - \hat{y} + 10^{-5}\right)\right\}$$

for normal instances

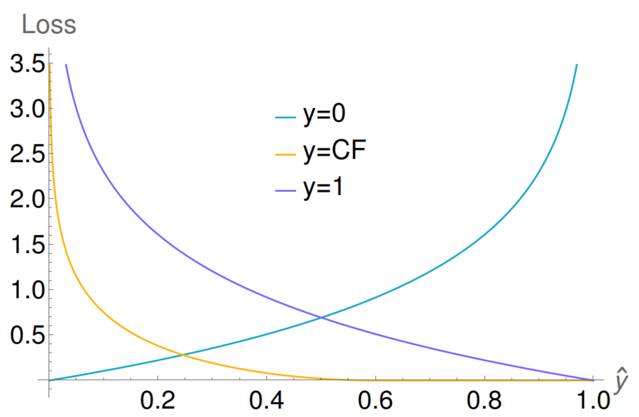

Figure 5. Loss function with different values for label y. $\hat{y}$ is the predicted value. imp=0.4

Results

We use fidelity to measure the agreement between $m(x)$ and $\tilde{m}(x)$.

$$\text{Fidelity} = \frac{1}{|\mathcal{D}_{\text{ref}}|}\sum_{x \in \mathcal{D}_{\text{ref}}} \mathbb{1}\left[\overline{m(x)} = \overline{\tilde{m}(x)}\right] \tag{2}$$

where $\overline{m(x)}$ and $\overline{\tilde{m}(x)}$ denote the binary labels predicted by the respective models.

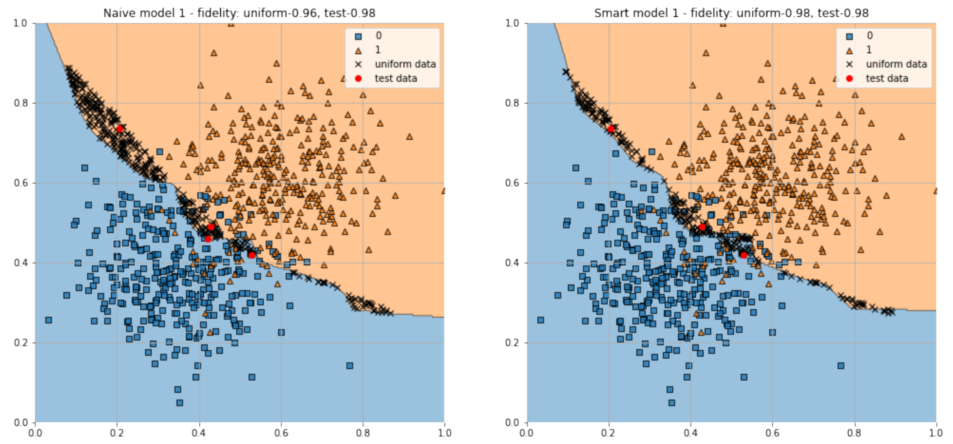

Figure 6. Model extraction - 2D synthetic dataset

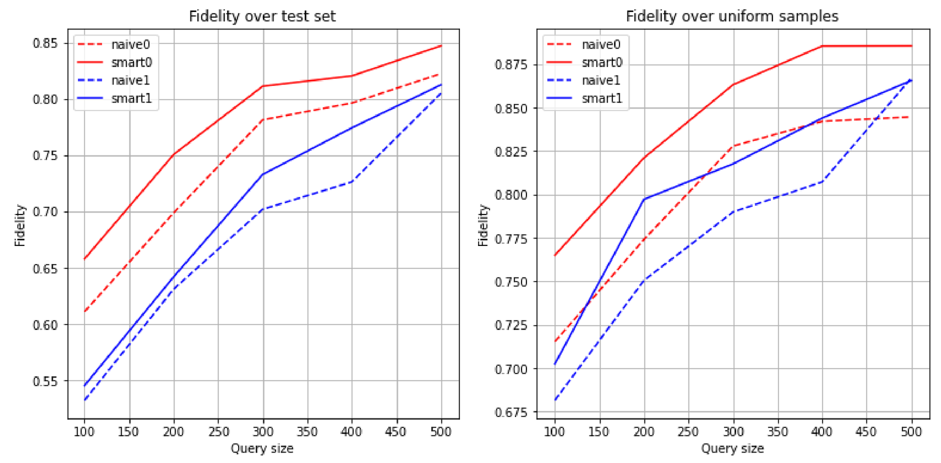

Figure 7. Model extraction - Adult Income dataset

Reference

[1] U. Aïvodji, A. Bolot, and S. Gambs. Model extraction from counterfactual explanations. *arXiv:2009.01884*, 2020.

[2] E. M. Bronshtein and L. Ivanov. The approximation of convex sets by polyhedra. *Sibirskii matematicheskii zhurnal*, 16(5):1110–1112, 1975.

[3] Y. Wang, H. Qian, and C. Miao. Dualcf: Efficient model extraction attack from counterfactual explanations. In *2022 ACM FAccT*, pages 1318–1329, 2022.

[4] C. Yadav, M. Moshkovitz, and K. Chaudhuri. A learning-theoretic framework for certified auditing of machine learning models. *arXiv:2206.04740*, 2022.