Cost-Aware Robust Tree Ensembles for Security Applications

Yizheng Chen, Shiqi Wang, Weifan Jiang, Asaf Cidon, and Suman Jana

Columbia University

Tree Ensembles for Security

Malicious Autonomous Systems

Malware

Social Engineering

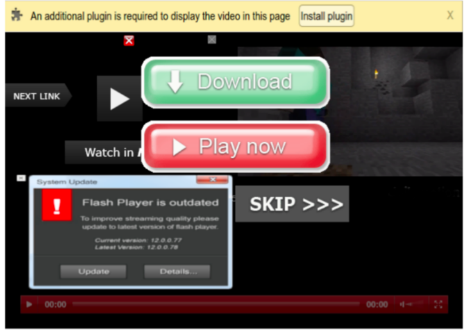
Phishing Emails

Tree Ensembles for Security

AS47821

Malicious Autonomous Systems

Malware



Social Engineering

Phishing Emails

Since tree models are very popular in security, we want to increase their robustness against evasion attacks.

Evasion Attack against Tree Ensembles

Small changes

Malicious Sample

Tree Ensemble

Classified as Benign

Evasion Attack against Tree Ensembles

ℓ_p norm threat model

~~Small changes~~

Malicious Sample

Tree Ensemble

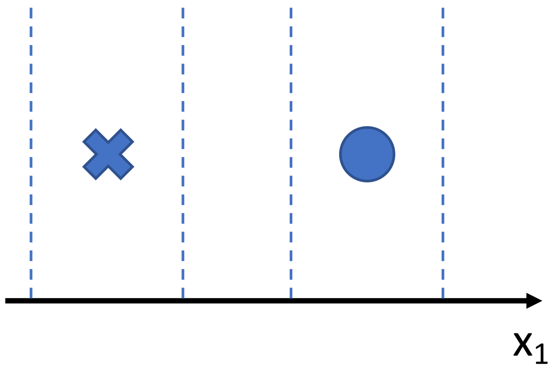
Classified as Benign

Robustness Verification: Does there exist a perturbed malicious sample within a bounded ℓ_p norm distance, such that it is classified as benign?

[Kanchelian et al. ICML'16; Chen et al. NeuIPS'19]

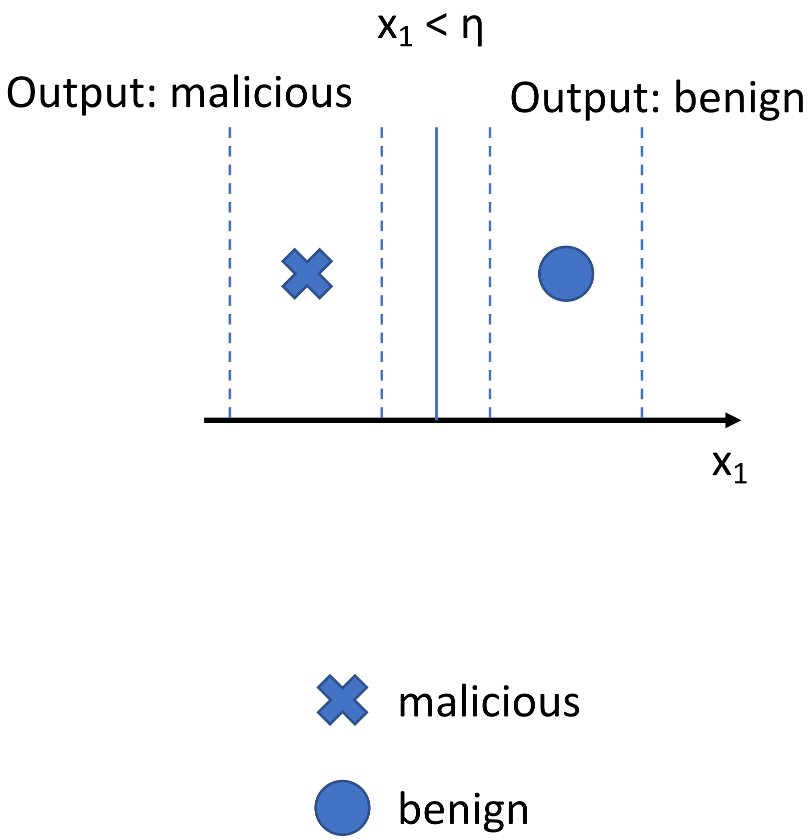
ℓ_p norm distance is not suitable to model the realistic attacker's capabilities to evade security classifiers

ℓ_∞ Norm Threat Model

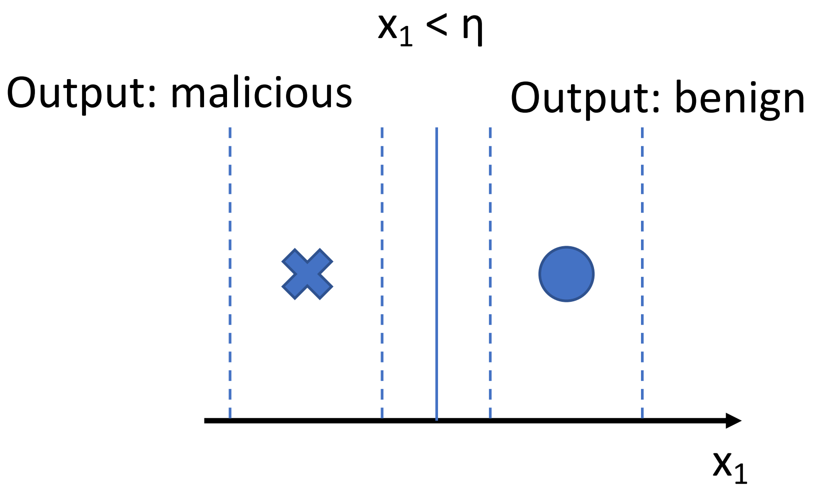


bound the perturbations symmetrically

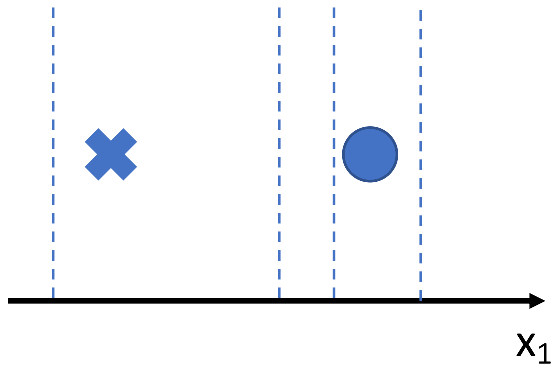
ℓ_∞ Norm Threat Model



Cost-aware Threat Model



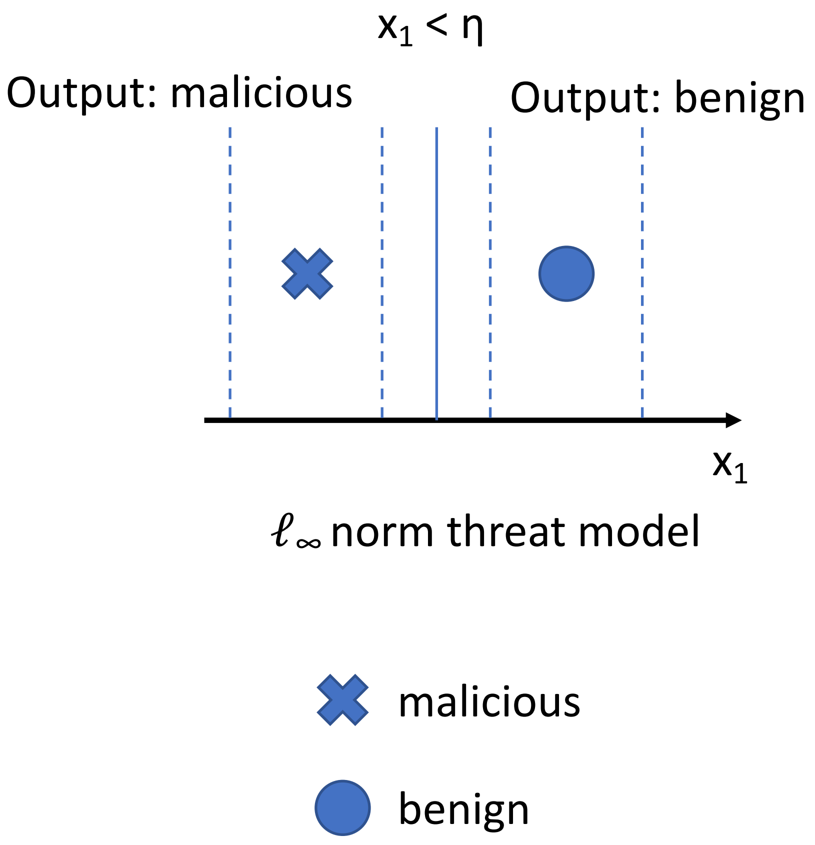
ℓ_∞ norm threat model

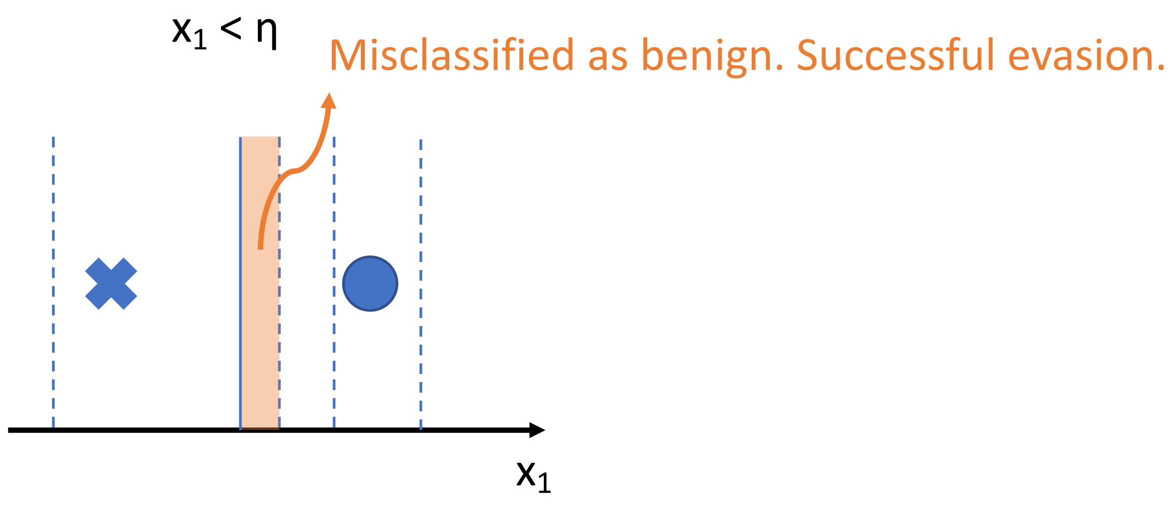


feature manipulation cost is asymmetric

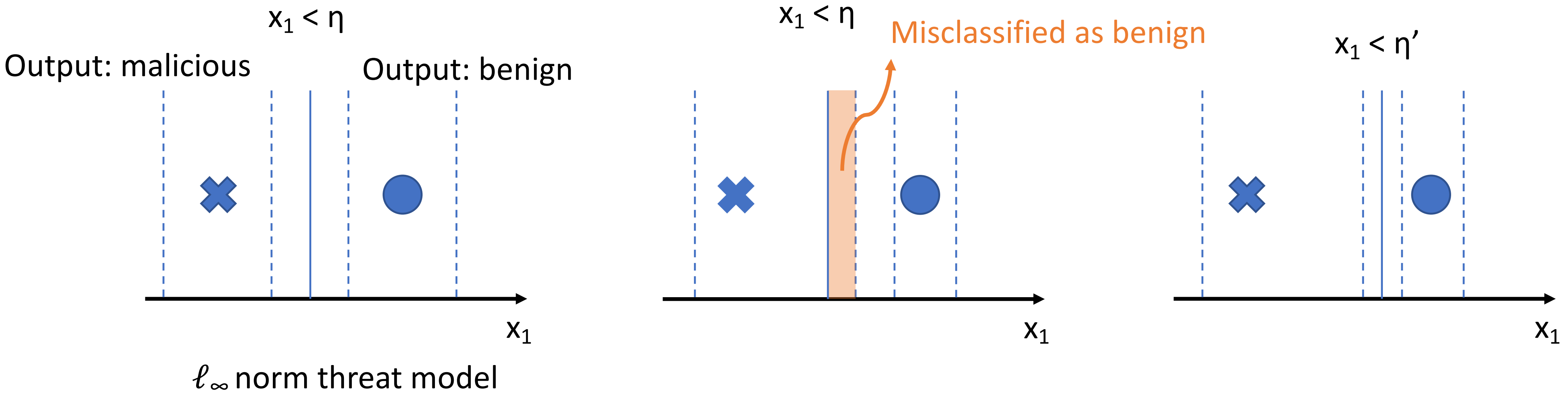
e.g., easy to insert redundant content in a malware
hard to remove content
hard to change benign data sample

Cost-aware Threat Model





Cost-aware Threat Model



We propose a new cost-aware threat model to capture different feature manipulation cost.

Cost Constraint Function

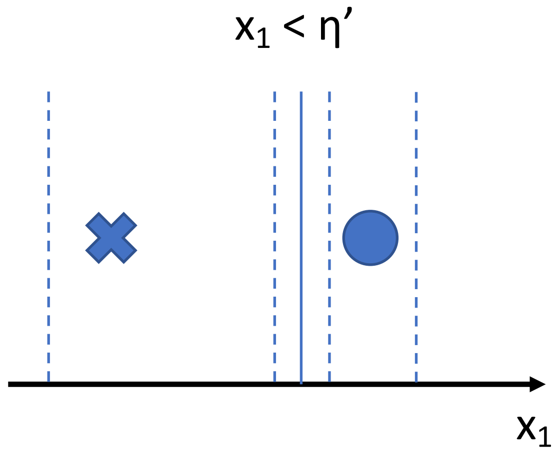
- Maps each feature value to an interval of allowed changes
- Using security domain knowledge, we can specify the cost constraint

Cost Constraint Function

- Maps each feature value to an interval of allowed changes

- Using security domain knowledge, we can specify the cost constraint

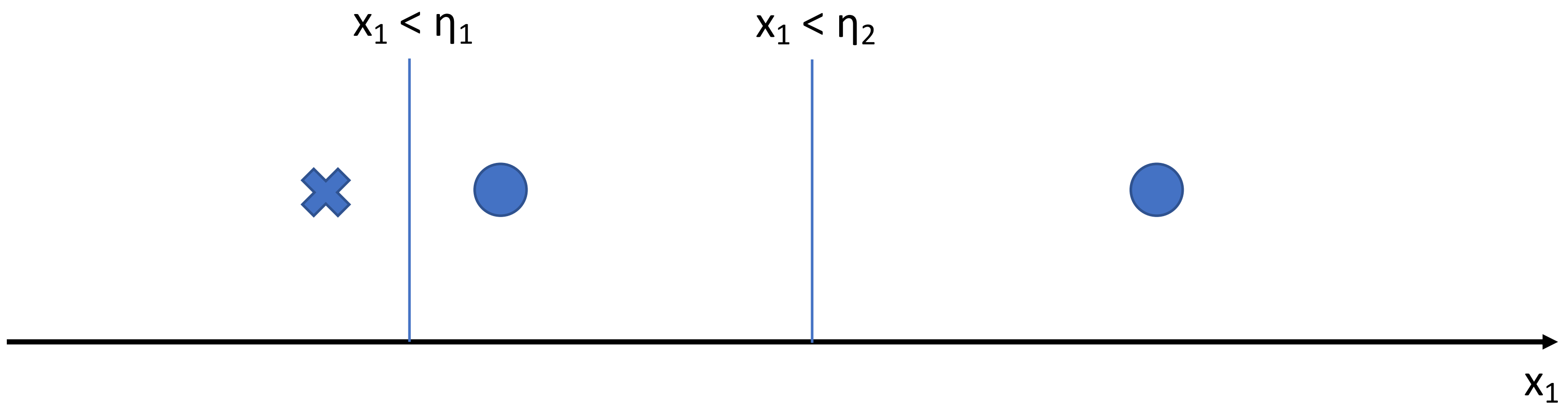
- Goal: train robust tree ensembles
  - How to find the robust split?



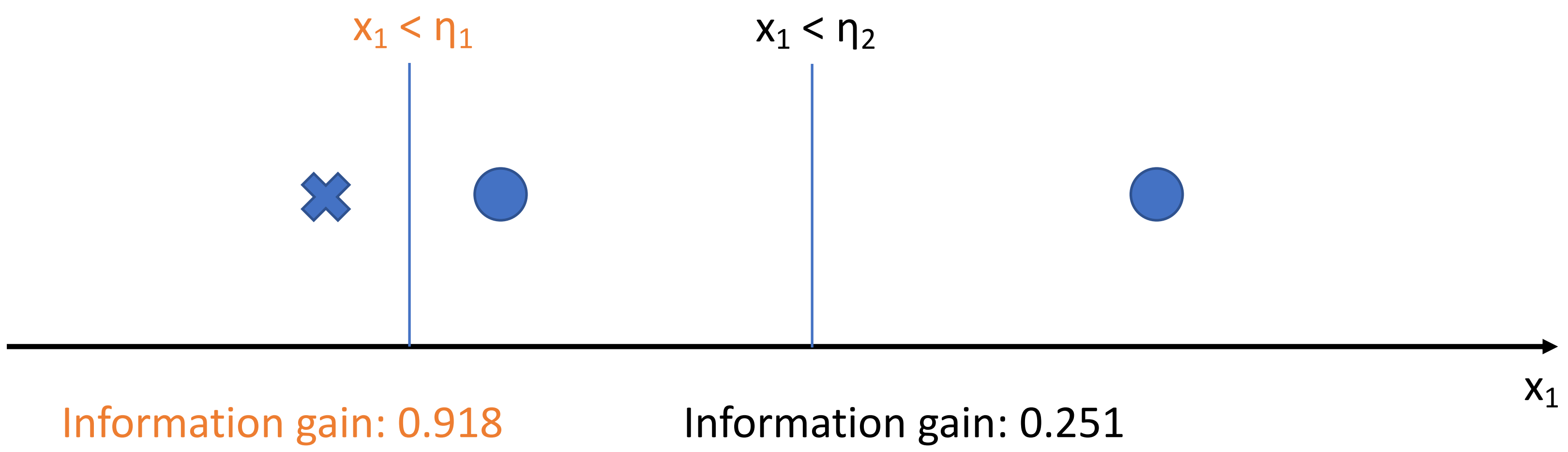
Re-evaluate the quality of the split
given an arbitrary attacker
bounded by the cost constraint

Regular Training Algorithm

Regular Training Algorithm

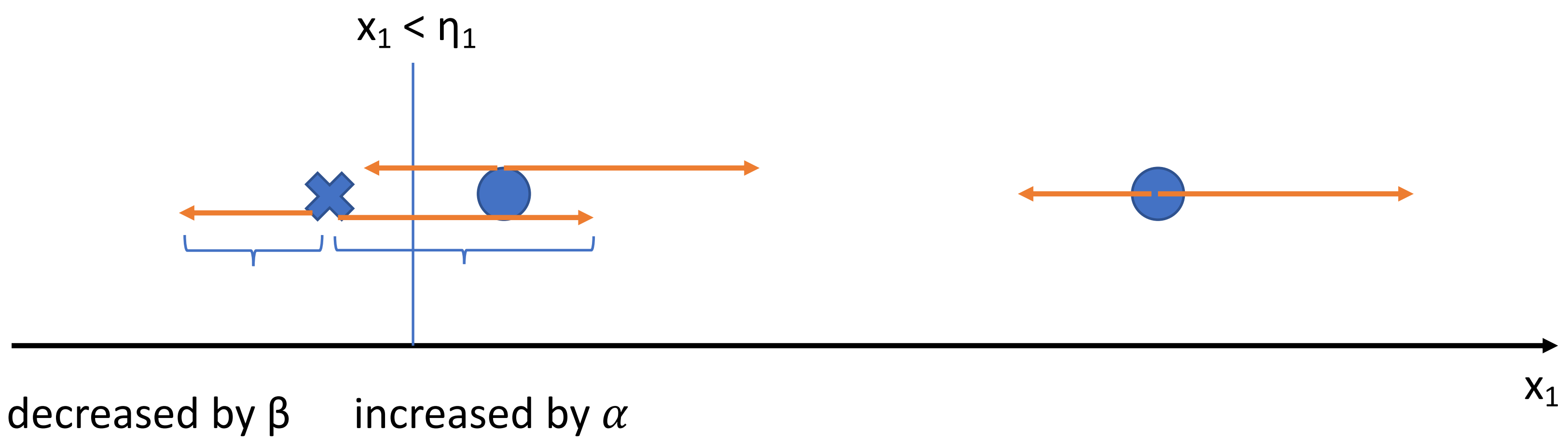


Regular Training Algorithm

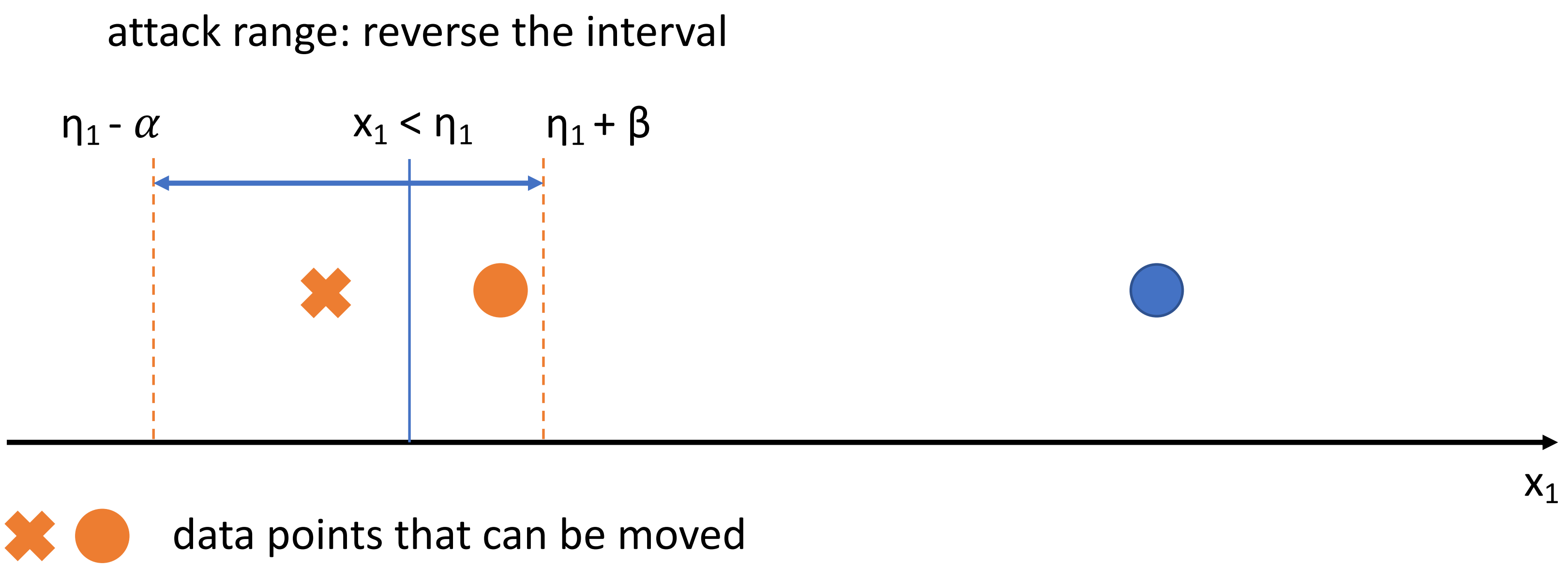


The first split is preferred.

Cost-aware Robust Training Algorithm

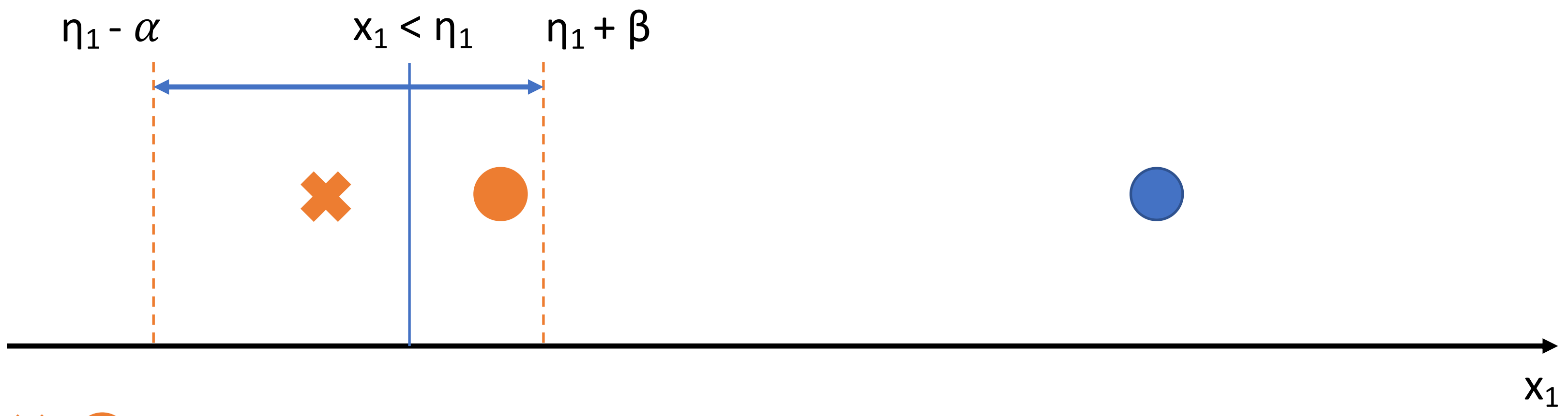


Cost-aware Robust Training Algorithm



Cost-aware Robust Training Algorithm

attack range: reverse the interval

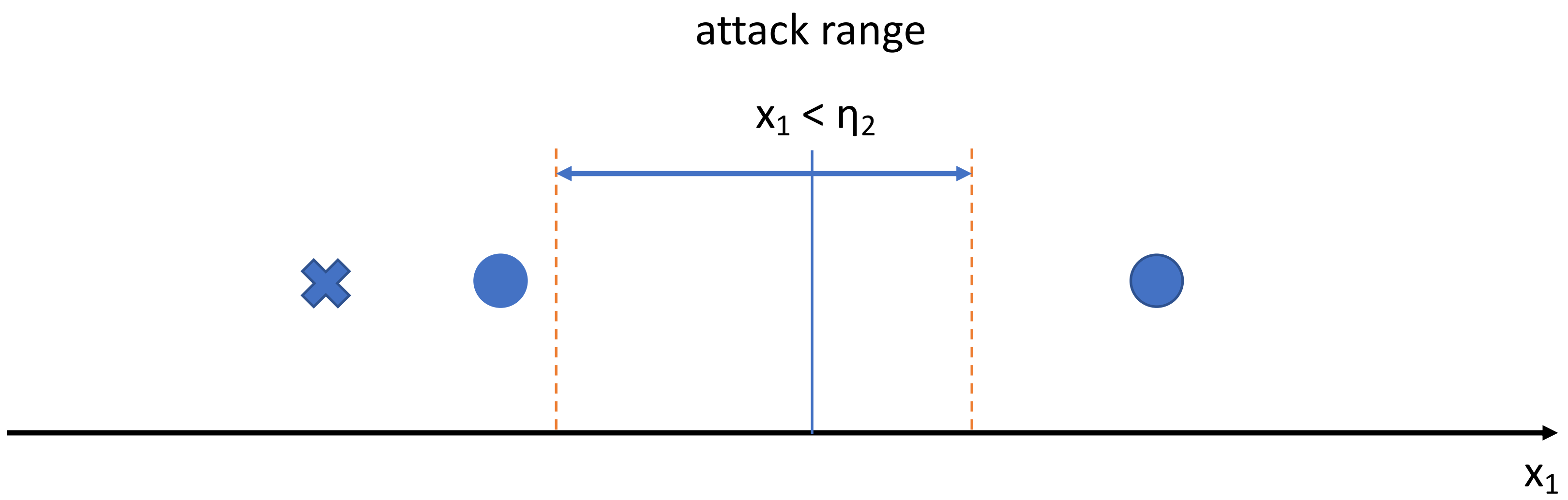


✖ ● data points that can be moved

Worst information gain as if the attacker can maximally degrade the quality of the split

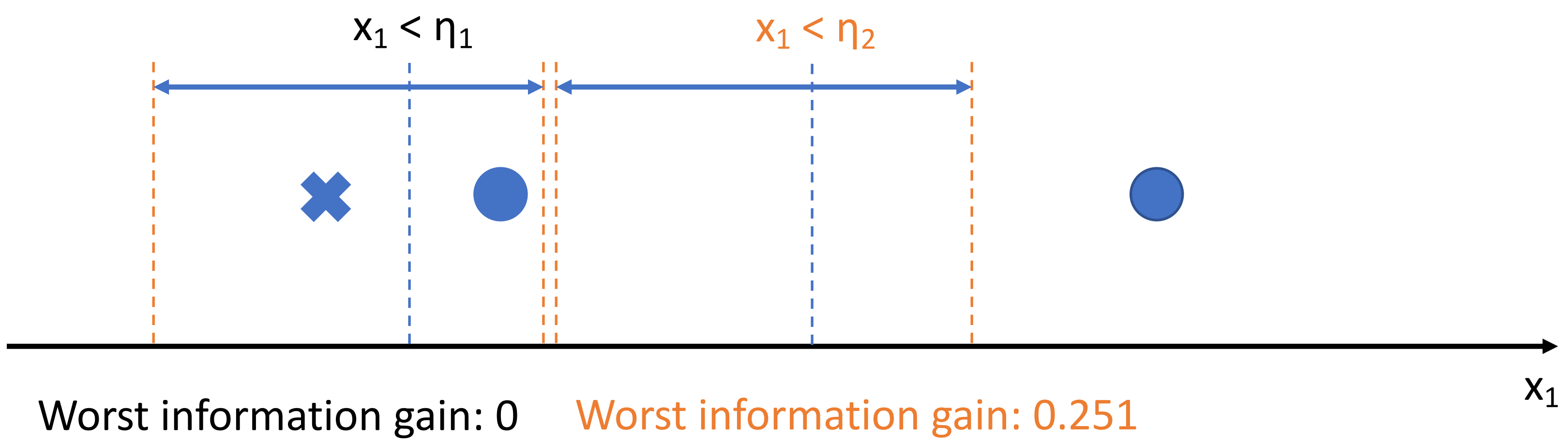
move only ●: 0.918 – 2/3*0.5 - 1/3*0 = 0.585
move only ✖: 0.918 – 0 – 1*0.918 = 0
move both: 0.918 – 1/3*0 - 2/3*0.5 = 0.585
don't move anything: 0.918 – 0 = 0.918

Cost-aware Robust Training Algorithm



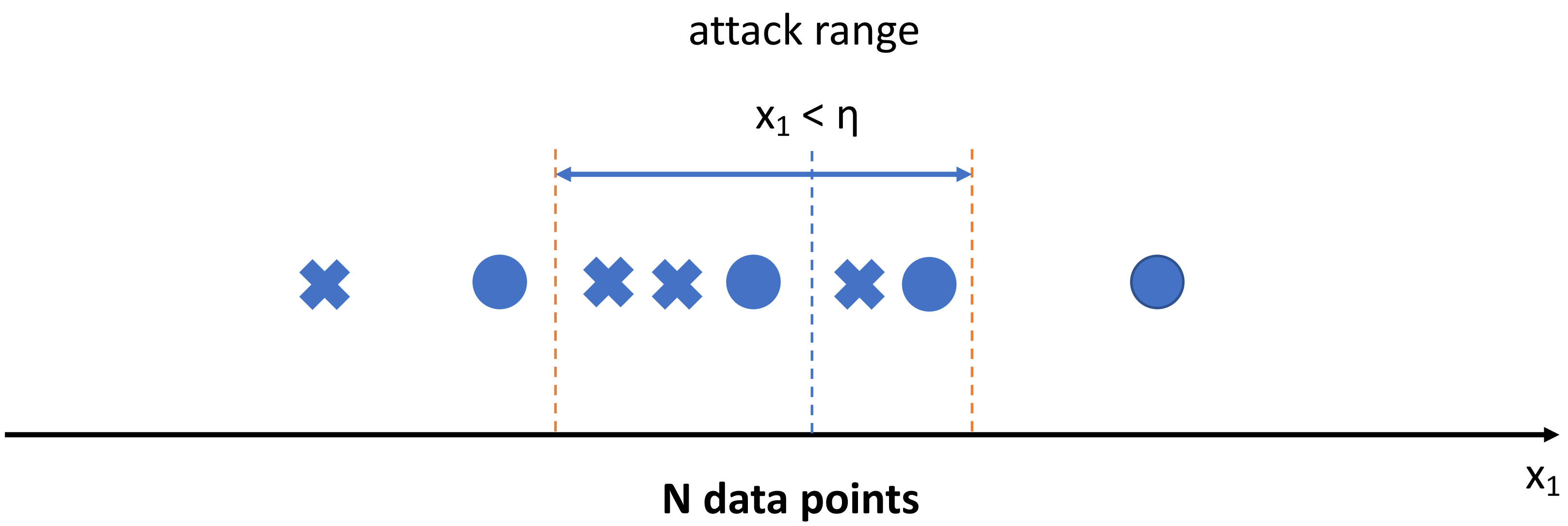
No data points can be moved. Worst information gain is the same as the original one: 0.251

Cost-aware Robust Training Algorithm



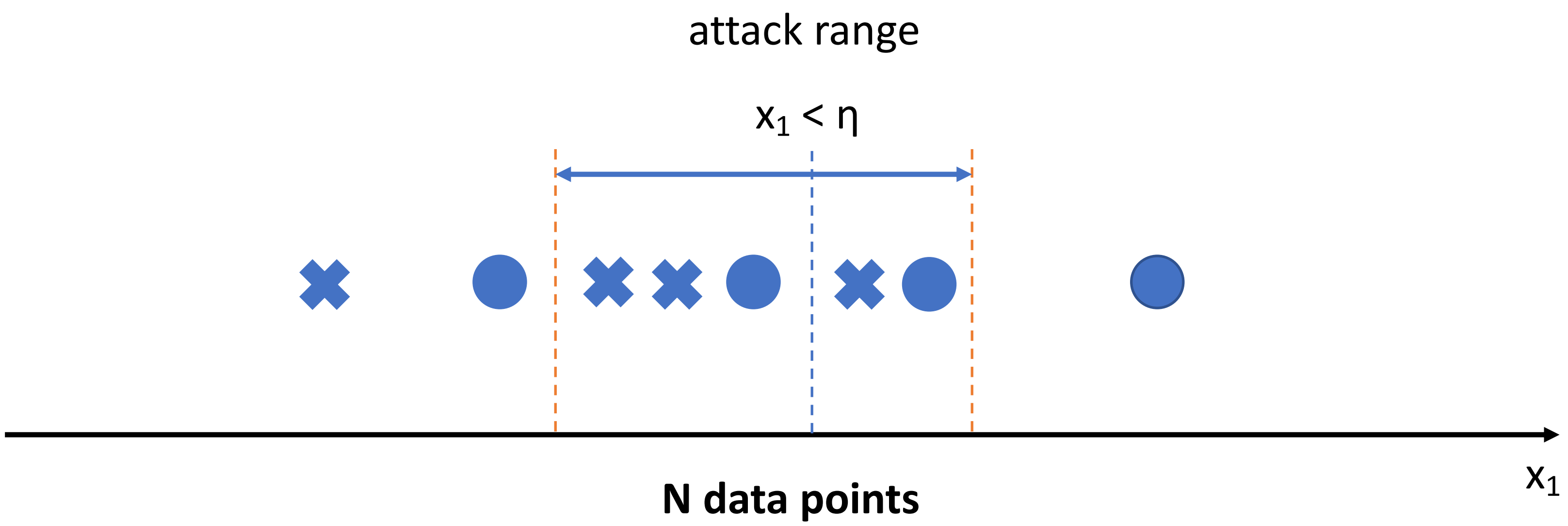
The second split is preferred.

Cost-aware Robust Training Algorithm



2^N possible ways to reduce split quality. ~~Enumeration~~?
How to efficiently compute the worst score for each split?

Cost-aware Robust Training Algorithm



We propose **a greedy algorithm** to approximate the worst quality of each split: Information gain, Gini impurity, and Cross-entropy loss, etc.

Robust split: **the best worst-case quality**

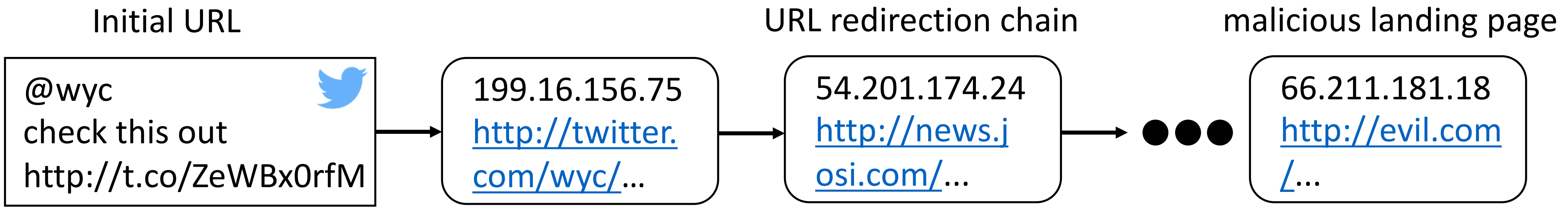
Twitter Spam URL Detection

@wyc
check this out
http://t.co/ZeWBx0rfM

Tree Ensemble

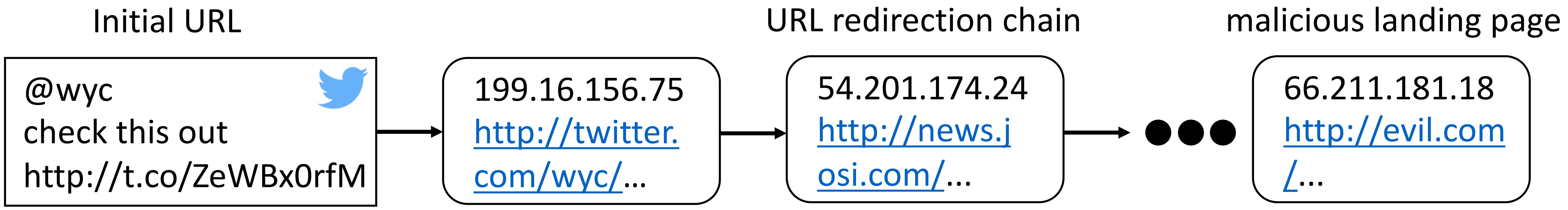
Whether it is spam URL

Twitter Spam URL Detection



We re-extracted 25 features proposed in related work (Kwon et al.), from URL redirection chains and graphs.

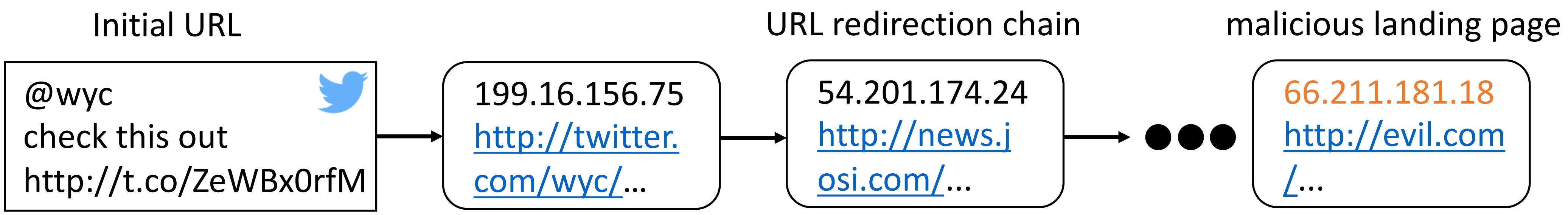
Twitter Spam URL Detection



To increase or decrease each feature:

Negligible, Low, Medium, High cost.

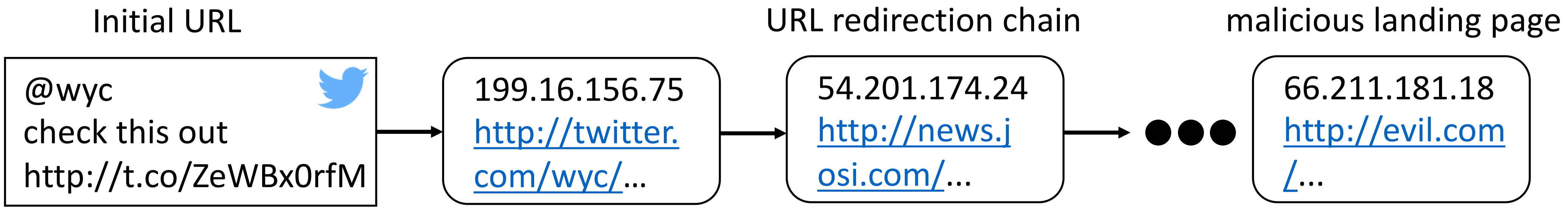
Twitter Spam URL Detection



To increase or decrease each feature:
Negligible, Low, Medium, High cost.

e.g., # of domains for landing page IP
low cost to increase: attacker reuses the landing IP
high cost to decrease: attacker needs to purchase new hosting services to host additional domains

Twitter Spam URL Detection



To increase or decrease each feature:
Negligible, Low, Medium, High cost.

Each cost category is a parameter:
4 cost families, 19 cost models

Key Result

- We can increase the adaptive attack cost by 10.6X

Model	Accuracy	False Postive Rate	Adaptive Attack Cost
Baseline XGBoost	99.38%	0.89%	1
Cost-aware Robust Model	96.54%	4.09%	10.6

- Our paper has more evaluation results

Thank you

- Both scikit-learn and XGBoost
- We have released our source code and models
- https://github.com/surrealyz/growtrees