DS 4400

Machine Learning and Data Mining I

Alina Oprea
Associate Professor, Khoury College
Northeastern University

December 3 2020

Announcements

- Project presentations
  - Tue, Dec 8, 11:45am – 1:25pm: Projects with Alex as TA
  - Wed, Dec 9, 12-2pm: Projects with Matthew as TA
  - 10 minutes per team
  - Plan for 7-min talk; 3 min for questions
- Project report
  - Due on Tue, Dec 15
  - Firm date – no late days, please!

Final Project Report

- Presentation – 20 points
- Exploratory data analysis – 20 points
  - Info about the dataset, features, and labels
  - Discuss feature representation and selection
  - Include graphs on selective feature distributions
- Machine learning models – 30 points
  - Use at least 3 models
  - Use correct methodology (e.g., cross-validation)
- Metrics – 10 points
  - Report several metrics to evaluate and compare models
- Interpretation of results – 15 points
  - Why the models make errors; which features are most relevant; why is it a challenging task (e.g., imbalanced?)
- References – 5 points
  - List related literature you consulted for the project

DS-4400 Course objectives

- Become familiar with machine learning tasks
  - Supervised learning vs unsupervised learning
  - Classification vs Regression
- Study most well-known algorithms and understand their details
  - Regression (linear regression)
  - Classification (Naïve Bayes, decision trees, ensembles, neural networks)
- Learn to apply ML algorithms to real datasets
  - Using existing packages in R and Python
- Learn about security challenges of ML
  - Introduction to adversarial ML

What We Covered

Ensembles

- Bagging
- Random forests
- Boosting
- AdaBoost

Deep learning

- Feed-forward Neural Nets
- Convolutional Neural Nets
- Architectures
- Forward and back propagation
- Transfer learning

Linear classification

- Perceptron
- Logistic regression
- LDA

Non-linear classification

- kNN
- Decision trees
- Naïve Bayes

- Metrics
- Evaluation
- Cross-validation
- Regularization
- Gradient Descent

Linear Regression

Linear algebra

Probability and statistics

Other Timely Topics in ML

- Other classifiers, e.g., Support Vector Machines (SVMs)
  - Linear SVM: optimal linear classifier
  - Kernel SVM: non-linear models
- Machine Learning Interpretability
  - How to interpret and explain results generated by ML
- Fairness in Machine Learning
- Privacy in Machine Learning
  - How to use Differential Privacy to train models
  - Tradeoff between privacy and utility
- Federated learning
  - Training ML in a distributed fashion to protect user data
- Application-specific ML models: NLP (GPT-2, GPT-3, BERT)
- Unsupervised learning: embeddings, autoencoders, clustering, anomaly detection
- Reinforcement Learning
- Adversarial Machine Learning

Adversarial ML

- Attacks
  - Studies how can Machine Learning Fail
  - Different attack models
    - Attack objective and knowledge about the ML system
- Defenses
  - How to defend Machine Learning against different failures and improve their robustness
  - What are the tradeoffs between accuracy and robustness

Adversarial Machine Learning: Taxonomy

Attacker's Objective

Learning stage

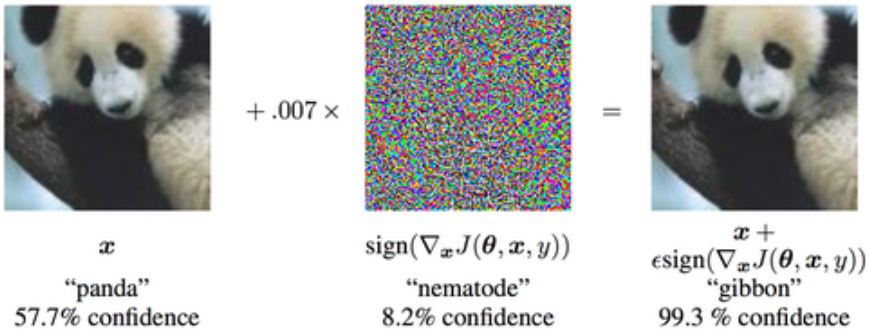
	Targeted Target small set of points	**Availability** Target majority of points	**Privacy** Learn sensitive information
Training	Targeted Poisoning Backdoor Trojan Attacks	Poisoning Availability Model Poisoning	-
Testing	Evasion Attacks Adversarial Examples	-	Membership Inference Model Extraction

Adversarial Machine Learning: Taxonomy

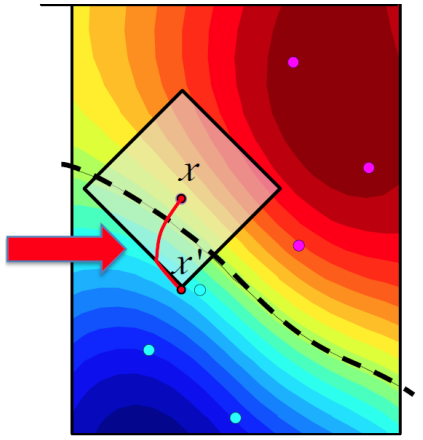
Attacker's Objective

Learning stage	**Targeted** Target small set of points	**Availability** Target majority of points	**Privacy** Learn sensitive information
Training	Targeted Poisoning Backdoor Trojan Attacks	Poisoning Availability Model Poisoning	-
Testing	Evasion Attacks Adversarial Examples	-	Membership Inference Model Extraction

Evasion Attacks



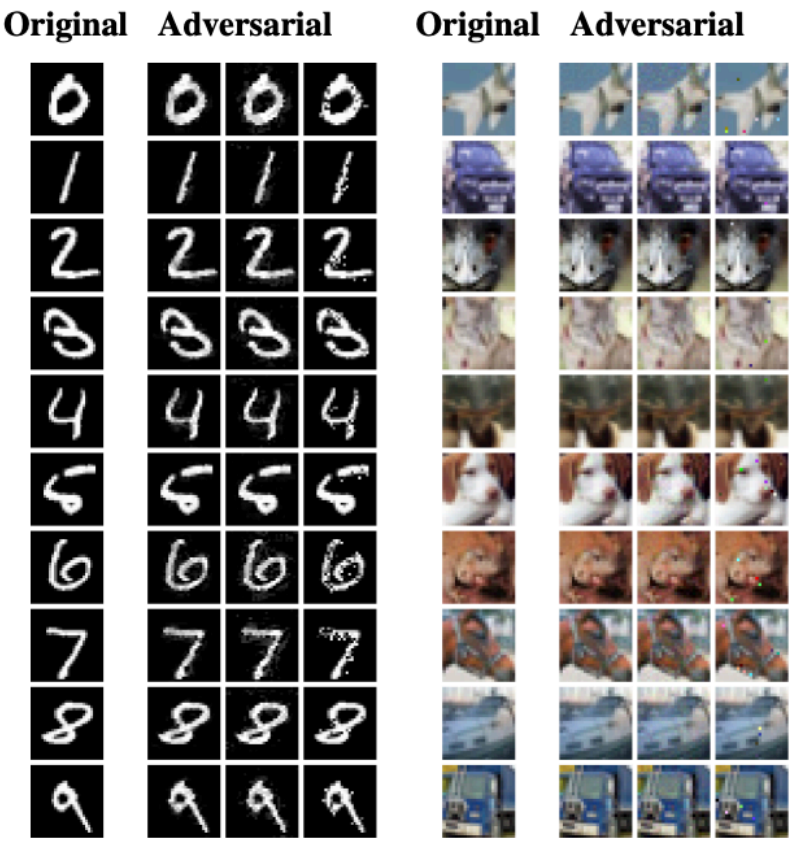
Adversarial example



- Evasion attack: attack against ML at testing time
- Implications
  - Small (imperceptible) modification at testing time can change the classification of any data point to any targeted class

- Szegedy et al. *Intriguing properties of neural networks*. 2014 https://arxiv.org/abs/1312.6199
- Goodfellow et al. *Explaining and Harnessing Adversarial Examples*. 2014. https://arxiv.org/abs/1412.6572

Adversarial Examples

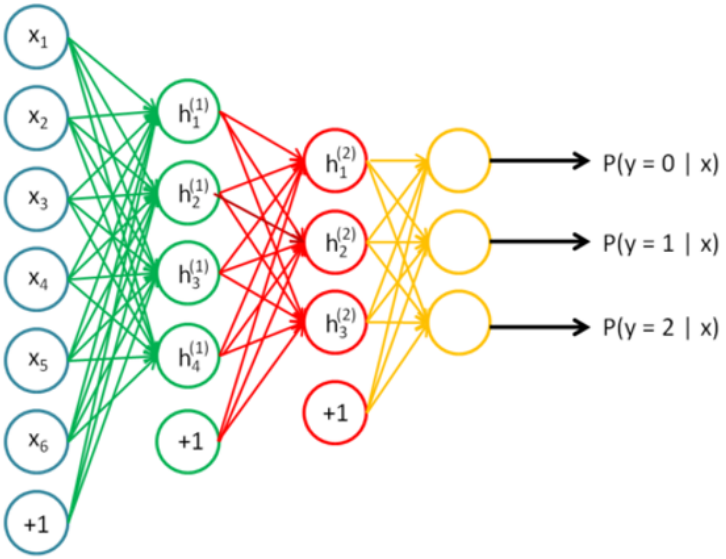


- N. Carlini and D. Wagner. *Towards Evaluating the Robustness of Neural Networks*. In IEEE Security and Privacy Symposium 2017

  https://arxiv.org/abs/1608.04644
- Goal: create minimum perturbations for adversarial examples
- They always exist!
- Application domains: image recognition, videos classification, text models, speech recognition

Evasion Attacks For Neural Networks

Input: Images represented as feature vectors



Optimization Formulation

Given input x

Find adversarial example

$$x' = x + \delta$$

$$\min_{\delta} c\left|\left|\delta\right|\right|_2^2 + L_t(x+\delta)$$

Min distance

Change class

[Carlini, Wagner 17]

- Most existing attacks are in continuous domains
- Images represented as matrix of pixels with continuous values
- How to solve optimization problem?

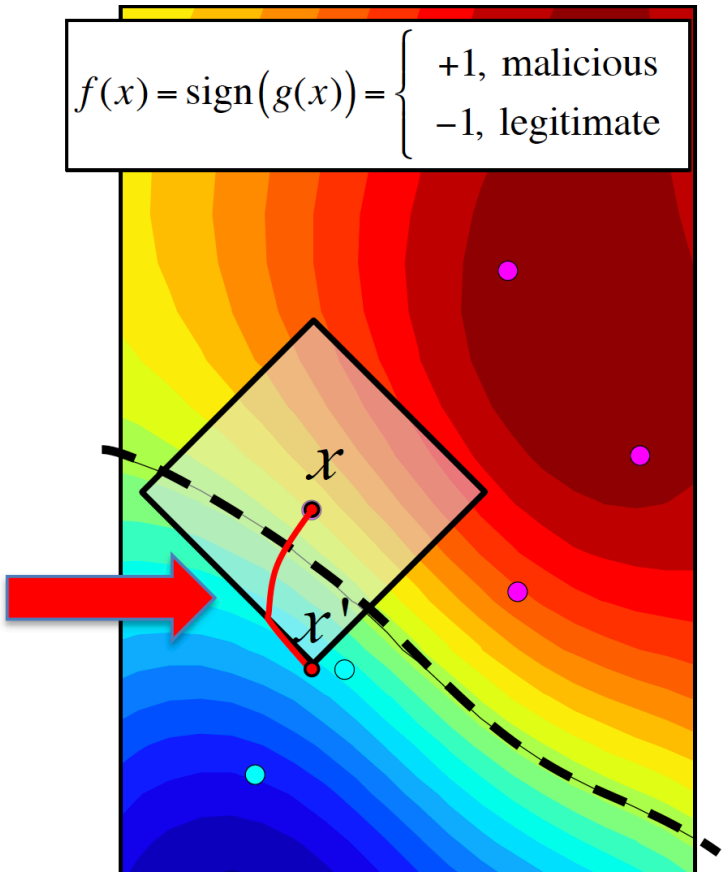
Projected Gradient Descent (PGD)

- **Goal:** maximum-confidence *evasion*
- **Knowledge:** *perfect (white-box attack)*
- **Attack strategy:**

$$\min_{x'} g(x')$$

$$\text{s.t. } \|x - x'\|_p \leq d_{\max}$$

- Non-linear, constrained optimization
  - **Projected gradient descent**: approximate solution for *smooth* functions
- Gradients of g(x) can be analytically computed in many cases
  - SVMs, Neural networks



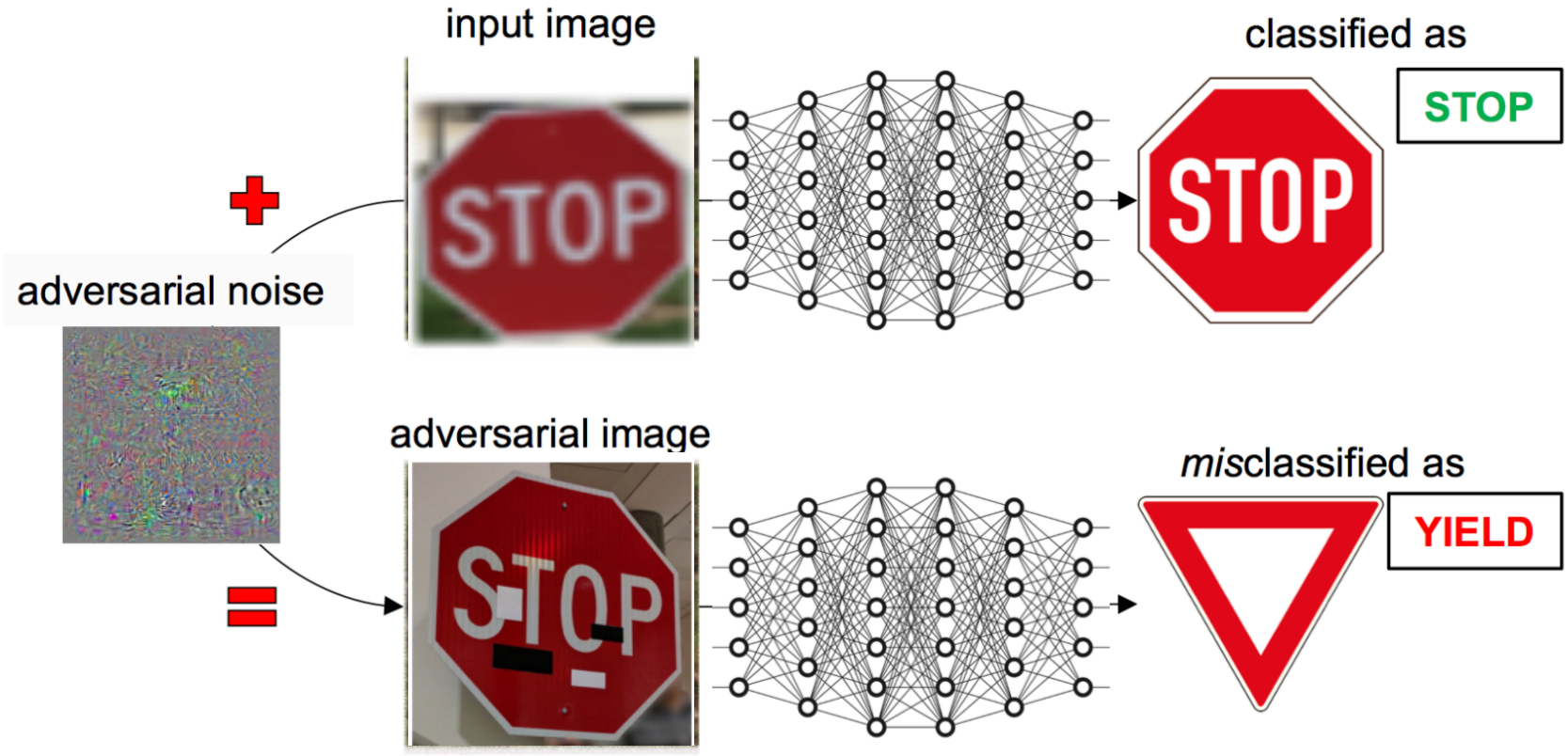
- In each iteration of gradient descent, perform a projection to feasible space
- Madry et al. *Towards Deep Learning Models Resistant to Adversarial Attacks*. 2018. https://arxiv.org/pdf/1706.06083.pdf

Feasible Adversarial Examples

Adversarial Glasses

- M. Sharif et al. (ACM CCS 2016) attacked deep neural networks for face recognition with carefully-fabricated eyeglass frames
- When worn by a 41-year-old white male (left image), the glasses mislead the deep network into believing that the face belongs to the famous actress Milla Jovovich

Adversarial Attacks on Road Signs



Eykholt et al. *Robust Physical-World Attacks on Deep Learning Visual Classification*. In CVPR 2018

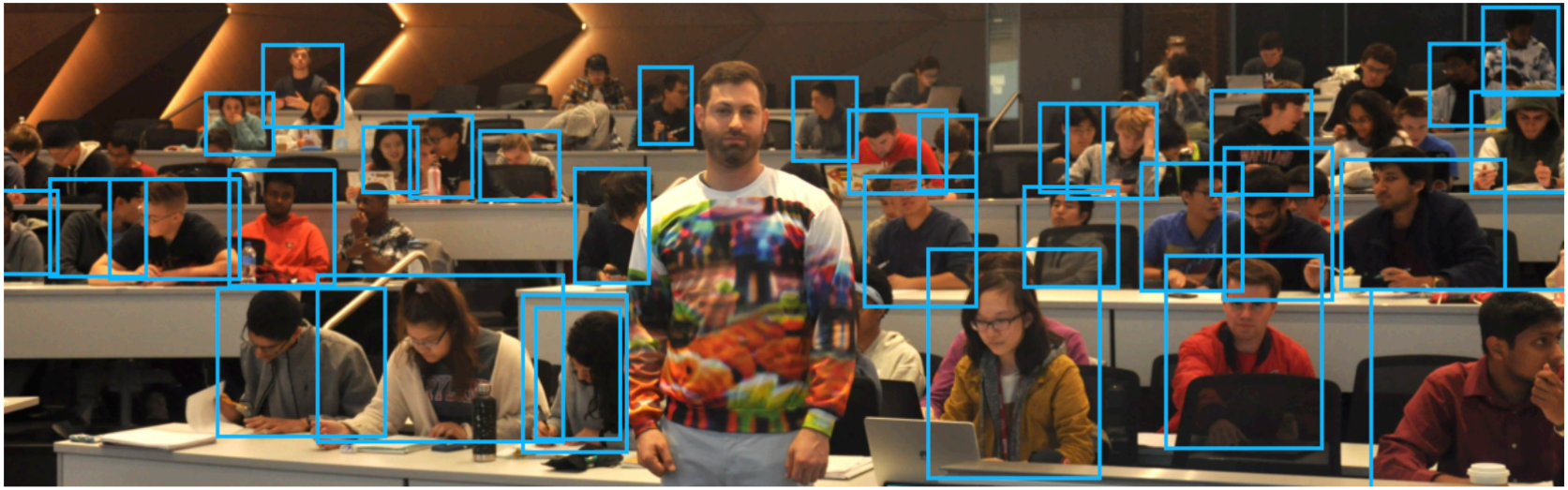
Speech Recognition

Audio Adversarial Examples

Audio	Transcription by Mozilla DeepSpeech
	“without the dataset the article is useless”
	“okay google browse to evil dot com”

https://nicholas.carlini.com/code/audio_adversarial_examples/

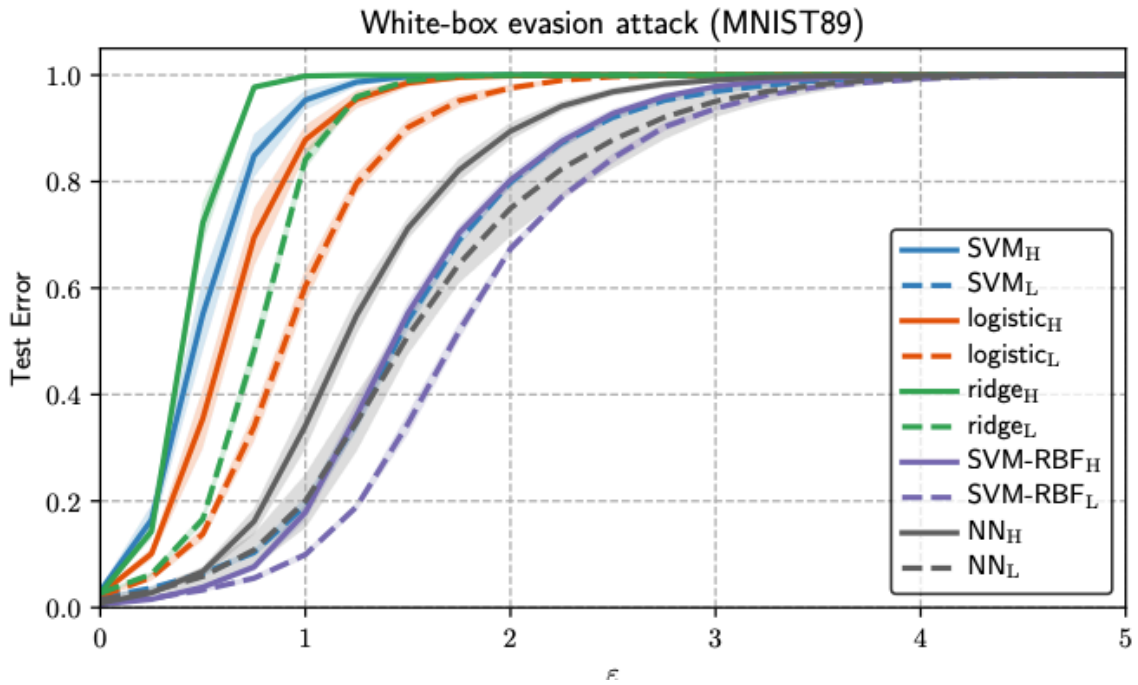
Attacking Object Detectors



This stylish pullover is a great way to stay warm this winter, whether in the office or on-the-go. It features a stay-dry microfleece lining, a modern fit, and adversarial patterns the evade most common object detectors. In this demonstration, the YOLOv2 detector is evaded using a pattern trained on the COCO dataset with a carefully constructed objective.

https://www.cs.umd.edu/~tomg/projects/invisible/

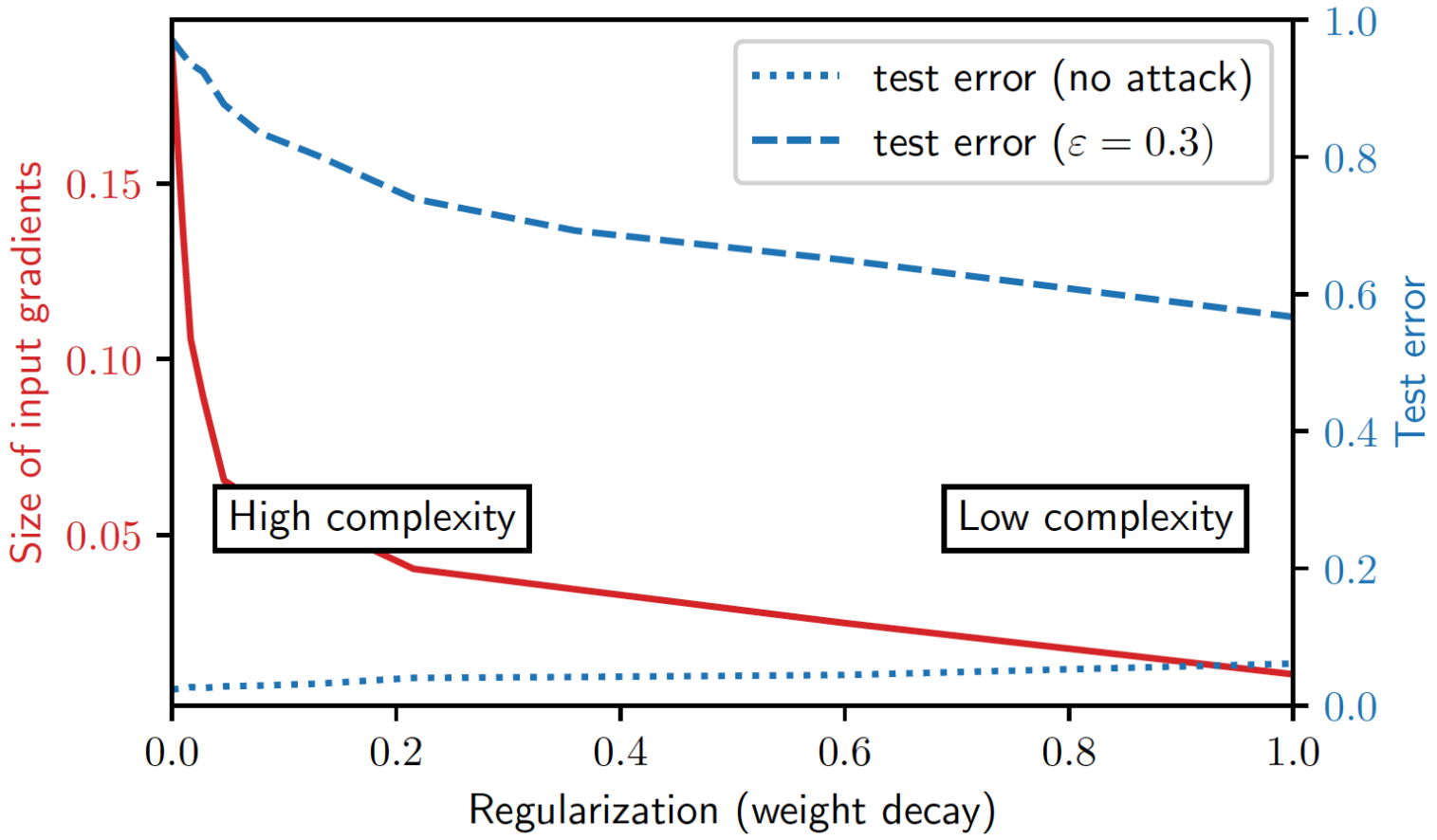
Multiple Classifiers Fail under Evasion



- Classifier test error as a function of perturbation budget on MNIST dataset
- Linear classifiers: SVM, logistic regression, ridge
- Non-linear classifiers: SVM-RBF, Feed-forward neural network

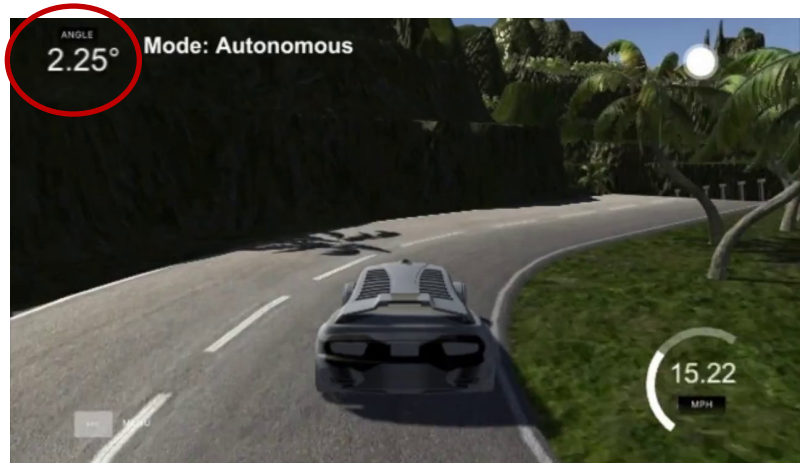
A. Demontis, M. Melis, M. Pintor, M. Jagielski, B. Biggio, A. Oprea, C. Nita-Rotaru, F. Roli. *Why Do Adversarial Attacks Transfer? Explaining Transferability of Evasion and Poisoning Attacks*. USENIX Security, 2019

Impact of Regularization



Evasion Attacks in Connected Cars

- Udacity challenge: Predict steering angle from camera images, 2014
- Actions
  - Turn left (negative steering angle)
  - Turn right (positive steering angle)
  - Straight (steering angle in [-T,T])
- Dataset has 33,608 images and steering angle values (70GB of data)



Predict direction: Straight, Left, Right

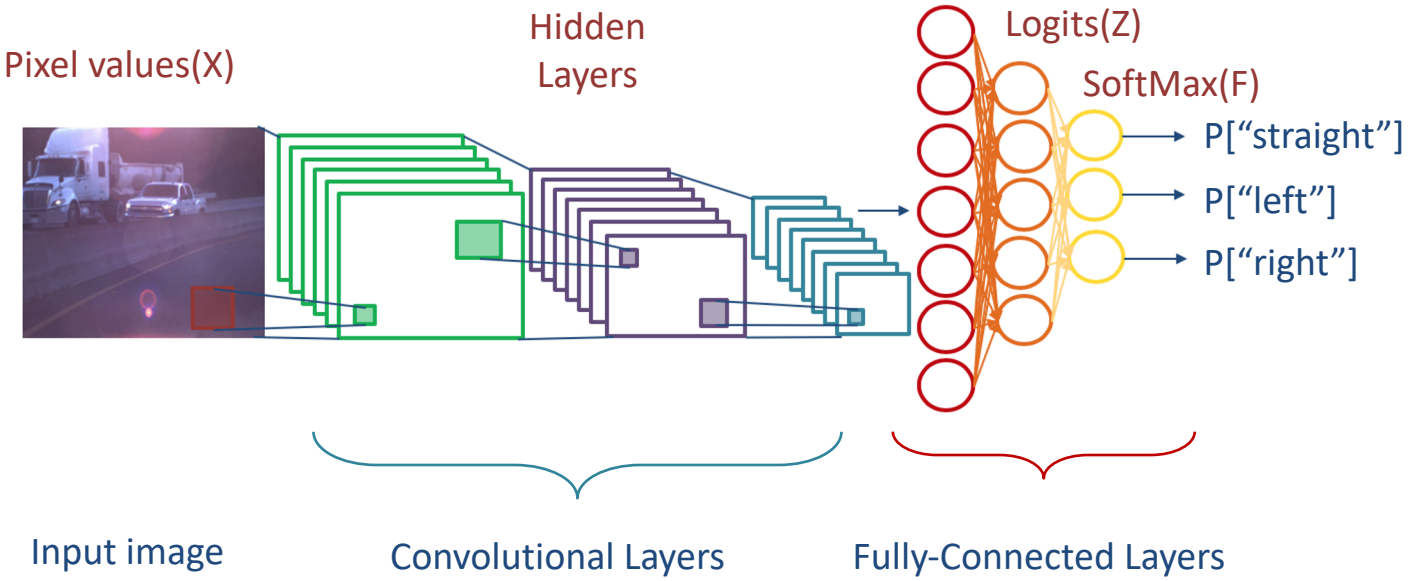
Predict steering angle

A. Chernikova, A. Oprea, C. Nita-Rotaru, and B. Kim.

Are Self-Driving Cars Secure? Evasion Attacks against Deep Neural Networks for Self-Driving Cars.

In IEEE SafeThings 2019. https://arxiv.org/abs/1904.07370

CNN for Direction Prediction

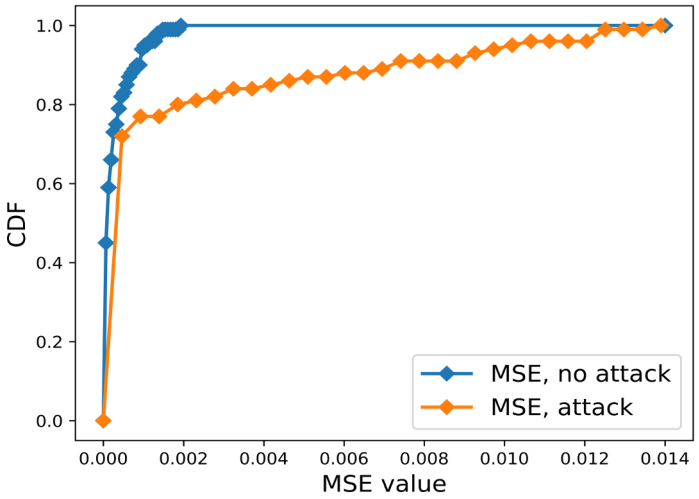


- Two CNN architectures: 25 million and 467 million parameters
- For Regression, exclude the last softmax layer
- Architectures used in the Udacity challenge

Evasion Attack against Regression

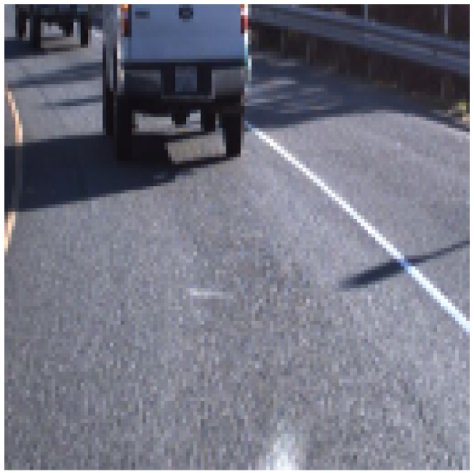
- First evasion attack for CNNs for regression
- New objective function
  - Minimize adversarial perturbation
  - Maximize the square residuals (difference between the predicted and true response)

$$\min_{\delta} c||\delta||_2^2 - G(x+\delta, y)$$
such that $x + \delta \in [0,1]^d$
$$G(x+\delta, y) = [F(x+\delta) - y]^2$$

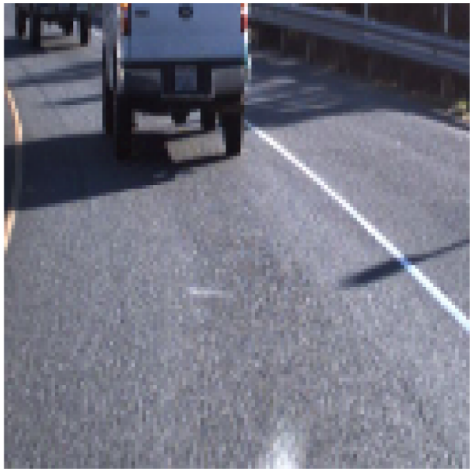


- 10% of adversarial images have 20 times higher MSE
- The maximum ratio of adversarial to legitimate MSE reaches 69

Adversarial Example for Regression



Original Image
Steering angle = -4.25; MSE = 0.0016

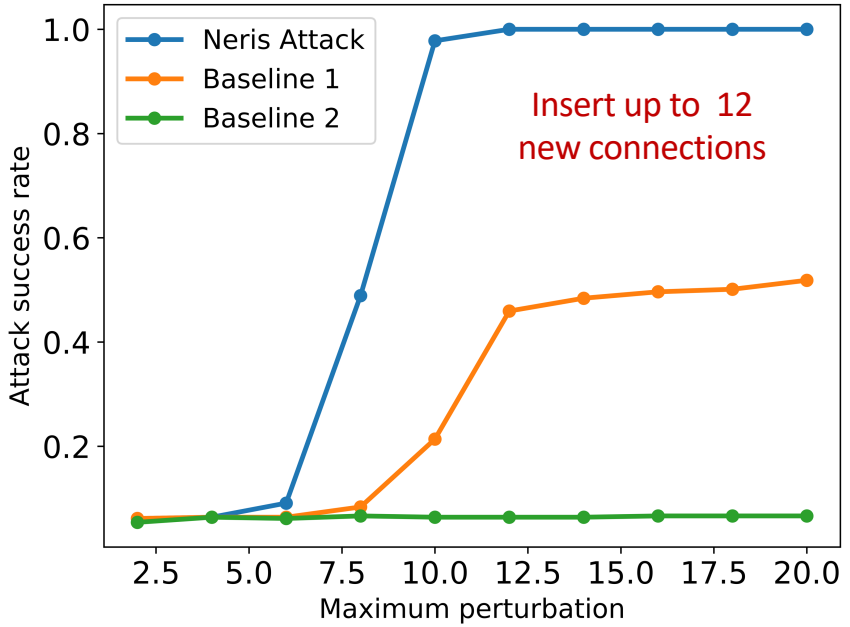


Adversarial Image
Steering angle = -2.25; MSE = 0.05

- Significant degradation of CNN classifiers in connected cars
- Small amount of perturbation is effective
- Models for both classification and regression are vulnerable

How Effective are Evasion Attacks in Security?

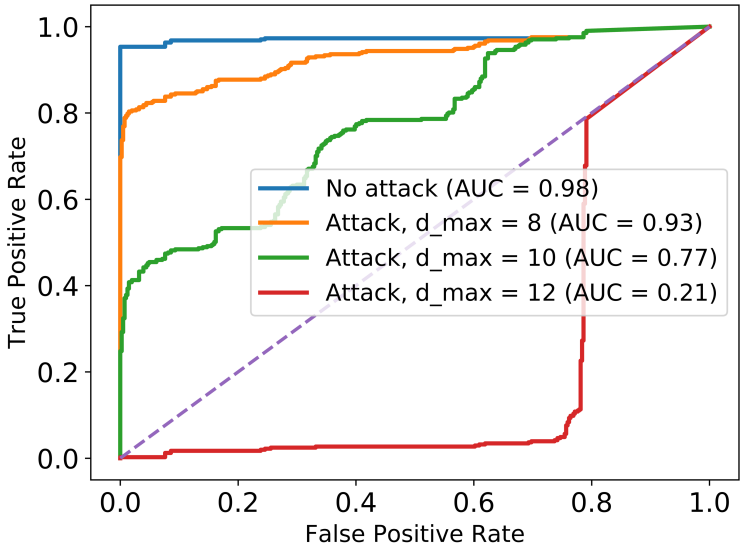
- Dataset: CTU-13, Neris botnet, highly imbalanced
  - 194K benign
  - 3869 malicious
- Features: 756 on 17 ports
- Model: Feed-forward neural network (3 layers), F1: 0.96



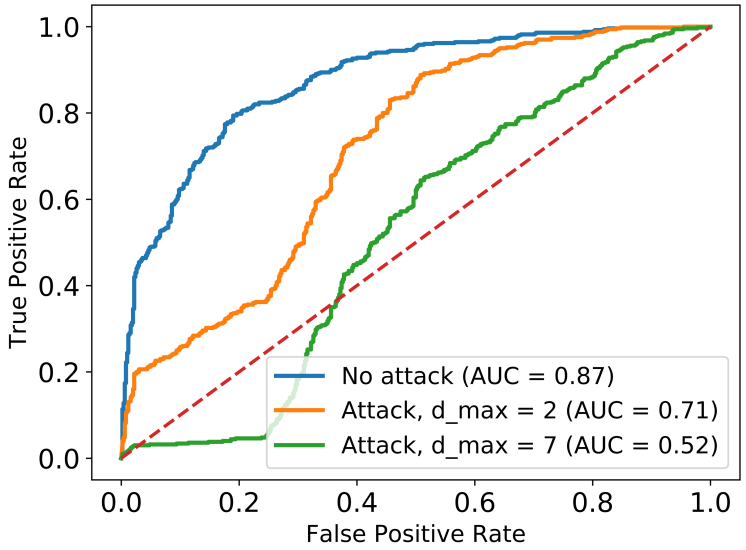
A. Chernikova and A. Oprea. *FENCE: Feasible Evasion Attacks on Neural Networks in Constrained Environments*

http://arxiv.org/abs/1909.10480, 2019.

How Effective are Evasion Attacks in Security?



Malicious connection classifier

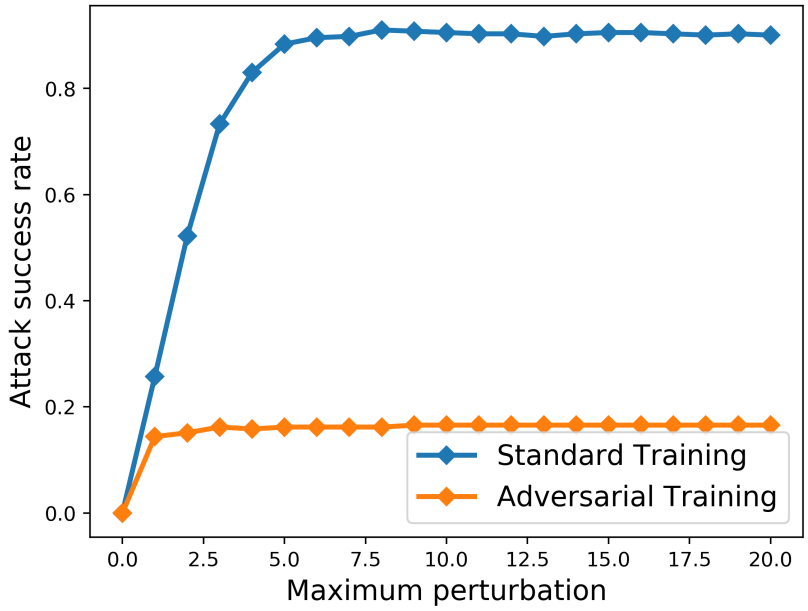


Malicious domain classifier

- Significant degradation of ML classifiers in security
- Small amount of perturbation is effective
- General framework for adversarial testing in discrete domains

Defense: Adversarial Training

- Adversarial Training
  - Train model iteratively
  - In each iteration, generate adversarial examples and add to training with correct label
- Implications
  - Adversarial training can improve ML robustness
- Challenges
  - Computationally expensive
  - Specific to certain attacks
  - Does it generalize to other attacks?



Malicious domain classifier

Taxonomy

Attacker's Objective

Learning stage

	Targeted Target small set of points	**Availability** Target majority of points	**Privacy** Learn sensitive information
Training	Targeted Poisoning Backdoor Trojan Attacks	Poisoning Availability	Membership Inference
Testing	Evasion Attacks Adversarial Examples	-	Membership Inference Model Extraction

Training-Time Attacks

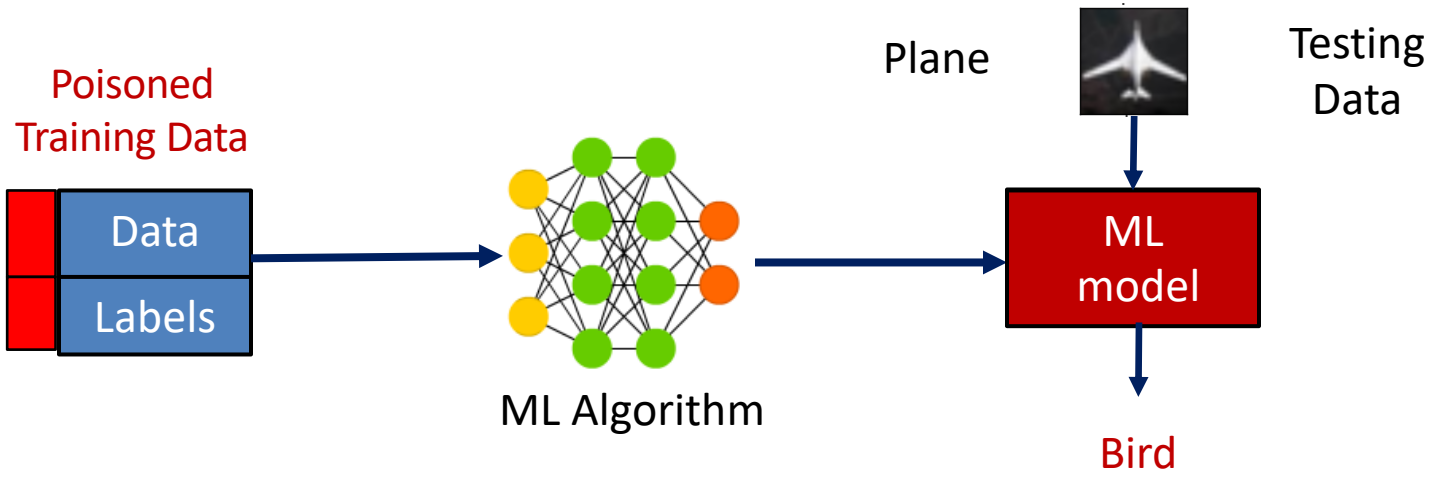
- ML is trained by crowdsourcing data in many applications

- Social networks
- News articles
- Tweets

- Navigation systems
- Face recognition
- Mobile sensors

- Cannot fully trust training data!

Poisoning Availability Attacks



- Attacker Objective:
  - Corrupt the predictions by the ML model significantly
- Attacker Capability:
  - Insert fraction of poisoning points in training
  - Find the points that cause the maximum impact

M. Jagielski, A. Oprea, B. Biggio, C. Liu, C. Nita-Rotaru, and B. Li. *Manipulating Machine Learning: Poisoning Attacks and Countermeasures for Regression Learning*. In IEEE S&P 2018

Optimization Formulation

Given a training set D find a set of poisoning data points D_p

that maximizes the adversary objective A on validation set D_{val}

where corrupted model $\boldsymbol{\theta}_p$ is learned by minimizing the loss L on $D \cup D_p$

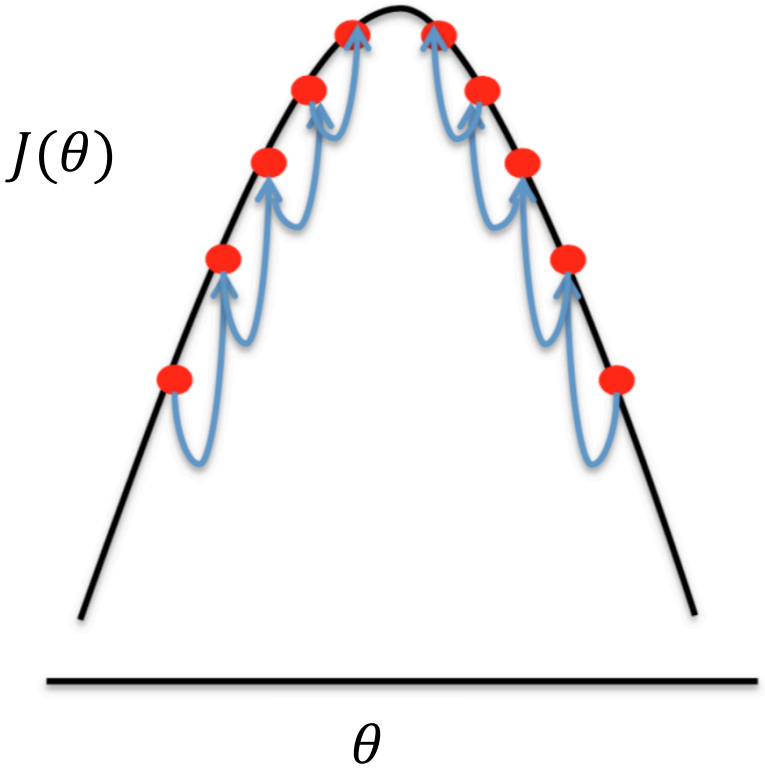
$$\underset{D_p}{\operatorname{argmax}}\, A(D_{val}, \boldsymbol{\theta}_p)\; s.t.$$
$$\boldsymbol{\theta}_p \in \underset{\boldsymbol{\theta}}{\operatorname{argmin}}\, L(D \cup D_p, \boldsymbol{\theta})$$

Bilevel Optimization
NP-Hard!

First white-box attack for linear regression [Jagielski et al. 18]

- Determine optimal poisoning point $(\boldsymbol{x}_c, y_c)$
- Optimize by both $\boldsymbol{x}_c$ and y_c
- How to optimize this?

Gradient Ascent



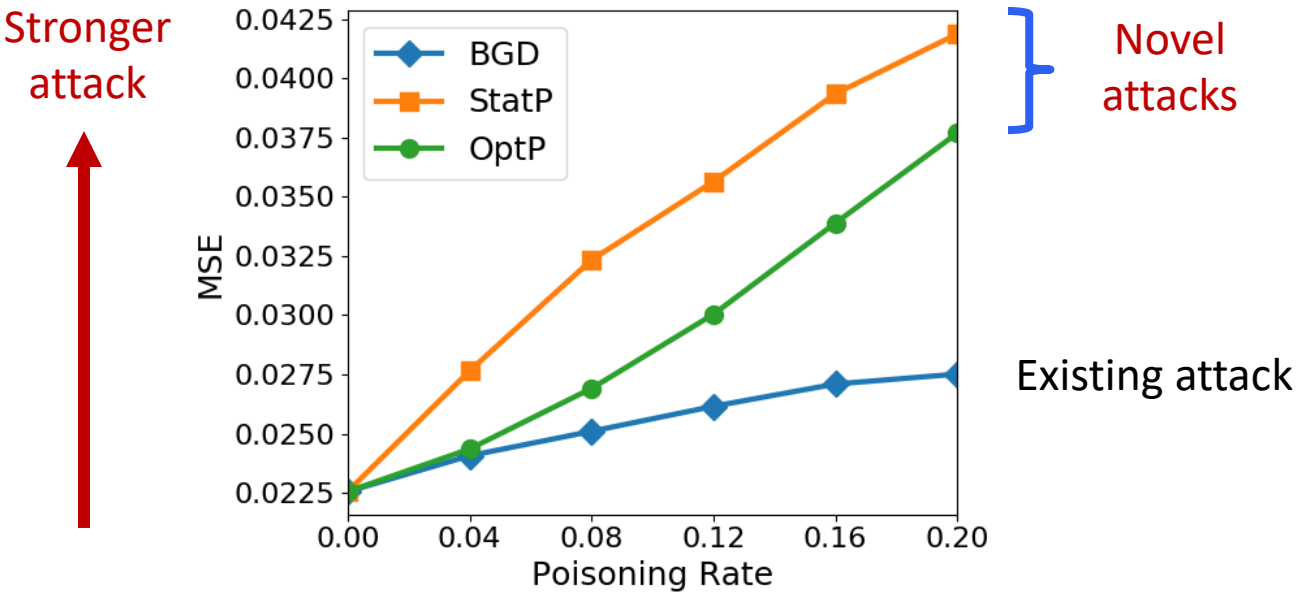
$\max J(\theta)$

Same as Gradient Descent, but with update rule:

$$\theta \leftarrow \theta + \frac{\partial J(\theta)}{\partial \theta}$$

Poisoning Regression

- Improve existing attacks by a factor of at most 6.83



Predict loan rate with ridge regression
(L2 regularization)

Is It Really a Threat?

- Case study on healthcare dataset (predict Warfarin medicine dosage)
- At 20% poisoning rate
  - Modifies 75% of patients' dosages by 93.49% for LASSO
  - Modifies 10% of patients' dosages by a factor of 4.59 for Ridge
- At 8% poisoning rate
  - Modifies 50% of the patients' dosages by 75.06%

Quantile	Initial Dosage	Ridge Difference	LASSO Difference
0.1	15.5 mg/wk	31.54%	37.20%
0.25	21 mg/wk	87.50%	93.49%
0.5	30 mg/wk	150.99%	139.31%
0.75	41.53 mg/wk	274.18%	224.08%
0.9	52.5 mg/wk	459.63%	358.89%

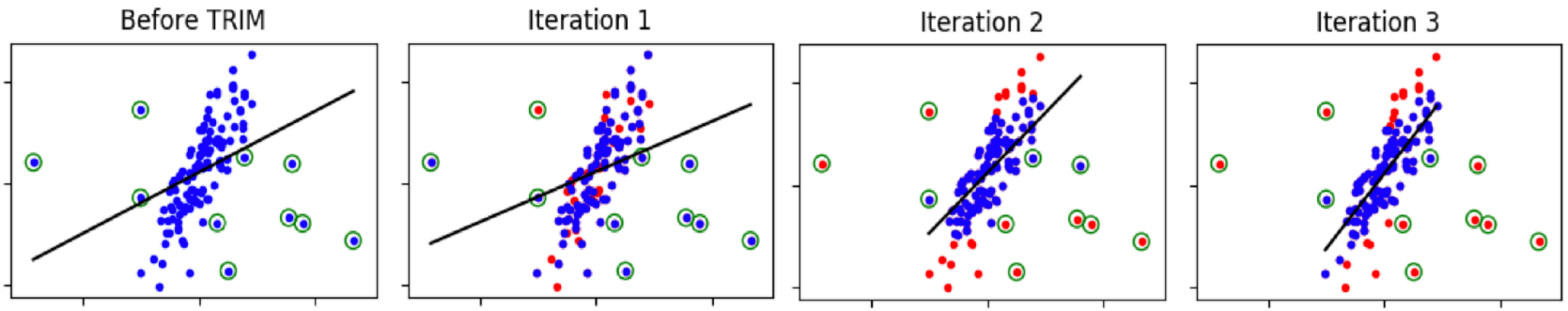
Defenses via Robust Optimization

Robust Regression with TRIM

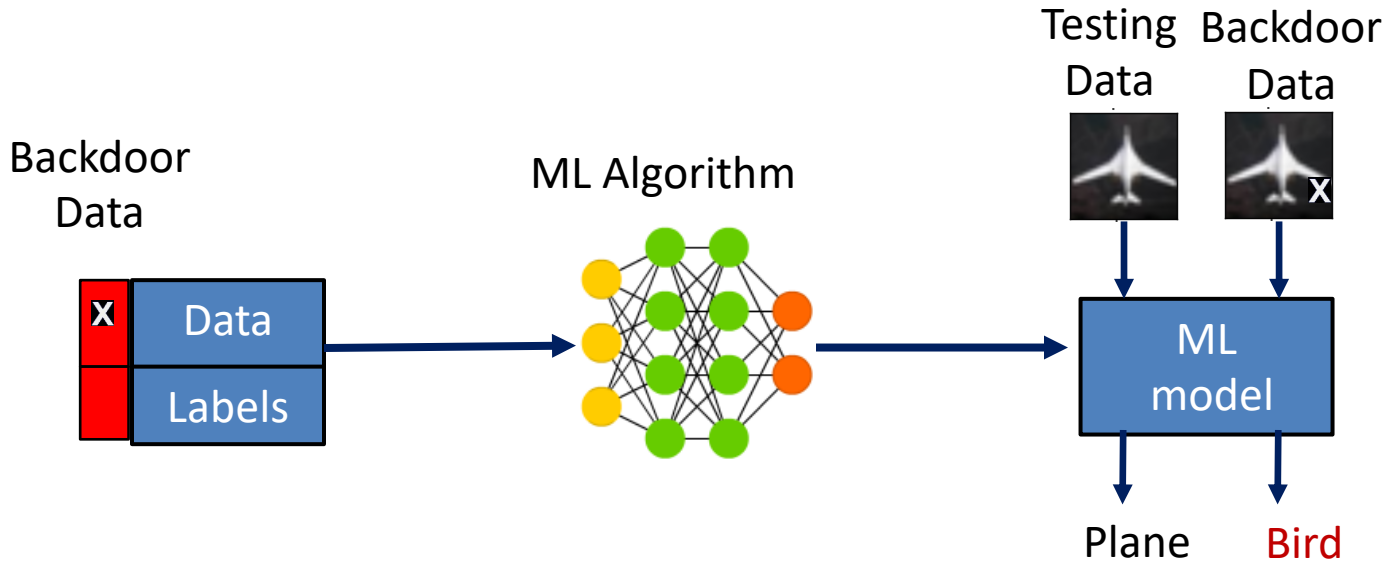
- TRIM learns the model by retaining only training points with the smallest residuals

$$\underset{w,b,I}{\operatorname{argmin}} L(w,b,I) = \frac{1}{|I|}\sum_{i\in I}(f(\boldsymbol{x}_i) - y_i)^2 + \lambda\Omega(\boldsymbol{w})$$

$$N = (1+\alpha)n, \qquad I \subset [1,\ldots,N], \qquad |I| = n$$



Backdoor Poisoning Attacks



- Attacker Objective:
  - Prediction on clean data is unchanged
  - Change prediction of *backdoor data* in testing
- Attacker Capability:
  - Add backdoored poisoning points in training
  - Add backdoor pattern in testing
- [Gu et al. 17], [Chen et al. 17], [Turner et al. 18], [Shafahi et al. 18]

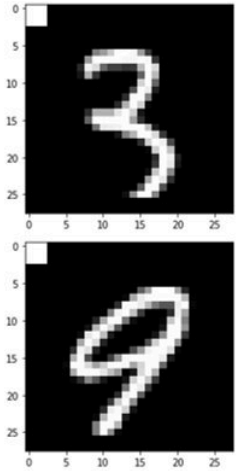
BadNets

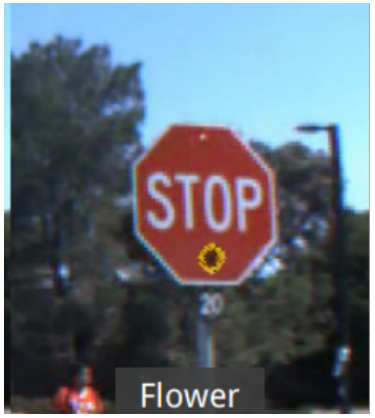
Gu et al. *BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain*. 2017. https://arxiv.org/abs/1708.06733

Backdoor Attacks on Feature-Based Models

Computer vision

- A fixed pixel pattern.





Feature space

- Fixed assignment of numerical values to features.

Feature	LightGBM	EmberNN
major_image_version	1704	14
major_linker_version	15	13
major_operating_system_version	38078	8
minor_image_version	1506	12
minor_linker_version	15	6
minor_operating_system_version	5	4
minor_subsystem_version	5	20

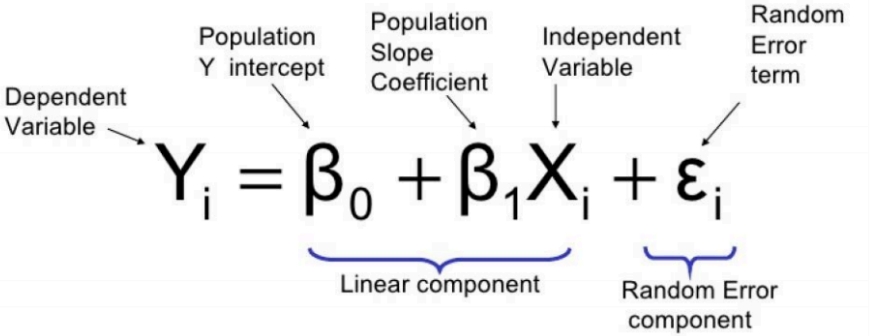
- Identify most relevant features that point to target class
- Equivalent to variable importance, but model-agnostic
- Use techniques from ML explainability to identify relevant features
- G. Severi, J. Meyer, S. Coull, A. Oprea. *Exploring Backdoor Poisoning Attacks Against Malware Classifiers*. 2020. https://arxiv.org/abs/2003.01031

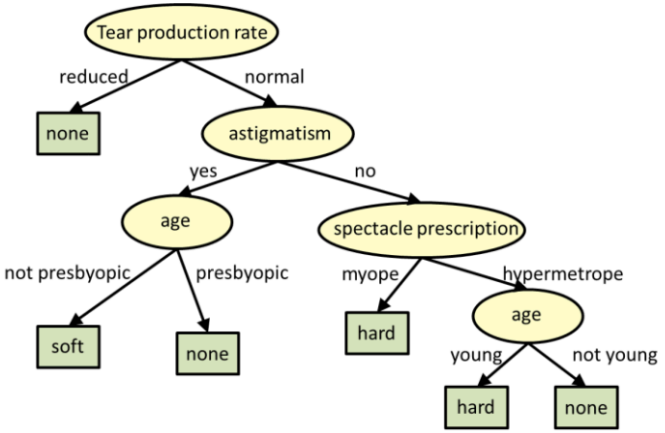
ML Interpretability

Goals

- Explain why models makes a prediction
- Which features and values contribute to the prediction
- Which features are most important

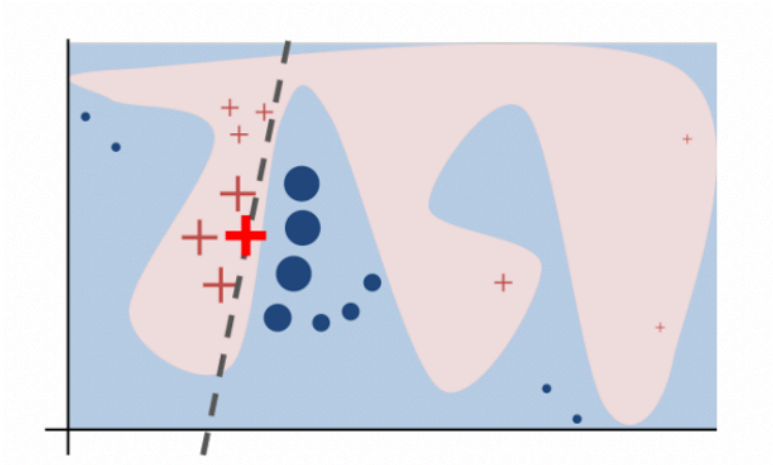
- In pre-deep learning models, some models are considered "interpretable"





Interpretability for Neural Networks

- Hard to explain a complex model in its entirety
  - How about explaining smaller regions?



LIME (Ribeiro et. al.)

  - Explains decisions of any model in a local region around a particular point

  - Learns sparse linear model

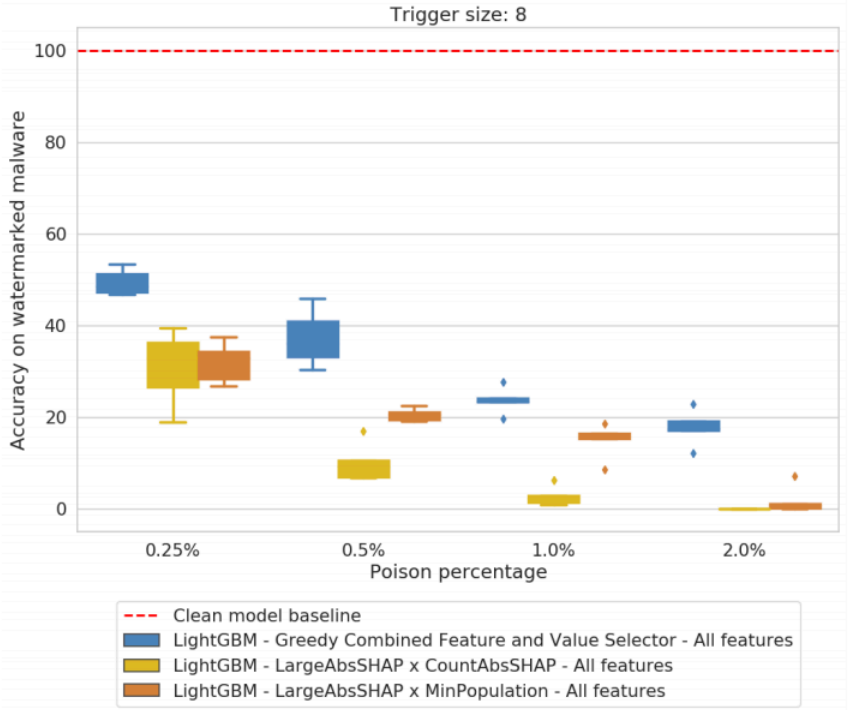
Example LIME

(a) Original Image (b) Explaining *Electric guitar* (c) Explaining *Acoustic guitar* (d) Explaining *Labrador*

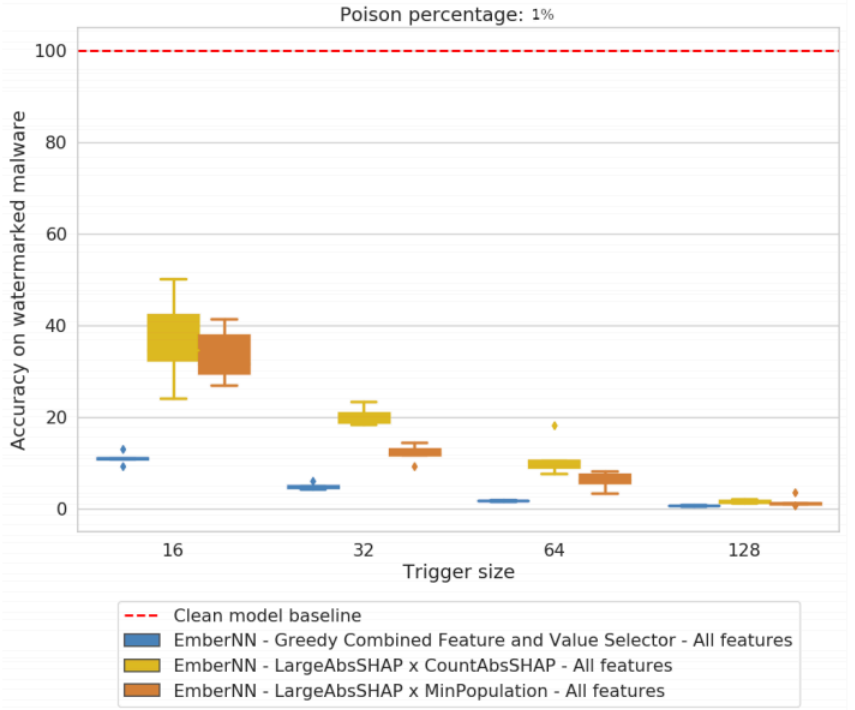
Figure 4: Explaining an image classification prediction made by Google's Inception network, highlighting positive pixels. The top 3 classes predicted are "Electric Guitar" ($p = 0.32$)**, "Acoustic guitar"** ($p = 0.24$) **and "Labrador"** ($p = 0.21$)

- LIME: Local Interpretable Model-Agnostic Explanations.
  - Ribiero et al. *"Why Should I Trust You?" Explaining the Predictions of Any Classifier*. 2016
- SHAP values: Integrates LIME and other interpretability methods
  - Lundberg and Lee. *A Unified Approach to Interpreting Model Predictions*. NeurIPS 2017.
  - Provides model-agnostic feature importance

Attack Effectiveness



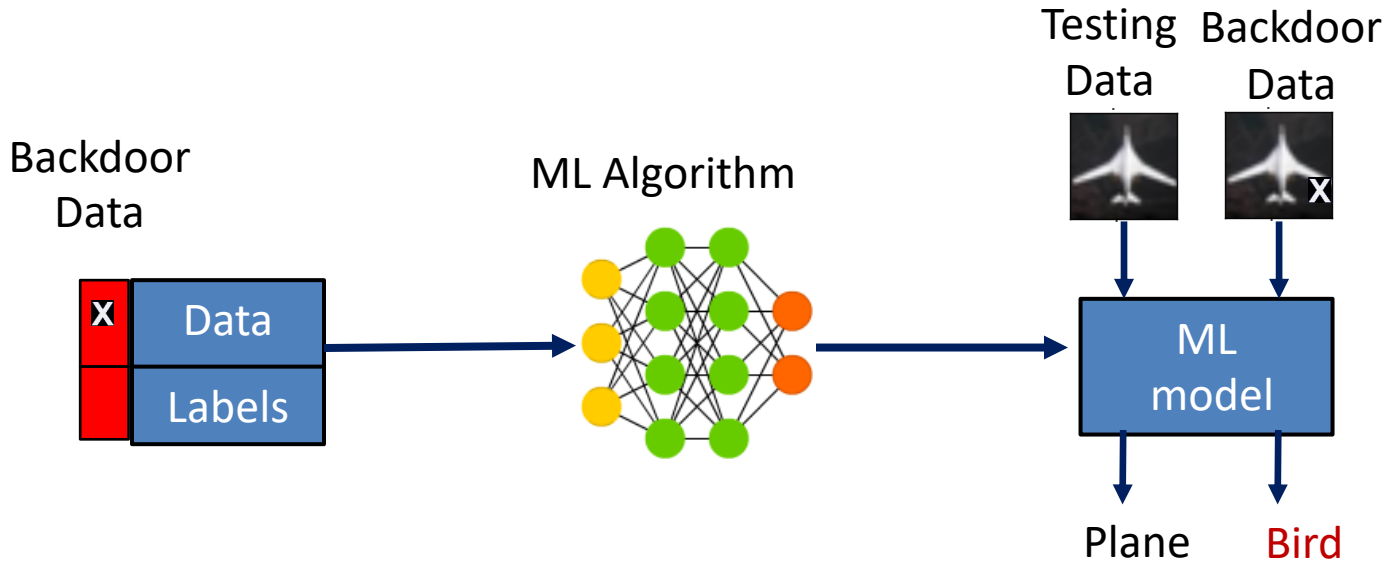
(a) LightGBM target



(b) EmberNN target

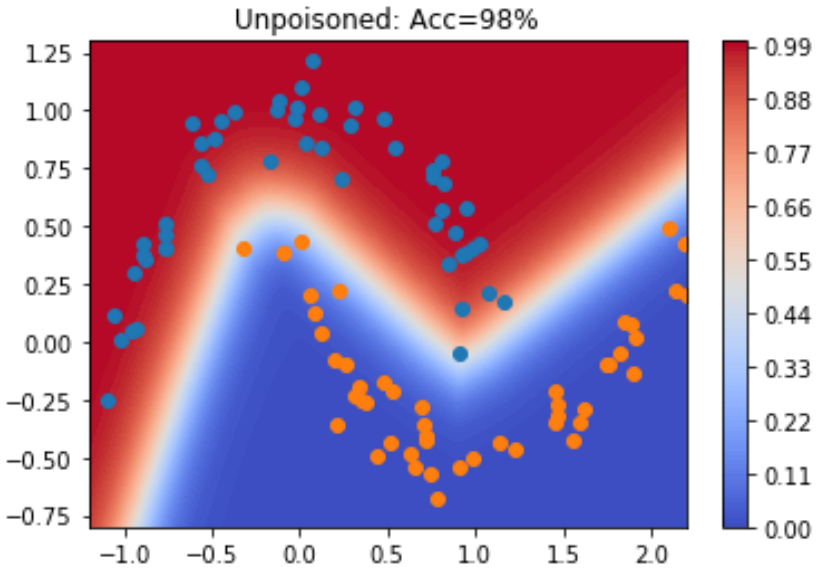
- Malware classifiers: Windows, Android, PDF files
- A small percentage of backdoor data and a small number of features in the trigger are sufficient for attack

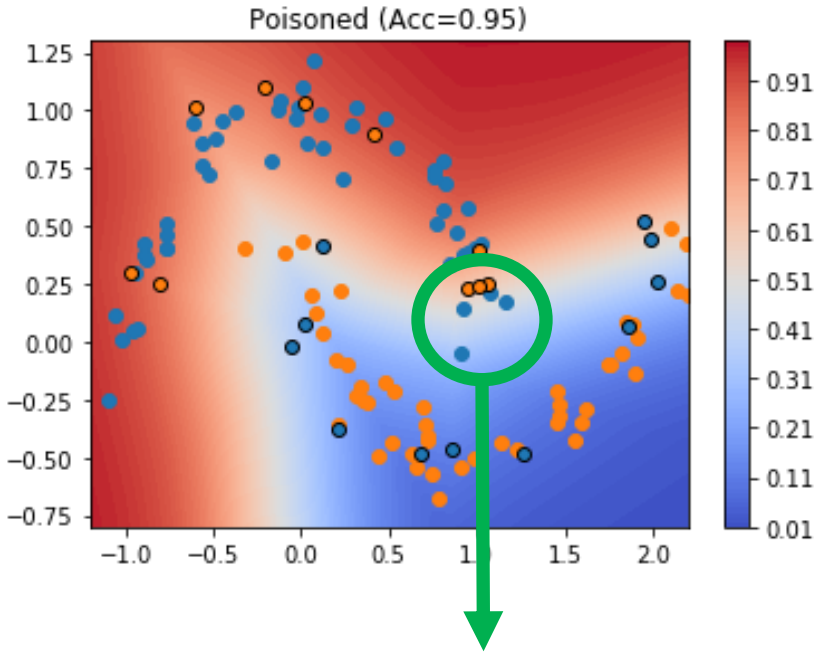
Backdoor Poisoning Attacks



- Attacker Objective:
  - Prediction on clean data is unchanged
  - Change prediction of *backdoor data* in testing
- Attacker Capability:
  - Add backdoored poisoning points in training
  - Add backdoor pattern in testing
- [Gu et al. 17], [Chen et al. 17], [Turner et al. 18], [Shafahi et al. 18]
- Strong assumption: Attacker controls both training and testing phases

New Attack: Subpopulation Poisoning





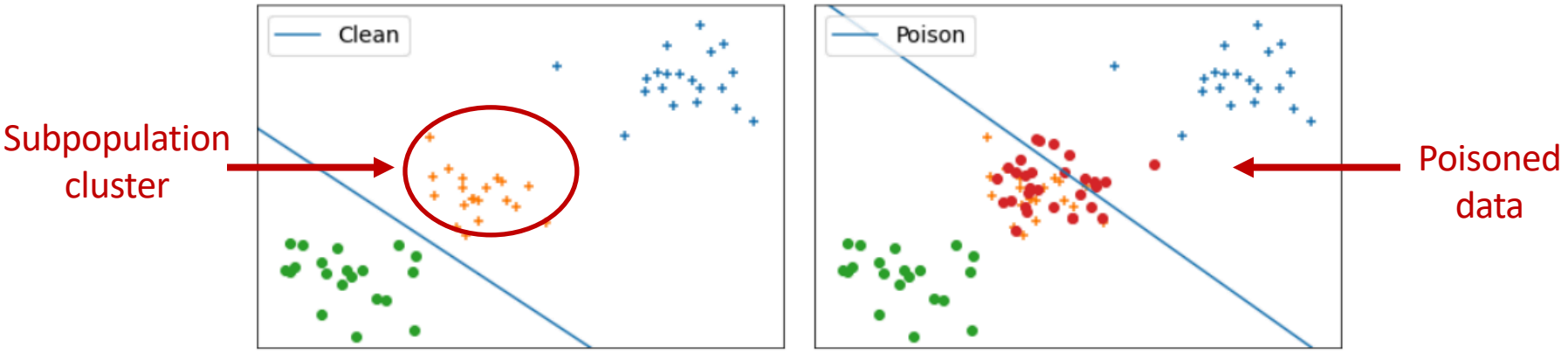
Key Insights

- Data has natural clusters (subpopulations)
- Some subpopulations are more vulnerable
- Minority populations are affected more!

Attack can be mounted stealthily!

Subpopulation Poisoning Attack

- Subpopulations can be attacked independently of each other
- Identify best subpopulations to attack
  - Via feature matching or clustering
- Add points from the subpopulation with target label and perform optimization



How Effective are Subpopulation Attacks?

Two metrics

- Accuracy on target subpopulations
- Collateral: damage on remaining subpopulations in the data
- Vary size of poisoning set

Dataset	Original Accuracy	Poisoned Accuracy Worst 5 Populations	Mean Collateral	Attack Size
CIFAR-10 + VGG	86.3%	36.3%	1.3%	181
UCI Adult	83.7%	62.8%	1.4%	45
IMDB + BERT	91.3%	66.1%	0.05%	160
UTKFace + VGG	96.3%	48.5%	2.9%	95

- Evaluated end-to-end and transfer learning models

Defending against Poisoning Attacks

- New subpopulation poisoning attack
  - Attack is stealthy (difficult to detect)
  - Insert a small number of poisoned points in training
  - Does not require change of testing data
- Research questions
  - Which subpopulations are more vulnerable?
  - Are defenses possible? We show some impossibility results!
  - How can we train end-to-end robust ML?

M. Jagielski, G. Severi, N. Pousette-Harger, A. Oprea. *Subpopulation Data Poisoning Attacks*. 2020. https://arxiv.org/abs/2006.14026

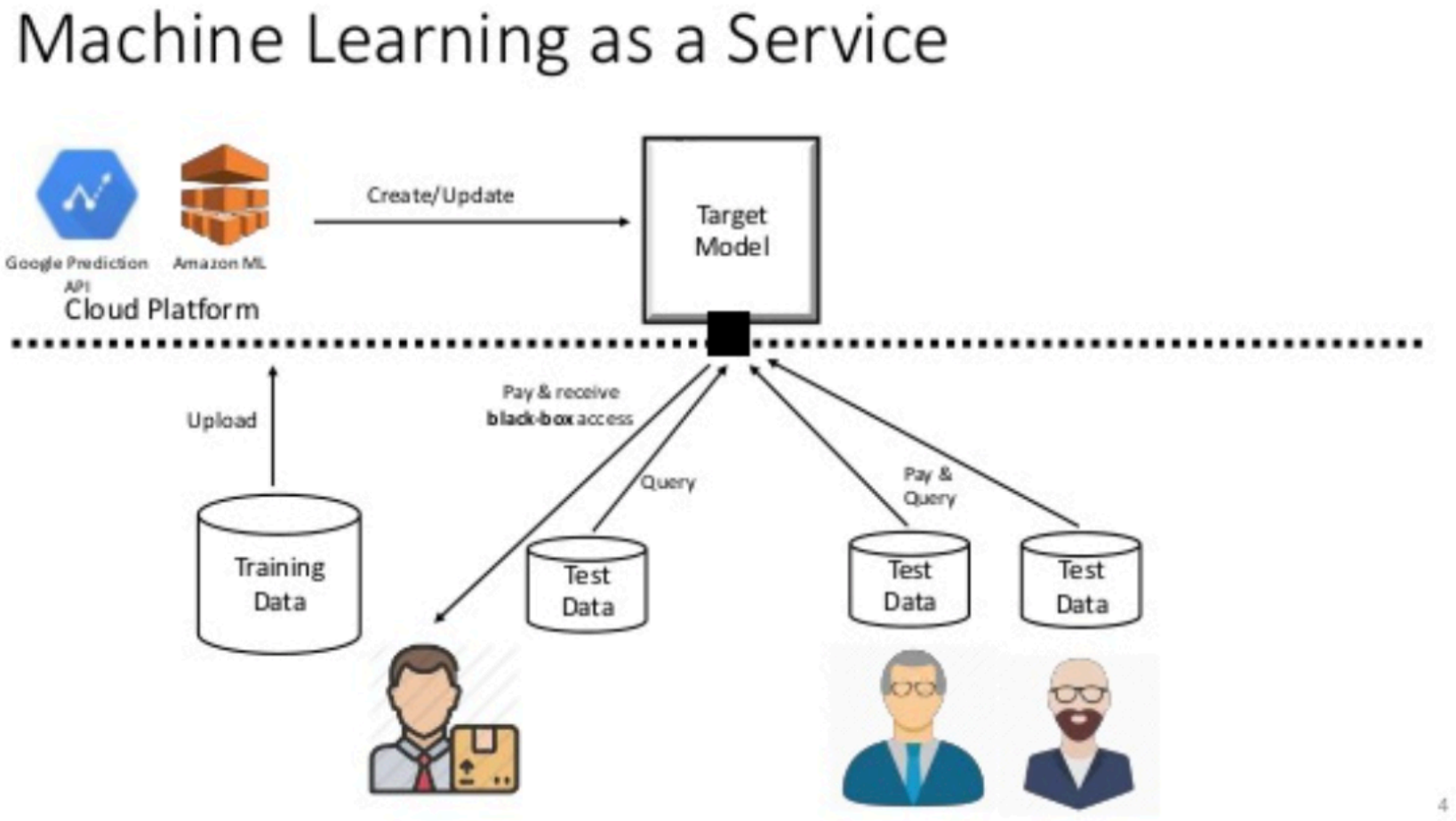
Adversarial Machine Learning: Taxonomy

Attacker's Objective

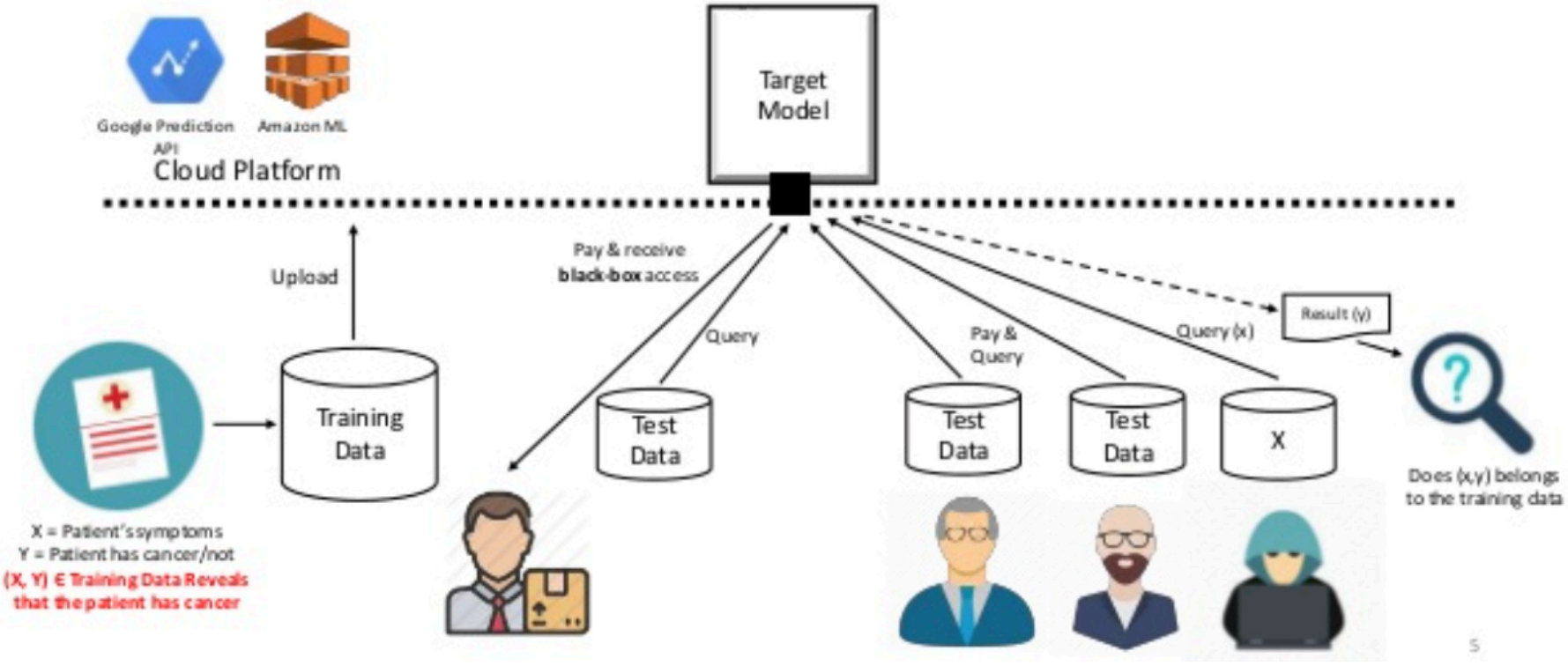
Learning stage

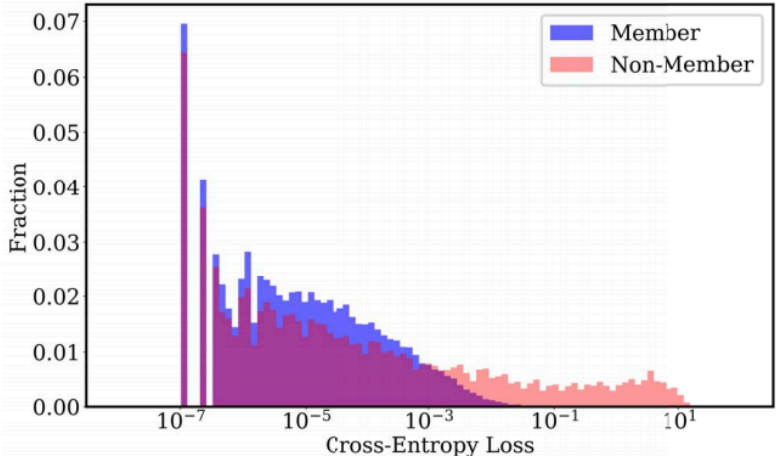
	Targeted Target small set of points	**Availability** Target majority of points	**Privacy** Learn sensitive information
Training	Targeted Poisoning Backdoor Trojan Attacks	Poisoning Availability Model Poisoning	-
Testing	Evasion Attacks Adversarial Examples	-	Membership Inference Model Extraction

Privacy Attacks against ML



Membership Inference Attack





- There is difference in the loss between member and non-member
- Due to over-fitting of ML to some extent

Model Extraction

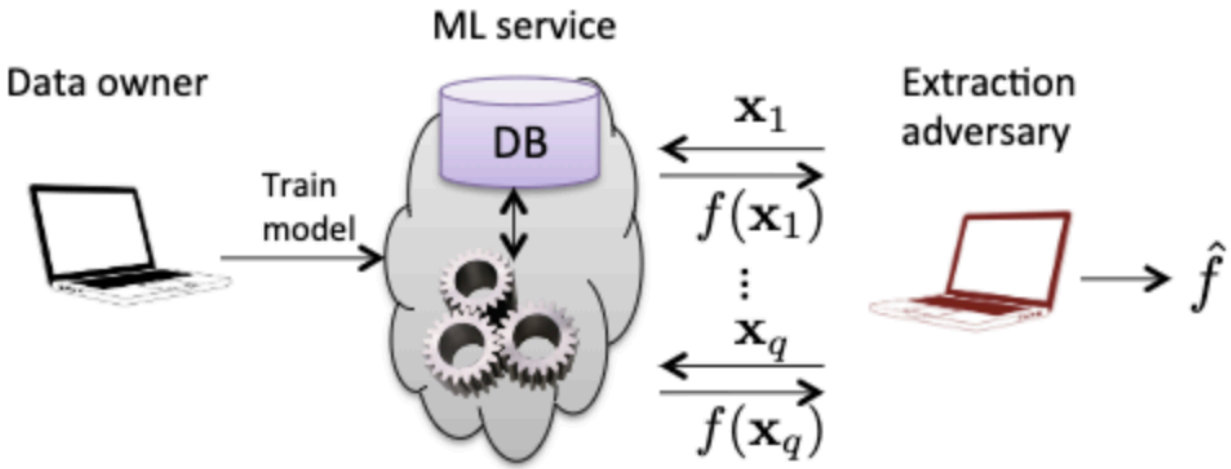


Figure 1: **Diagram of ML model extraction attacks.** A data owner has a model f trained on its data and allows others to make prediction queries. An adversary uses q prediction queries to extract an $\hat{f} \approx f$.

Defense: Differential Privacy

The output distribution of a differentially private algorithm changes very little whether or not any individual's data is included in the input – so you should contribute your data

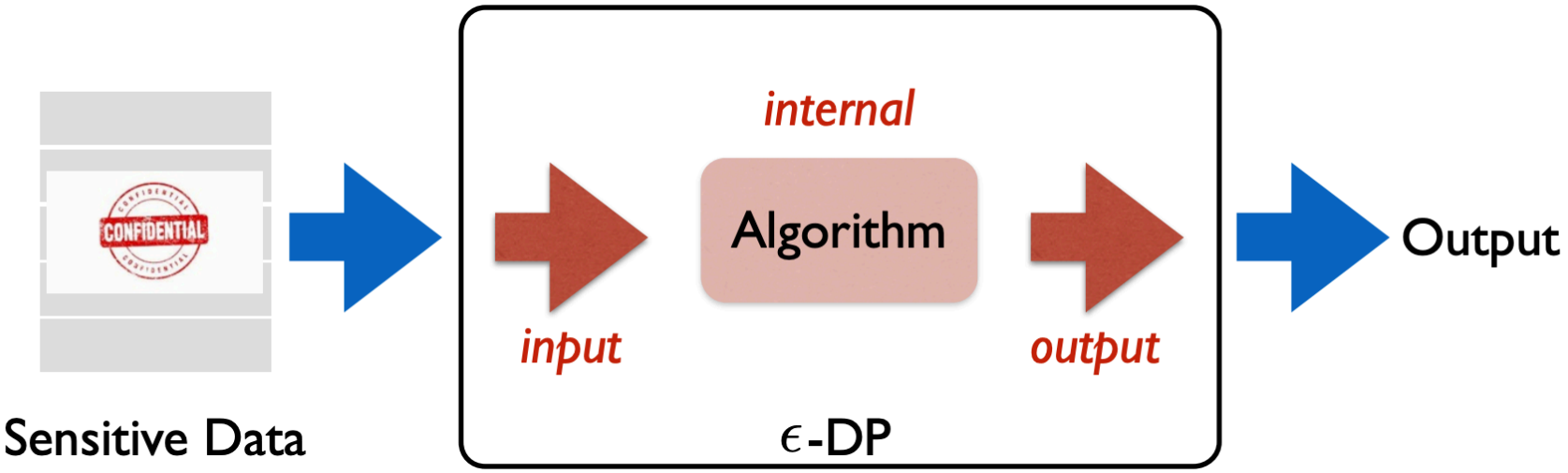
A randomized algorithm K satisfies ε-differential privacy if:

Given any pair of neighboring data sets,
D and D', and S in Range(K):

$$Pr[K(D) = S] \leq e^{\varepsilon} Pr[K(D') = S]$$

Neighboring datasets differ in one individual: we say $|D-D'|=1$

How to Achieve DP



- *input perturbation*: add noise to the input before running algorithm
- *output perturbation*: run algorithm, then add noise (sensitivity)
- *internal perturbation*: randomize the internals of the algorithm

Adversarial Machine Learning: Taxonomy

Attacker's Objective

Learning stage

	Targeted Target small set of points	**Availability** Target majority of points	**Privacy** Learn sensitive information
Training	Targeted Poisoning Backdoor Trojan Attacks	Poisoning Availability Model Poisoning	-
Testing	Evasion Attacks Adversarial Examples	-	Membership Inference Model Extraction

Open Problem: Design Robust AI

- Most AI models are vulnerable in face of attacks!
- This holds for many applications
  - Evasion (testing-time) attacks
  - Poisoning (training-time) attacks
  - Privacy attacks
- How to design AI algorithms robust to attacks?

Acknowledgements

- Thank the TAs
  - Alex and Matthew
- Thanks Everyone for a great semester!
- Stay safe and enjoy the holidays!

Acknowledgements

- Slides made using some resources from:
  - Battista Biggio
  - Byron Wallace
  - Reza Shokri
- Alesia Chernikova, Matthew Jagielski, and Giorgio Severi from the NDS2 Lab at the Khoury College contributed slides