Notes

- Call for actions
  - Homework 2 due (on 11/07, 1-week extension)
  - Switch to online from 11/07
  - Checkpoint Presentation II (online, on 11/07)
    - 12-min presentation + 3 min Q&A
    - Presentation MUST cover:
      - 1 slide on your research topic
      - 1-2 slides on your goals and ideas (how do you plan to achieve your goals)
      - 1-2 slides on your *experimental design*
      - 1-2 slides on your *preliminary results* [*very important*]
      - 1 slide on your *next steps* until the final presentation

CS 499/579: Trustworthy ML

Indiscriminate Poisoning attacks

Tu/Th 4:00 – 5:50 pm

Sanghyun Hong

sanghyun.hong@oregonstate.edu

Oregon State University

SAIL
Secure AI Systems Lab

Poisoning threat model

- Goal
  - Manipulate a ML model's behavior by **compromising the training data**
  - Harm the integrity of the training data

- Capability
  - Perturb a subset of samples (D_p) in the training data
  - Inject a few malicious samples (D_p) into the training data

- Knowledge
  - D_{train}: training data
  - D_{test}: test-set data
  - f: a model architecture and its parameters θ
  - A: training algorithm (*e.g.*, SGD)

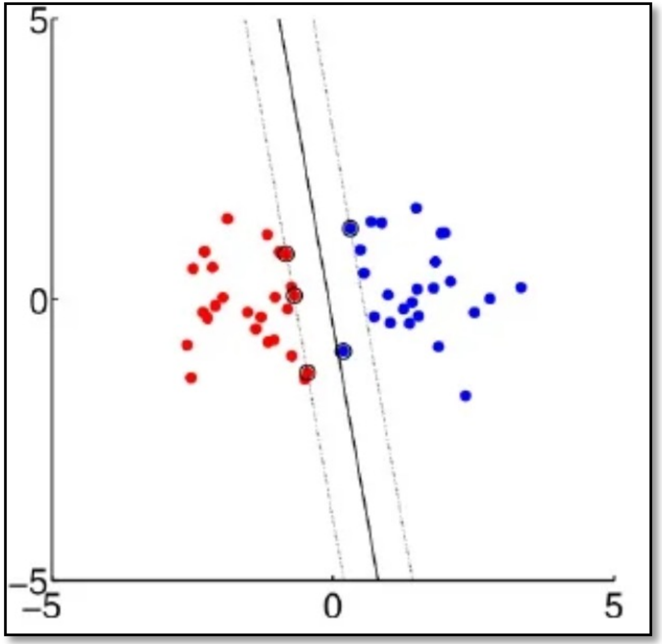
Poisoning threat model: goals

- Goal
  - Manipulate a ML model's behavior by **contaminating the training data**
  - Harm the integrity of the training data
- Two well-studied objectives
  - Indiscriminate attack: I want to degrade a model's accuracy!
  - Targeted attack: I want misclassification of a specific test-time data!

Topics for Part II – Data Poisoning

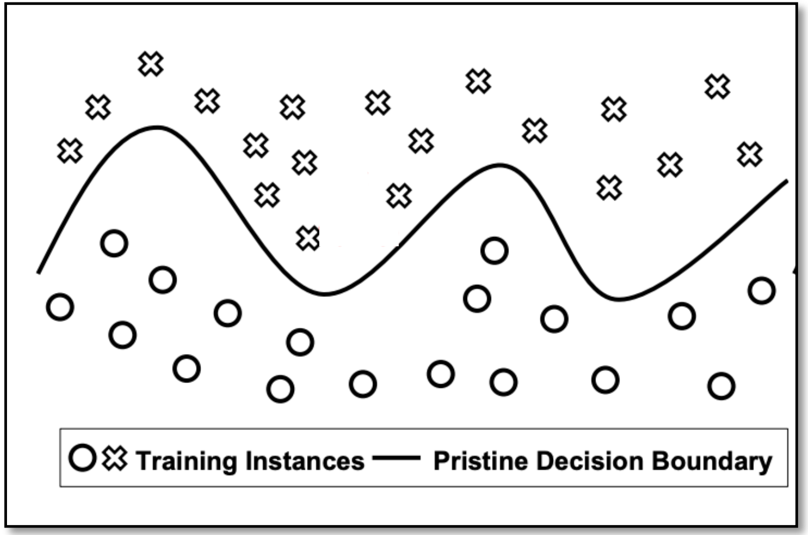
- Research questions
  - What are some examples of poisoning attacks?
  - How can we generate *indiscriminate* poisoning examples?
  - How can we synthesize poisoning samples for *targeted* attacks?
  - How can we mitigate data poisoning attacks?

Conceptual analysis of the poisoning vulnerability

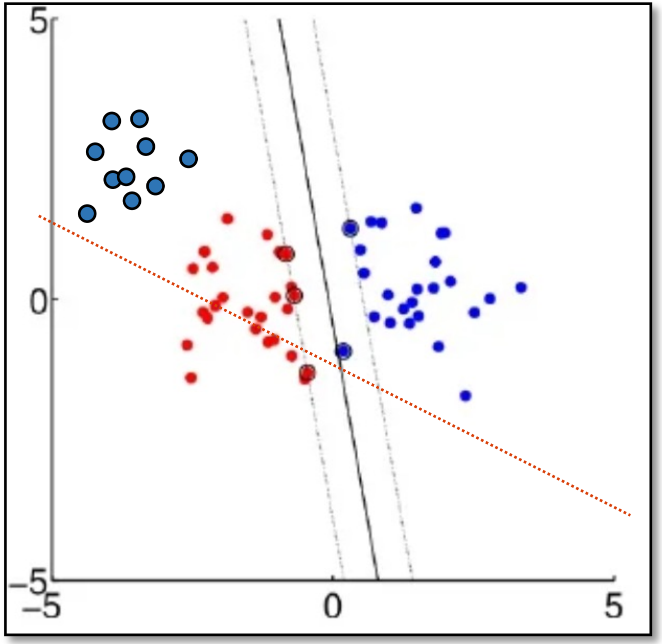


← Linear model (SVM)

Neural Network →

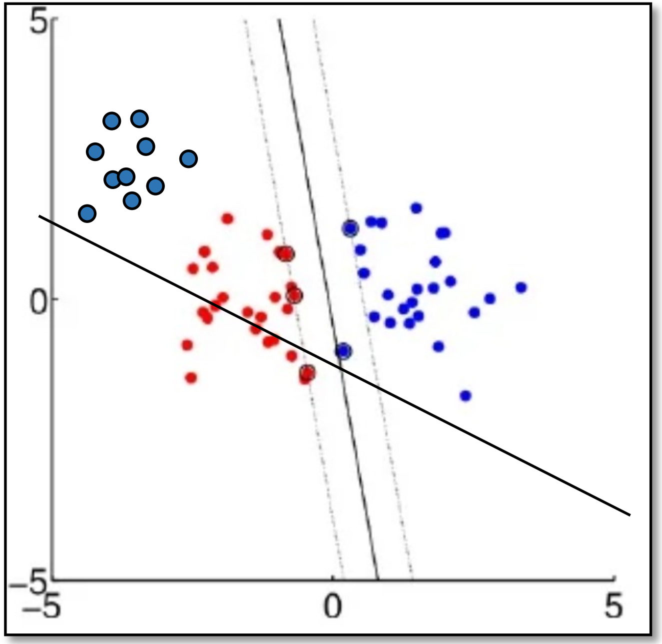


Conceptual analysis of the poisoning vulnerability

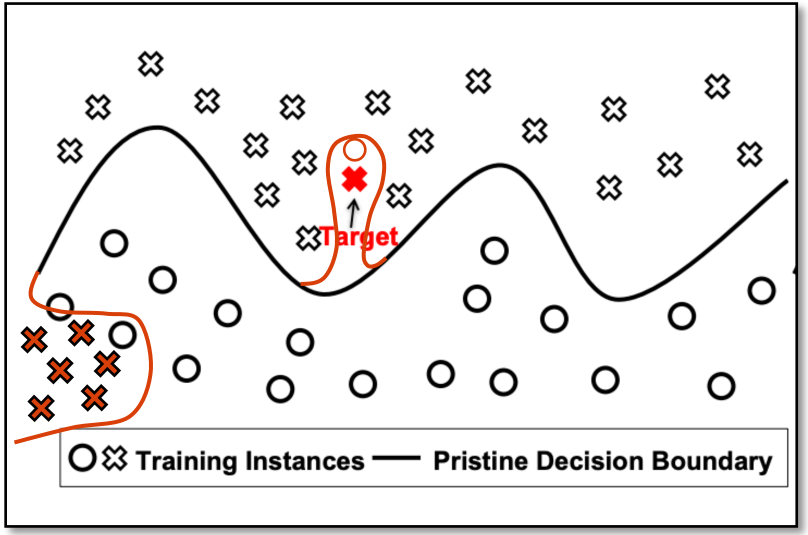


← Linear model (SVM)

Conceptual illustration of the vulnerability to poisoning



← Linear model (SVM)



Neural Network →

How can we perform indiscriminate attacks?

Poisoning attacks against support vector machines, Biggio et al., ICML 2012

Preliminaries: support vector machine

- DIT [Link]
  - 1: let's put green points
  - 2: let's put red points on the other side
  - 3: let's put red points closer to the green cluster
  - 4: let's put red points in the middle of the green cluster
  - 5: let's use another kernel.

Poisoning threat model

- Goal
  - Manipulate a ML model's **accuracy** by compromising the training data
  - In short: **indiscriminate** attack

- Capability
  - Pick a set of test-time samples and craft poisons (x_c, y_c)
  - Inject them into the training data

- Knowledge
  - D_{tr} : training data
  - D_{test}: test-set data (validation data)
  - f: a linear SVM and its parameters θ
  - A: training algorithm (*e.g.*, Sub-gradient descent)

Poisoning threat model

- Label noise in ImageNet[1]

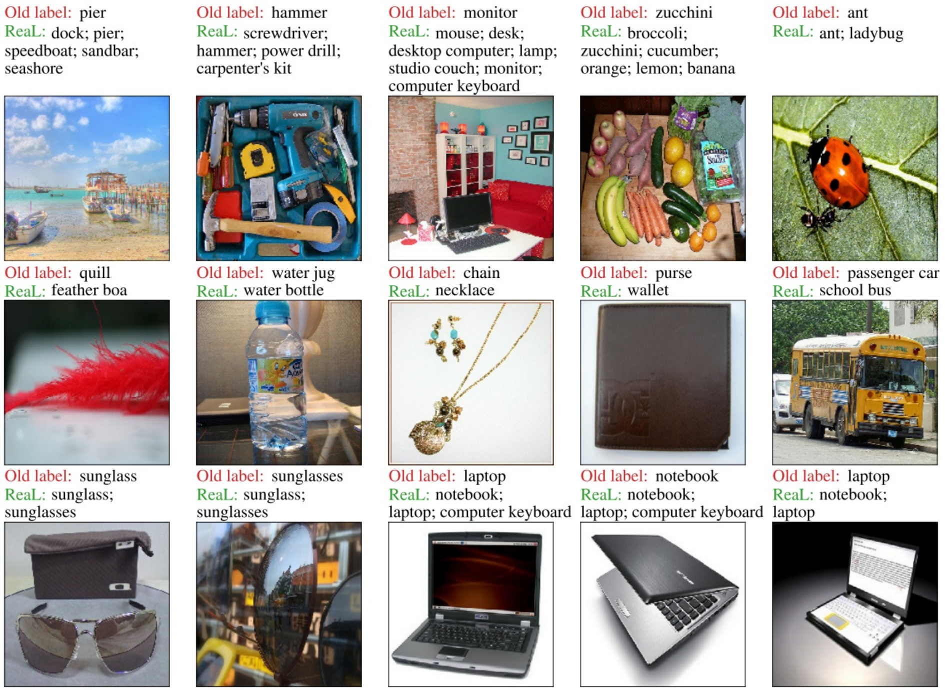


Figure 2: Example failures of the ImageNet labeling procedure. Red: original ImageNet label, green: proposed ReaL labels. **Top row**: ImageNet currently assigns a single label per image, yet these often contain several equally prominent objects. **Middle row**: Even when a single object is present, ImageNet labels present systematic inaccuracies due to their labeling procedure. **Bottom row**: ImageNet classes contain a few unresolvable distinctions.

[1]Beyer et al., Are we done with ImageNet? arXiv 2020

Proposed attack on support vector machine

- Indiscriminate attack procedure
  - Draw a set of poison candidates from the validation data
  - Craft poisoning samples
  - Inject them into the original training data
  - Increase the loss of the model trained on the compromised data

Proposed attack on support vector machine

Algorithm 1 Poisoning attack against SVM

Input: $\mathcal{D}_{\text{tr}}$, the training data; $\mathcal{D}_{\text{val}}$, the validation data; y_c, the class label of the attack point; $x_c^{(0)}$, the initial attack point; t, the step size.

Output: x_c, the final attack point.

1: $\{\alpha_i, b\} \leftarrow$ learn an SVM on $\mathcal{D}_{\text{tr}}$. // train an SVM on the clean data

2: $k \leftarrow 0$.

3: **repeat**

4: Re-compute the SVM solution on $\mathcal{D}_{\text{tr}} \cup \{x_c^{(p)}, y_c\}$ using incremental SVM (*e.g.*, Cauwenberghs & Poggio, 2001). This step requires $\{\alpha_i, b\}$. // train an SVM with the poison

5: Compute $\frac{\partial L}{\partial u}$ on $\mathcal{D}_{\text{val}}$ according to Eq. (10). // compute the gradient

6: Set u to a unit vector aligned with $\frac{\partial L}{\partial u}$.

7: $k \leftarrow k + 1$ and $x_c^{(p)} \leftarrow x_c^{(p-1)} + tu$ // update the poison, to increase the loss

8: **until** $L\left(x_c^{(p)}\right) - L\left(x_c^{(p-1)}\right) < \epsilon$ // stop if the loss doesn't increase more than ϵ

9: **return:** $x_c = x_c^{(p)}$

Proposed attack on support vector machine

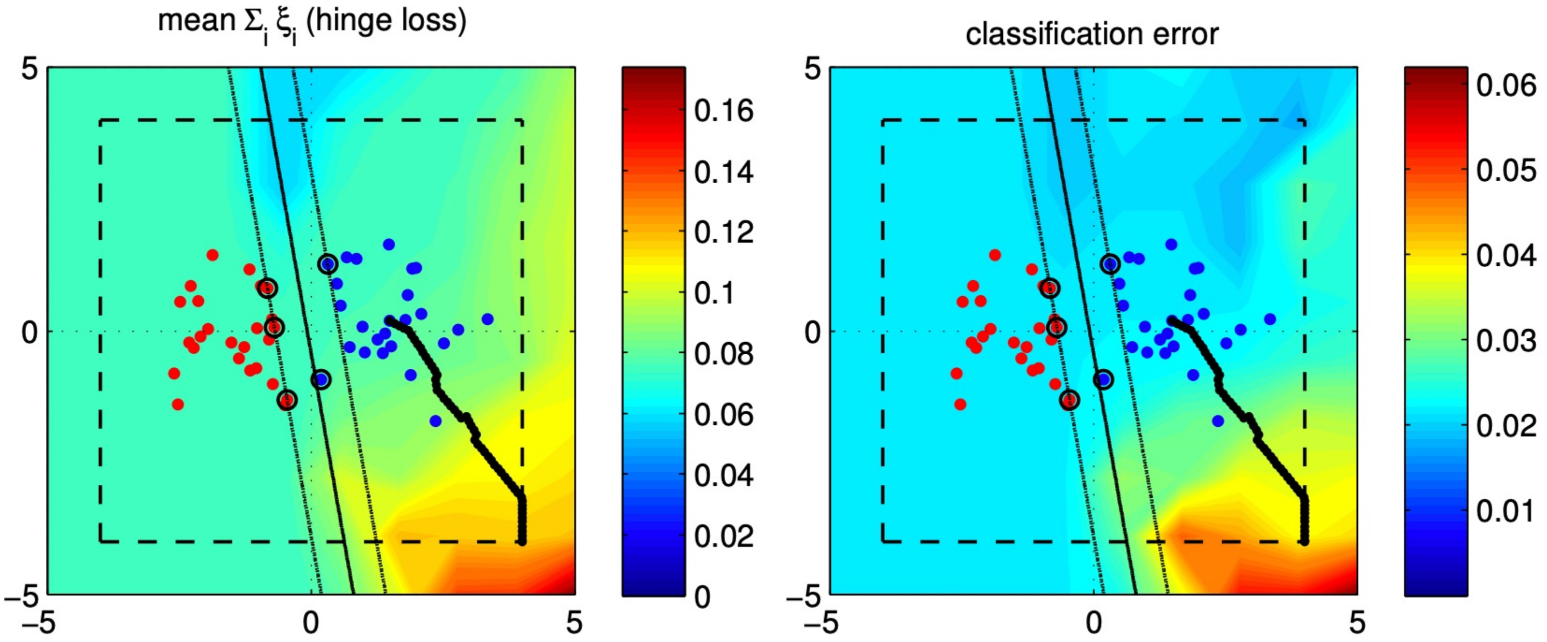
- Indiscriminate attack procedure
  - Draw a set of poison candidates from the validation data
  - Craft poisoning samples
  - Inject them into the original training data
  - Increase the loss of the model trained on the compromised data

EVALUATION

- Setup
  - Datasets
    - Artificial data:
      - Binary classification: Gaussian dist. [$N(-1.5, 0.6^2)$ and $N(1.5, 0.6^2)$]
      - Training data : 50 samples, 25 per class
      - Validation data: 1k samples, 500 per class
    - Real data: MNIST
  - Model(s)
    - SVM [Linear vs. RBF-Kernel]

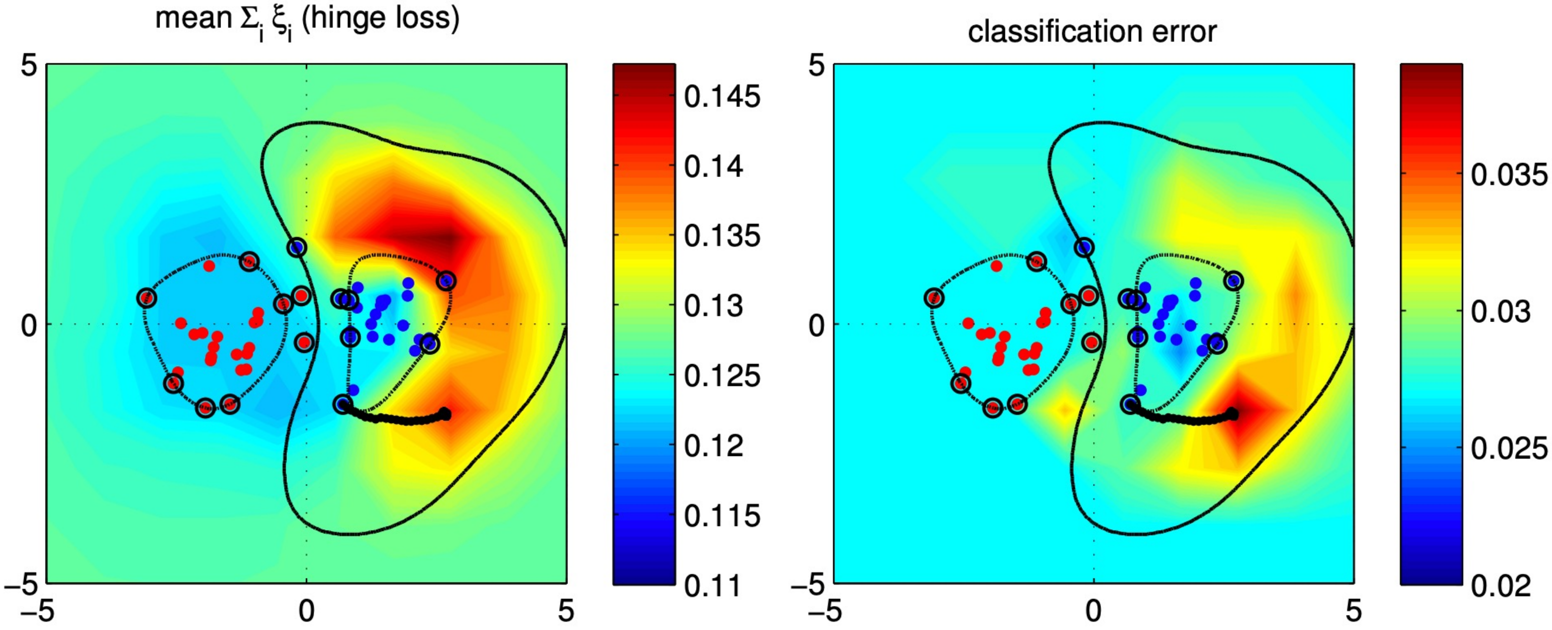
Evaluation: poison crafting in artificial data

- Linear SVM



Evaluation: poison crafting in artificial data

- SVM with RBF Kernel

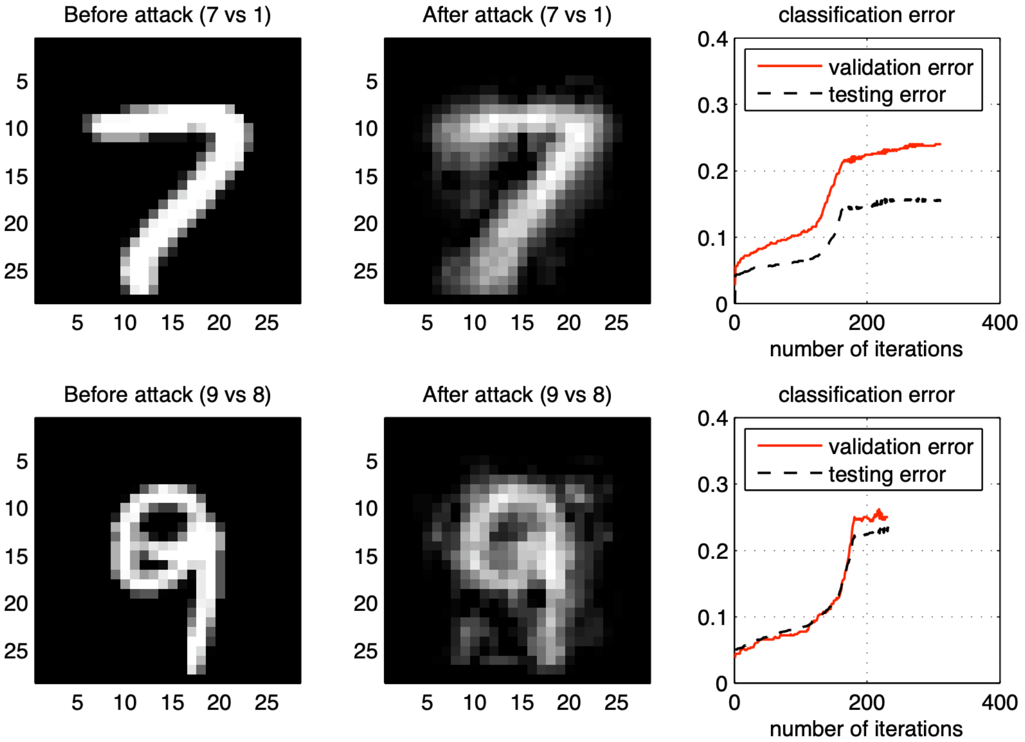


Evaluation

- Setup
  - Datasets
    - Artificial data:
      - Binary classification: Gaussian dist. $[N(-1.5, 0.6^2)$ and $N(1.5, 0.6^2)]$
      - Training data : 50 samples, 25 per class
      - Validation data: 1k samples, 500 per class
    - Real data: MNIST
      - 7 vs 1 | 9 vs 8 | 4 vs 0
      - Training data : 200 samples, 100 per class
      - Validation data: 1k samples, 500 per class
      - Testing data : 4k samples, 2k per class
  - Model(s)
    - SVM [Linear vs. RBF-Kernel]

Evaluation: real-data (MNIST)

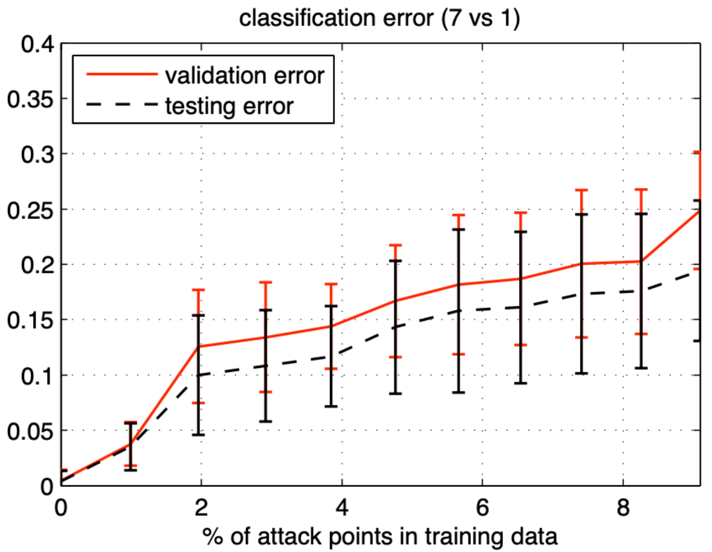
- Linear SVM



- Results
  - Use a *single* poison
  - Error increases by 15 – 20%

Evaluation: real-data (MNIST)

- Linear SVM



- Results
  - Use a *single* poison
  - Error increases by 15 – 20%
  - Increasing # poisons leads to a higher error

How can we perform indiscriminate attacks?

Manipulating machine learning: poisoning attacks and countermeasures for regression leaning, jagielski et al., IEEE Security and privacy symposium 2018

Thank You!

Tu/Th 4:00 – 5:50 pm

Sanghyun Hong

https://secure-ai.systems/courses/MLSec/F23

Oregon State University

SAIL

Secure AI Systems Lab