Lecture 26: Machine Learning Security

CS 138

Spring 2026

Background: Machine Learning







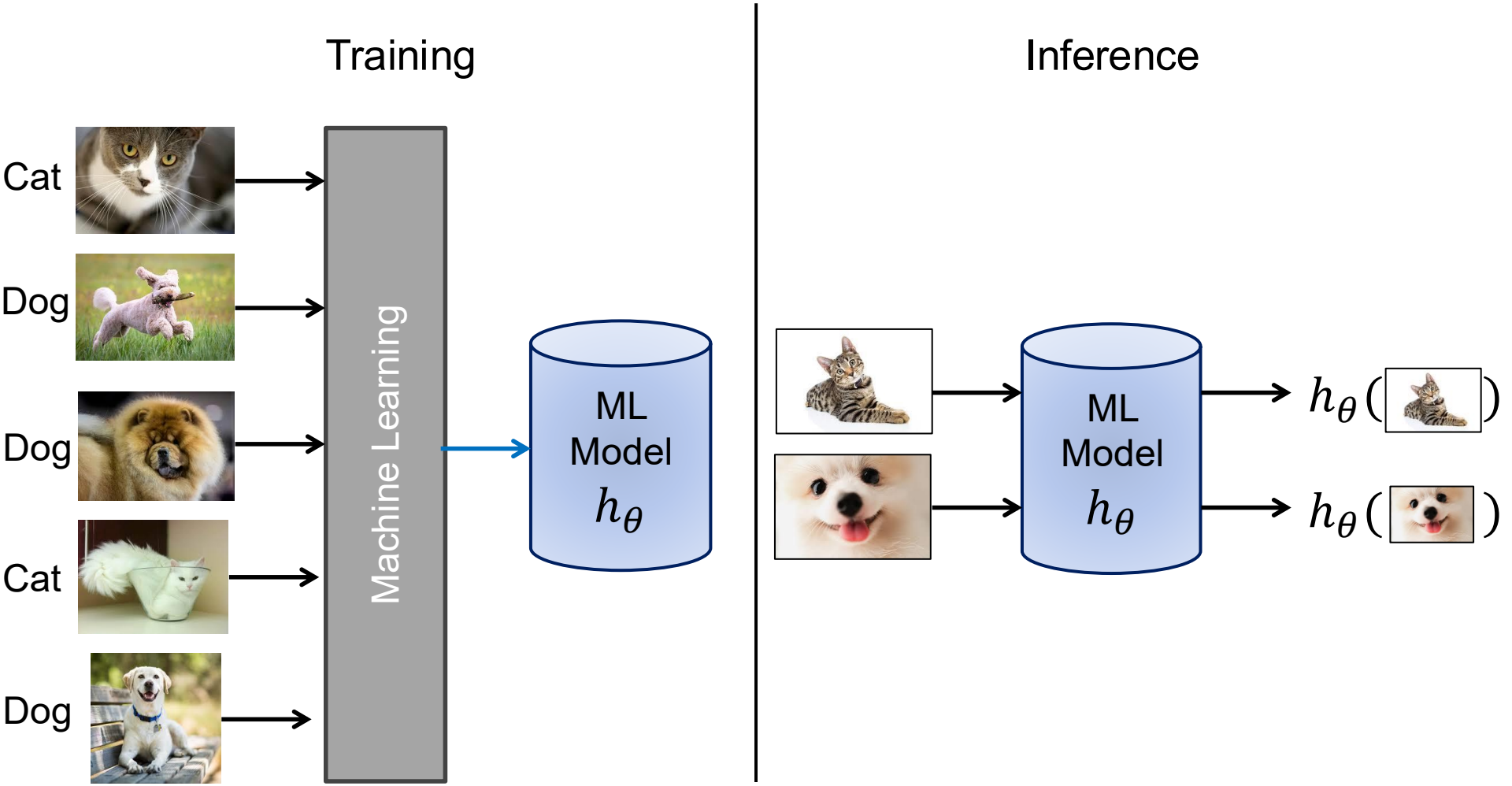









Background: ML Stages



Exercise: Security Goals

- What Confidentiality and/or Integrity goals would you like to have for the training phase?

Training-Stage Attacks

Confidentiality

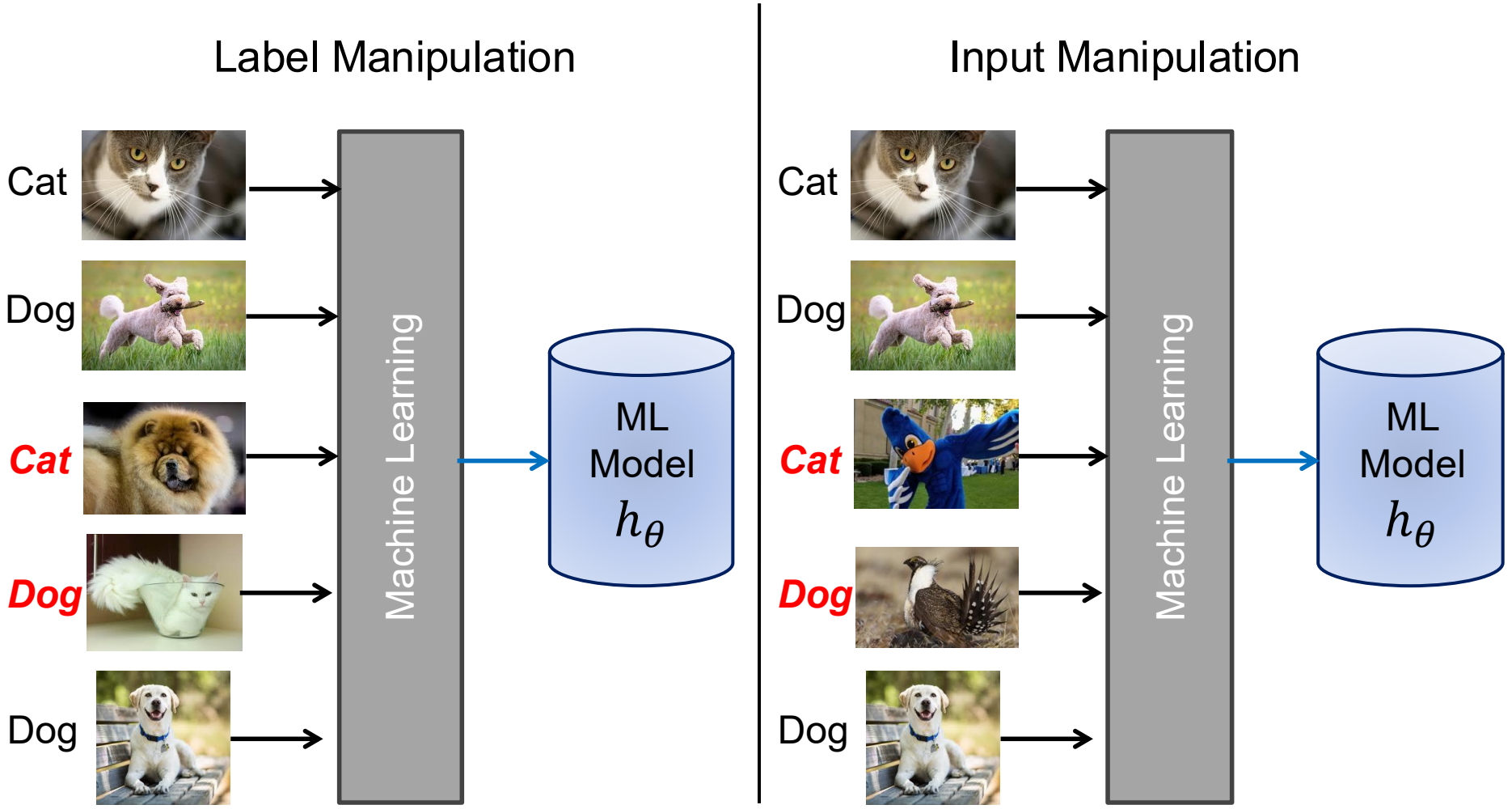
- training data
- model parameters

Access Control

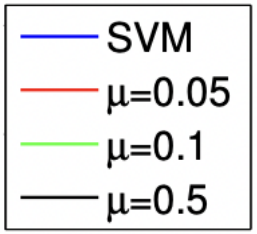
Integrity

- Model poisoning

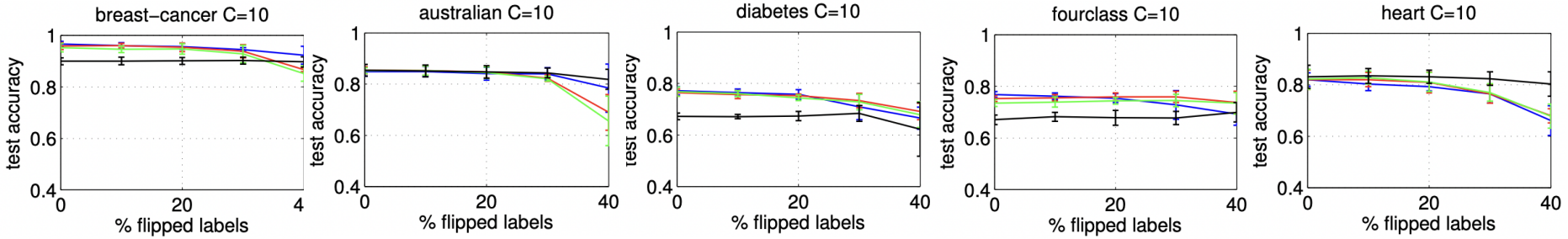
Model Poisoning Attacks



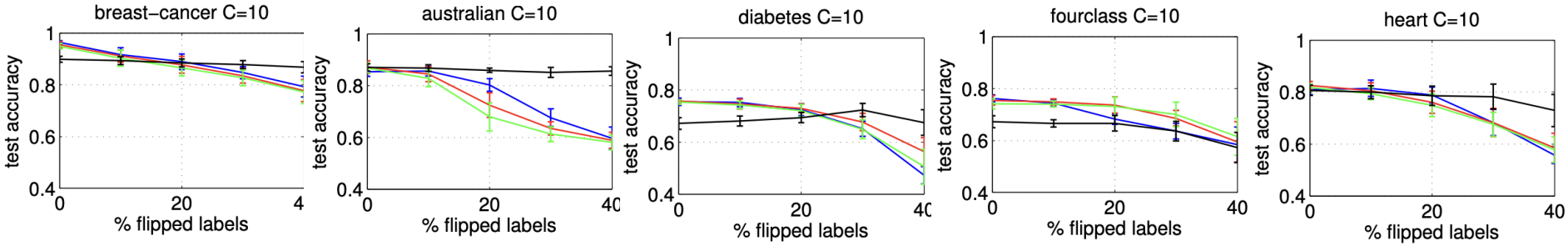
Label Manipulation Attacks



- Random Flipping: 40% labels -> Accuracy significantly reduced



- Heuristic Flipping: bias sample towards high-confidence training values improves effectiveness and robustness

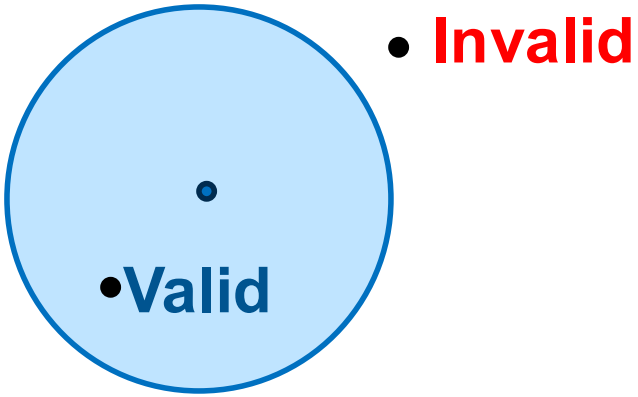


Input Manipulation: Anomaly Detection

- **Anomaly Detection:** Given a dataset X, goal is to determine whether a new sample x is drawn from the same distribution as X
- **Centroid Anomaly Detection:** use Euclidean distance from empirical mean as metric

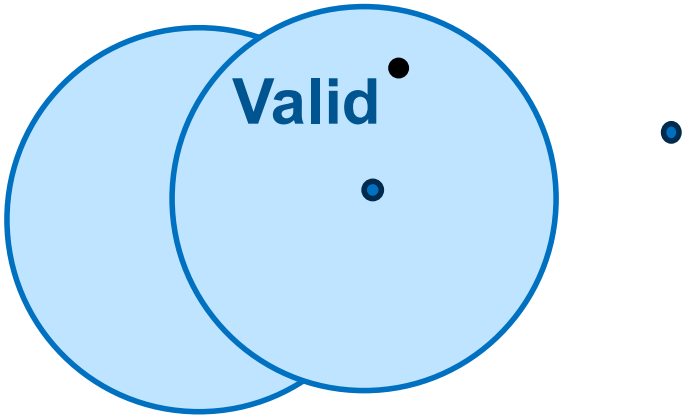
$$f(x) = \left\| x - \frac{1}{n} \sum_{i=1}^{n} x_i \right\|$$

  - reject inputs above threshold r



Input Manipulation: Anomaly Detection

- **Online Anomaly Detection:** update normality model
  - Update mean by adding new valid datapoint
  - Remove random old point and add new datapoint
  - Remove old point nearest to new datapoint
  - Remove one point at old mean and add new datapoint
- **Adversarial Input Manipulation:**

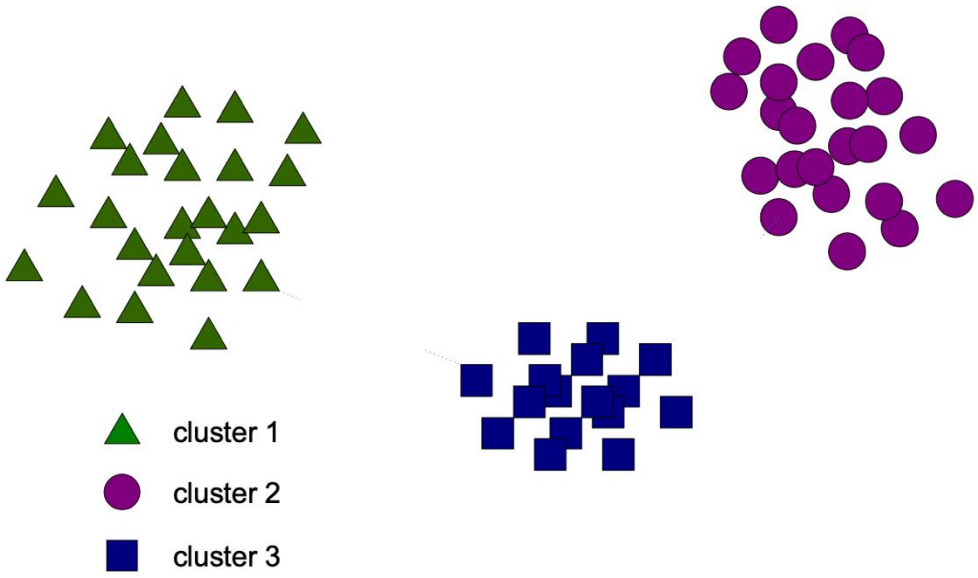


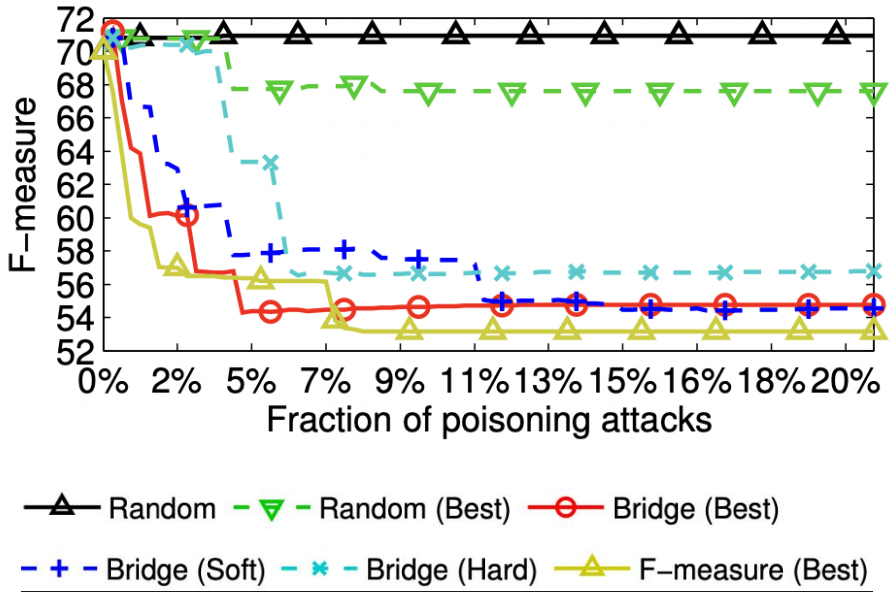
  - Effective at poisoning anomaly detector for HTTP traffic

Input Manipulation: Malware Clustering

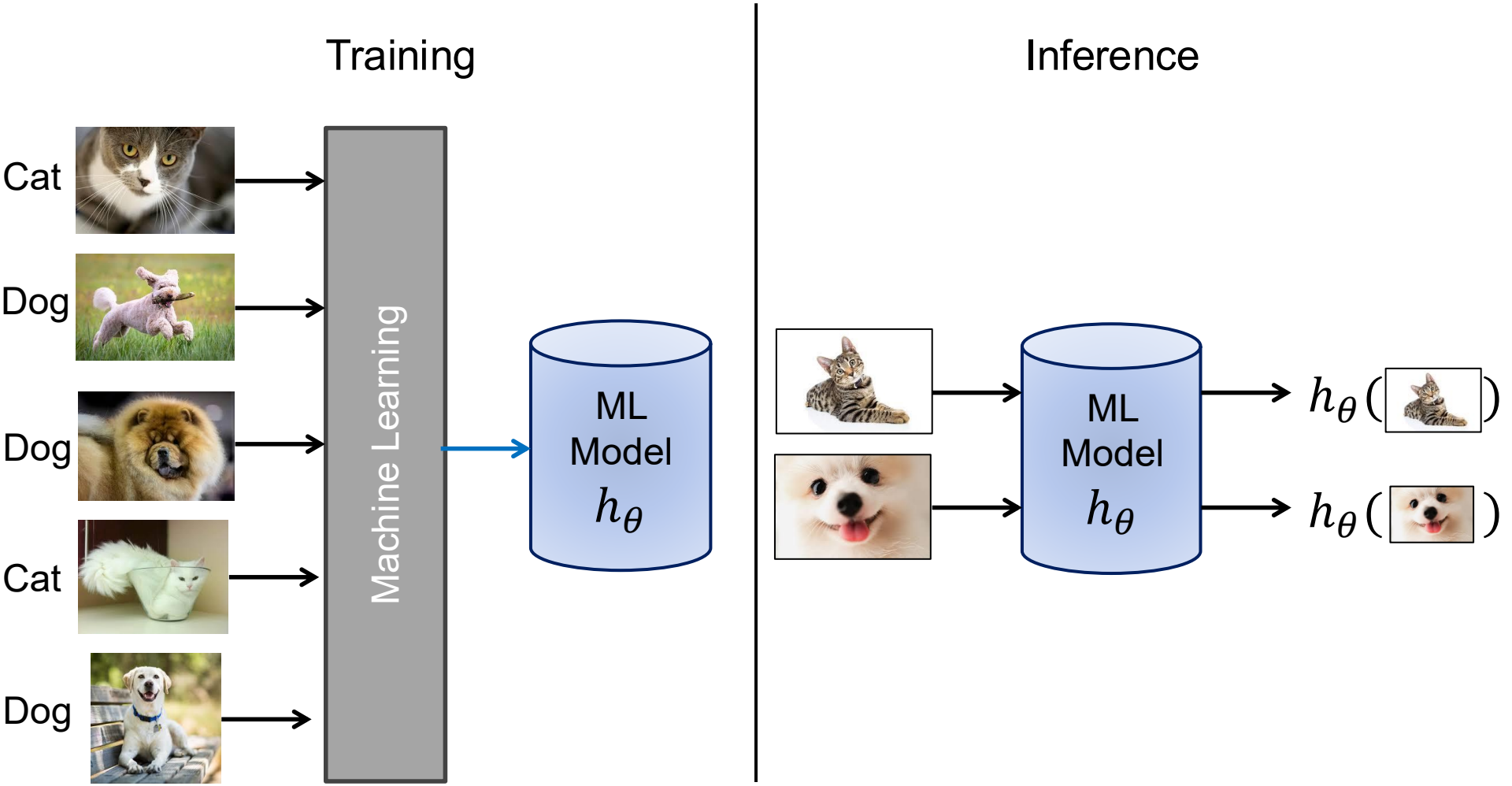
- Clustering is used to characterize related malware and generate network signatures

- Poisoning attacks can prevent ML from accurately identifying clusters





Background: ML Stages



Exercise: Security Goals

- What Confidentiality and/or Integrity goals would you like to have for the inference phase?

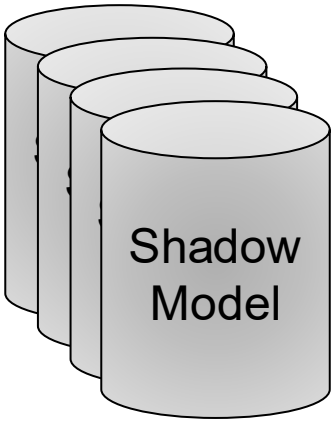
Inference-Stage Attacks

Confidentiality	Integrity
• Membership inference • Model inversion • Model extraction	

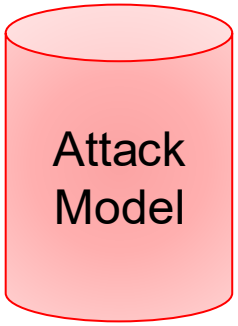
Membership Inference

Goal: Given a ML model and a data record, determine whether record was used to train that model

1. Train shadow models on same task



2. Using shadow models as training set, train attack model on classification task: was x in training set for model M

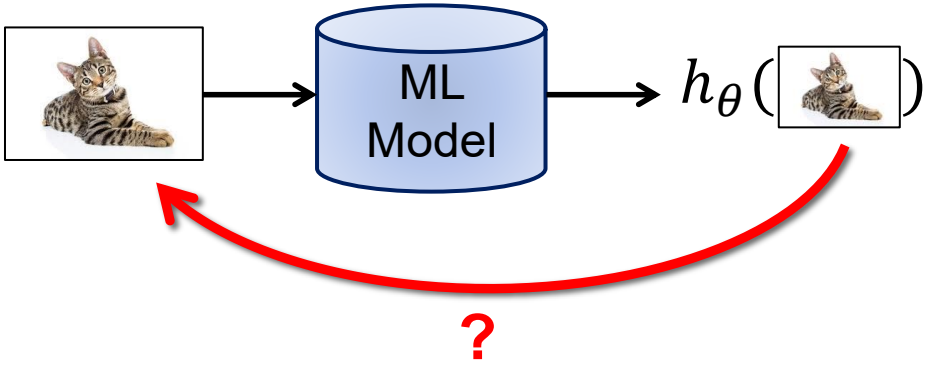


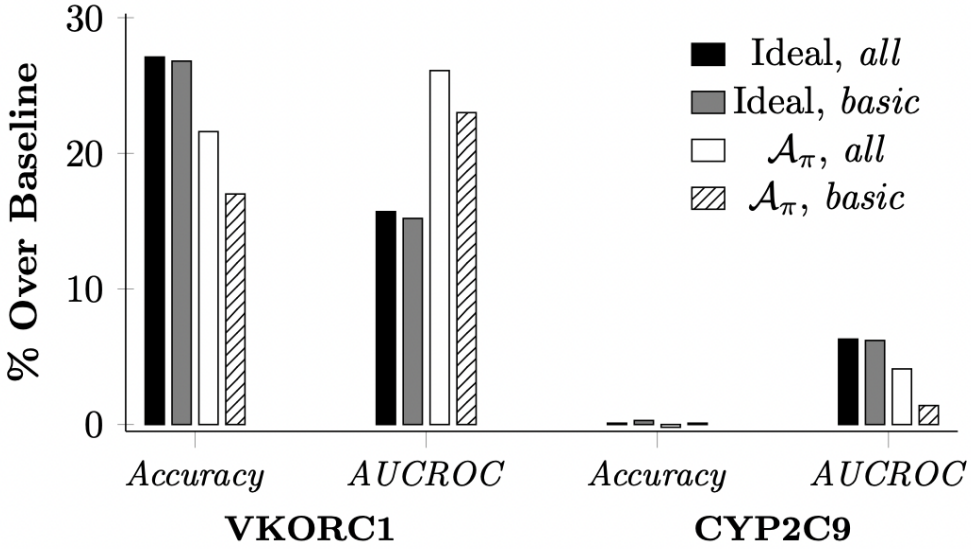
83-92% accuracy

3. Use attack model to decide whether record was used to train target model

Model Inversion

- Goal: Learn (private) training data from ML outputs





Model Extraction

- Goal: Learn model parameters given black-box access

- For logistic regressions w/ confidence values: ask multiple queries, solve system of equations
- For decision trees: for each leaf, search for constraints that stay on leaf

Model	Unknowns	Queries	$1 - R_{\text{test}}$
Softmax	530	265	99.96%
		530	100.00%
OvR	530	265	99.98%
		530	100.00%
MLP	2,225	1,112	98.17%
		2,225	98.68%
		4,450	99.89%
		11,125	99.96%

Model	Leaves	Depth	$1 - R_{\text{test}}$	Queries
IRS Tax Patterns	318	8	100.00%	101,057
Steak Survey	193	17	92.45%	3,652
GSS Survey	159	8	99.98%	7,434
Email Importance	109	17	99.13%	12,888
Email Spam	219	29	87.20%	42,324
German Credit	26	11	100.00%	1,722
Medical Cover	49	11	100.00%	5,966
Bitcoin Price	155	9	100.00%	31,956

Inference-Stage Attacks

Confidentiality

- Membership inference
- Training data extraction
- Model extraction

Integrity

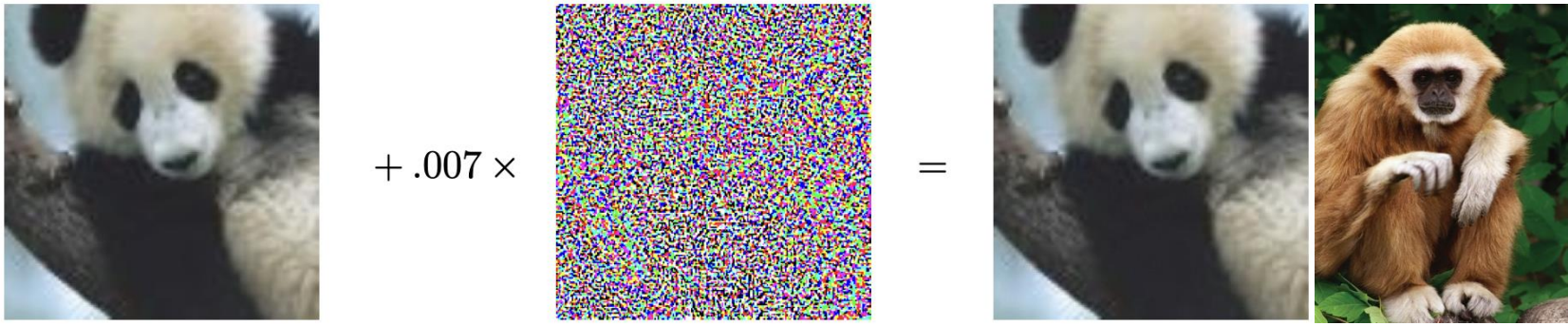
- Adversarial Examples

Direct Adversarial Examples

- Consider a linear model:

$$h_\theta(\vec{x}) = \underbrace{\begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix}}_{\theta} \cdot \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}}_{\vec{x}}$$

- want $\vec{x}'$ such that $\|\vec{x} - \vec{x}'\|_\infty < \epsilon$, but $h_\theta(\vec{x})$ and $h_\theta(\vec{x}')$ differ



$\boldsymbol{x}$

"panda"
57.7% confidence

$\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$

"nematode"
8.2% confidence

$\boldsymbol{x} + \epsilon\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
"gibbon"
99.3 % confidence

Real-World Adversarial Examples

- Goal: Modify $\vec{x}$ s.t. $h_\theta(\vec{x}') =$ 

100%

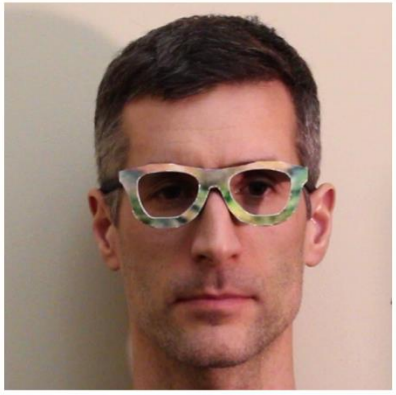
77.3%

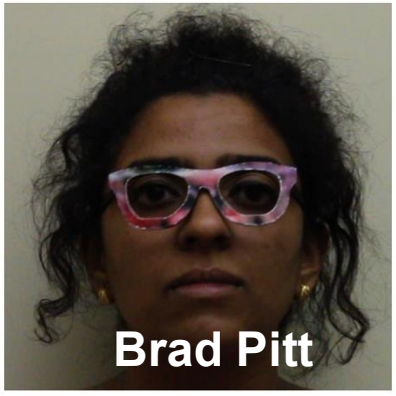
66.7%

100%

80%

- Goal: Defeat facial recognition





Review: Security Attacks on ML

Training-Stage Attacks

Confidentiality	Integrity
• training data • model parameters	• Model poisoning

Inference-Stage Attacks

Confidentiality	Integrity
• Membership inference • Training data extraction • Model extraction	• Adversarial Examples

Defending against ML Attacks

- **Outlier mitigation:** detect examples outside normal distribution and mitigate their impact on final model
- **Differentially-private training:** ensure that there is no significant difference if datapoint is in training set
- **Gradient masking:** minimize model sensitivity during training
- **Explainable AI:** justify decisions to (human) auditor
- Active area of research

Machine Learning Security