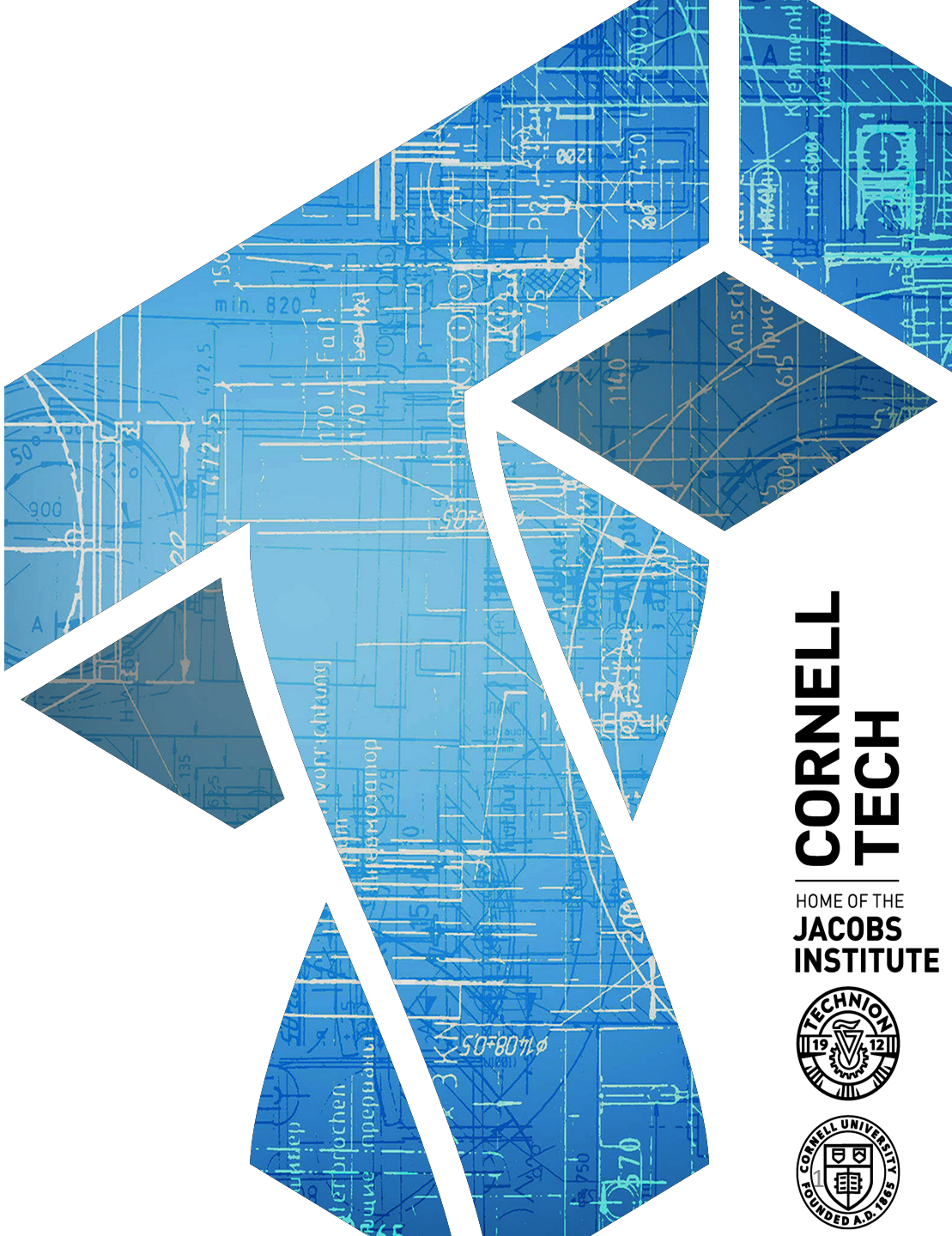
CS 6431:
ML Privacy

Instructor: Tom Ristenpart

https://github.com/tomrist/cs6431-fall2021

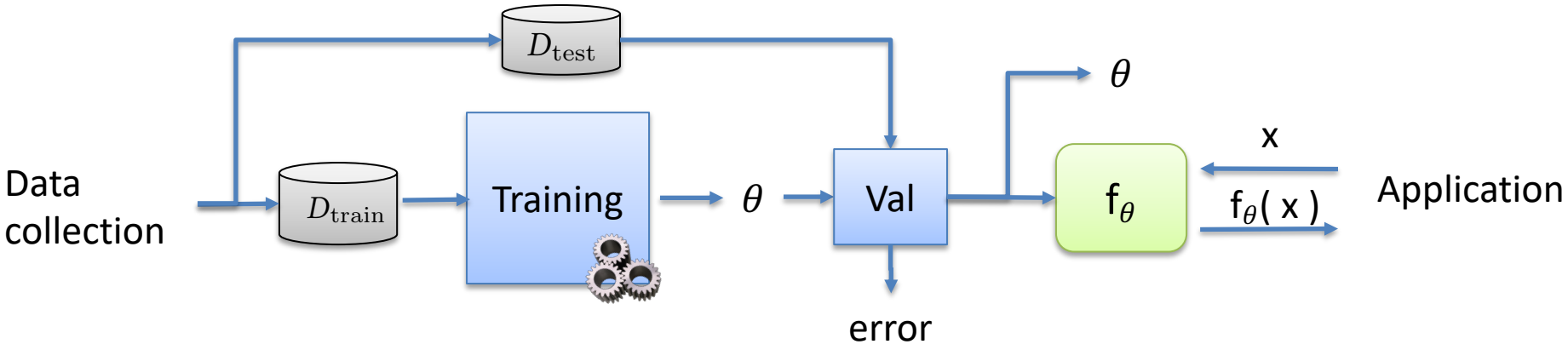


CORNELL TECH
HOME OF THE JACOBS INSTITUTE

ML security last time

- Overview of landscape of threat models
- Model extraction attacks:
- Evasion attacks
- Today we will look at ***privacy*** issues

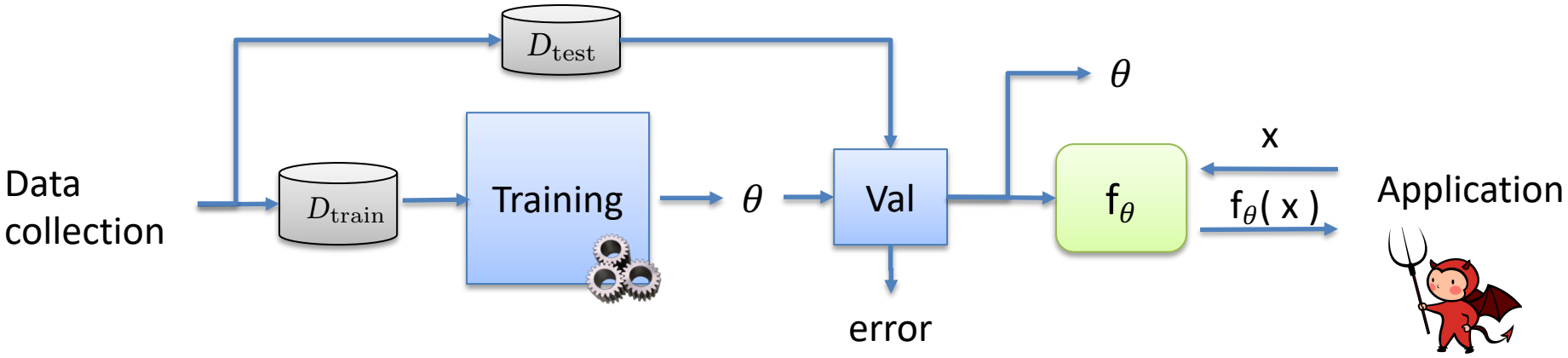
Supervised machine learning (ML) pipeline



What privacy issues related to ML pipelines do you see?

Membership inference

[Shokri, Stronati, Song, Shmatikov 2017]

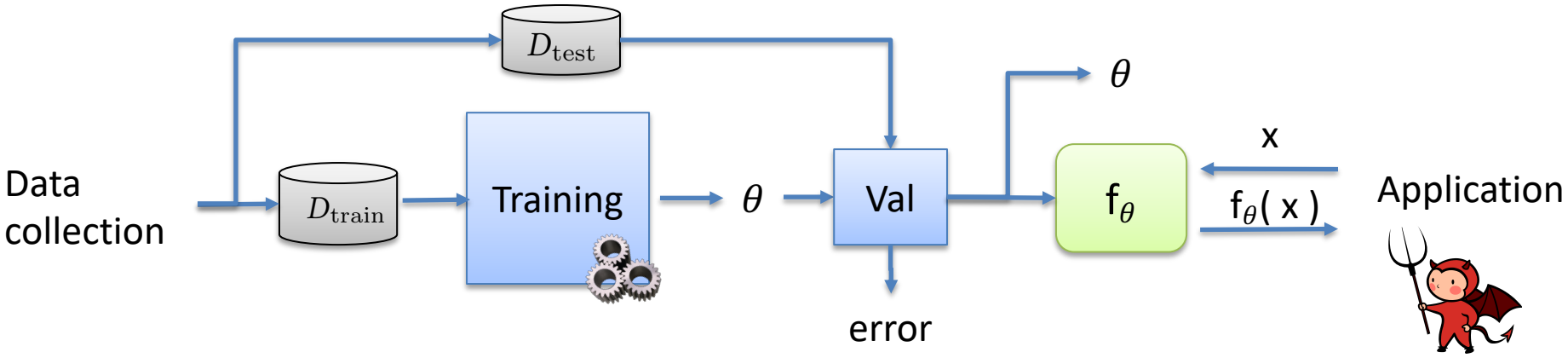


Adversary given (x,y) and access to f_θ
Is (x,y) a member of D_{train}?

Is this a privacy risk?

Membership inference

[Shokri, Stronati, Song, Shmatikov 2017]



Adversary given (x,y) and access to f_θ
Is (x,y) a member of D_{train}?

<u>Baseline attack:</u>

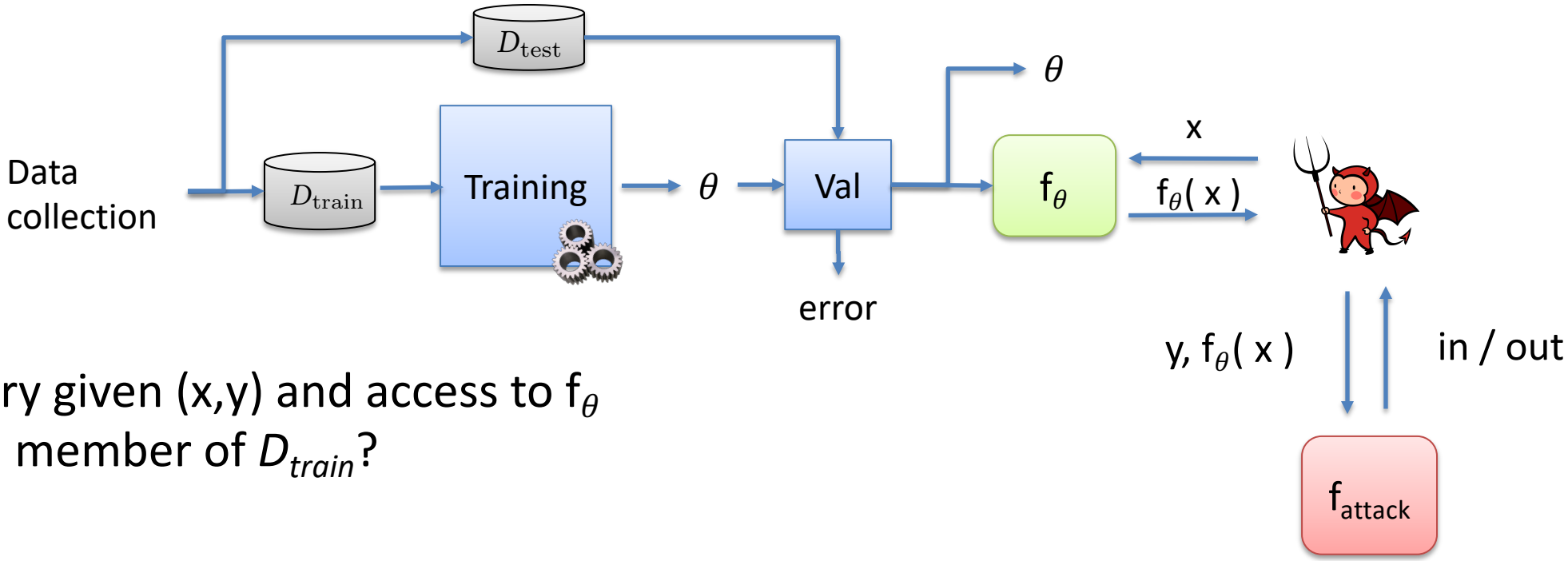
1. Query to get y' = $f_\theta(x)$
2. Output "yes" if y' = y, otherwise "no"

Exploits train-test gap (overfitting)

Only using label. What about confidence values?

Membership inference

[Shokri, Stronati, Song, Shmatikov 2017]



Adversary given (x,y) and access to f_θ
Is (x,y) a member of D_{train}?

<u>Their suggested approach:</u>

1. Train attack model f_{attack} to predict membership using confidence vectors
2. Get confidence vectors for target, apply f_{attack}

Training an attack model

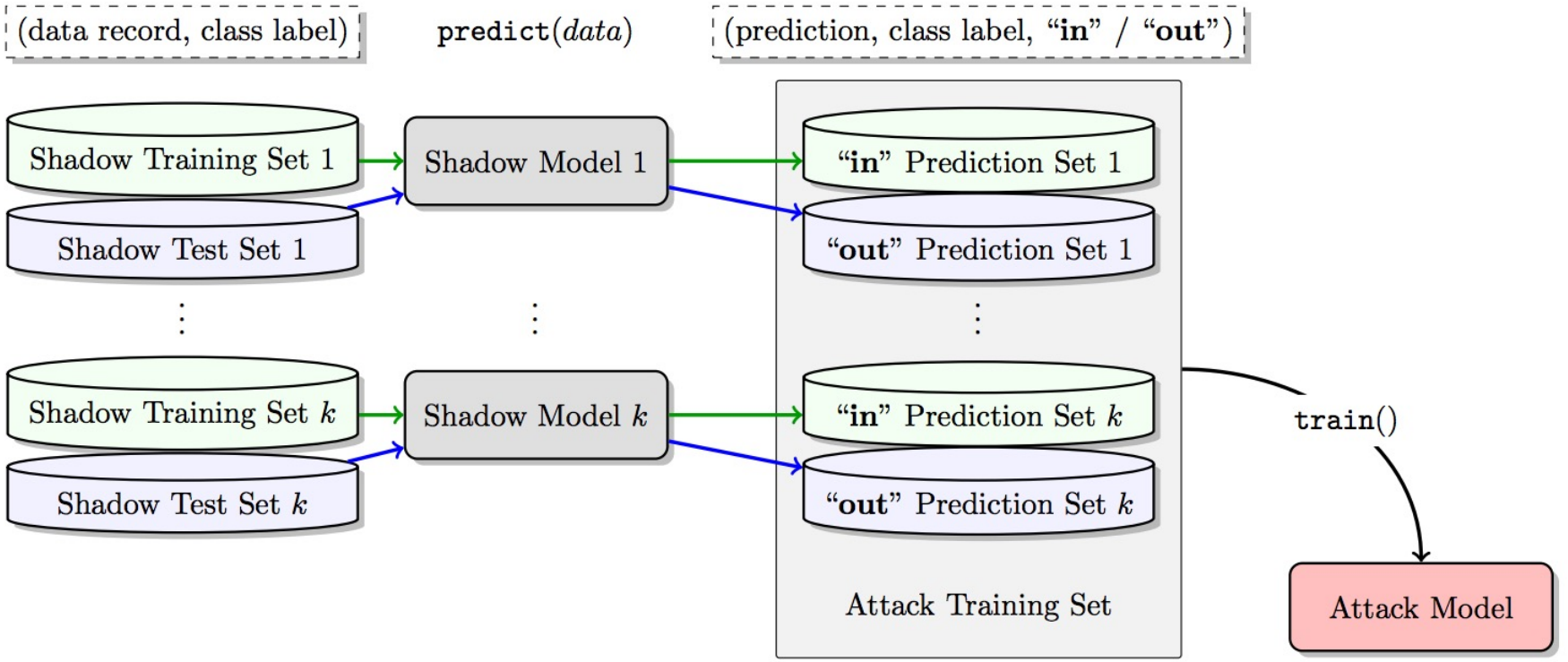
How do we train an attack model?

- Synthesize training data set using black-box access to f_θ
- Use marginals for features
  - $\Pr[\, A \cap B \,]$ measures joint probability of events A and B

    $\Pr[\, A \,] = \sum_b \Pr[\, A \cap B = b \,]$ is marginal distribution of event A
  - In words: ignore B's outcome, just look at A's probability
- Use in-distribution training data, possibly noised

Shadow model approach

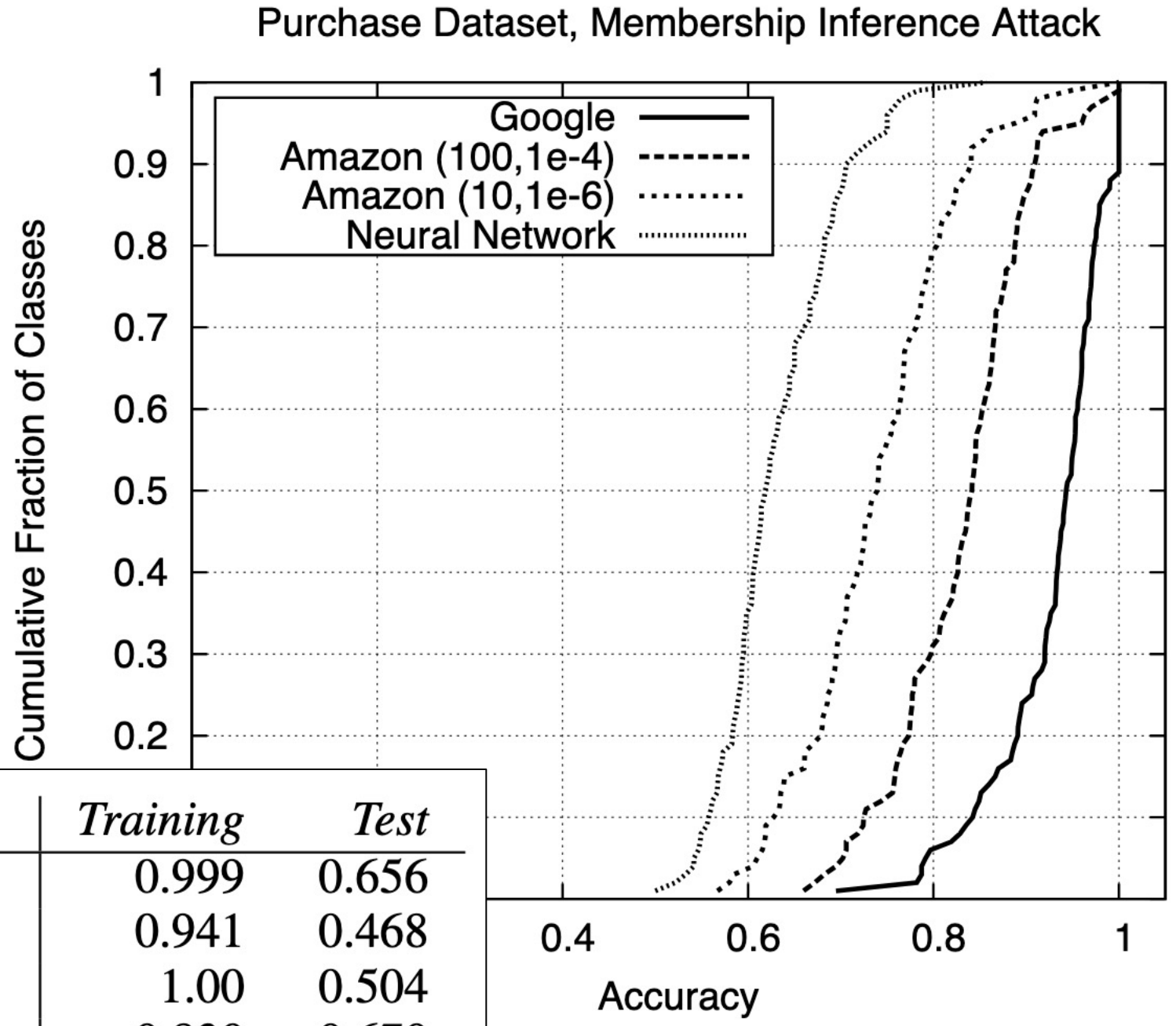
Diagram from [Shokri et al. 2017]



- **Intuition: Confidence vectors contain some information about membership**
- Shadow models used to generate examples of member and non-member confidence vectors
- Examples used to train attack model
- Attack queries $y' = f_\theta(x)$ (now interpreted as confidence vector)
- Runs attack model on (y',y)

Accuracy of attack using in-distribution training data against different target architectures

Train-test accuracy gap of target models



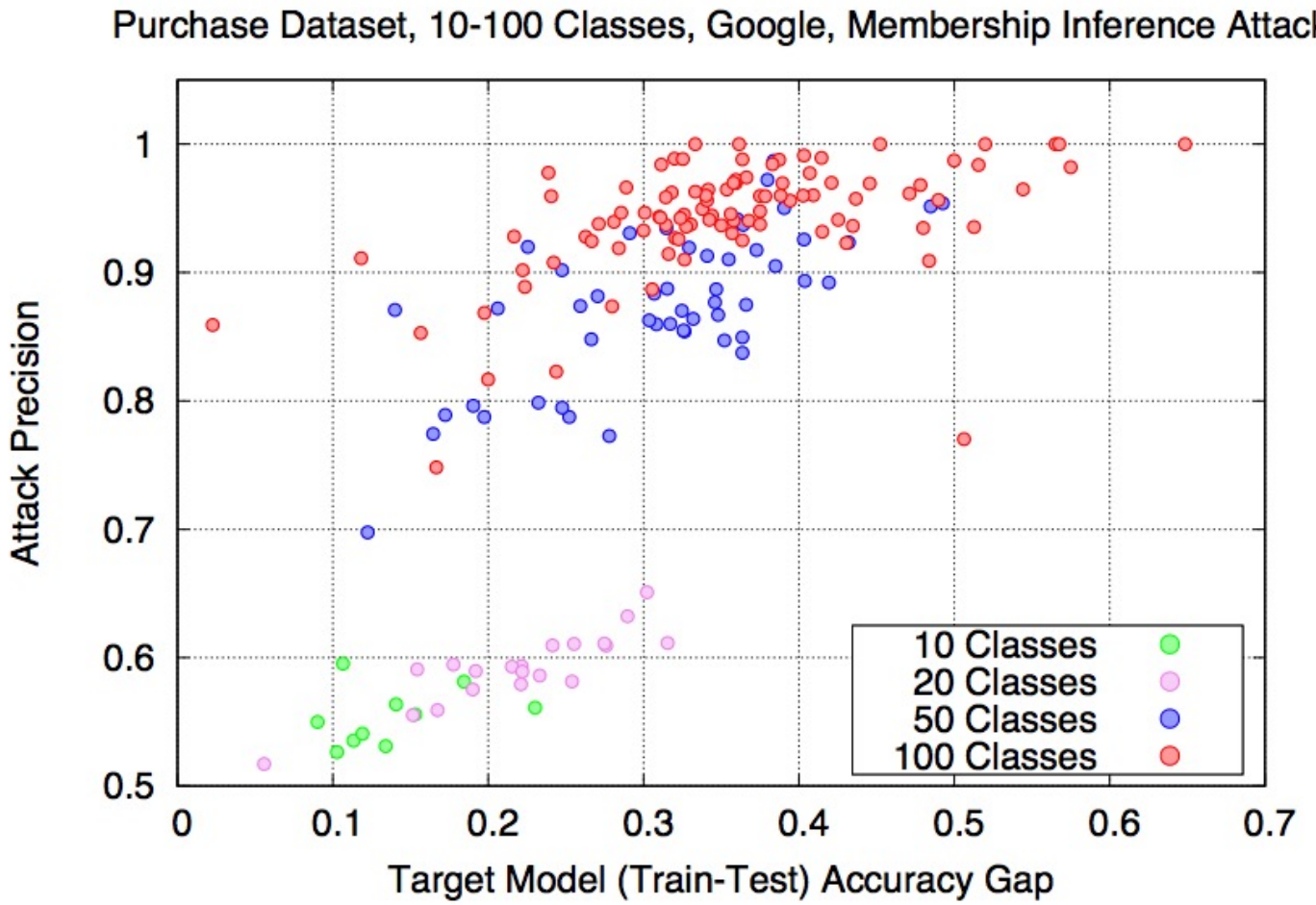
ML Platform	*Training*	*Test*
Google	0.999	0.656
Amazon (10,1e-6)	0.941	0.468
Amazon (100,1e-4)	1.00	0.504
Neural network	0.830	0.670

Precision: when predicting “in”, how often correct?

Dataset	*Training Accuracy*	*Testing Accuracy*	*Attack Precision*
Adult	0.848	0.842	0.503
MNIST	0.984	0.928	0.517
Location	1.000	0.673	0.678
Purchase (2)	0.999	0.984	0.505
Purchase (10)	0.999	0.866	0.550
Purchase (20)	1.000	0.781	0.590
Purchase (50)	1.000	0.693	0.860
Purchase (100)	0.999	0.659	0.935
TX hospital stays	0.668	0.517	0.657

What level of precision is sufficient to be concerned?

Train-test accuracy gap versus attack precision



Take-away: works better when models overfit, but target model architecture plays a role

Mitigations explored

- Reducing output of f_θ --- in limit just class label
  - They show this is ***not*** effective at eliminating attacks.
  - What does this mean?
- Regularization seems useful
  - Why does this make sense?

Paper discussion

- Limitations of their results?
- Experiments they might have added?
- Open questions they left?

Paper in context

- Prior works had looked at other threat models related to machine learning
- Note the style of intro: no reference to prior work before late in intro, mostly to distance from prior model inversion papers
- Related work spends a lot of energy distancing from model inversion. Included a figure!

Model inversion attacks

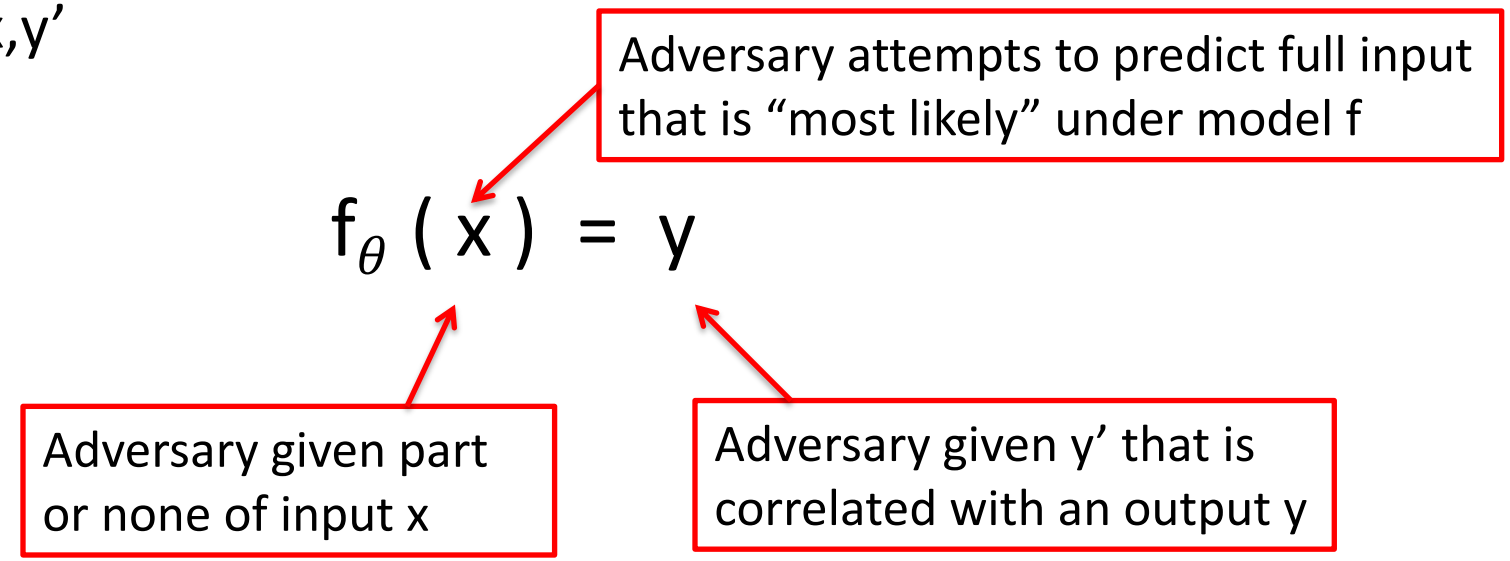
Adversary uses θ or f_θ to infer information about training set members

[Ateniese et al. 2015]: Guess one bit about full training data set
[Shokri et al. 2017]: Determine if x,y pair was in training set

Model inversion attack setting:

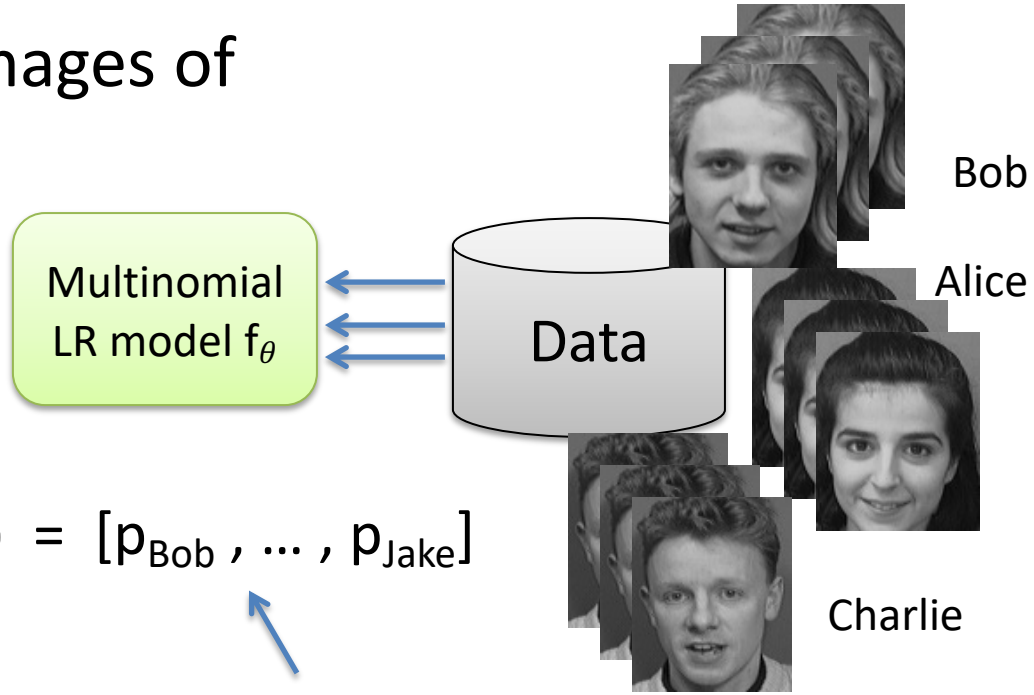
[Fredrikson, et al. 2014, 2015]

Training set entry x,y'



Model inversion and facial recognition

Can adversary use θ to recover images of training member's faces?



Approach (slightly simplified):

Given θ, y' = "Bob", find input x that is most likely to match "Bob"

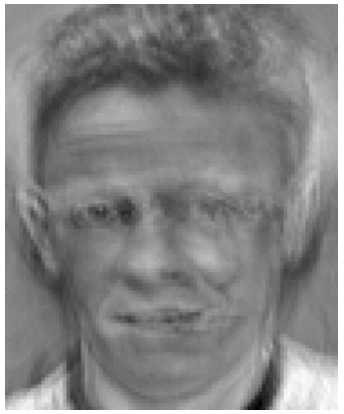
Search for x that maximizes p_{Bob}

Can search efficiently using gradient descent

Can repeat for all class labels

Model inversion and facial recognition

Target



Softmax

MLP

DAE

Trained on AT&T faces dataset (40 individuals, 400 images)

Inversion for three neural-network classifiers :

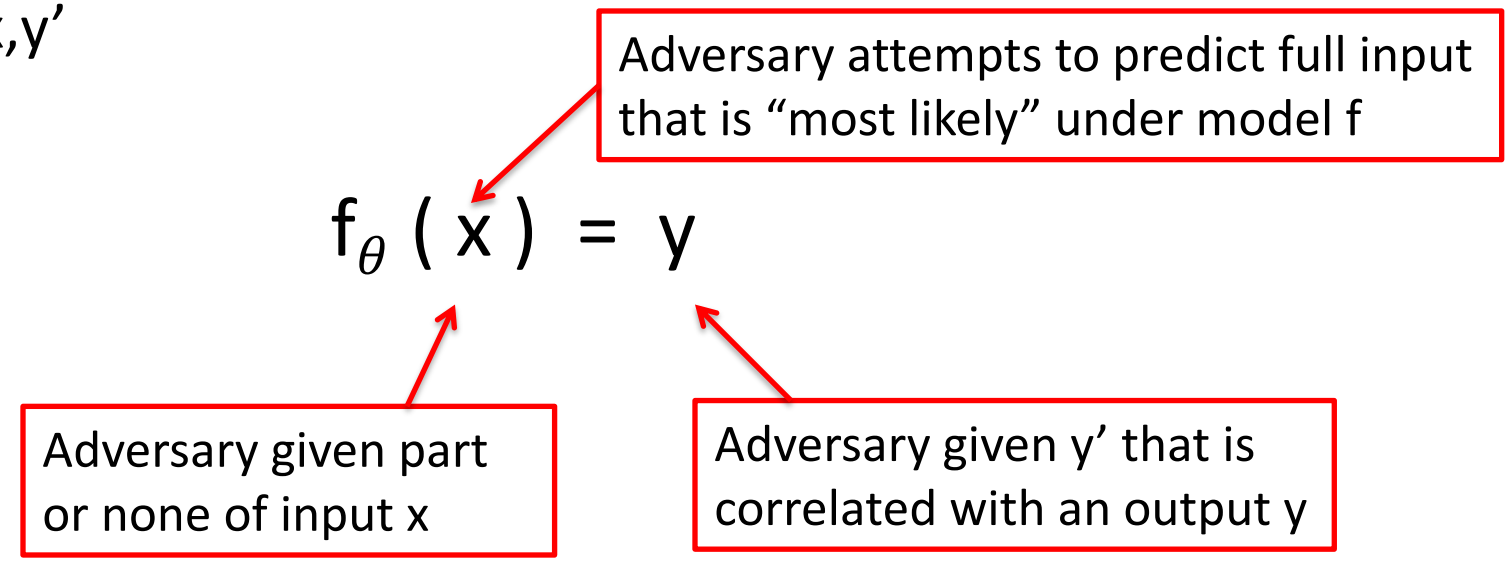
Multinomial LR, Multi-layer perceptron, Denoising auto-encoder

Mechanical Turk experiments: re-identify person up to 95% accuracy

Model inversion vs membership inference?

Model inversion attack setting: [Fredrikson, et al. 2014, 2015]

Training set entry x,y'



It could be that attacks work equally well for members of training set vs. not

What does this mean in how we think about privacy?

Disclosure risk versus membership privacy

- “A plausible notion of privacy, known in statistical disclosure control as the “Dalenius desideratum,” states that the model should reveal no more about the input to which it is applied than would have been known about this input without applying the model. This cannot be achieved by any useful model [14].”
- “Spooky action at a distance” point
- What do you think?

Next we'll go back in time to pre-ML privacy