ETH zürich UNIVERSITÄT ZU LÜBECK

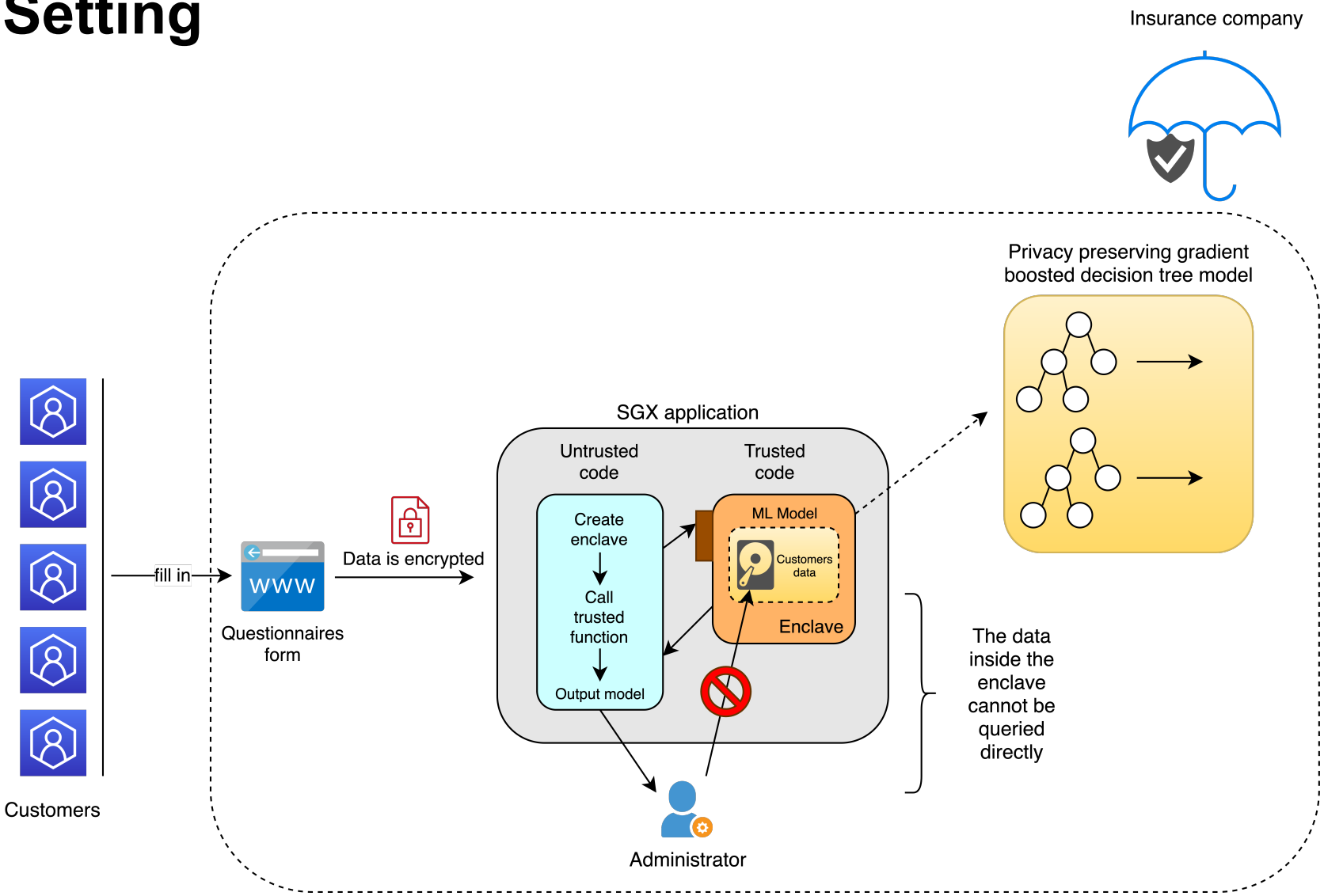
# Privacy-Preserving Machine Learning for Cyber Insurance

Theo Giovanna (ETH), Prof. Dr. Esfandiar Mohammadi (Lübeck), Dr. Kari Kostiainen (ETH)

# Motivation

- Insurance companies have come up with cyber insurance products
  - Evaluating cyber risk is hard
    - Use of questionnaires
  - Customers are not so willing to provide full details about their security practices
    - They need privacy guarantees
  - Lack of any historical data
    - About 10 real samples

# Example Setting



# Chosen Approach

- Differentially Private Gradient Boosted Decision Trees (DP-GBDT)
  - Great performance
  - Appropriate for low memory environments such as within an SGX's enclave
  - Easily explainable due to the decision trees' structure
  - Can work with little data (unlike e.g. DNN)
  - Provable privacy guarantees through differential privacy
  - Existing state of the art for DP-GBDT
    - AAAI - Qinbin et al. - Privacy-Preserving Gradient Boosting Decision Trees
    - https://arxiv.org/pdf/1911.04209.pdf

# Thesis Focus

- Shed some light on the cyber insurance problem
  - **Contribution 1.1:** Implemented state of the art DP-GBDT from the literature, and tuned it specifically to cyber insurance
  - **Contribution 1.2:** Proposed a new decision tree induction algorithm to better performances over small datasets

- Since historical data is lacking, generate synthetic data that mimics cyber insurance questionnaire's answers
  - **Contribution 2:** Proposed a synthetic data generation algorithm based on Bayesian networks

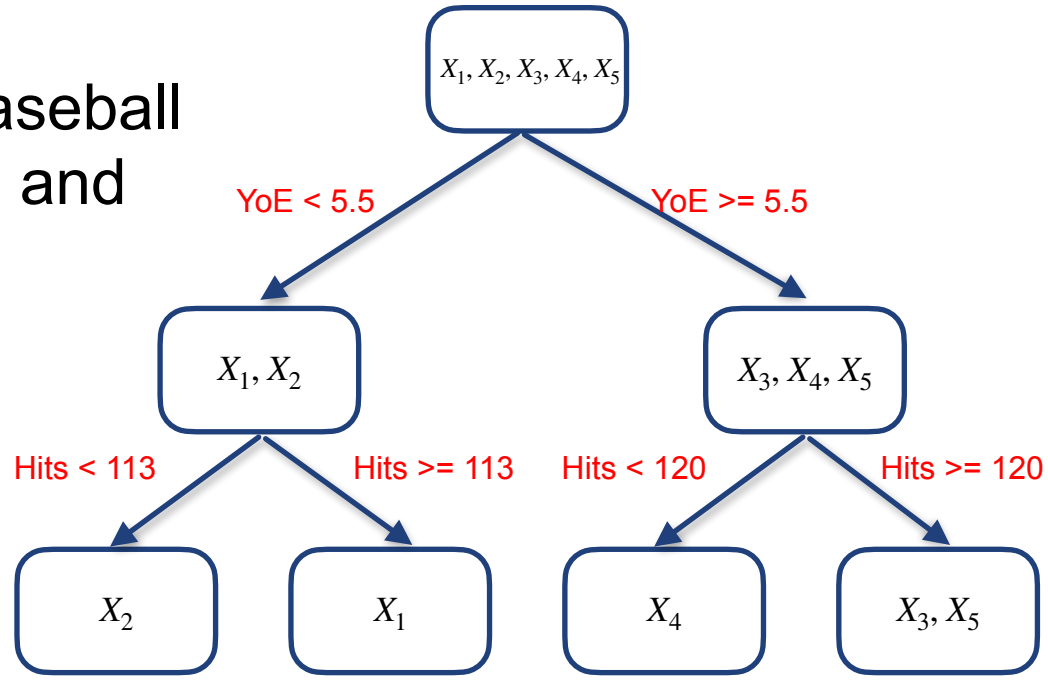
- Results:
  - **Contribution 3.1:** Accuracy evaluation
  - **Contribution 3.2:** Privacy evaluation

# Background

# Background - Decision Trees

- Decision Tree
  - Goal: given a set of inputs $X_1, \ldots, X_n$, predict a response or class $Y$ for an unseen input $X_i$
  - Example (adapted from RCS46): predict a baseball player's salary based on years of experience and number of hits

| | YoE | Hits | Salary |
|---|---|---|---|
| X_1 | 4 | 113 | 5.11 |
| X_2 | 2 | 90 | 4.5 |
| X_3 | 7 | 162 | 7.9 |
| X_4 | 5.5 | 105 | 6.1 |
| X_5 | 6 | 120 | 6.5 |



# Background - Decision Trees

- Problems: decision trees are
  - Prone to overfitting
  - Can get complex when there's many class labels or when the tree depth is very large
  - A change in the training set might result in a complete different decision tree

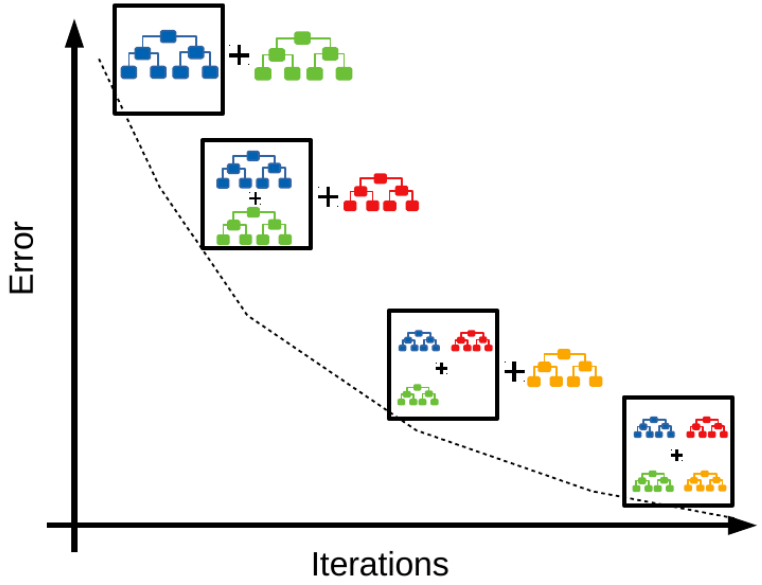
# Background - GBDT

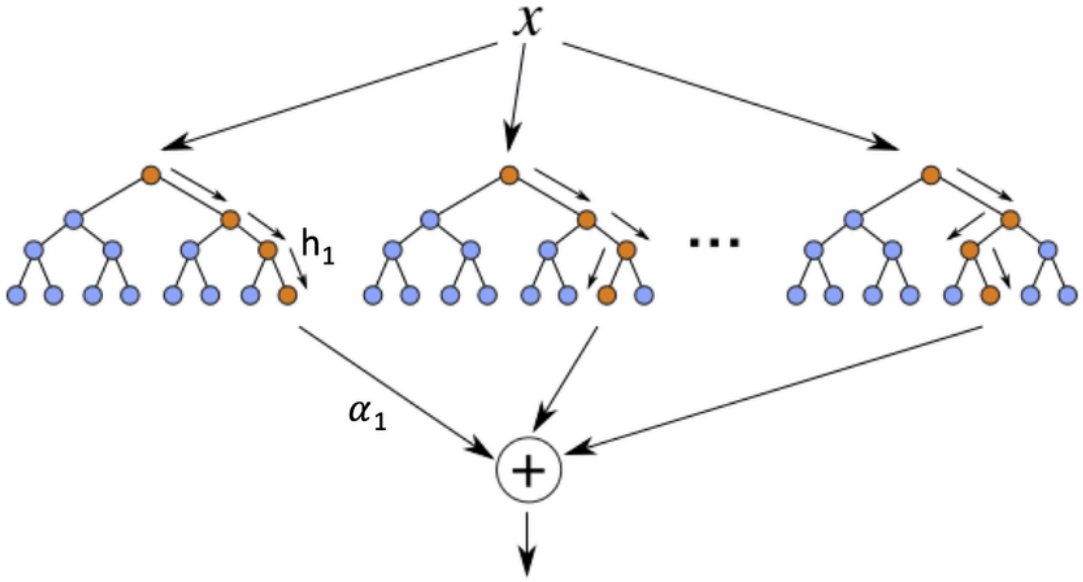
- Solution: Gradient Boosted Decision Trees
  - *Ensemble* method: combine many weak learners (here, many simple decision trees) to build the final model - more trees allow for error correction
  - Each tree is fit on the residual of the previous tree

- Trees stop growing when termination criteria is reached
  - Depth
  - Gain
  - Number of samples in node

# Background - GBDT

- Each new tree helps in reducing the overall error
- Use all trees to make final prediction



# Background - Differential Privacy

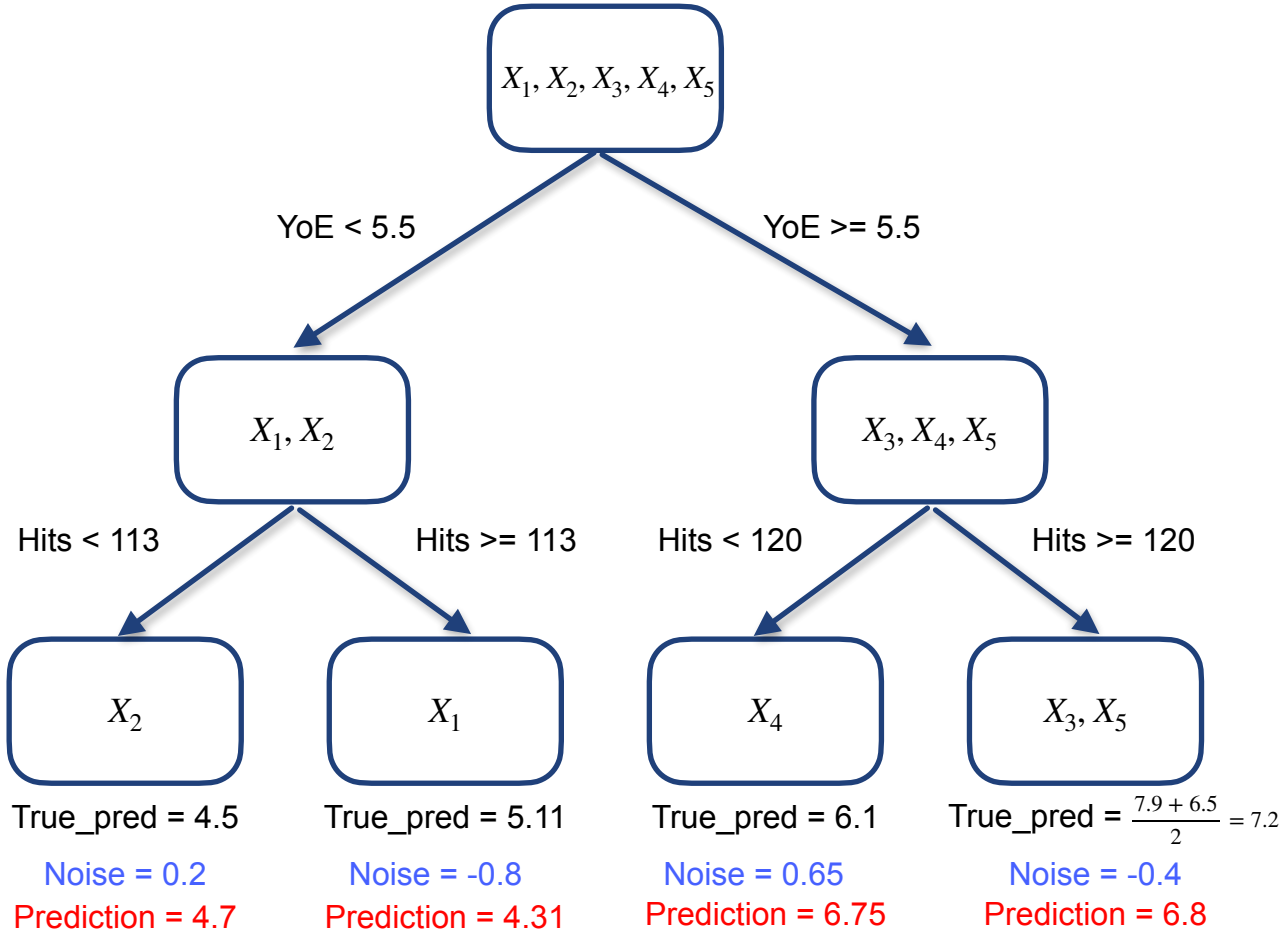
**Definition 1.** ($\varepsilon$-Differential Privacy) Let $\varepsilon$ be a positive real number and $f$ be a randomized function. The function $f$ is said to provide $\varepsilon$-differential privacy if, for any two datasets $D$ and $D'$ that differ in a single record and any output $O$ of function $f$,

$$Pr[f(D) \in O] \leq e^{\varepsilon} \cdot Pr[f(D') \in O] \qquad (5)$$

- The ratio between the output probability distributions for $D$ and $D'$ is strictly bounded by $e^{\epsilon}$
- More intuitively, we aim to learn about a dataset, yet don't learn anything about a particular datapoint
- The influence of every single data point on the learned model is *deniable*
- Privacy 'level' controlled by *privacy budget* $\epsilon$

# Background - Differential Privacy

- Applying this to GBDTs: select the attribute and attribute's value to split on with the exponential mechanism:
  - Essentially, the exponential mechanism is a randomised arg_max which outputs high gain splits with high probability
- Once the termination criteria is reached within a branch, draw some random noise from a Laplacian distribution, and add it to the leaf node's prediction



# Background - Differential Privacy

- Challenges:
  - How much noise is too little? Too much?
  - How to design models that query the data as less as possible while learning sufficient information
    - Each query consumes some of the privacy budget
  - The noise should not outweigh the data itself, which happens when the privacy budget is too low or when there's too little data

# Contribution 1: Implementing and Improving DP-GBDT

# Implementing DP-GBDT (Qinbin et al., 2020)

**Algorithm 2:** Differentially private GBDTs training process [25]

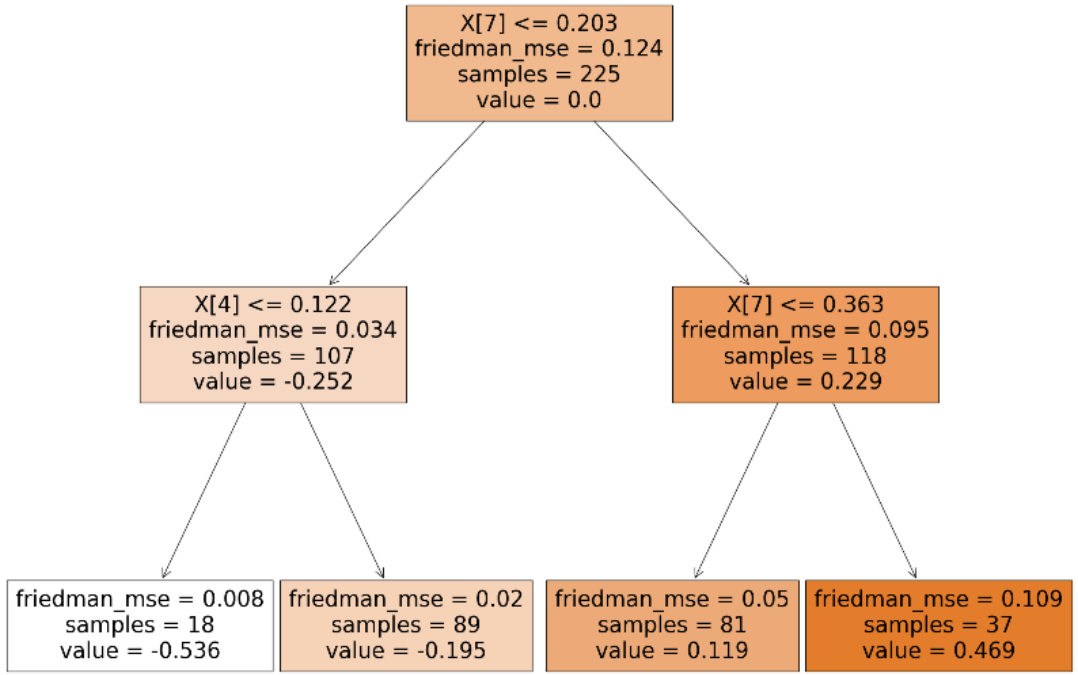
**Input:** $\mathbf{X} = X_1, \ldots, X_n$: instances, $\mathbf{y} = y_1, \ldots, y_n$: labels
**Input:** $\lambda$: regularisation parameter, $d_{max}$: maximum depth, $\eta$: learning rate
**Input:** $T$: total number of trees, $l$: loss function, $\varepsilon$: privacy budget
**Output:** An ensemble of trained differentially private decision trees.

$\varepsilon_t = \varepsilon$ ▷ *Each tree is trained on a disjoint subset of the dataset, so we can apply Theorem 2.8*
**for** $t = 1$ ***to*** $T$ **do**
  Update gradients of all training instances on loss $l$
  $\varepsilon_{leaf} = \frac{\varepsilon_t}{2}$, $\varepsilon_{node} = \frac{\varepsilon_t}{2d_{max}}$
  **for** $d = 1$ ***to*** $d_{max}$ **do**
    **for** *each node in current depth* **do**
      **for** *each split value* $i$ **do**
        $G_i \leftarrow \frac{(\sum_{i \in I_L} g_i)^2}{|I_L|+\lambda} + \frac{(\sum_{i \in I_R} g_i)^2}{|I_R|+\lambda}$ ▷ *Equation 2.3*
        $P_i \leftarrow \exp(\frac{\varepsilon_{node} \cdot G_i}{2\Delta G})$ ▷ *Theorem 2.6*
      Split node on split value $i$, where $i$ is chosen with probability $P_i / \sum_j P_j$
  **for** *each leaf node* $i$ **do**
    $V_i \leftarrow \eta \left( -\frac{\sum_{i \in I} g_i}{|I|+\lambda} + Lap(0, \Delta V / \varepsilon_{leaf}) \right)$ ▷ *Equation 2.4 and Theorem 2.5*

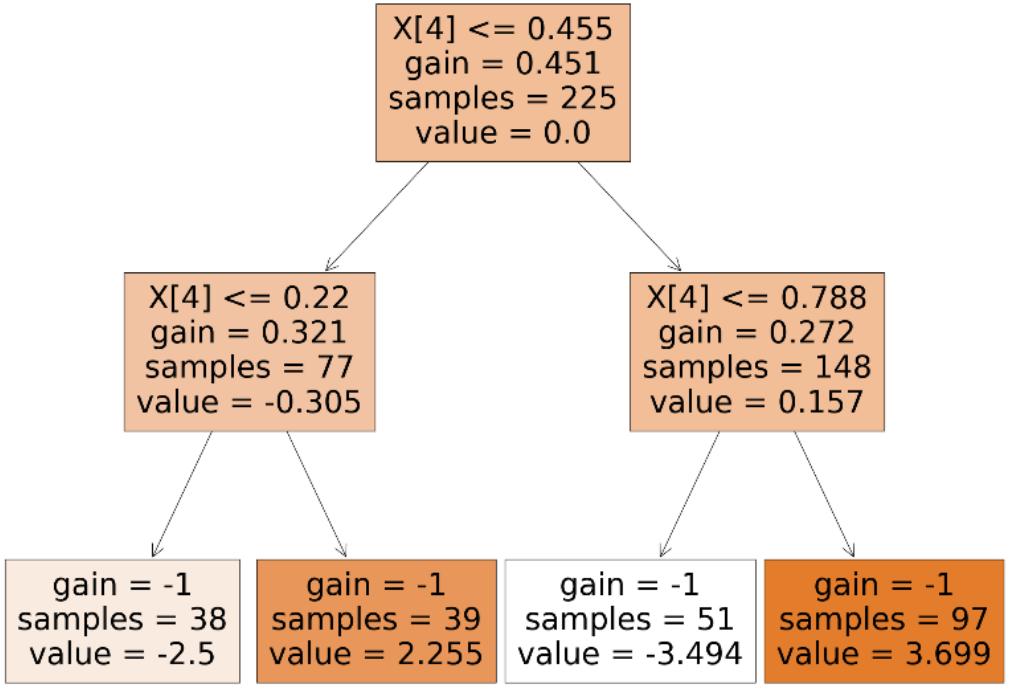
- Additionally:
  - Gradient Based Data Filtering
  - Geometric Leaf Clipping

# Impact of DP

Dataset: abalone, samples = 300



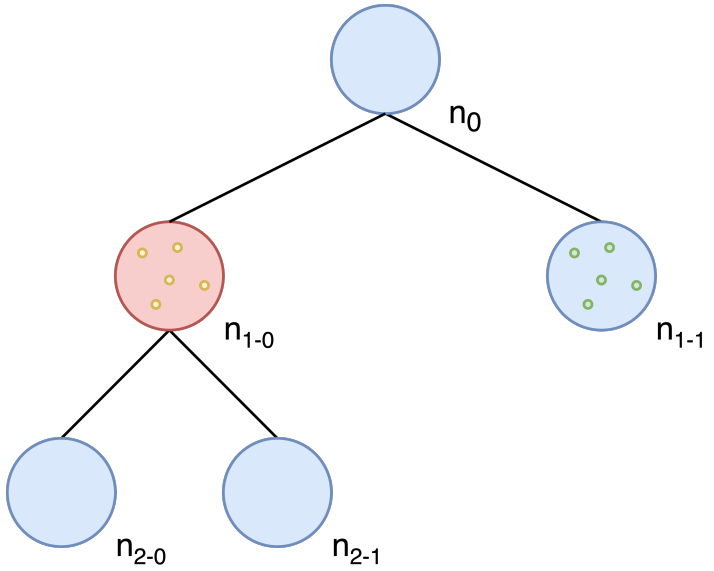

Vanilla

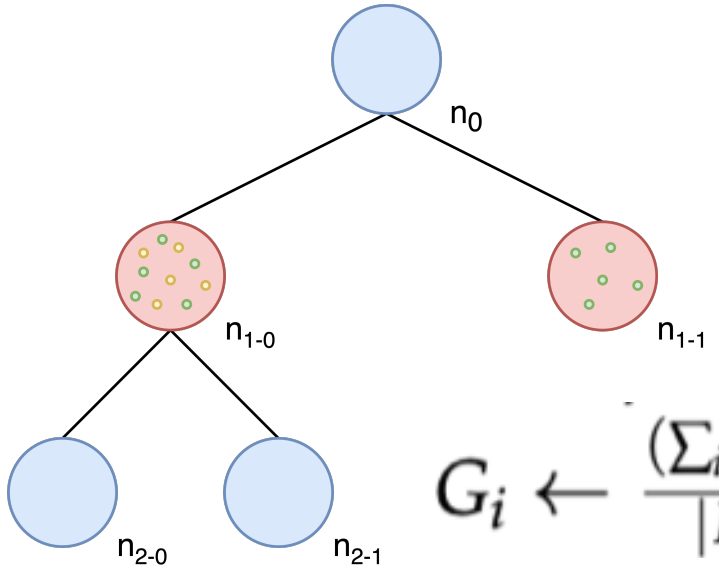



Differentially private

# Improving DP-GBDT

- How can we enhance the quality of the splits when there isn't much data available?
  - Look for more data!
    - '2-nodes' algorithm: instead of splitting a node based on its samples only, split it based on its samples + the ones from its sibling

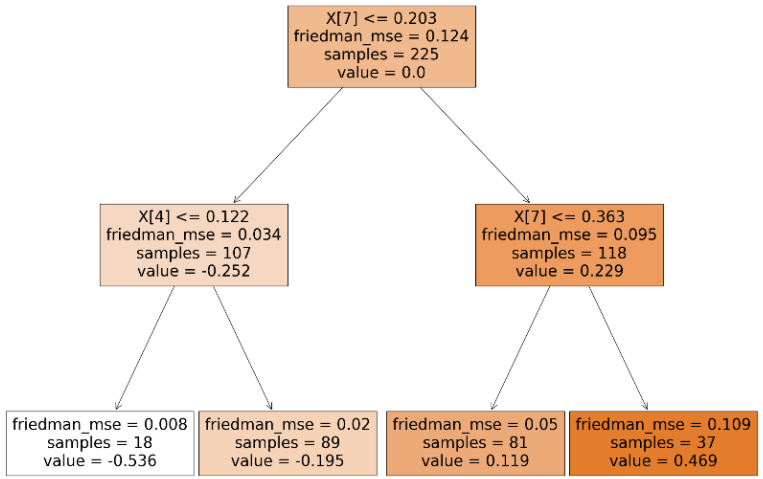


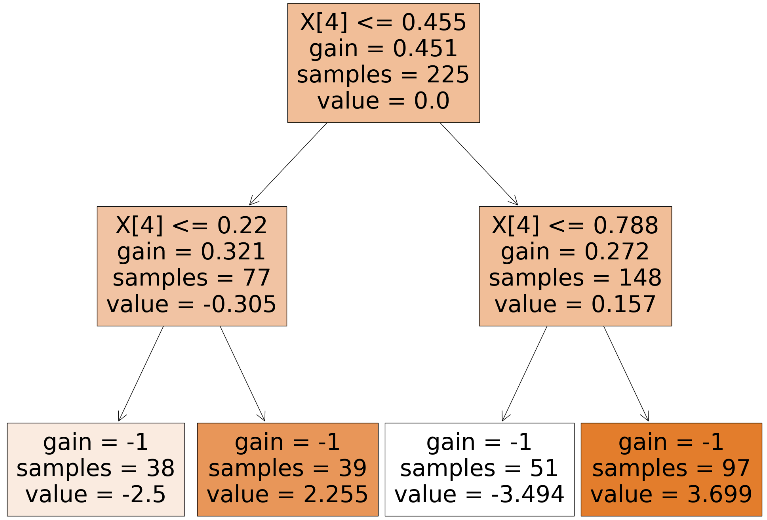

$$G_i \leftarrow \frac{(\sum_{i \in I_L} g_i)^2}{|I_L| + \lambda} + \frac{(\sum_{i \in I_R} g_i)^2}{|I_R| + \lambda}$$

# Improving DP-GBDT

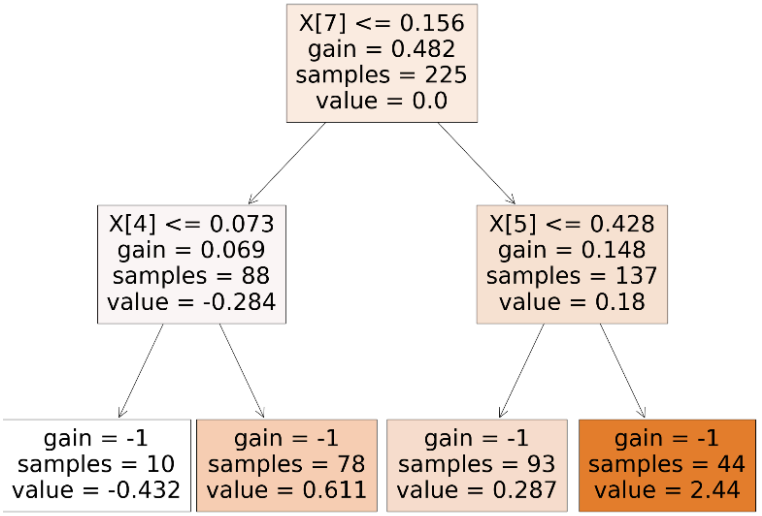
Dataset: abalone, samples = 300




Vanilla




Differentially private (e=1)




Differentially private - 2-nodes (e=1)

- What to look for:
  - Attribute selected for the split
  - Leaf value

# Improving DP-GBDT

- Leaf values:


Vanilla


Differentially private (e=1)

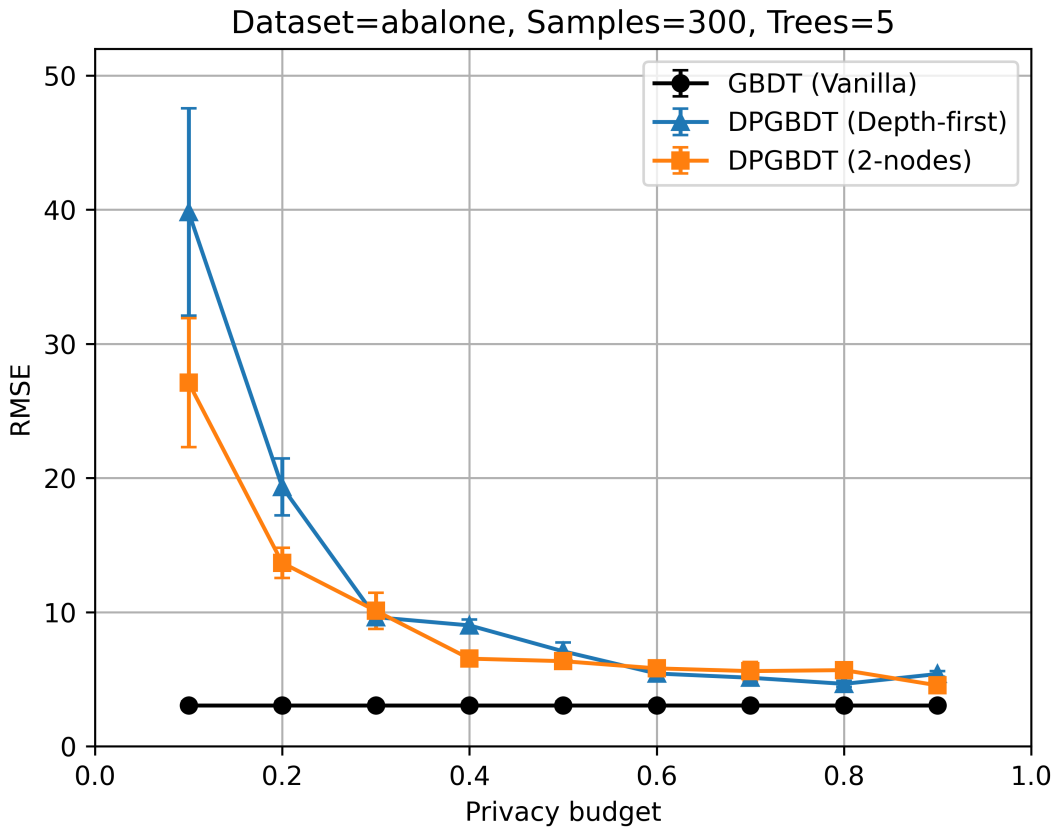

Differentially private - 2-nodes (e=1)

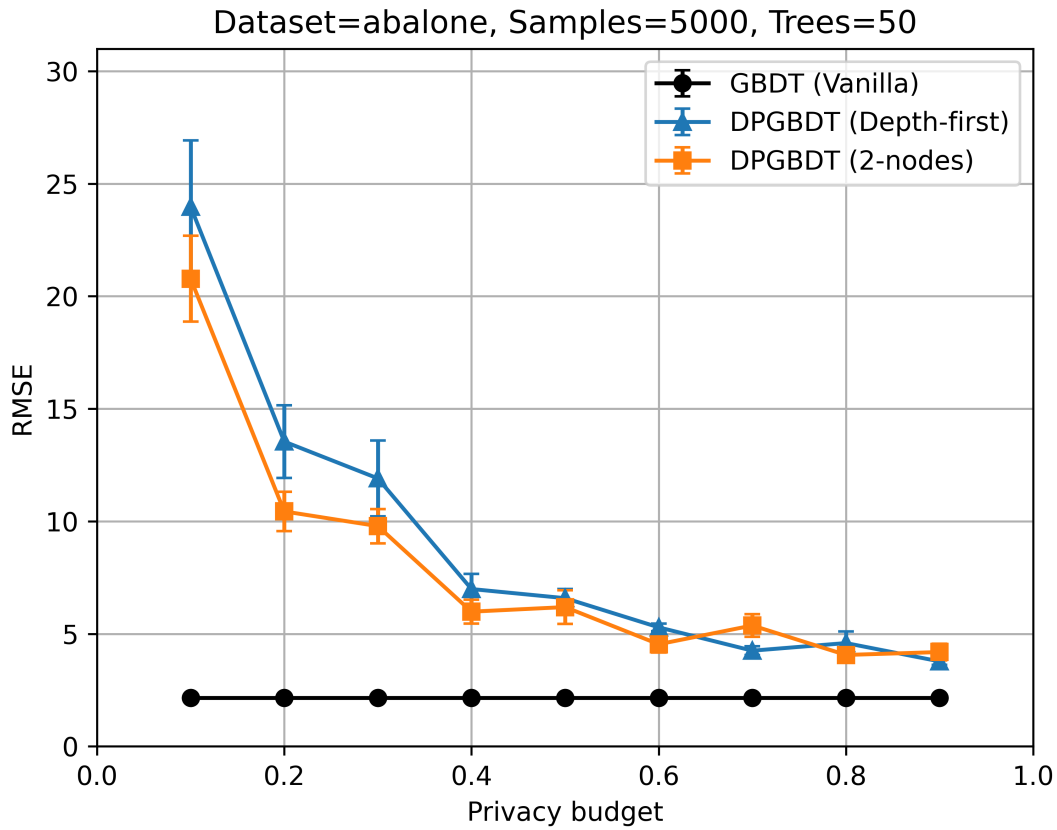
# Improving DP-GBDT

- Why not look into more nodes?
  - More data querying, privacy budget would be consumed too rapidly
  - Nodes in other sub-trees may share complete different attributes compared to the direct sibling
    - Until that point in the tree, the node and its sibling share identical features

- Is it still differentially private?
  - We query the data twice as much compared to the original implementation, hence we only need to adapt our privacy budget consumption accordingly

# Improving DP-GBDT



# Contribution 2: Synthetic Data Generation

# Synthetic Data Generation

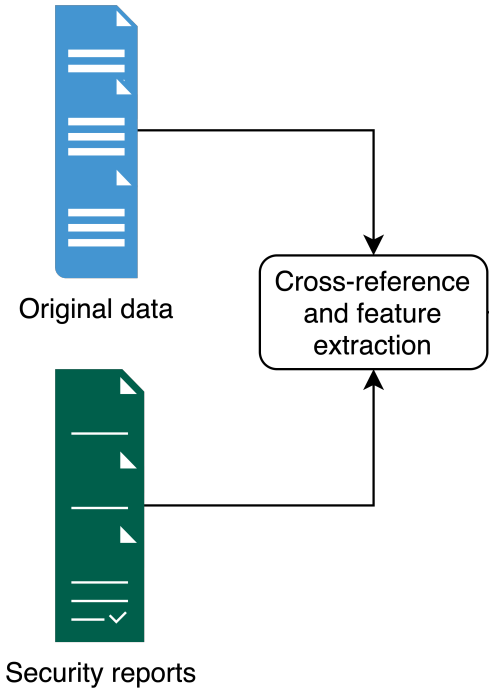
- Real data only available in small amount, so we have to extrapolate them
  - 10 samples provided
  - Several thousands needed to evaluate the model

# Synthetic Data Generation

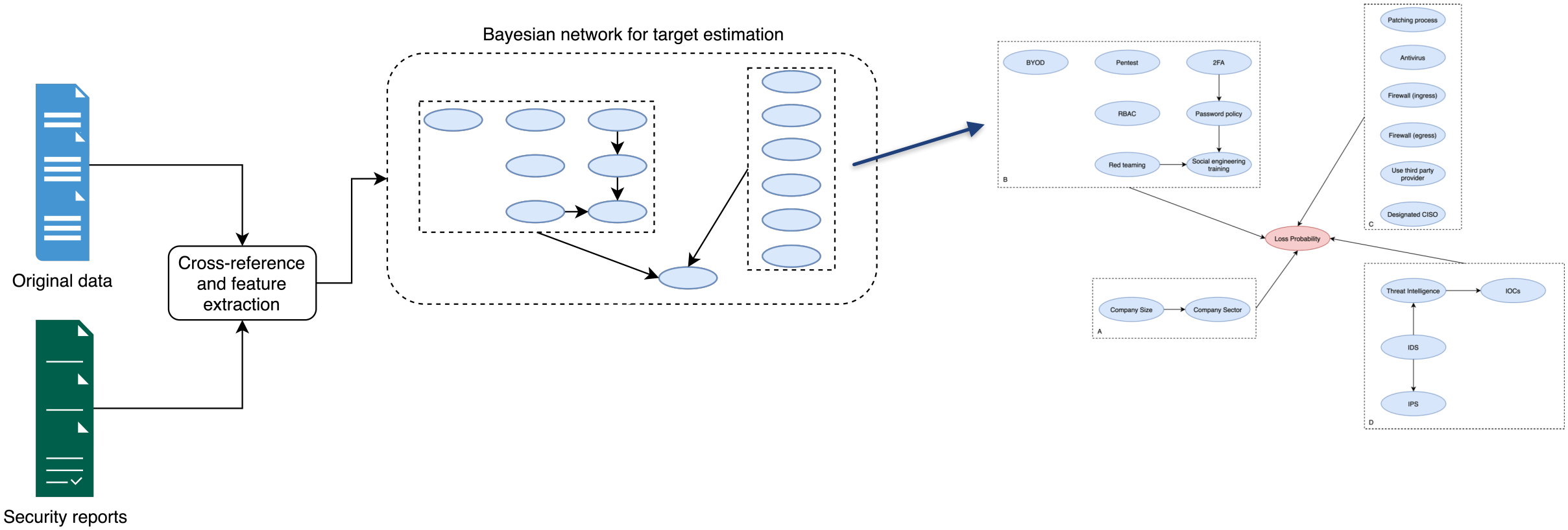
- Real data only available in small amount, so we have to extrapolate them
  - 10 samples provided
  - Several thousands needed to evaluate the model

- Goal: build a dataset which:
  - features mimic questionnaire answers
  - targets mimic
    - 1) the probability for a customer of suffering a loss
    - 2) the potential cost involved if a customer has a security incident
  - is sufficiently complex for evaluation

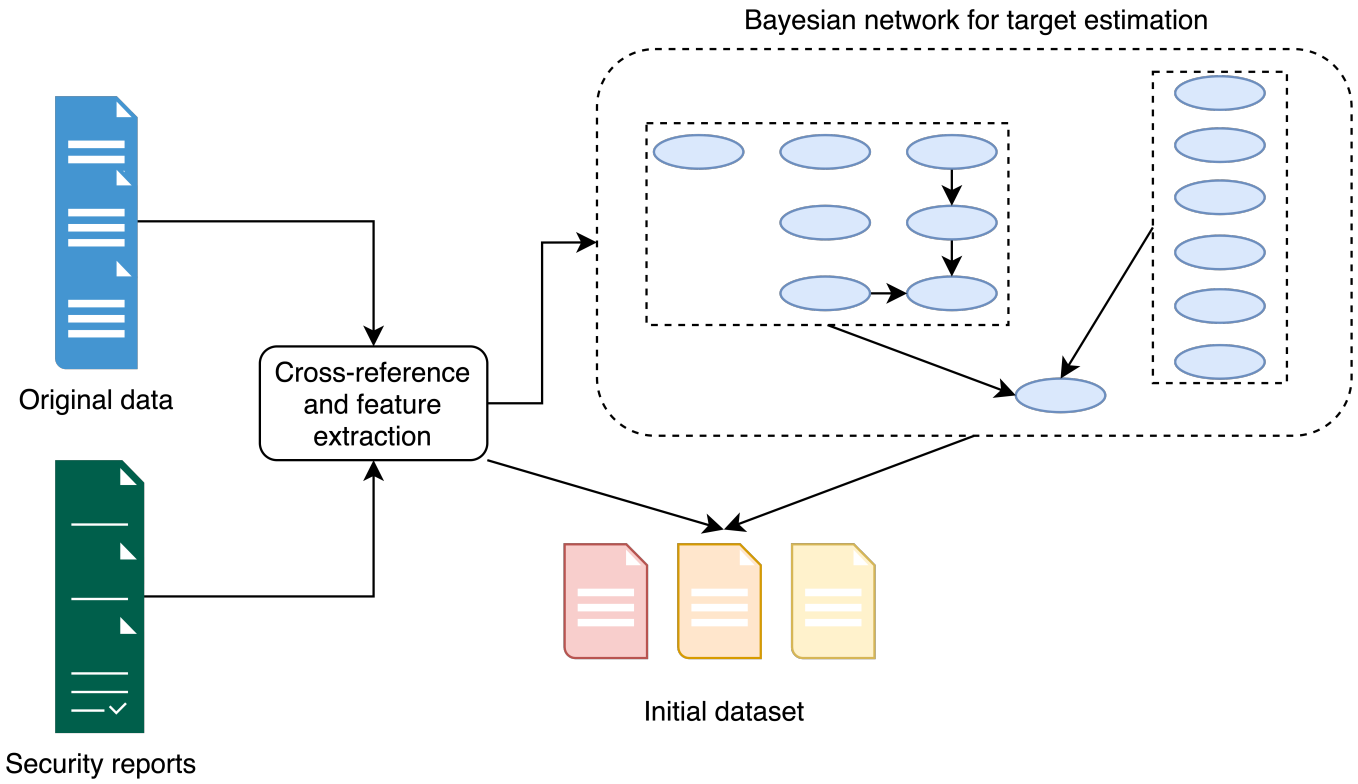
# Synthetic Data Generation



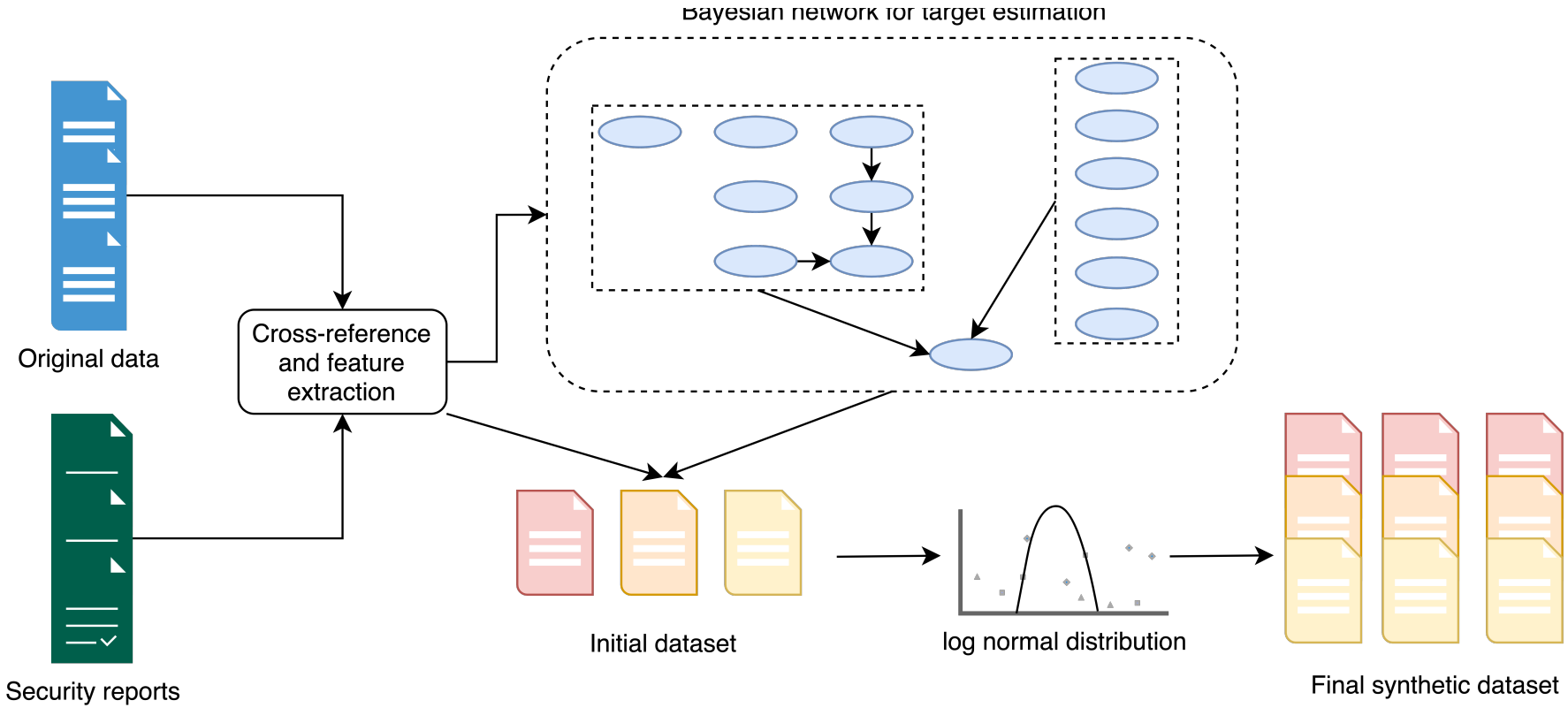
# Synthetic Data Generation



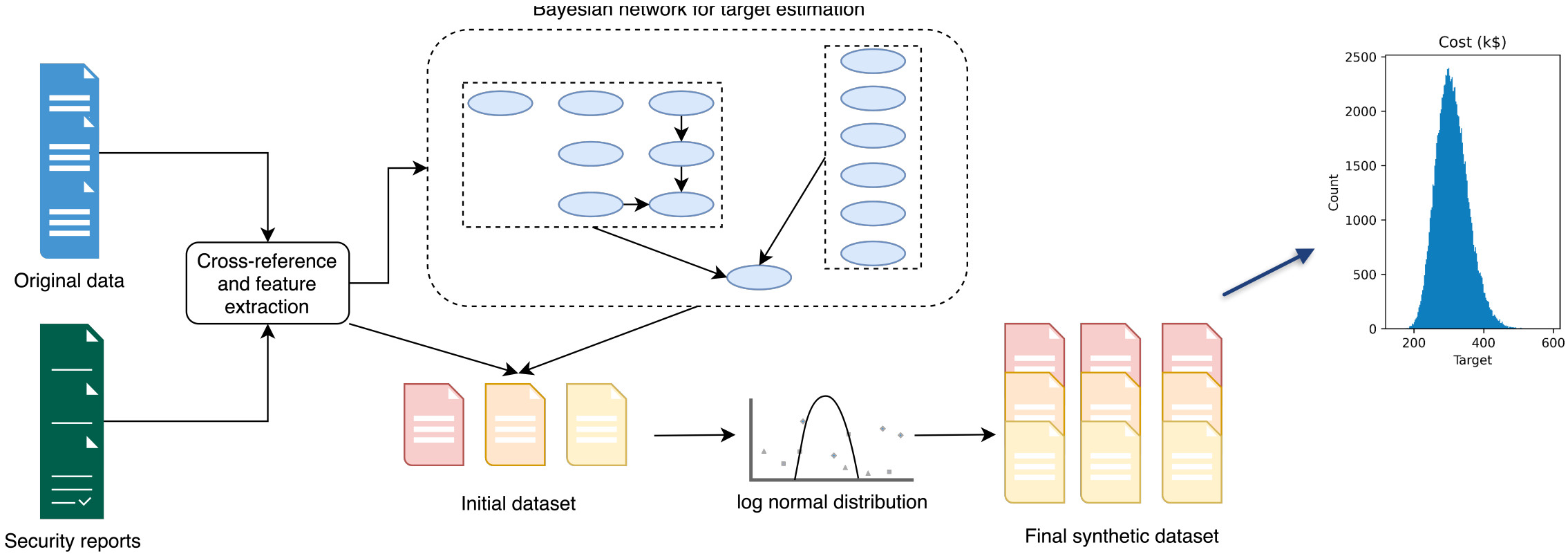
# Synthetic Data Generation



# Synthetic Data Generation



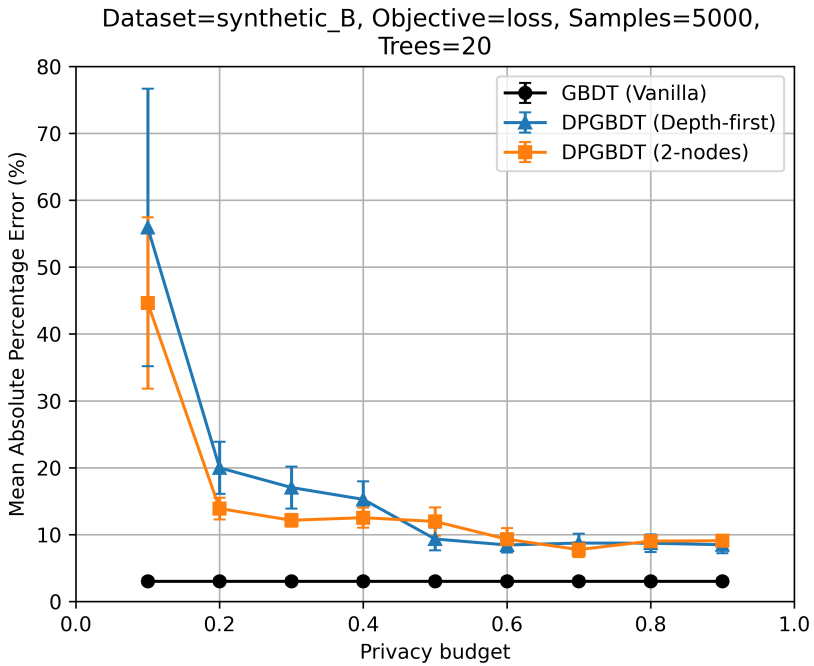
# Synthetic Data Generation

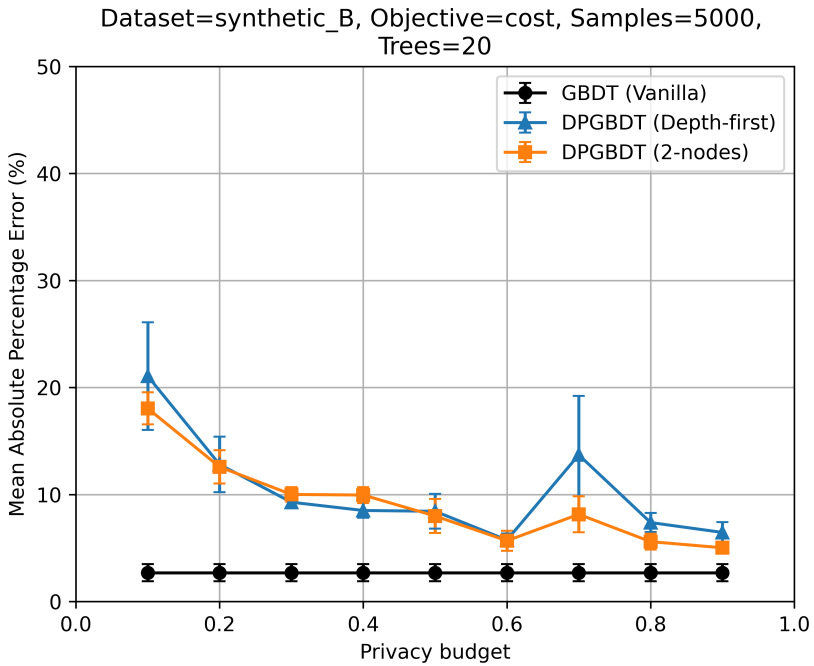


# Contribution 3: Accuracy and Privacy Evaluation

# Evaluation

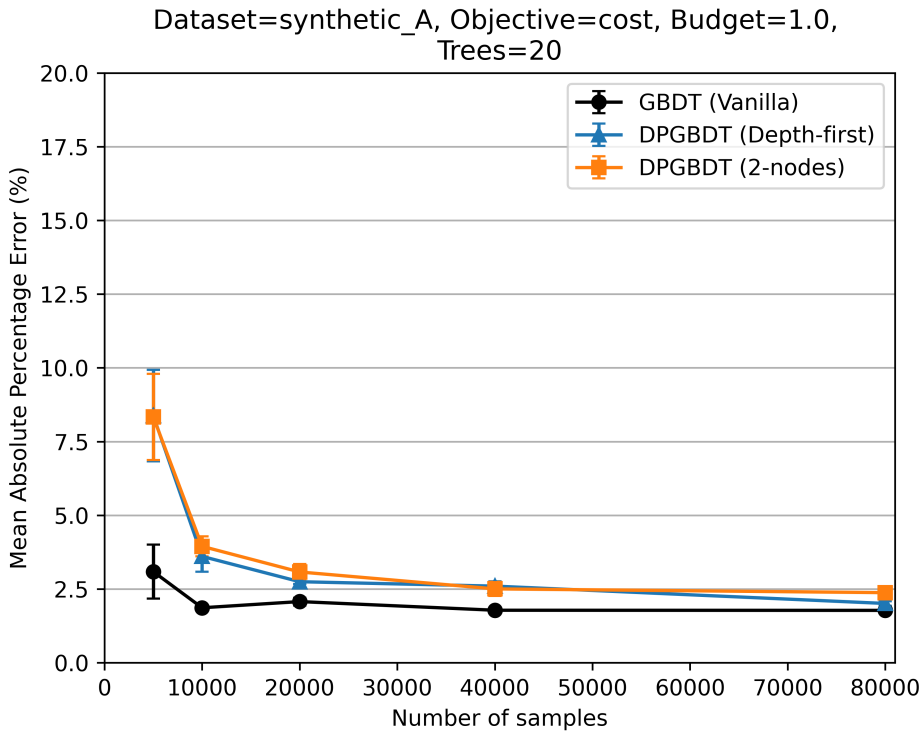
- For the evaluation, we report the Mean Average Percentage Error (MAPE) for each target.
  - E.g.: if the true value is a 100, and the predicted value is 110, then the MAPE score would be 10%.

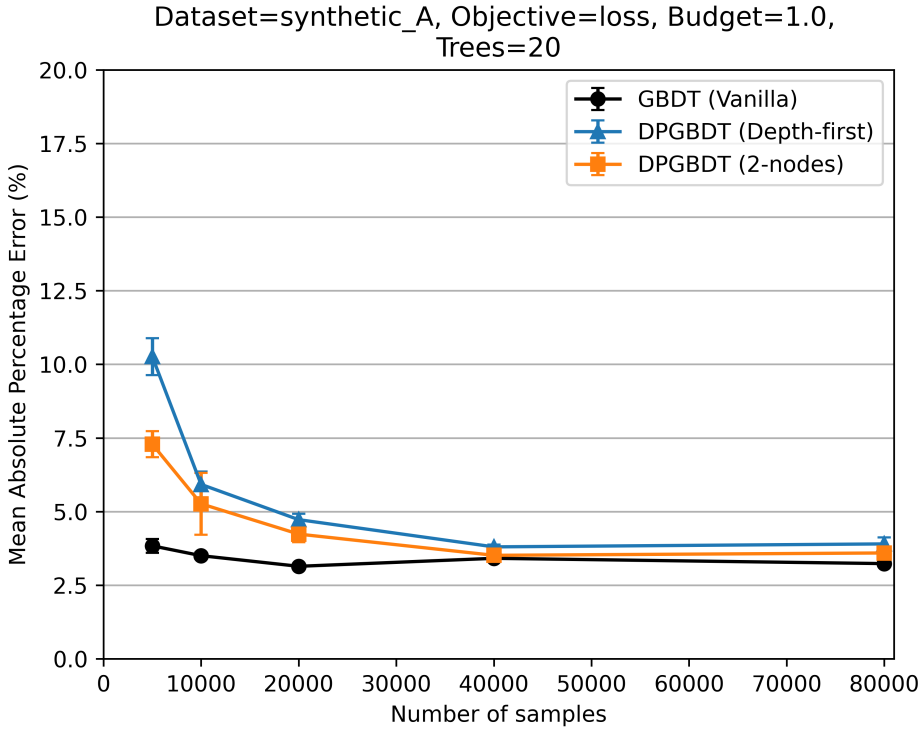


# Evaluation

- The more samples, the better the model can approach non-DP results

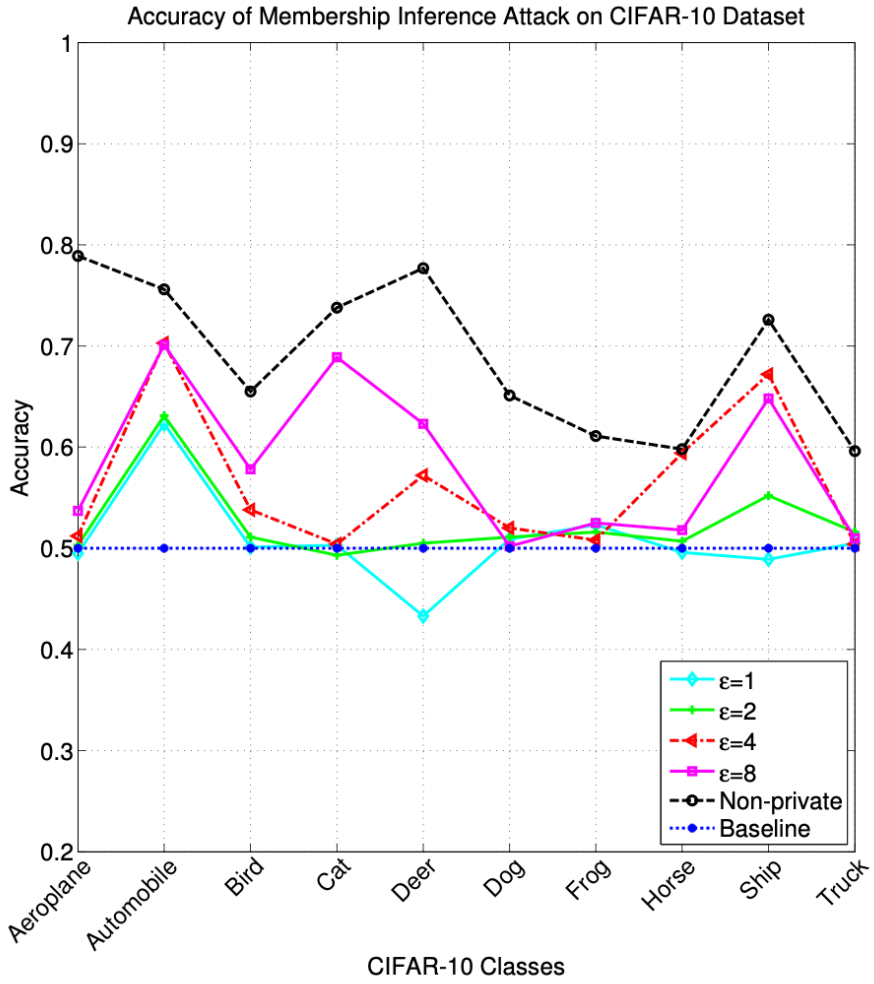


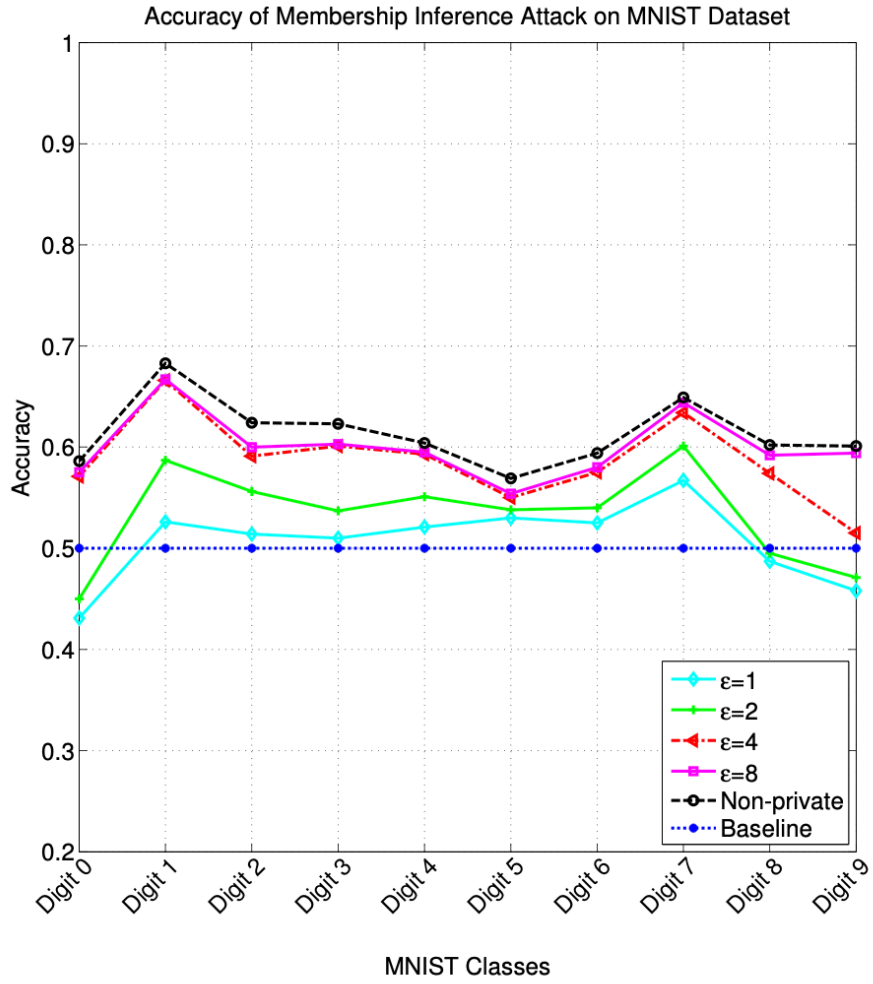
# Privacy Attacks

- Example attacks:
  - Model inversion attack: attacker tries to extract data points from the model
    - Very hard
  - **Membership Inference Attack (MIA)**: attacker tries to identify if a given instance was part of the training set or not
    - Easier from the attacker's perspective, harder to defend against from the defender's perspective
    - We want to achieve 50% attack accuracy (i.e. random guess, no effectiveness)

# Privacy Attacks




(a)




(b)

[1] Membership Inference Attack against Differentially Private Deep Learning Model

# Conclusion

- We showed that state of the art DP can be applied to a specific business problem and provide good results
  - Accuracy enhanced for small datasets
- Trade-off accuracy vs. size of the dataset and privacy constraints
- There's a clear need for proper privacy attacks that target regression models

# Future Work & Research Direction

- How to deploy such system
  - Retraining the model will ultimately break privacy, as querying the data again is a budget-consuming task
  - Enclave needs to be data-aware to block re-training
  - SGX side-channel resilience

- How to properly evaluate the model's privacy preserving features
  - Develop new attacks. Membership inference might be too strong for Zurich, but perhaps other attacks such as attribute inference attack could be developed

- GDF raises questions
  - Bug in Qinbin et al.'s proof found post-submission
  - Preliminary research indicates that this could be fixed with a cost of $\epsilon * 1.5$

# Q&A

# Appendix - Feature extraction

| Feature ($F$) | $P(F)$ | $P(F\|loss)$ |
|---|---|---|
| *company_size* | small: 0.68<br>large: 0.32 | |
| *company_sector* | See Table 5.1 | |
| *company_allows_BYOD* | 0.44 | 0.14 |
| *company_does_pentest* | 0.36 | 0.23 |
| *company_uses_RBAC* | 0.85 | 0.67 |
| *company_does_red_teaming* | 0.33 | |
| *company_has_2FA* | 0.78 | |
| *company_has_password_policy* | 0.88 | |
| *company_trains_employees_against_social_engineering* | 0.70 | |
| *company_has_antivirus* | 0.95 | 0.81 |
| *company_has_patching_process* | 0.60 | 0.828 |
| *company_has_ingress_firewall* | 0.83 | 0.84 |
| *company_has_egress_firewall* | 0.61 | 0.76 |
| *company_uses_3rd_party* | 0.87 | 0.79 |
| *company_has_ciso* | 0.67 | 0.68 |
| *company_has_threat_intelligence_team* | 0.38 | |
| *company_monitors_IOCs* | 0.44 | |
| *company_has_IDS* | 0.94 | |
| *company_has_IPS* | 0.81 | |
| *company_separates_systems* | 0.79 | |
| *company_does_daily_backup* | 0.91 | |
| *company_has_recovery_plan* | 0.33 | |
| *company_has_incident_response_team* | 0.46 | |

| Feature ($F$) | $Pr(F)$ | $Pr(F\|loss)$ | $Pr(F\|SME)$ | $Pr(F\|large)$ |
|---|---|---|---|---|
| *company_is_in_hospitality* | | | 0.03 | 0.04 |
| *company_is_a_public_entity* | | | 0.03 | |
| *company_is_in_other* | | | 0.11 | 0.09 |
| *company_is_a_nonprofit* | | | 0.05 | |
| *company_is_in_education* | | | 0.05 | 0.08 |
| *company_is_in_technology* | | | 0.06 | 0.03 |
| *company_is_in_manufacturing* | | | 0.08 | 0.03 |
| *company_is_in_financial_services* | | | 0.09 | 0.15 |
| *company_is_in_energy* | | | | 0.04 |
| *company_is_in_retail* | | | 0.09 | 0.24 |
| *company_is_in_healthcare* | | | 0.19 | 0.26 |

| Sector ($S$) | $Pr(PII\|S)$ | $Pr(PCI\|S)$ | $Pr(PHI\|S)$ | $Pr(credentials\|S)$ | $Pr(other\|S)$ |
|---|---|---|---|---|---|
| *Hospitality* | 0.44 | | | 0.34 | 0.23 |
| *Public Entity* | 0.51 | | | 0.33 | 0.34 |
| *Other* | 0.81 | | | 0.36 | 0.42 |
| *Education* | 0.75 | | | 0.30 | 0.23 |
| *Technology* | 0.69 | | | 0.41 | 0.34 |
| *Manufacturing* | 0.49 | 0.20 | | 0.55 | 0.25 |
| *Financial Services* | 0.77 | 0.32 | | 0.35 | 0.35 |
| *Energy* | 0.41 | 0.68 | | 0.41 | 0.35 |
| *Retail* | 0.49 | 0.47 | | 0.27 | 0.25 |
| *Healthcare* | 0.77 | | 0.67 | 0.18 | 0.18 |
| *Professional Services* | 0.75 | | | 0.45 | 0.32 |

# Appendix - Privacy Attacks

- Problems:
  - Current research work only focuses on classification models
    - Doesn’t apply to our task
    - Doesn’t apply to our model
  - Adapting the attacks to regression models is non-trivial
    - They rely on the class-probability vector outputted by classification models, which is lacking in regression decision trees
    - Some assumptions about the attacker might not hold
  - We can only look at the impact of DP in classification models from the literature