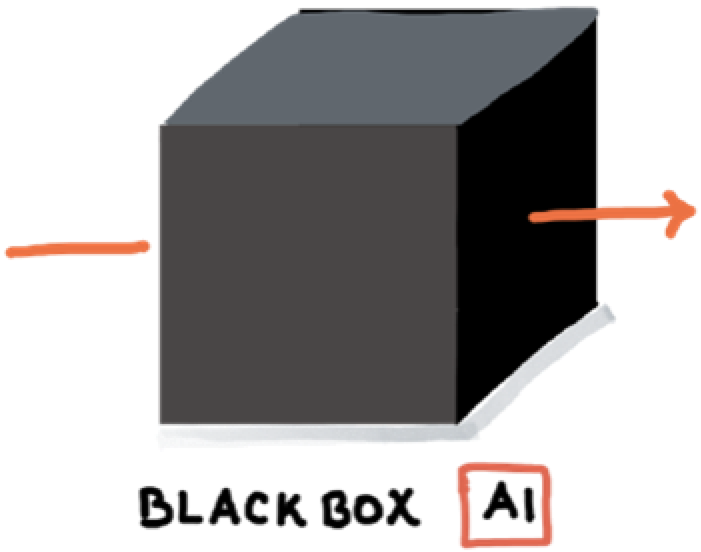
Privacy, Explainable AI

Francesca Naretto, Anna Monreale
University of Pisa
francesca.naretto@unipi.it

Context



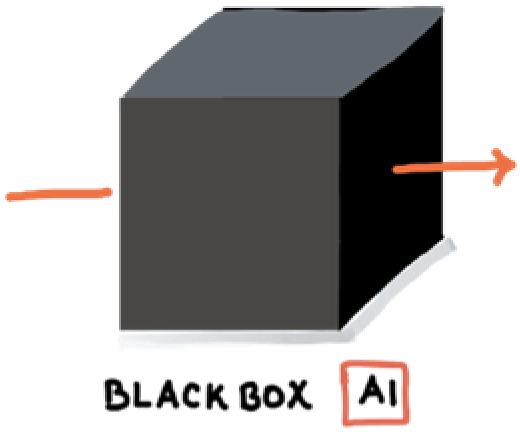
We want to explain the global behavior by using ***Global Explainers***

We want to explain the decision on a, instance by using ***Local Explainers***

A ***Machine Learning*** model whose internals are either ***unknown*** to the observer or they are known but **uninterpretable** by humans.

Potential Risk

Machine Learning models may enable privacy risks

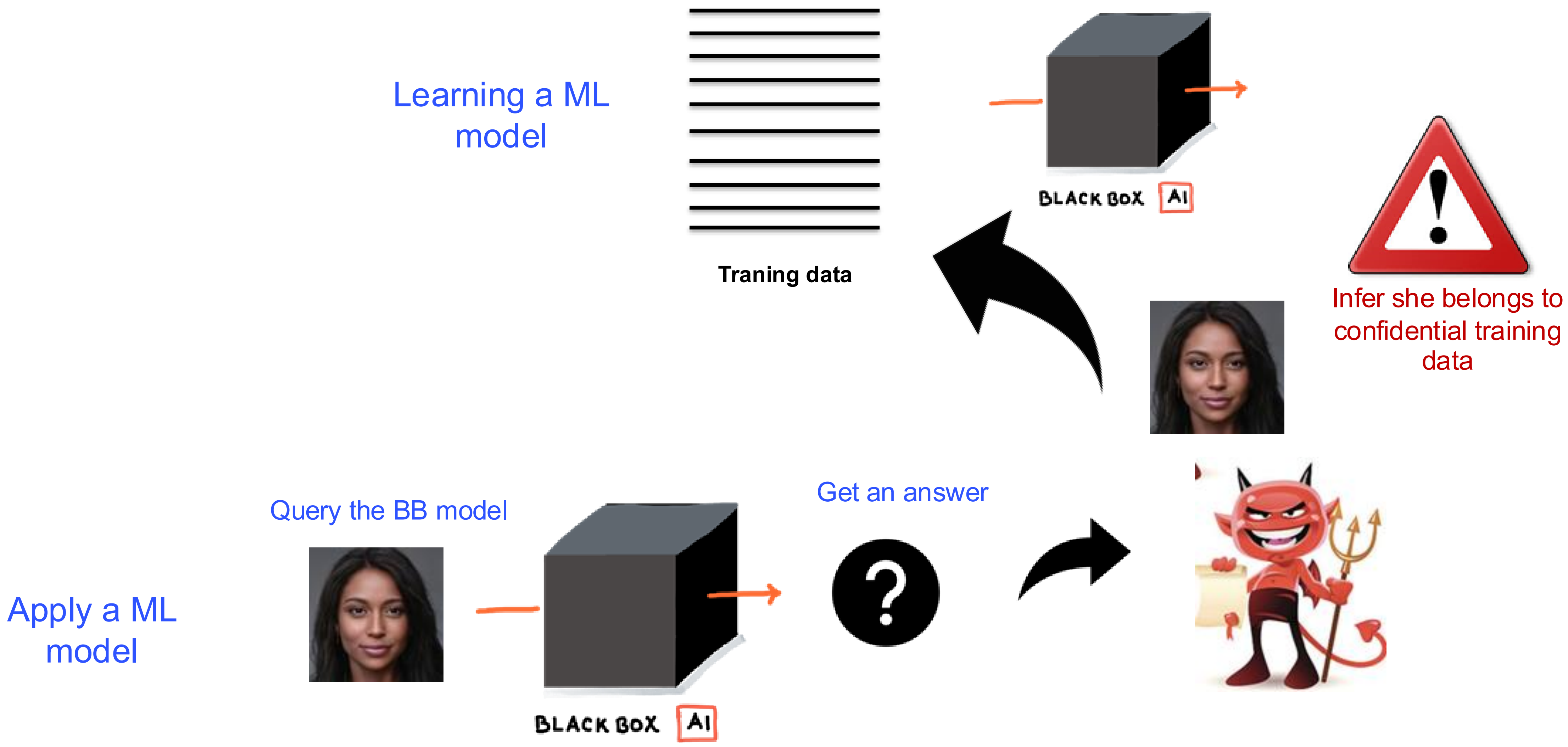


There are several privacy attacks design to attack Machine Learning models, such as:

1. Membership Inference attack
2. Reconstruction attack
3. Property inference attack

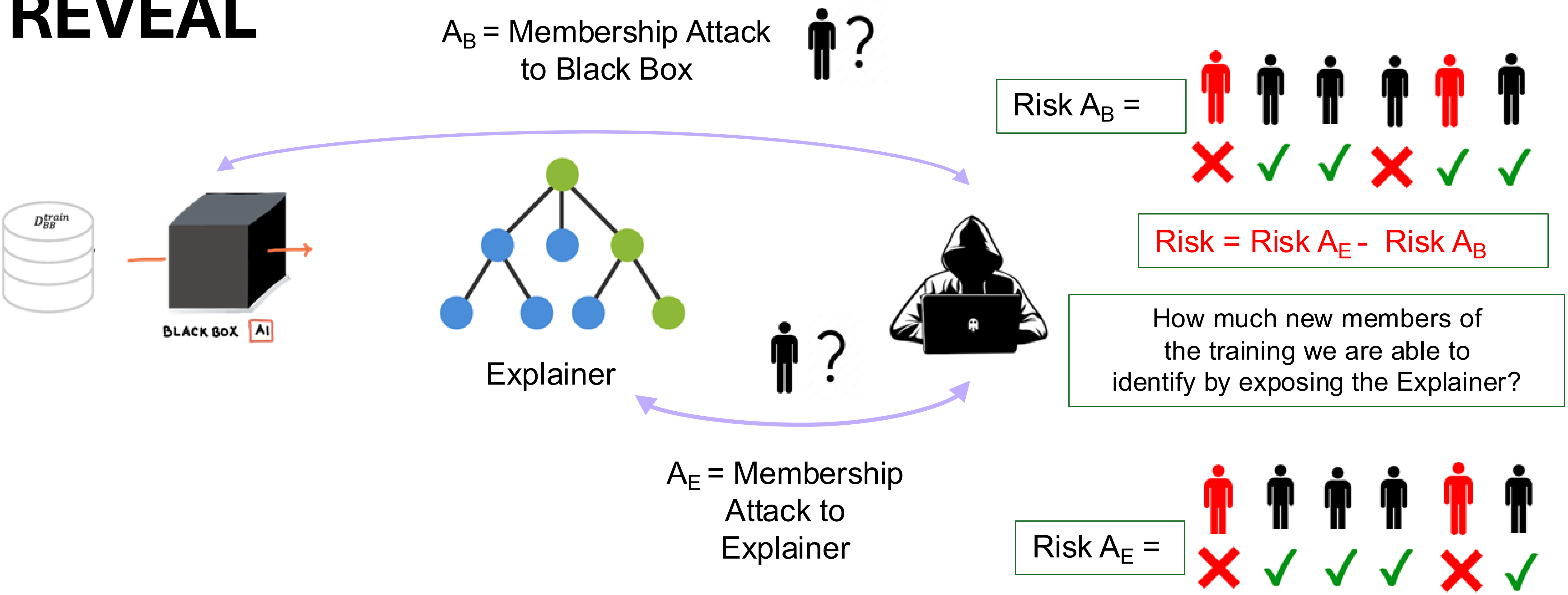
Explainers are themselves Machine Learning models, even if interpretable!
So, they can also be attacked.

Privacy Risk Assessment



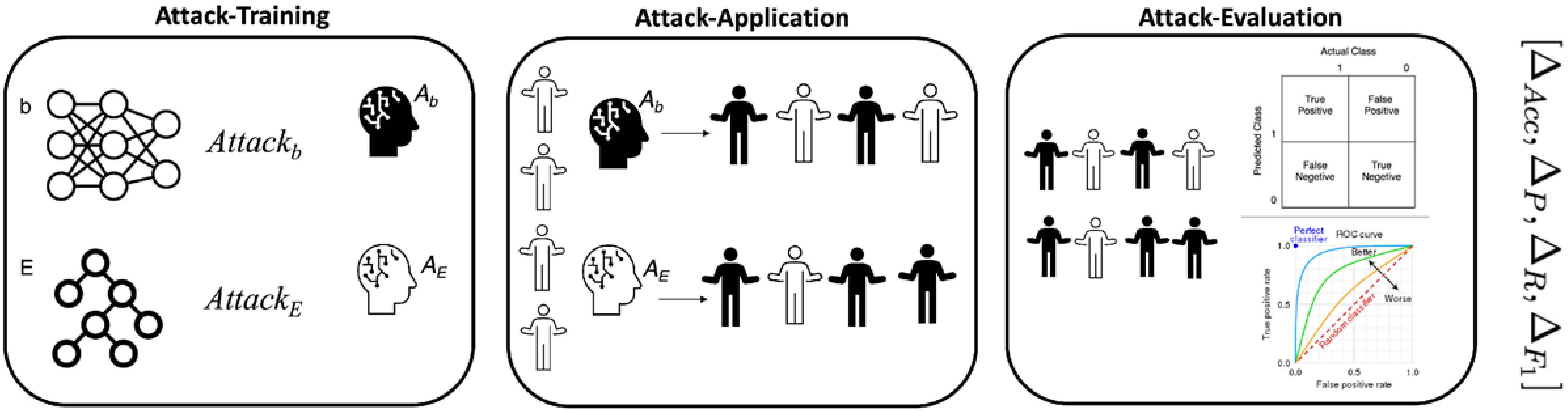
What about Explainers and privacy risks?

REVEAL

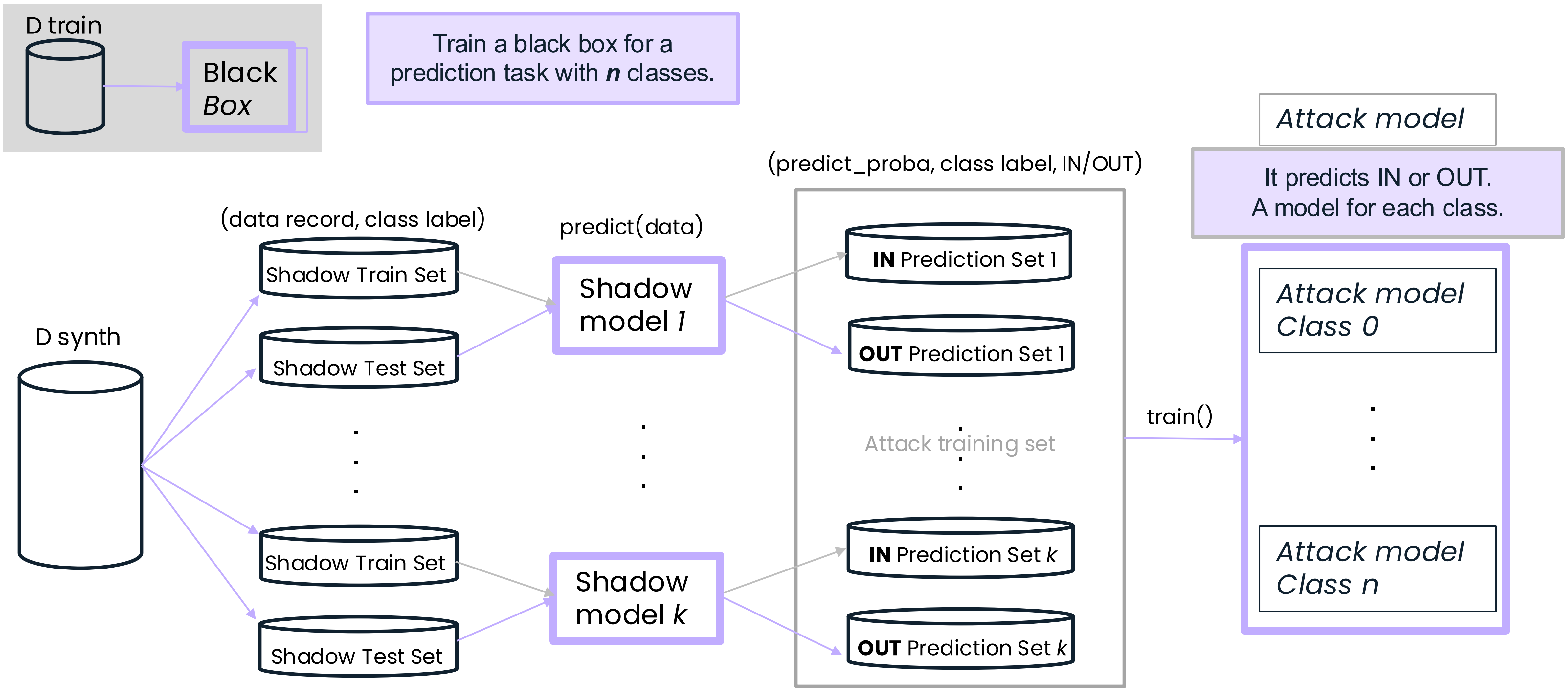


F. Naretto, A. Monreale, F. Giannotti: **Evaluating the Privacy Exposure of Interpretable Global and Local Explainers.** Trans. Data Priv. 18(2): 67-93 (2025)

REVEAL: privacy exposure of explainers

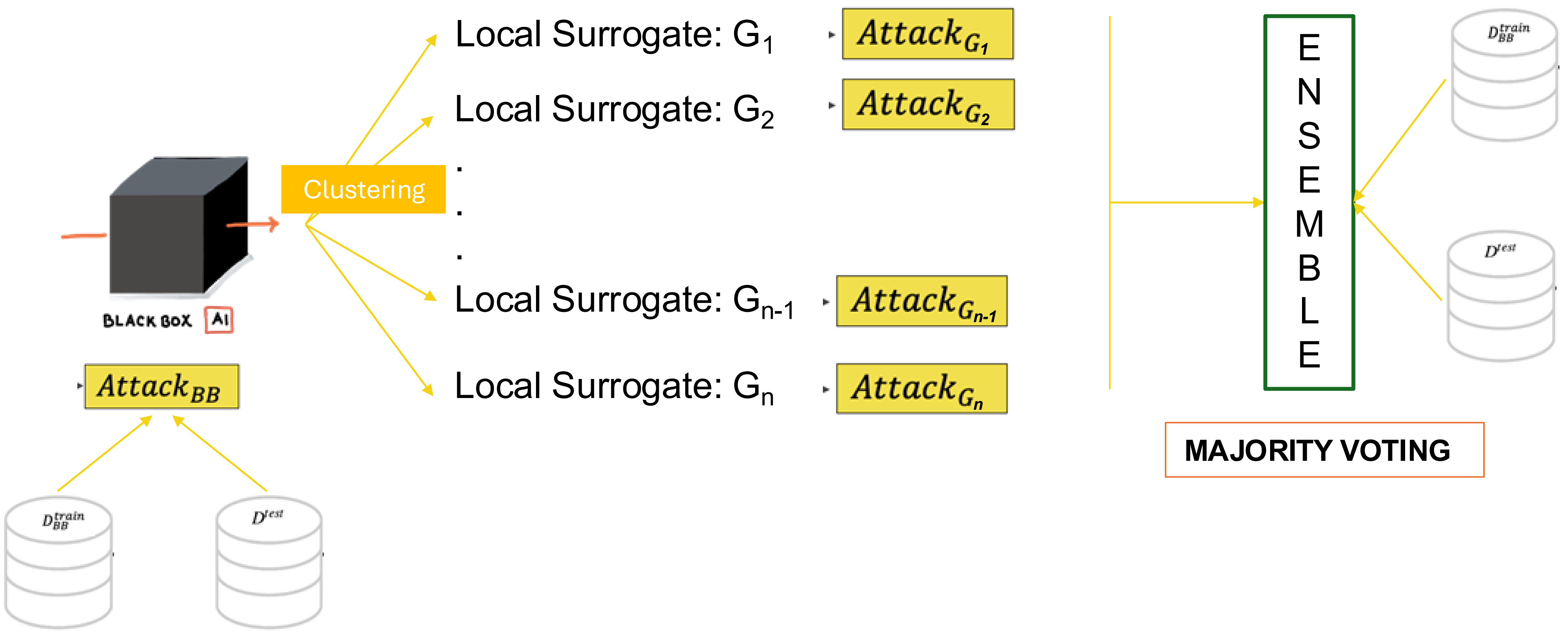


The privacy attack: MIA



Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models.
In 2017 IEEE Symposium on Security and Privacy

What about Local Explainers?



FastSHAP $^{++}$
A federated private explainer end-to-end

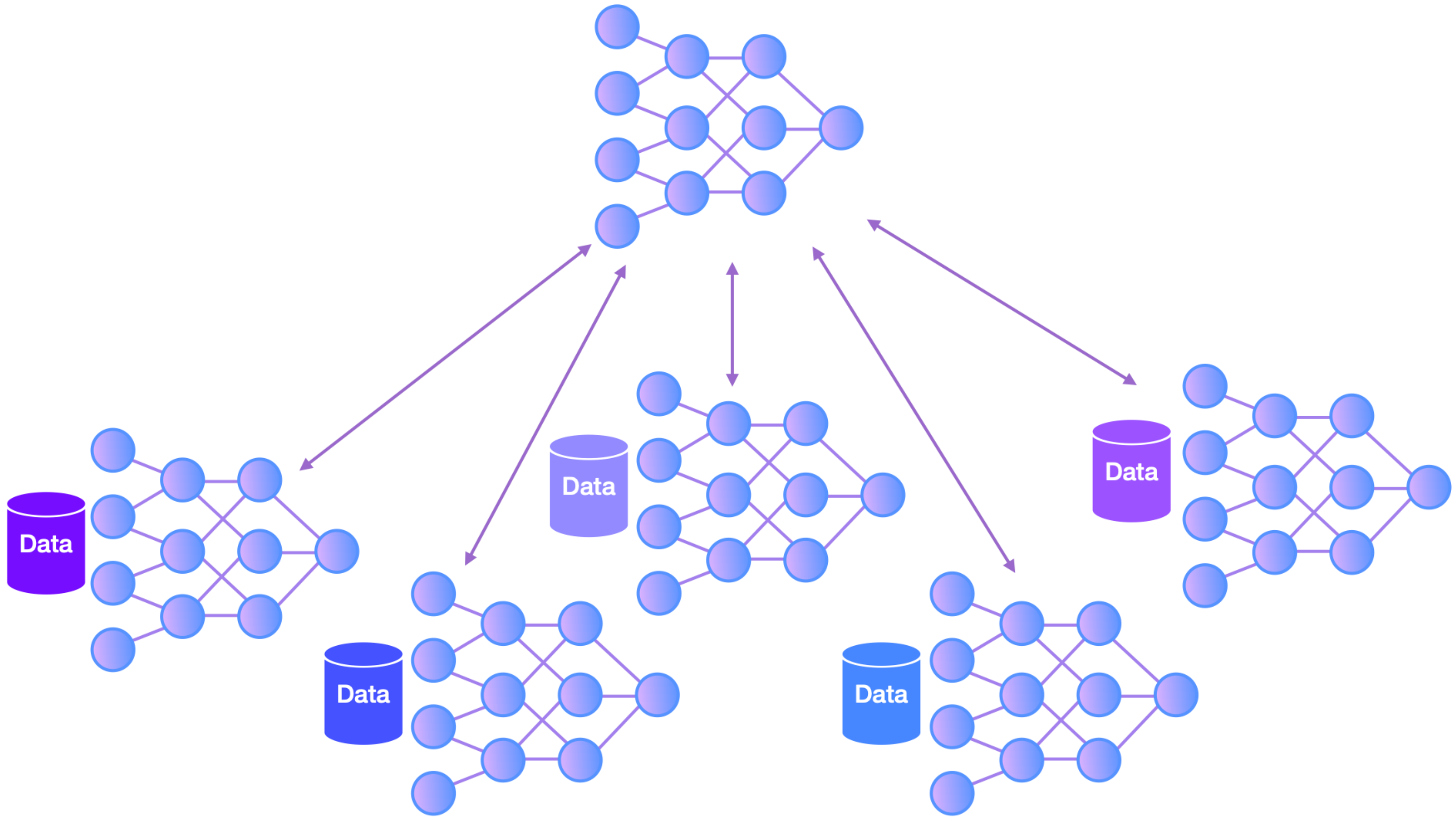
Valerio Bonsignori, Luca Corbucci, *Francesca Naretto*, Anna Monreale

Is it possible to explain Federated Learning models while preserving privacy and Federated Learning constraints?

Is it possible to explain **Federated Learning** models while preserving privacy and Federated Learning constraints?

Federated Learning

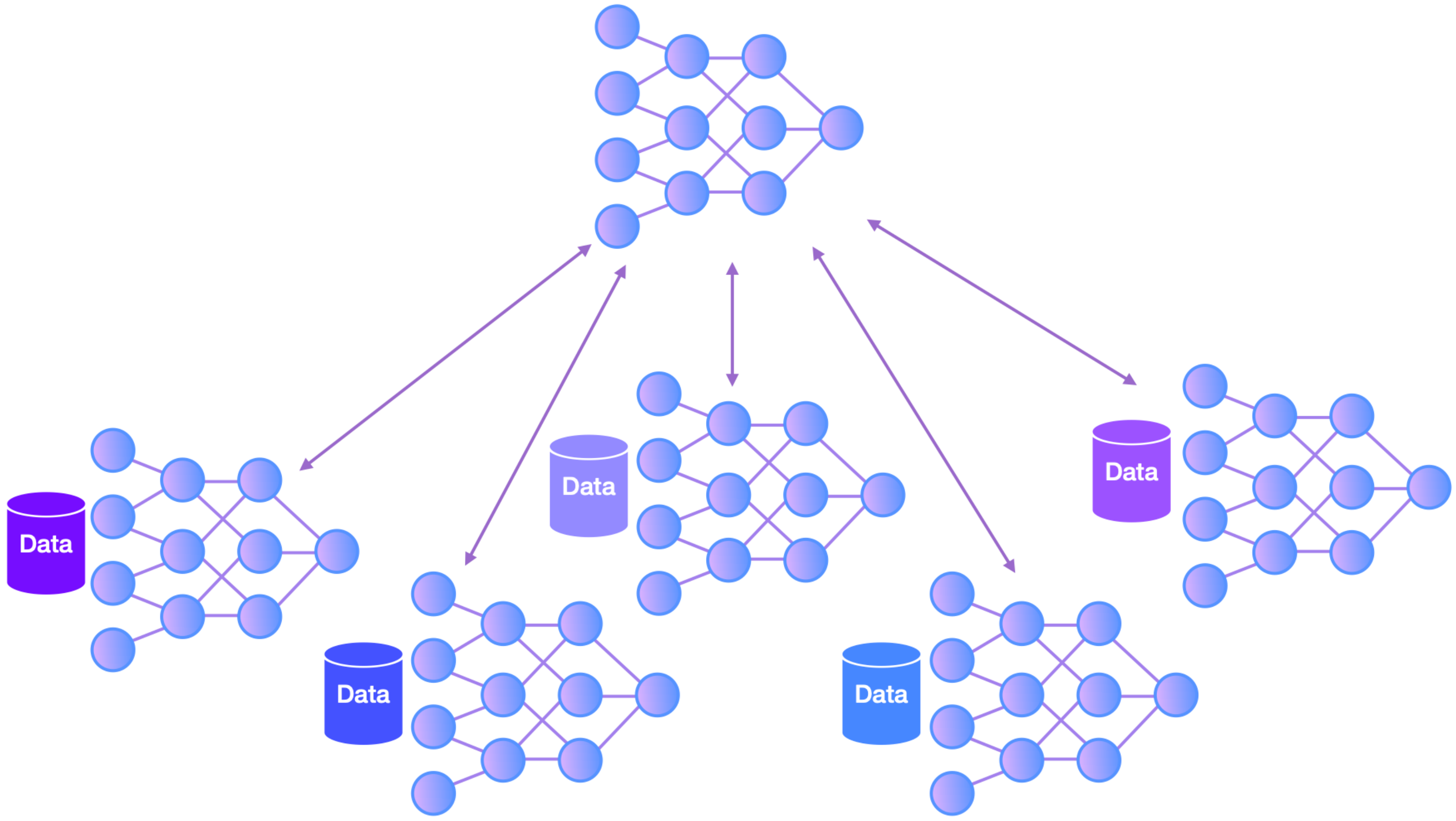
Clients don’t share their data, they only exchange model updates.



Federated Learning

FedAvg

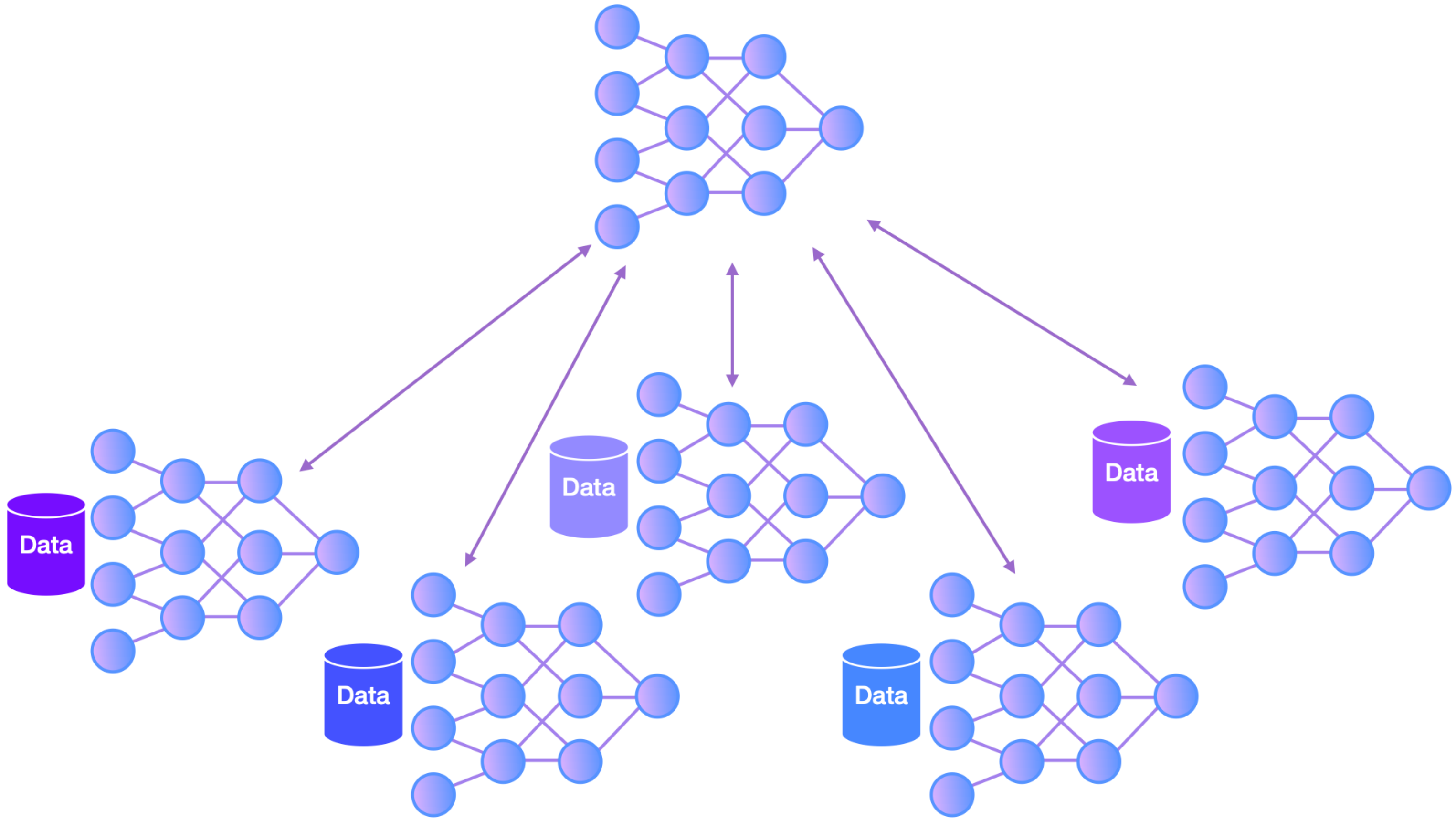
The server updates the global model by computing an average of the local parameter vectors returned by the participating clients after their local optimization steps.



Federated Learning

Clients don’t share their data, they only exchange model updates.

Good generalization capabilities.



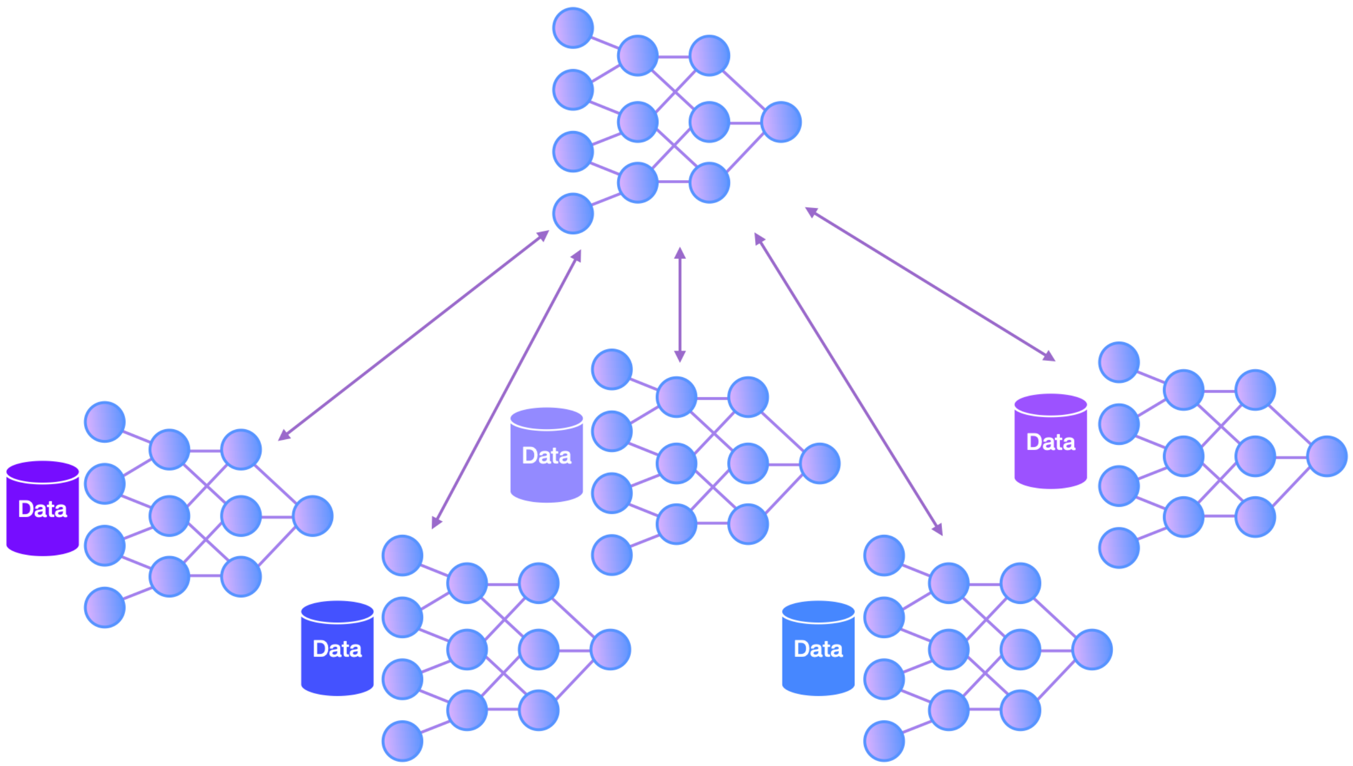
Is it possible to explain Federated Learning models while preserving **privacy** and Federated Learning constraints?

Federated Learning & Privacy

Privacy attacks are still possible.

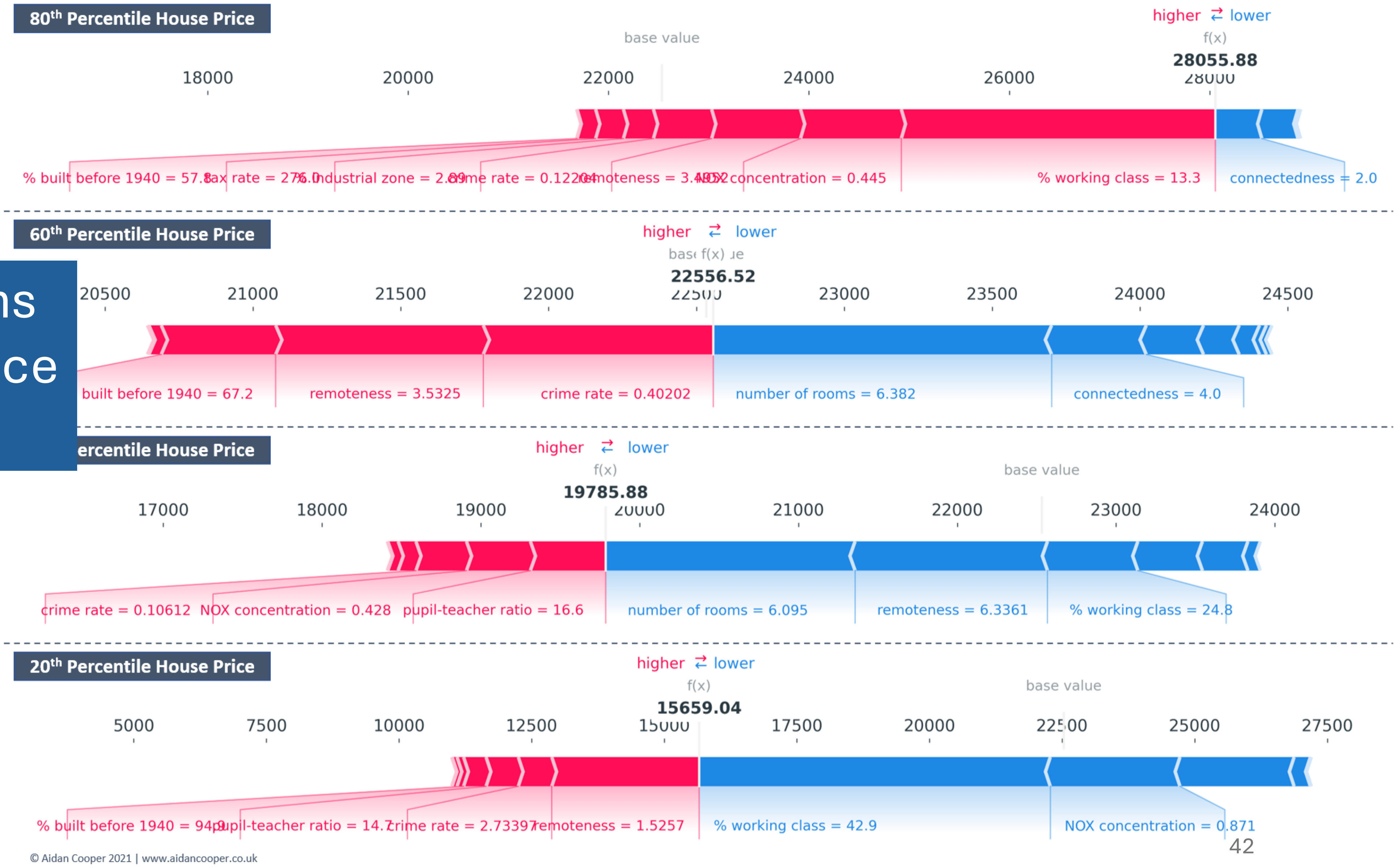
- Inverting gradients attacks
- Membership Inference Attacks
- Property Inference Attacks

Differential private learning of ML model can support privacy protection



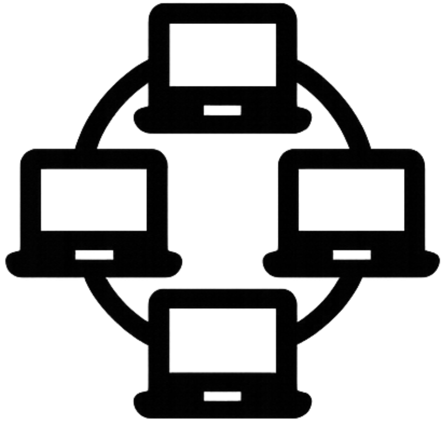
Is it possible to **explain** Federated Learning models while preserving privacy and Federated Learning constraints?

Explainable AI - SHAP



- Local explanations
- Feature importance
- Agnostic

We would like to:



Work in a Federated Learning scenario

Have local explanations

Preserve the privacy during all the steps of the pipeline

Limits

- ❌ SHAP can be slow
- ❌ Federated Learning VS explanations
    - SHAP requires data to be trained on
    - We don’t have them on the server side
    - Problems with privacy
- ❌ Overall… Limited privacy protection

FastSHAP

A possible solution is to use **FastSHAP**

- An explainer Neural Network
- Principles of SHAP values are still respected
- Good trade off between accuracy and speed up

FastSHAP

A possible solution is to use **FastSHAP**

- Not tailored for Federated Learning settings
- No privacy protection
- It uses original data for training

FastSHAP^{++}

Federated Learning
No exchange of data.

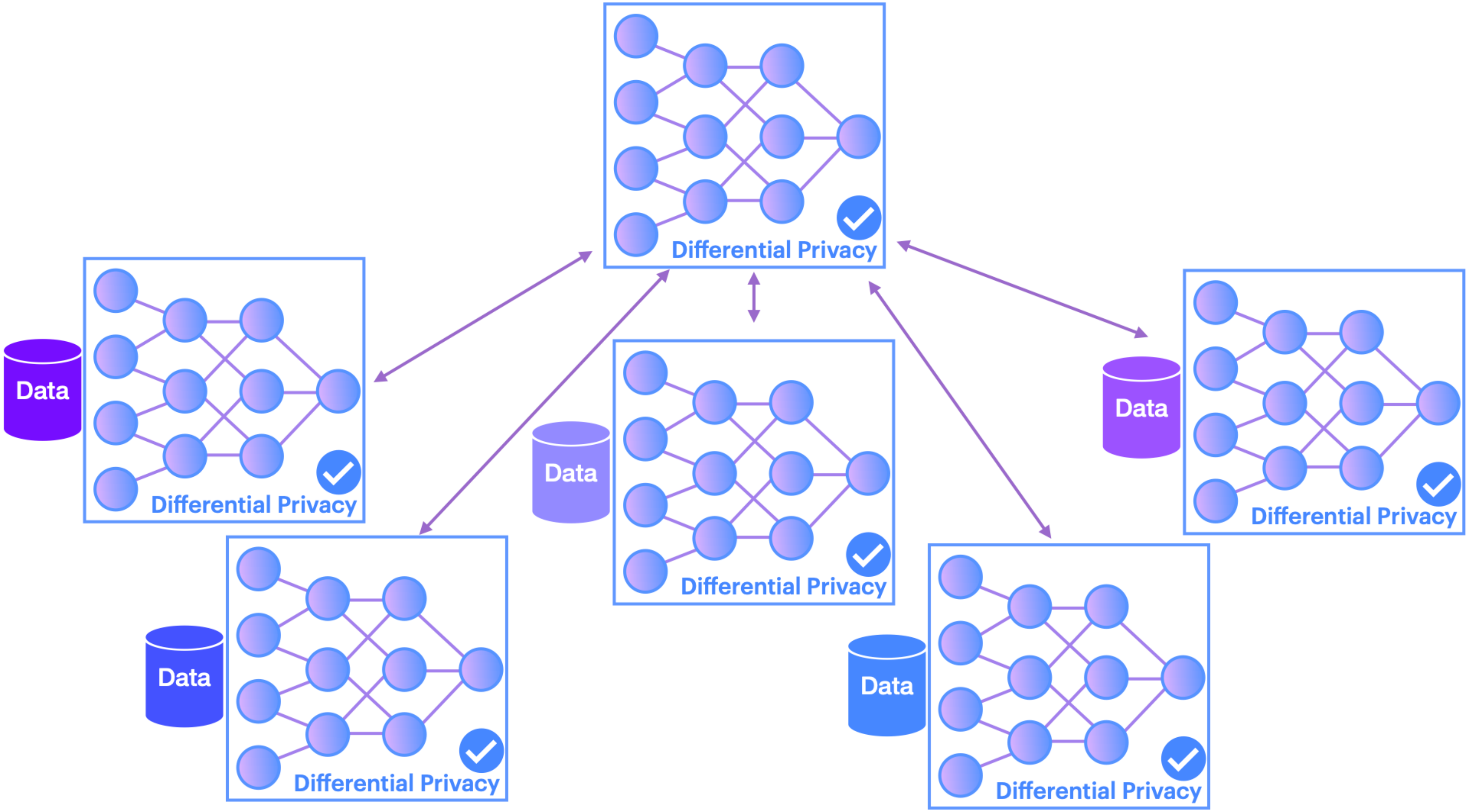
Fast Explanations
FastSHAP Explainer generates explanations in a forward step.

Local Explanation
Feature Importance Explanations using **FastSHAP**.

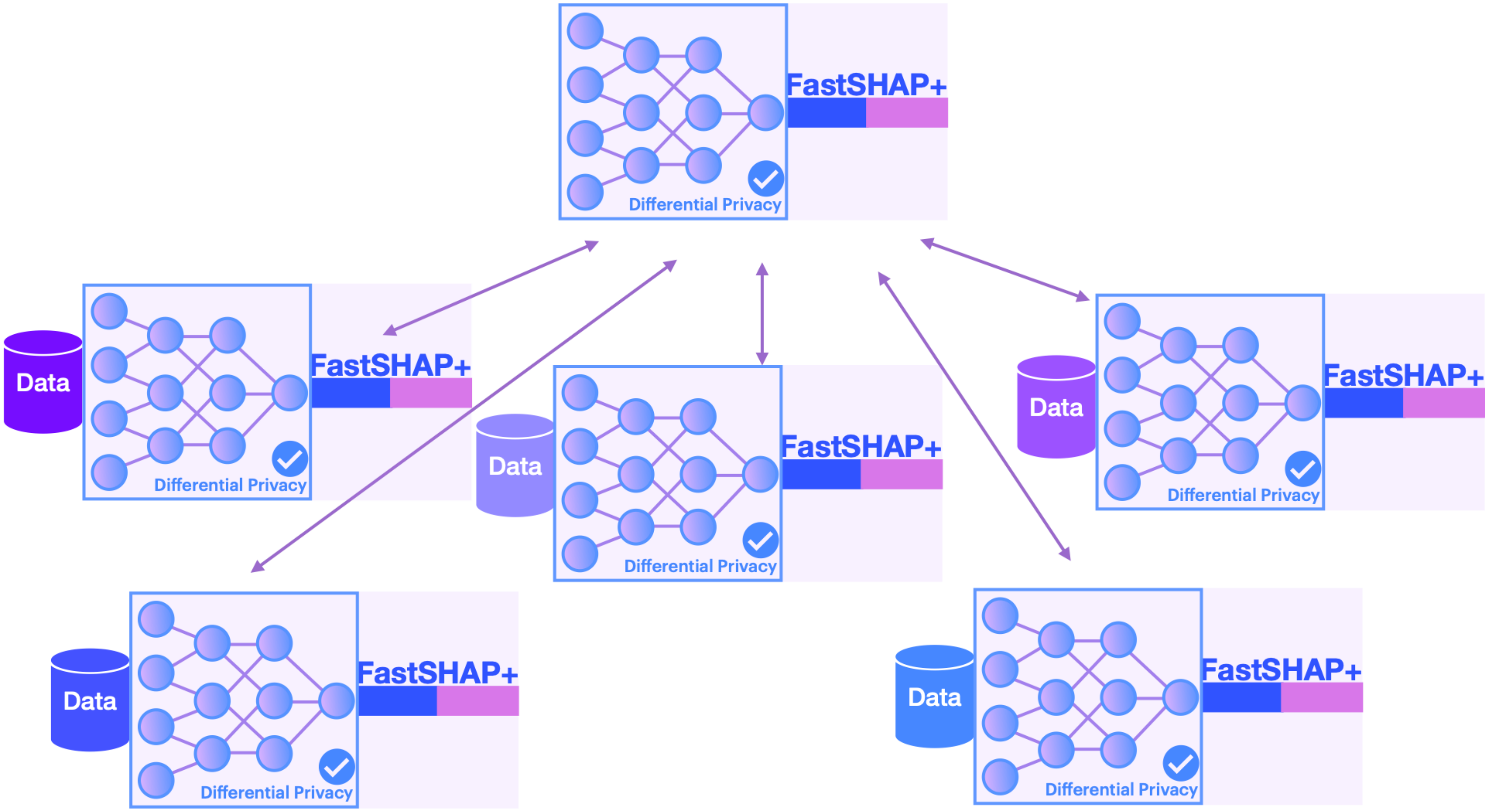
Distributed Explainer Training
Explainer complies with Federated Learning constraints.

Privacy Protection
A fully protected pipeline for privacy-guarantee explanations.

FastSHAP^{++}

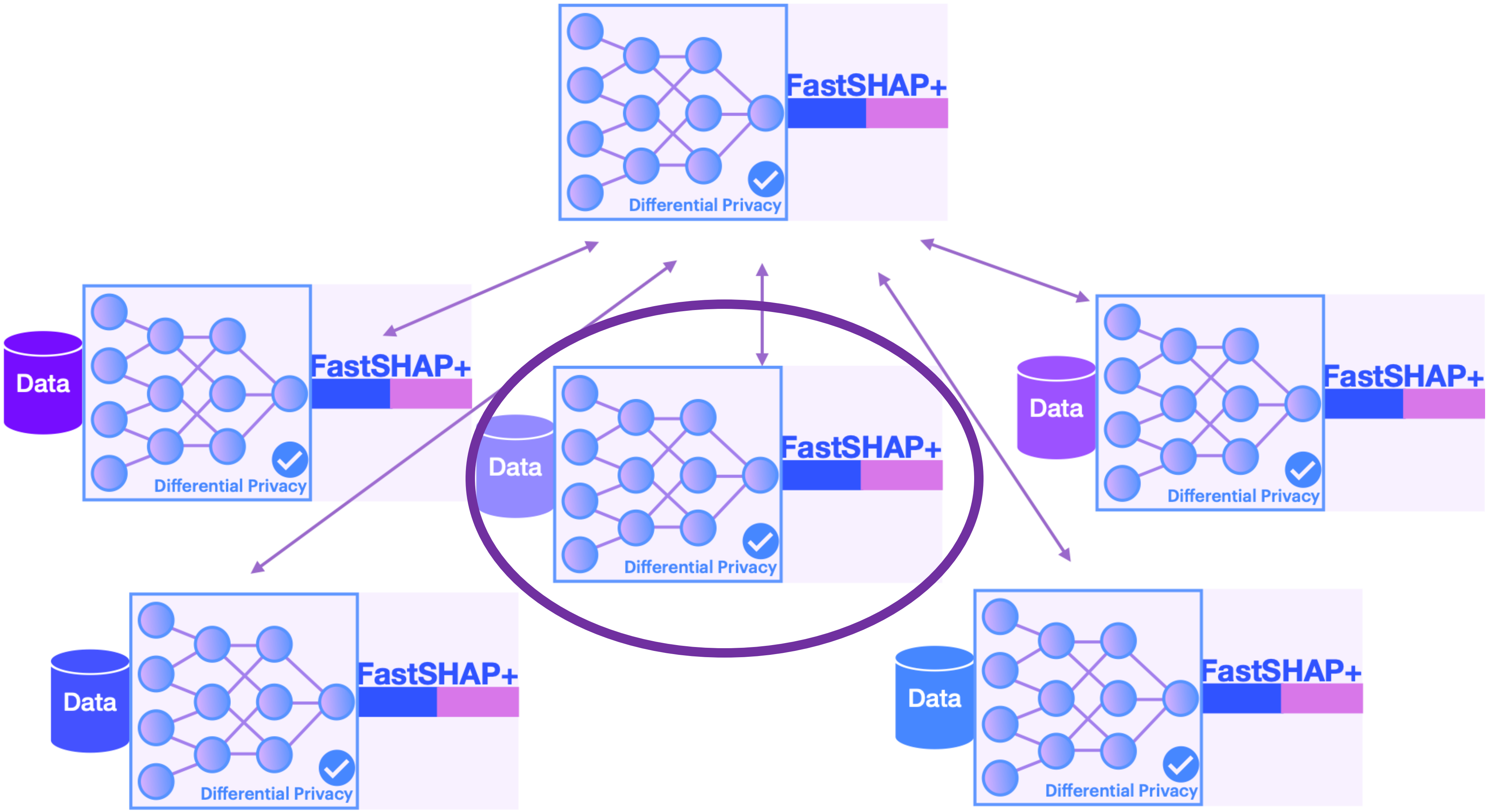

Differential Privacy
Data
Differential Privacy
Data
Differential Privacy
Data
Differential Privacy
Data
Differential Privacy
Data
Differential Privacy

FastSHAP^{++}



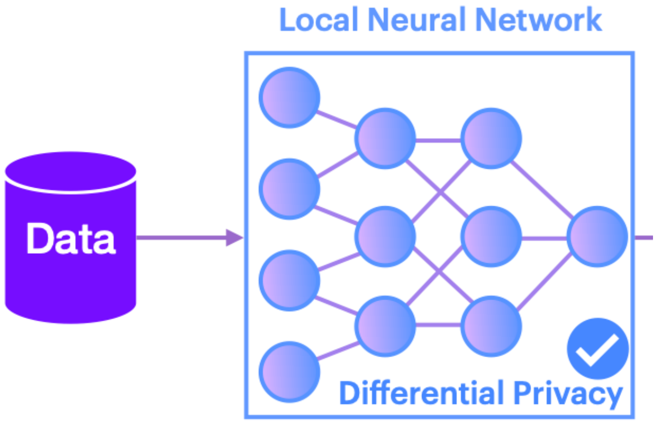
FastSHAP+
Differential Privacy
Data
FastSHAP+
Differential Privacy
Data
FastSHAP+
Differential Privacy
Data
FastSHAP+
Differential Privacy
Data
FastSHAP+
Differential Privacy
Data
FastSHAP+
Differential Privacy

FastSHAP^{++}

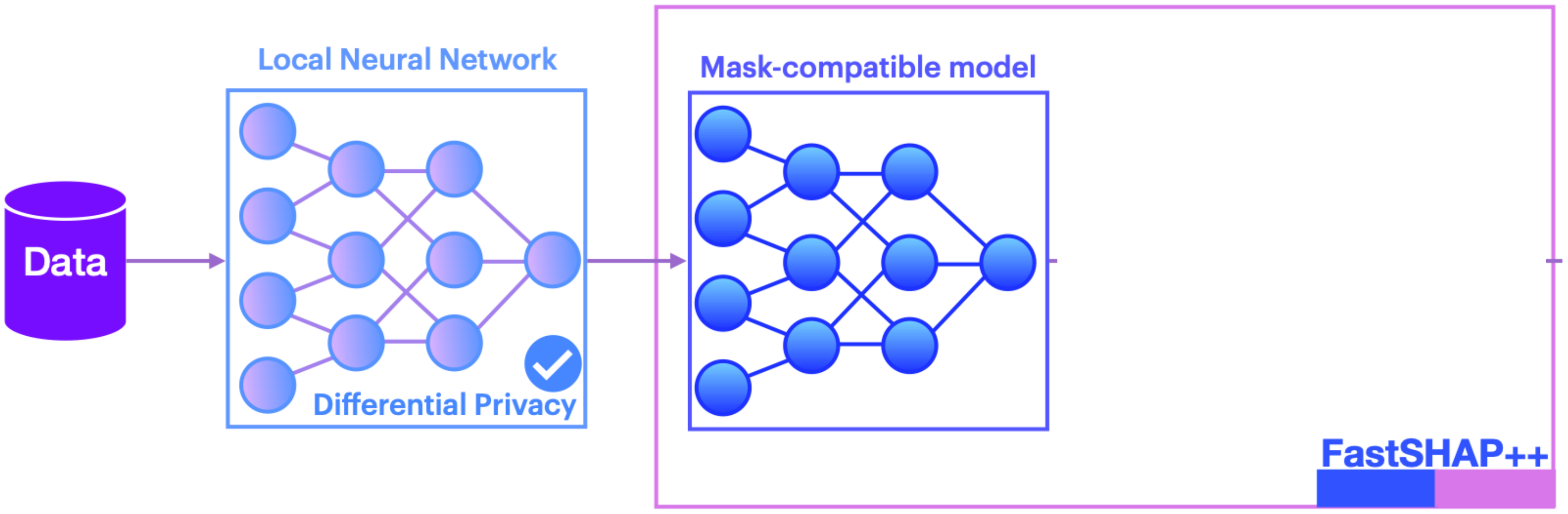


FastSHAP+
Differential Privacy
Data
FastSHAP+
Differential Privacy
Data
FastSHAP+
Differential Privacy
Data
FastSHAP+
Differential Privacy
Data
FastSHAP+
Differential Privacy
Data
FastSHAP+
Differential Privacy

FastSHAP^{++}

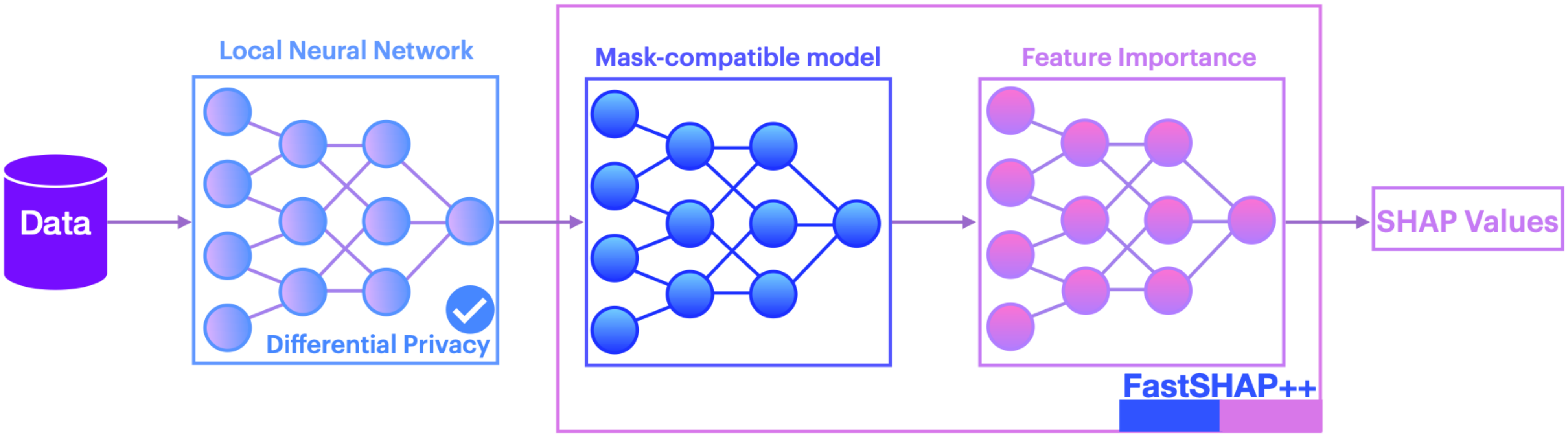


FastSHAP++



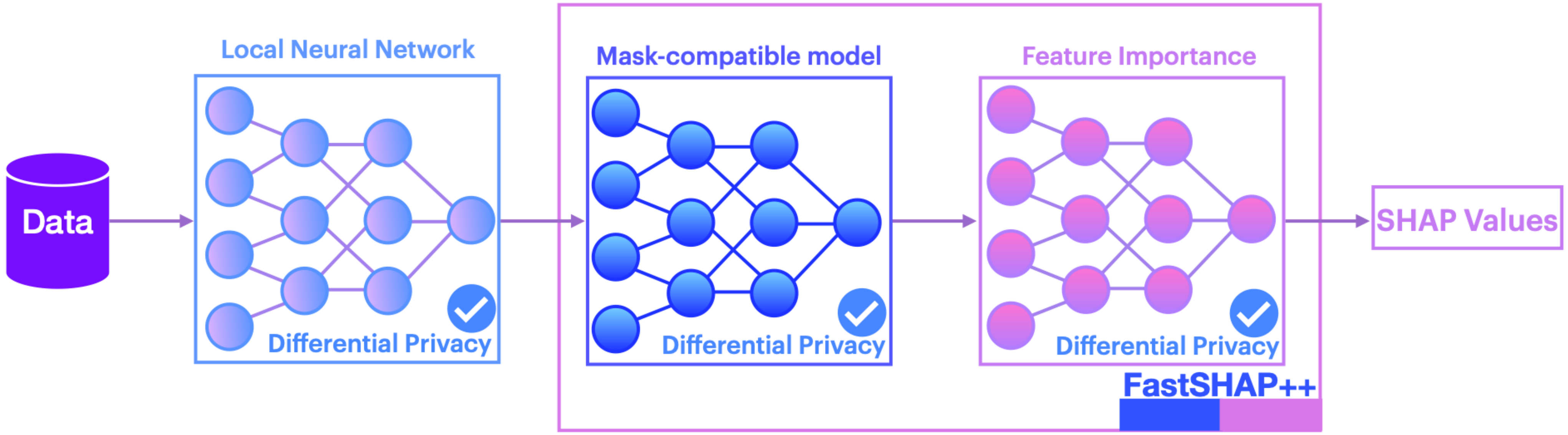
Data
Local Neural Network
Differential Privacy
Mask-compatible model
FastSHAP++

FastSHAP++

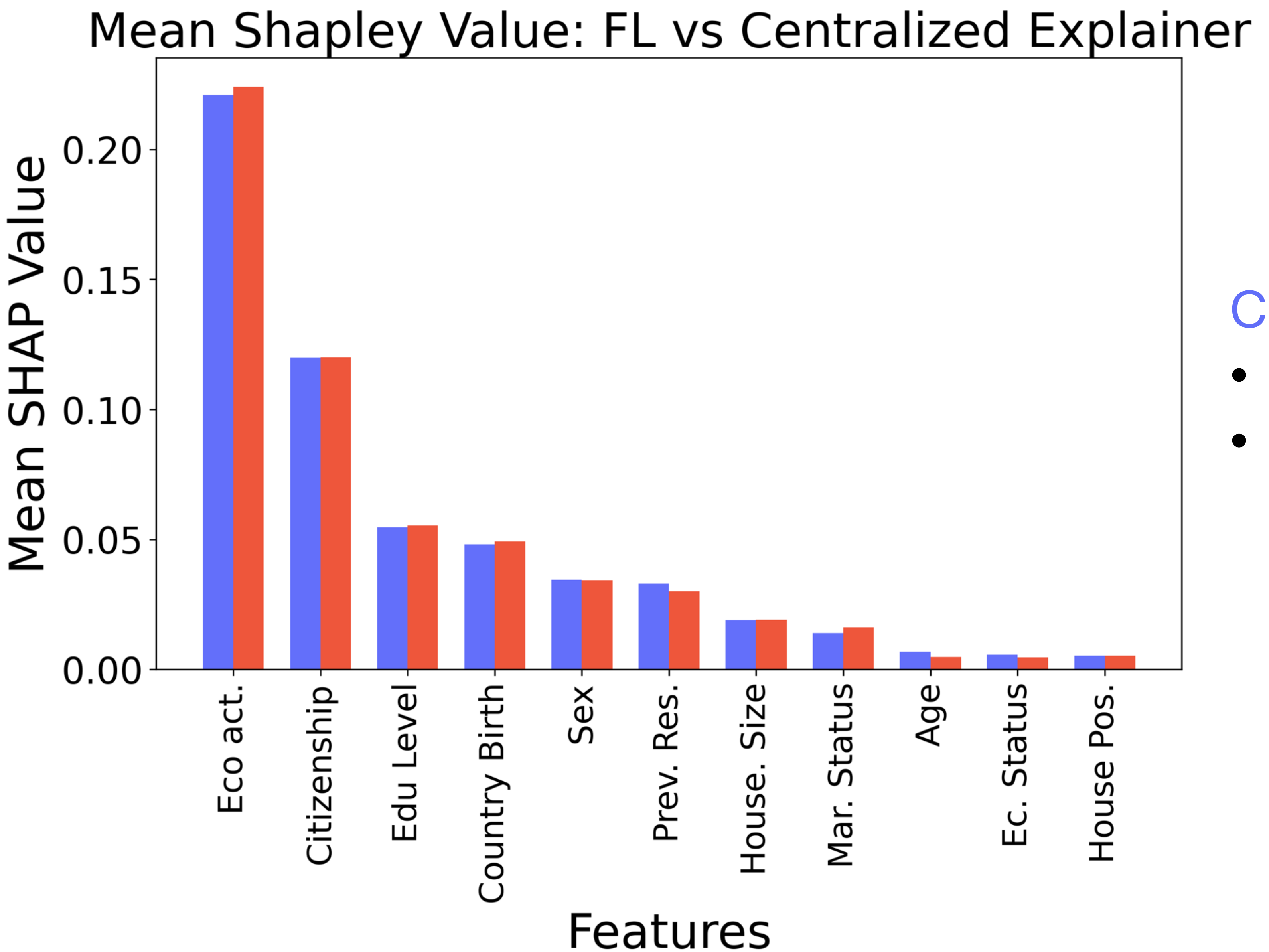


Data
Local Neural Network
Differential Privacy
Mask-compatible model
Feature Importance
FastSHAP++
SHAP Values

FastSHAP^{++}


Data
Local Neural Network
Differential Privacy
Mask-compatible model
Differential Privacy
Feature Importance
Differential Privacy
FastSHAP++
SHAP Values

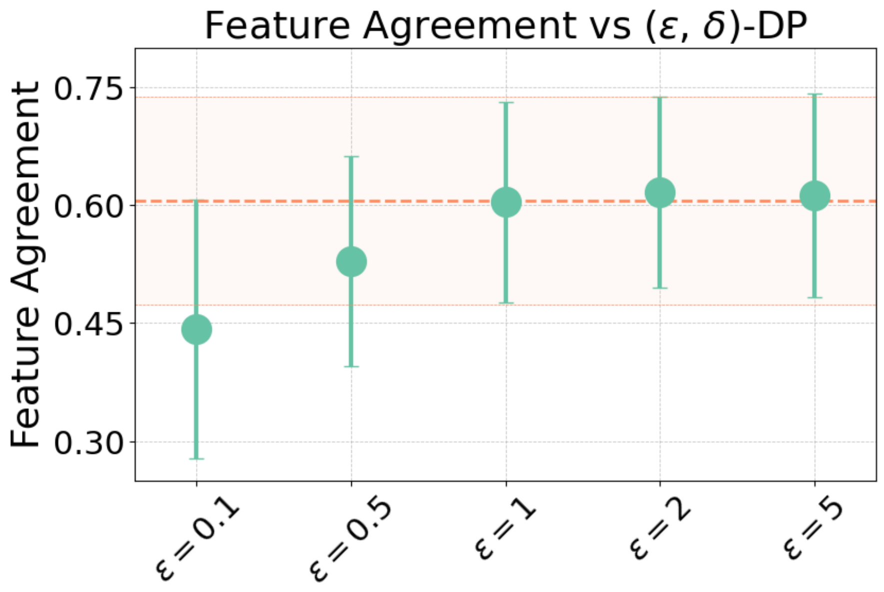
Are the explanations matching the centralized?

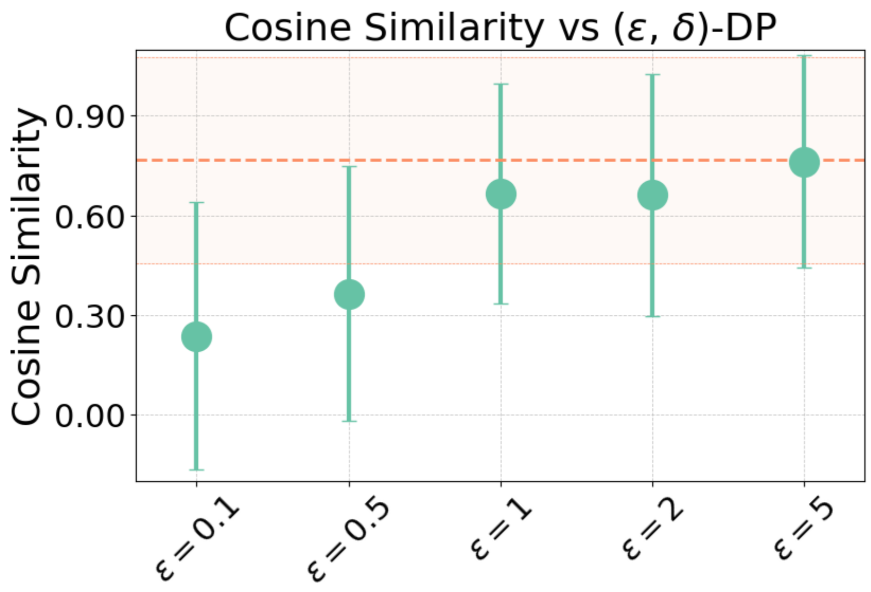


Centralized VS Federated

- Same direction of importance
- Similar magnitudes

Evaluation on Privacy





FastSHAP++ no privacy VS FastSHAP++ with privacy

The quality of the explanations is good with $\varepsilon \geq 1$.

FastSHAP^{++}

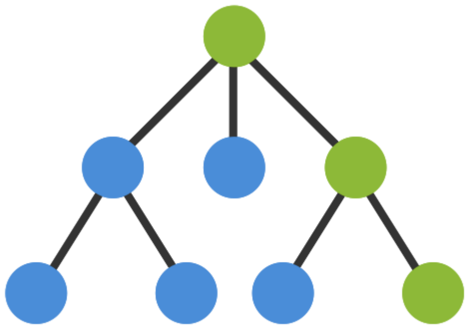
FastSHAP^{++} achieves centralized-level explanation quality while preserving clients' data privacy.

It achieves privacy without degrading the explanation quality too much, particularly when $\varepsilon \geq 1$.

Global Explainer: TREPAN

T = **root_of_the_tree**()
Q = <T, X, {}>
while Q ***not empty &*** **size**(T) < *limit*
    N, X_N, C_N = **pop**(Q)
    Z_N = **random**(XN, CN)
    y_Z = b(Z), y = b(XN)
    if **same_class**(y ∪ y_Z)
        continue
    S = **best_split**($X_N \cup Z_N$, y ∪ y_Z)
    S^1= **best_m-of-n_split**(S)
    N = **update_with_split**(N, S^1)
    for each condition c ***in*** S^1
        C = **new_child_of**(N)
        $C_C = C_N \cup \{c\}$
        X_C = **select_with_constraints**(X_N,C_N)
        put(Q, <C, X_C , C_C>)

- Enriches the training data
- Labels the data by using the BB
- Train a DT (surrogate model)

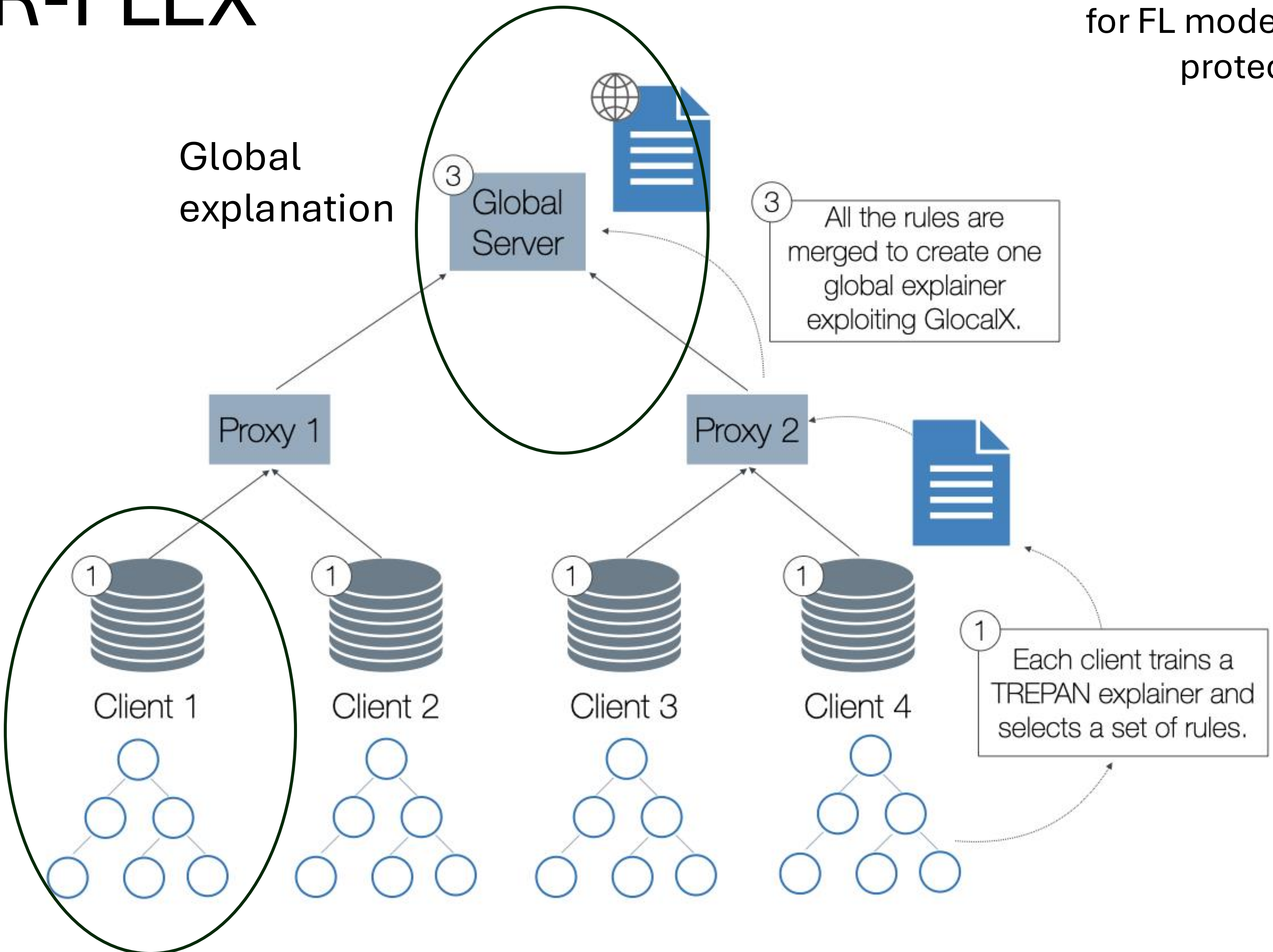


GLOR-FLEX

Define a rule-based explainer for FL models privacy-protected.

Global explanation

Global explanation in the locality of the client



How to merge rules

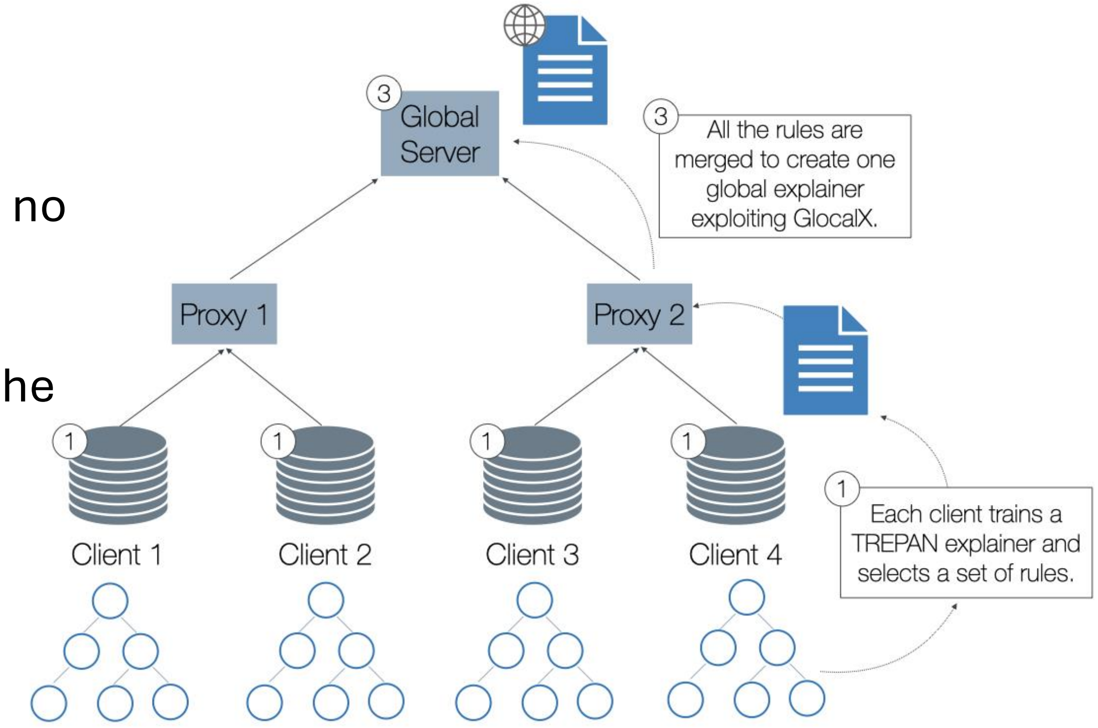
Input: $\mathbb{E}$ explanation theories, α filter threshold
Output: E explanation theory

$E \leftarrow \emptyset$
repeat
  $\mathbb{Q} \leftarrow \textsc{sort}(\mathbb{E})$ ▹ sort pairs of theories by similarity
  $\mathit{merged} \leftarrow \texttt{False}$
  $X' \leftarrow \text{batch}(X)$
  while $\neg\ \mathit{merged} \wedge \mathbb{Q} \neq \emptyset$ **do**
    $E_i, E_j \leftarrow \textsc{pop}(\mathbb{Q})$ ▹ select most similar theories
    $E_{i+j} \leftarrow \textsc{merge}(E_i,\ E_j,\ X')$ ▹ merge theories
    if $\textsc{bic}(E_{i+j}) \leq \textsc{bic}(E_i \cup E_j)$ **then** ▹ verify improvement
      $\mathit{merged} \leftarrow \texttt{True}$
      break
  if *merged* **then** ▹ merge occurred
    $\mathbb{E} \leftarrow \textsc{update}(E_i,\ E_j,\ E_{i+j})$ ▹ update hierarchy
until $\mid E \mid > 1 \wedge \mathit{merged}$ ▹ until the merge is successful
$E \leftarrow \textsc{filter}(E, \alpha)$ ▹ Filter final theory
return E

GLObal to loCAL eXplainer (**GlocalX**)
It hierarchically merges local explanations.
Explanations with lower fidelity are filtered out.

How to merge rules: drawback

- GlocalX needs data to perform its tasks.
- But if we use real data outside the clients, we are no longer respecting privacy
- Solution: generate synthetic data that resemble the original ones



Conclusion

GLOR-FLEX: A local-to-global post-hoc explanation method which generates rules for Federated Learning approaches.

- It uses TREPAN to generated global explanations at the client side;
- It uses GlocalX to merge the rules.

WRAP-UP:

- No private data exploited in the procedure
- Interpretability enhanced