# Privacy in AI

Presenters: Cohen Archbold, Usman Hassan
Contributors: Luna Zhu, Dongjie Chen
Advisor: Dr. Sen-Ching Cheung

# Overview

- Private Data [4]
- Ethical and Legal Considerations [5]
- Privacy Risks in AI [6-7]
  - Example Model Inversion [8]
- Current Solutions [9]
- Federated Learning [10-12]
  - Flower Tutorial [13-15]
- Differential Privacy [16-27]
  - DP-SGD Tutorial (Tensorflow Privacy) [28]
- Measuring Privacy using Attacks [30]
- Take-aways [31]

# **Big Picture**
# Private AI

Assess Privacy risk

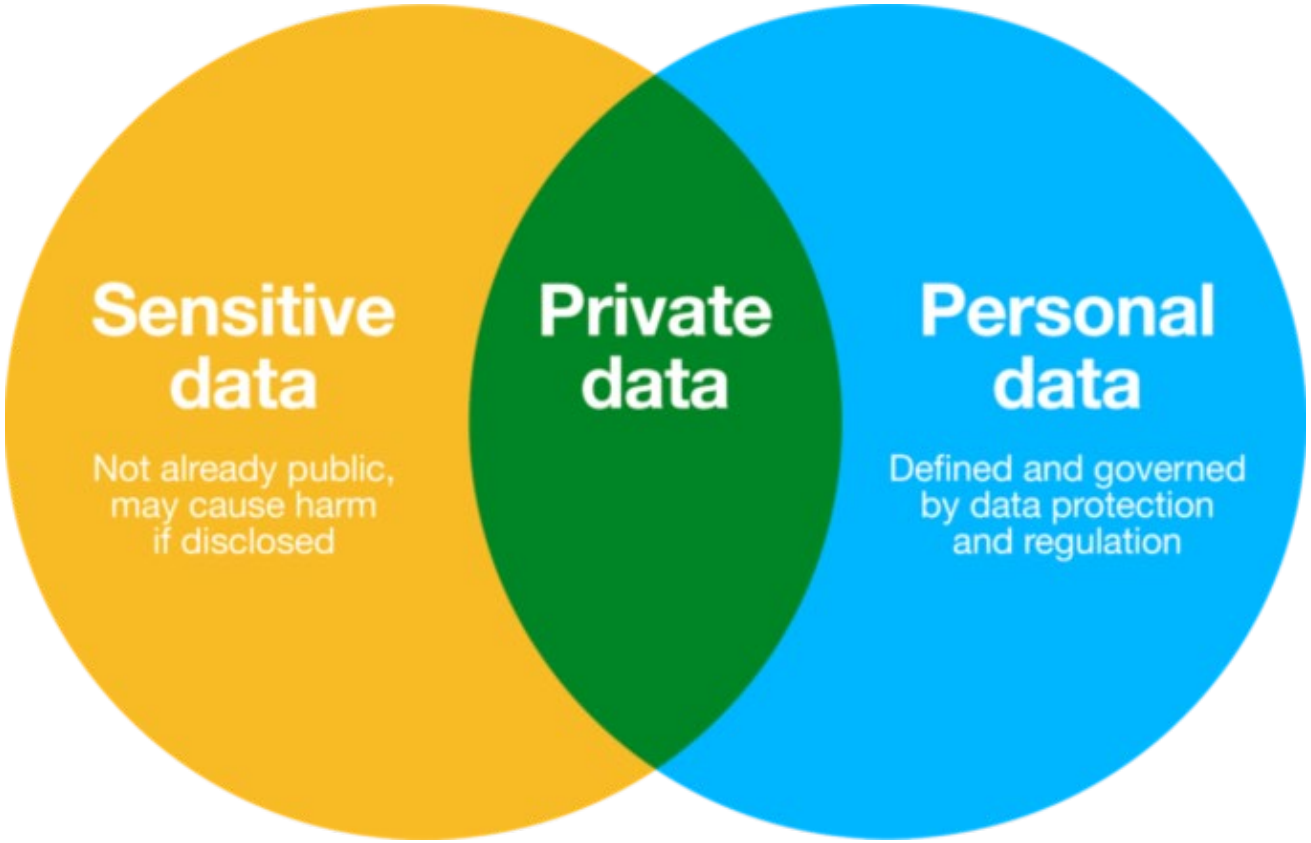
Select right solution

Implement solutions & measure performance

Validation & Ethical Hacking

# Private Data

- Sensitive data in AI refers to any information that could compromise *privacy, security, or confidentiality* if misused or exposed.
- This includes personal identifiers, health records, financial details, and any data that could be traced back to individuals.




Source(Cobalt)

# Ethical and Legal Considerations

- **Ethical considerations**
  - **Respect Privacy:** Handle sensitive data with integrity and avoid unauthorized access.
  - **Fairness:** Ensure models do not discriminate against protected groups.

- **Regulatory compliance**
  - **Key Regulations:** Comply with GDPR, HIPAA, and other applicable data protection laws.
  - **Data Minimization:** Collect only the data necessary for the intended purpose.

- **Transparency in Data Handling**
  - **Lifecycle Clarity:** Maintain transparency from data collection to model deployment.
  - **Foster Trust:** Clearly communicate practices to minimize risk and enhance user trust.

# Privacy Risks in Machine Learning

What are the real risks?

- Re-identification of individuals from training data
- Reconstruction of private data
- Memorization by generative models

*We focus on GPT-2 and find that at least 0.1% of its text generations (a very conservative estimate) contain long verbatim strings that are "copy-pasted" from a document in its training set.*

*Blog post: [Wallace, Tramer, Jagielski, and Herbert-Voss], 2020*

" A group of artists has filed a class-action complaint against the companies behind a trio of A.I. art generators, saying the services violated copyright and unfair competition laws.
The class action asserts that A.I. generators have allowed art to be generated “in the style” of particular artists, thus “siphoning commissions from the artists themselves.” "
-Harvard Business Review, April 2023

"55% of generative AI inputs comprised personally identifiable data" - Menlo Security, Feb 2024

# Privacy Risks in Machine Learning

Attacks on Machine Learning Models with Privacy Risk:

- Attribute Inference Attack
  - Extracts sensitive attributes from machine learning model outputs.
- Membership Inference Attack
  - Determines if a specific data point was in training data.
- Model Inversion Attack
  - Reconstructs original input data from the model's predictions or outputs.

# Example Model Inversion

**Figure 1: An image recovered using a new model inversion attack (left) and a training set image of the victim (right). The attacker is given only the person's name and access to a facial recognition system that returns a class confidence score.**

(Matt Fredrikson, Somesh Jha, Thomas Ristenpart, 2015, ACM SIGSAC)

# Current Solutions

Balancing insights and privacy in sensitive data handling is crucial. Key techniques include:

- **Encryption**: Secures data at rest and in transit.
- **Differential Privacy**: Adds noise to protect individual data points while allowing aggregate insights.
- **Federated Learning**: Enables model training across decentralized data sources without sharing raw data.

Despite advancements, privacy in AI remains an ongoing challenge, with continuous innovations in private learning.

# Federated Learning

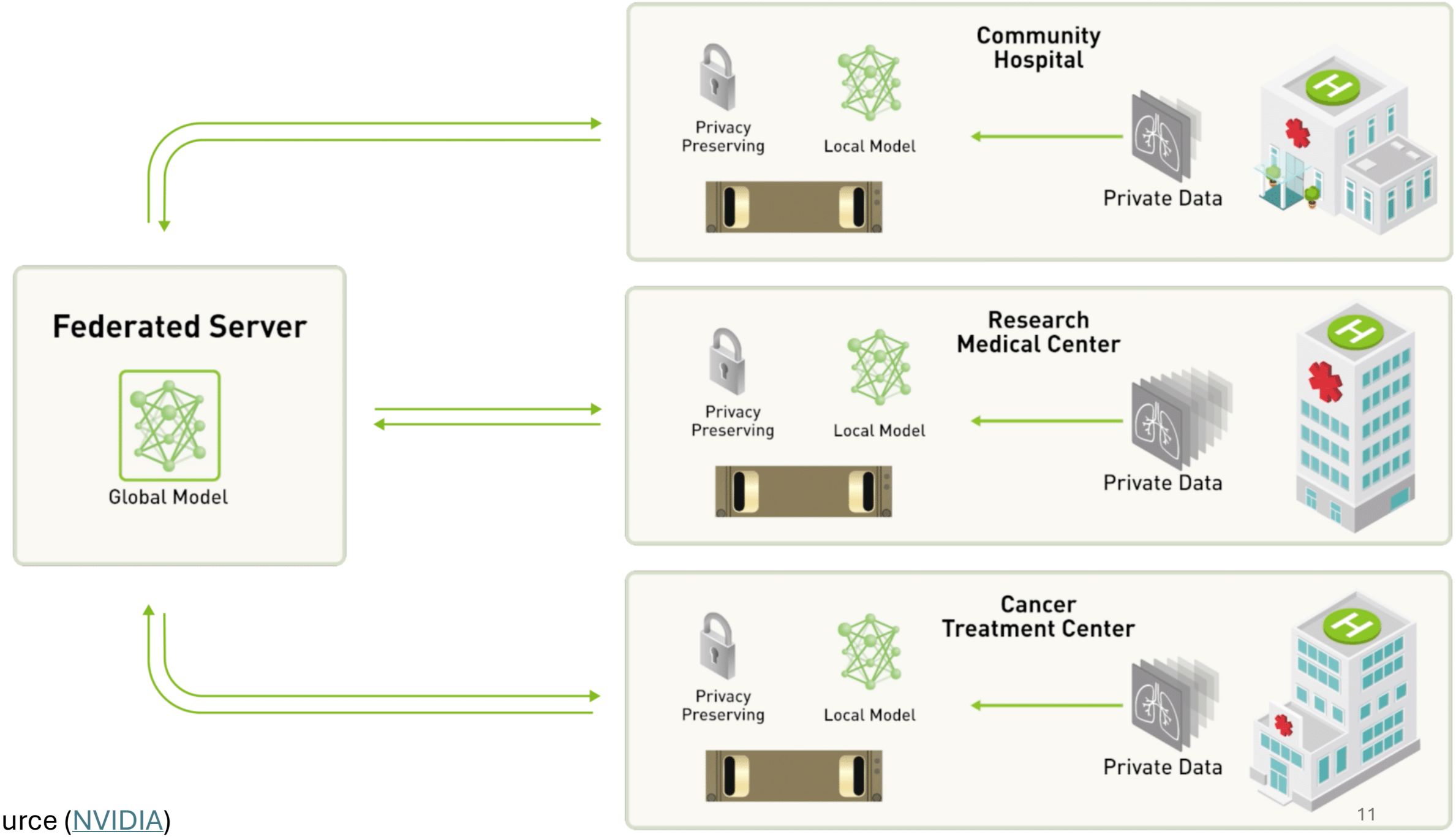
Federated Learning is a machine learning technique that enables decentralized training of models across multiple devices without sharing raw data.

- Allows devices to collaboratively train models while keeping data on local devices, enhancing **privacy** and **security**.
- Only the **model parameters** or **gradients** are shared, not the actual data.

Example Applications:

**Healthcare:**
Training predictive models across hospitals without sharing patient data.

**Smartphones:**
Improving keyboard suggestions or app personalization without compromising user privacy.




Source (NVIDIA)

# Federated Learning In Practice

Federated Learning Algorithm:

**Step 1**: Local Training — each device trains the model on its data.

**Step 2**: Update Aggregation — each device sends the model updates to the central server.

**Step 3**: Model Update — the server aggregates updates to improve the global model and sends it back to devices.

**Some optional / additional steps:** Differential Privacy, Secure (Encrypted) Aggregation, Compression, Re-sampling

# Flower.ai

- "A Friendly Federated Learning Framework"
- Simplifies implementation of server-client federated framework for simulation and real-world training + testing.
- Incorporates hooks for major machine learning libraries

Flower
Framework

Flower is compatible with:

- Pytorch/Tensorflow
- IOS/Android
- JAX
- And more!

# Flower Tutorial

**Breast Cancer Wisconsin (Diagnosis) Dataset**

- Samples: 569 cases of breast cancer
- Classes: Two classes – Malignant (M) and Benign (B)
  - M: 212 cases (~ 37%); B: 357 cases (~ 63%)
- 30 Numerical Features describing features of cell nuclei
  - Radius (mean distance from center to perimeter)
  - Texture (gray-scale variation)
  - Smoothness, compactness, concavity, symmetry, etc.
  - Features are computed as mean, standard error & "worst" value

# Flower Tutorial


Code Available at: https://github.com/cgarchbold/flwr_tutorial

Flower Documentation: https://flower.ai/docs/

# Differential Privacy

- Privacy breaches are inherent in large-scale ML and statistical analyses (Examples: Apple's iOS, U.S. Census Bureau's OnTheMap).
- Heuristic privacy measures like anonymization fall short.
  - Example: Governor William Weld's anonymized medical records were re-identified by cross-referencing with voter data.
- Robust privacy safeguards are essential to maintain user trust.

# Motivation
## The Problem of Personal Information Leakage

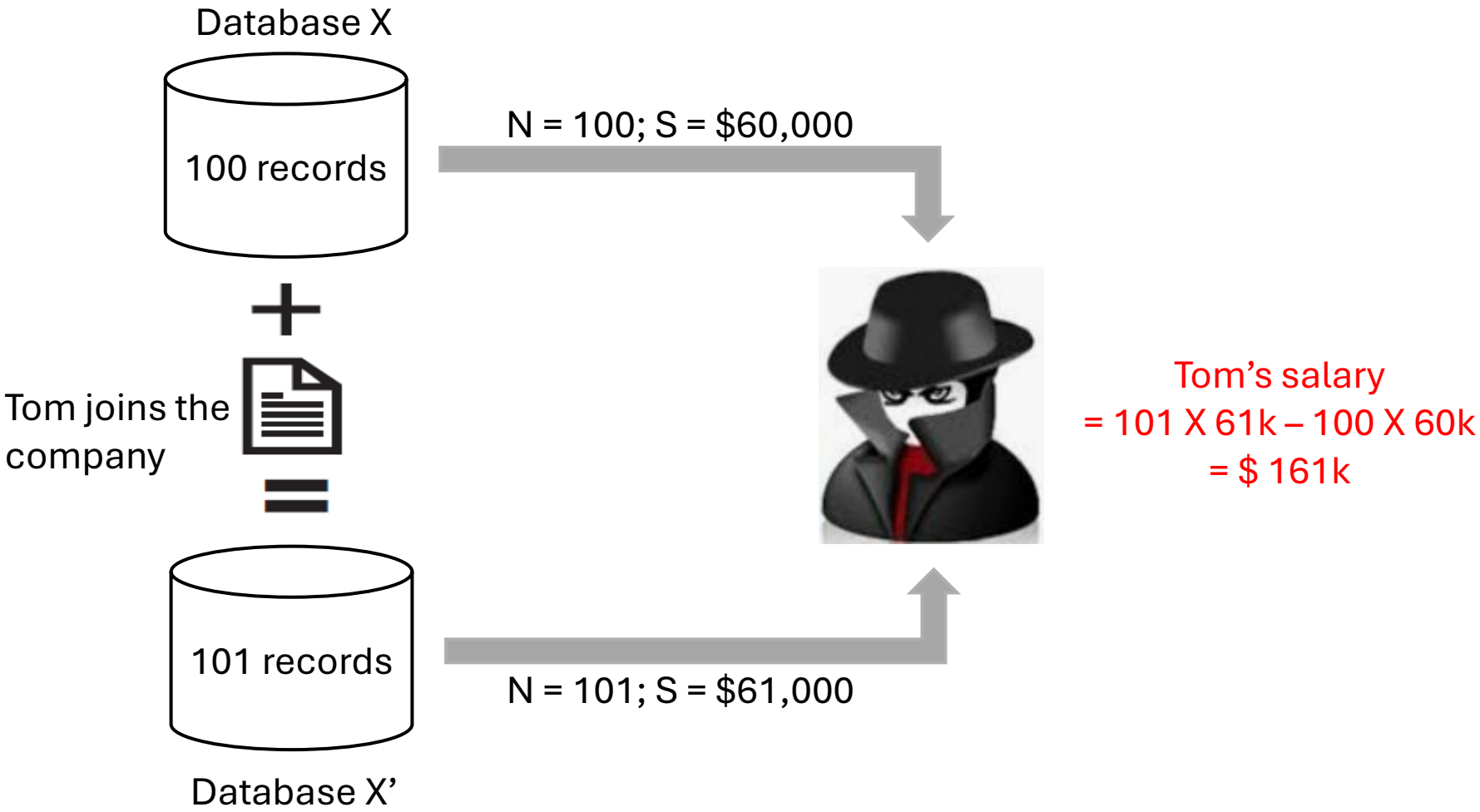



**Figure: Non-private publishing of average salary.**

# Motivation
## Protecting Privacy with Noise

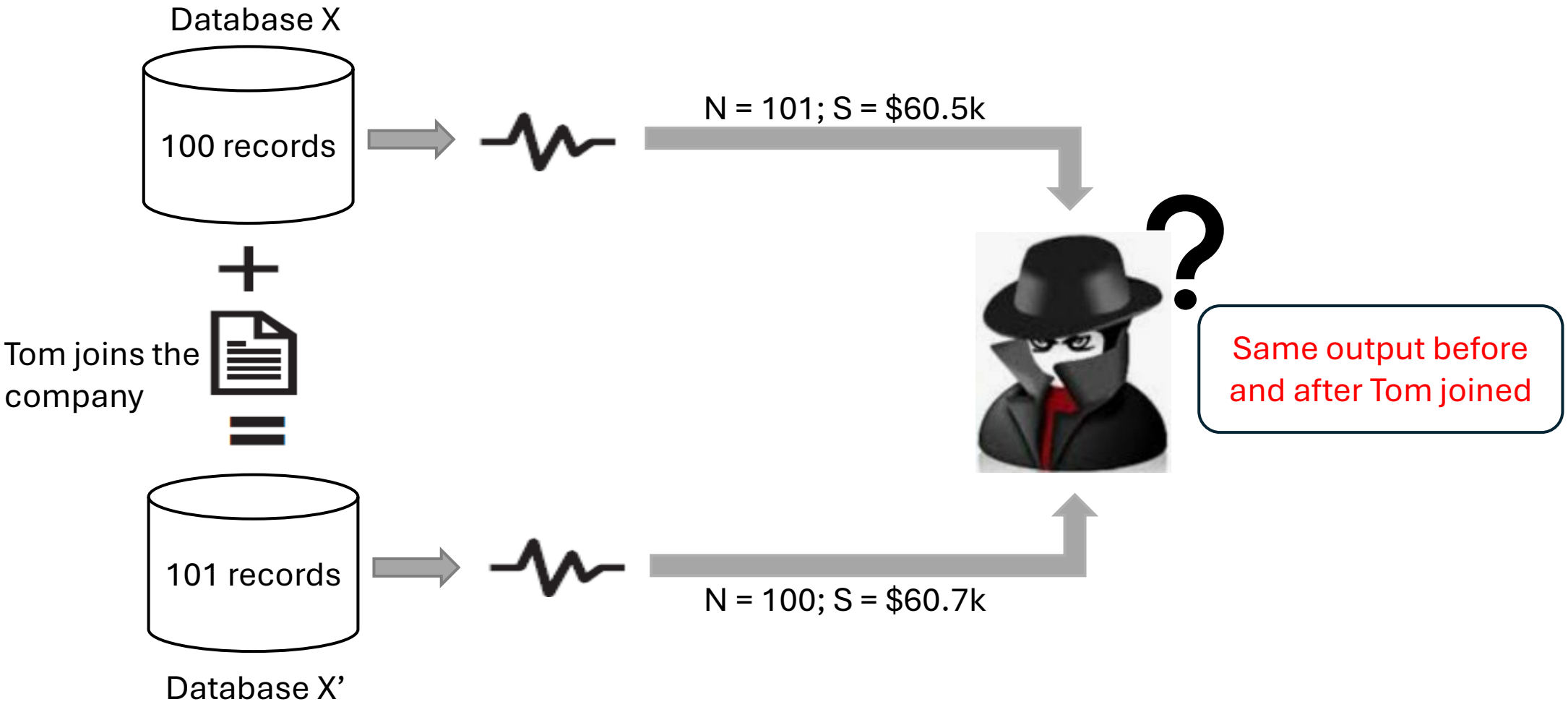



**Figure: Private publishing of average salary by adding noise.**

# Motivation
## High-Sensitivity Challenge

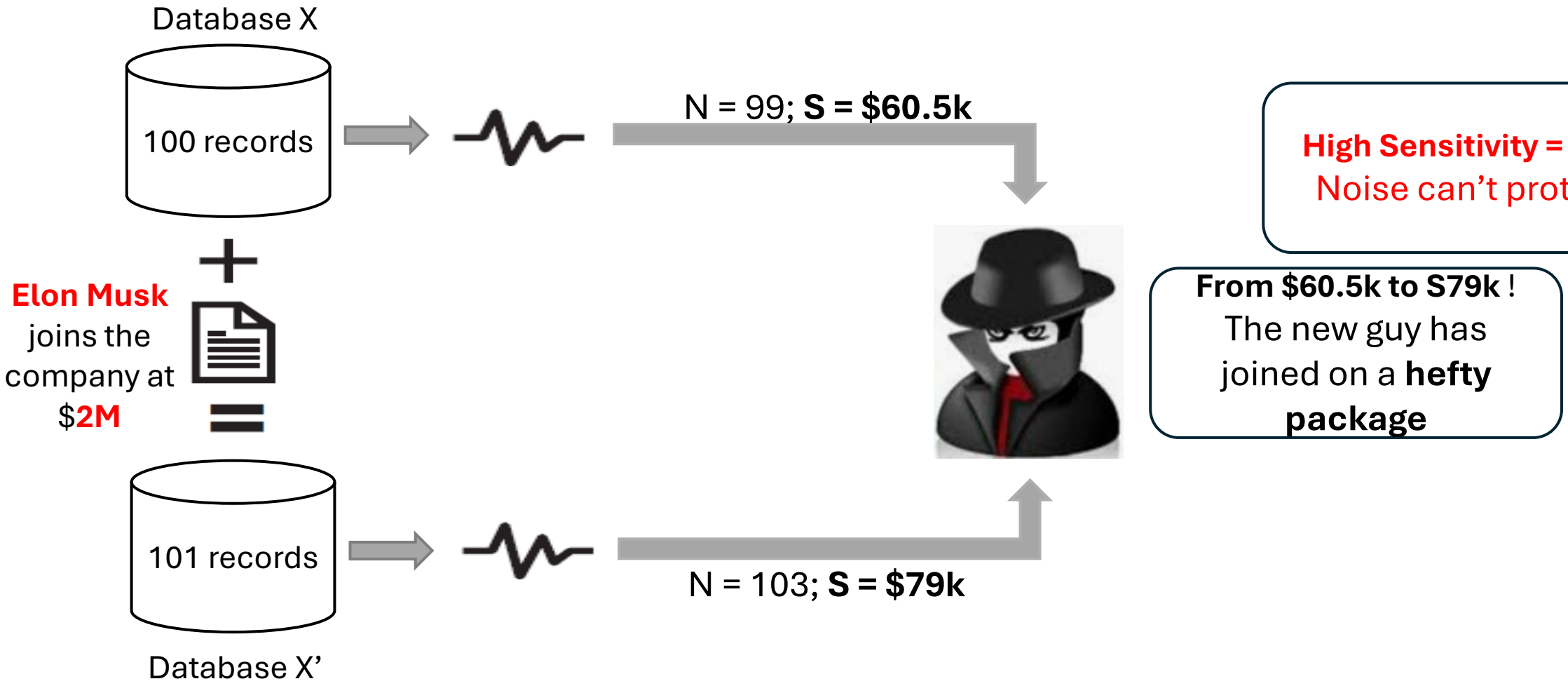



**Figure: Large amount of noise is required to protect highly sensitive data.**

# Differential Privacy: Framework

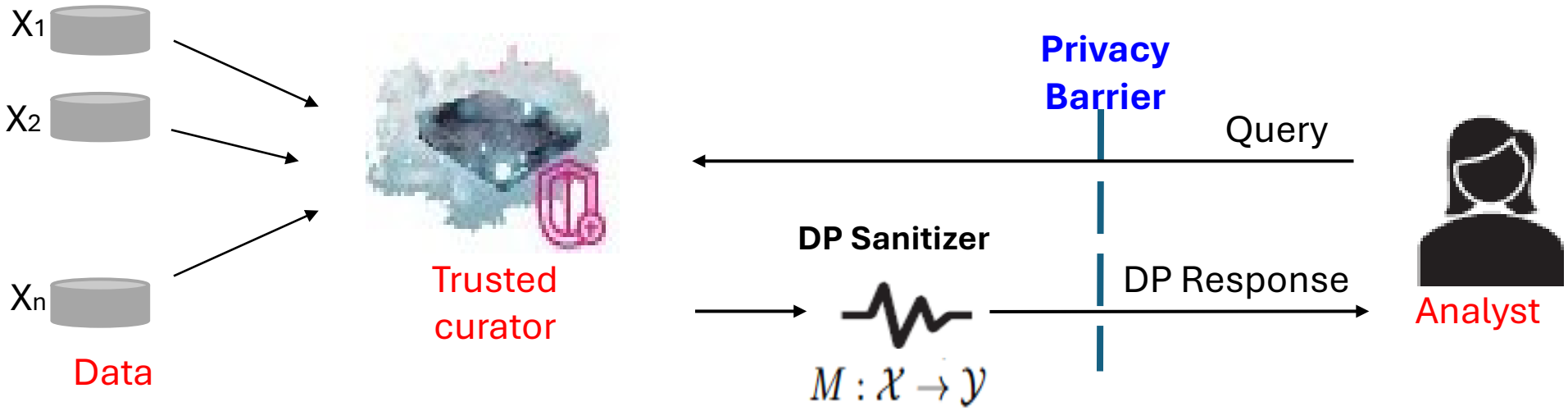



**Figure: Differential privacy framework.**

# Formal Definition

- An algorithm $M$ is $(\epsilon, \delta)$-DP if for all neighboring databases $X$ and $X'$ and for all events $E$

$$P[M(X) \in E] \leq e^{\epsilon} P[M(X') \in E] + \delta$$

# Formal Definition

- An algorithm $M$ is $(\epsilon, \delta)$-DP if for all neighboring databases $X$ and $X'$ and for all events $E$

$$P[M(X) \in E] \leq e^{\epsilon} P[M(X') \in E] + \delta$$

- Limits how far apart the probabilities of outcomes for neighboring datasets can be

# Formal Definition

- An algorithm $M$ is $(\epsilon, \delta)$-DP if for all neighboring databases $X$ and $X'$ and for all events $E$

$$P[M(X) \in E] \leq e^{\epsilon} P[M(X') \in E] + \delta$$

- Limits how far apart the probabilities of outcomes for neighboring datasets can be

# Formal Definition

- An algorithm $M$ is $(\epsilon, \delta)$-DP if for all neighboring databases $X$ and $X'$ and for all events $E$

$$P[M(X) \in E] \leq e^{\epsilon} P[M(X') \in E] + \delta$$

- Limits how far apart the probabilities of outcomes for neighboring datasets can be
  - With a small probability for failure
- $\epsilon \geq 0, \delta \geq 0$ define the *privacy budget*
- **Low** $\epsilon$, $\delta$ imply **more privacy**

# The Gaussian Mechanism for Differential Privacy

- DP output:
  - $M(X) = q(X) + N(0, \sigma^2)$

- Gaussian noise $\sigma^2$ is
  - Proportional to the **sensitivity** of the database w.r.t the query function
  - Decreases with $\epsilon$ and $\delta$

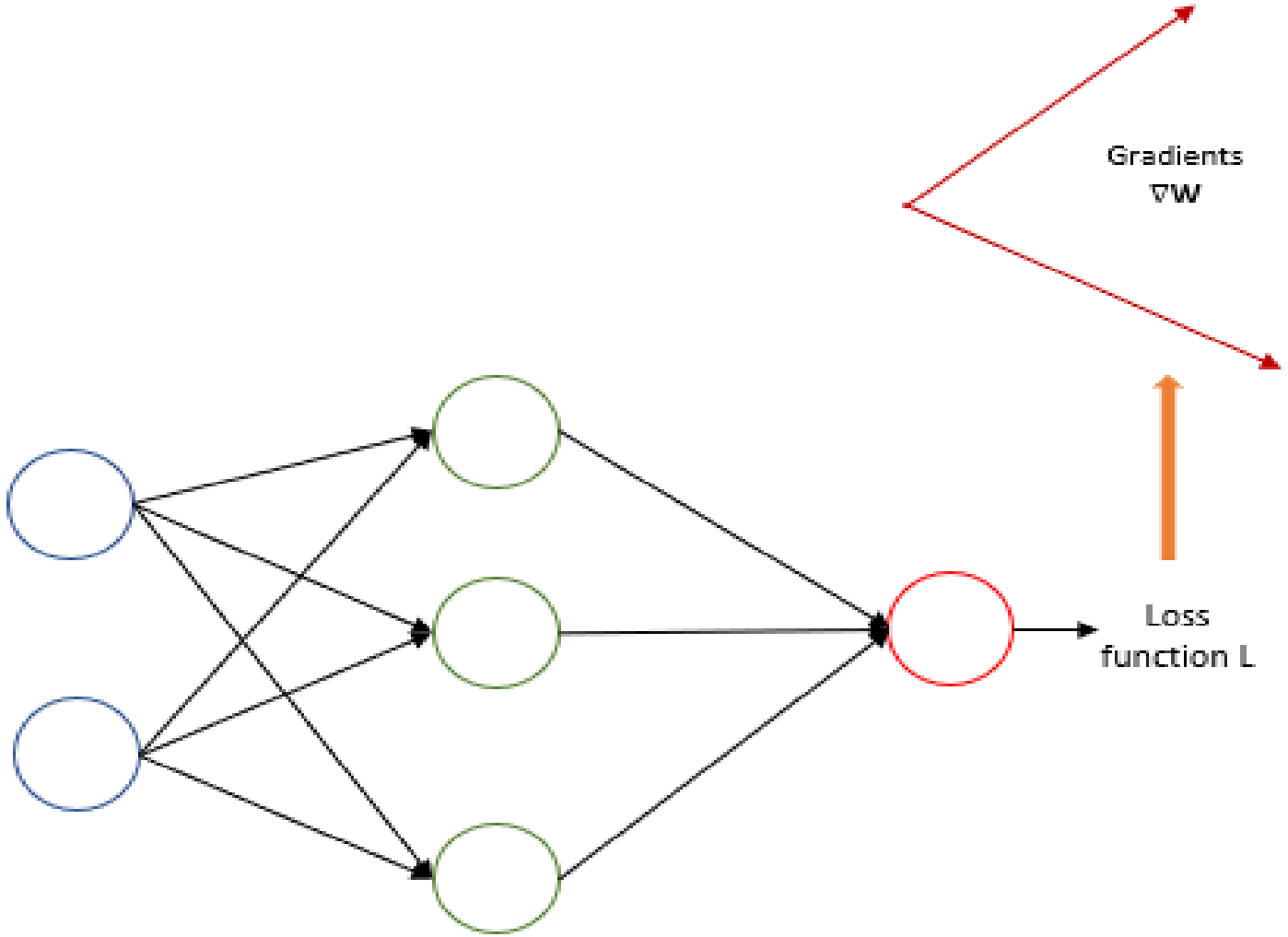
- Gaussian Mechanism
  - Very suitable when a lot of queries are made on the same dataset e.g., in ML applications

# Differential Privacy in Machine Learning

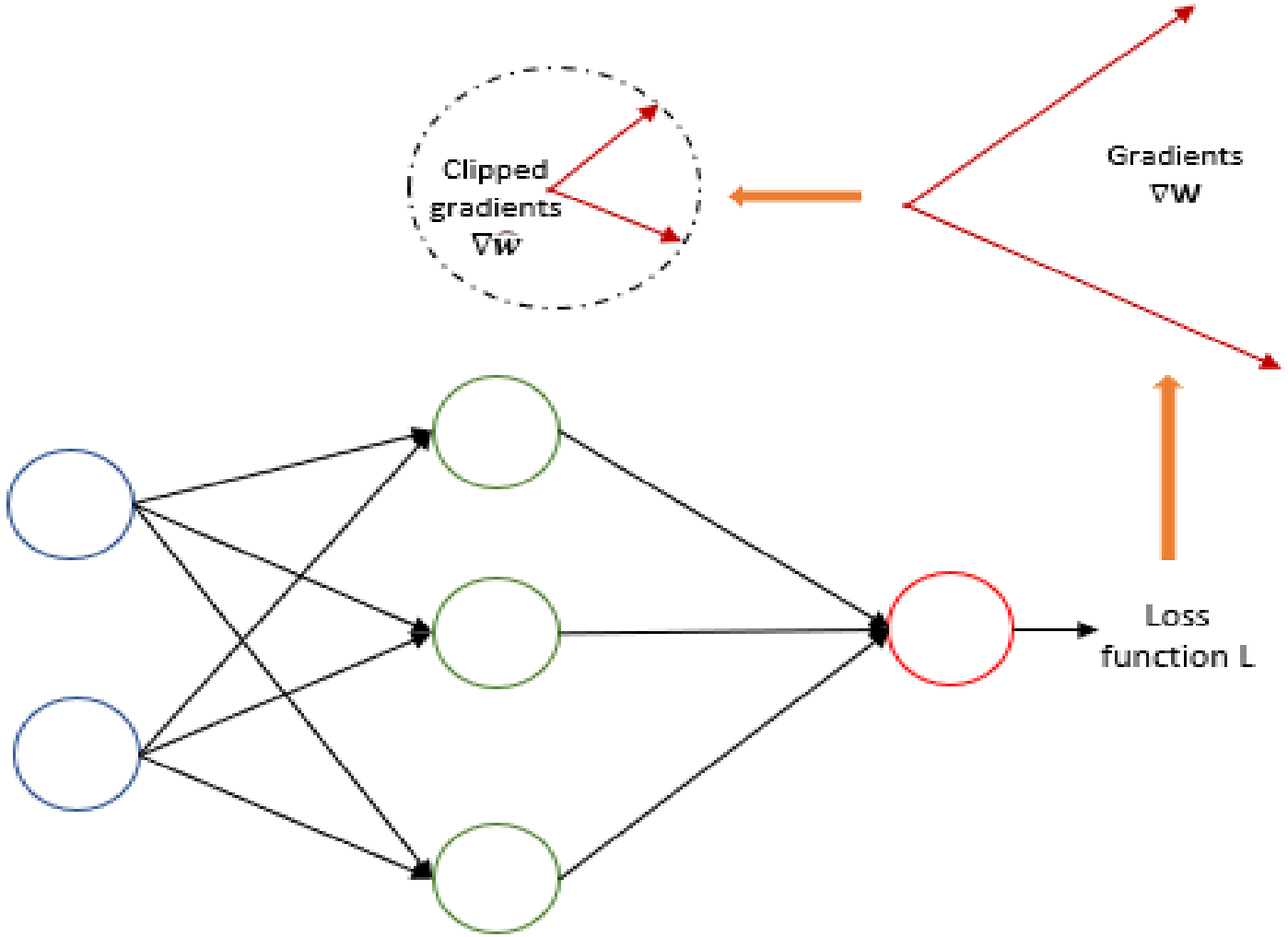
- Three main ways:
  - Privatizing input data
  - Training with differential privacy
  - Privatizing output

- Property: Differential Privacy Composition
  - Privacy degrades when the same query is repeated on the same sensitive dataset ($\epsilon\uparrow$, $\delta\uparrow$)
  - Advanced Composition Theorem: If one query is $\epsilon$-DP, then k queries are $\sqrt{\boldsymbol{k}}\,\epsilon$-DP (sublinear increase).

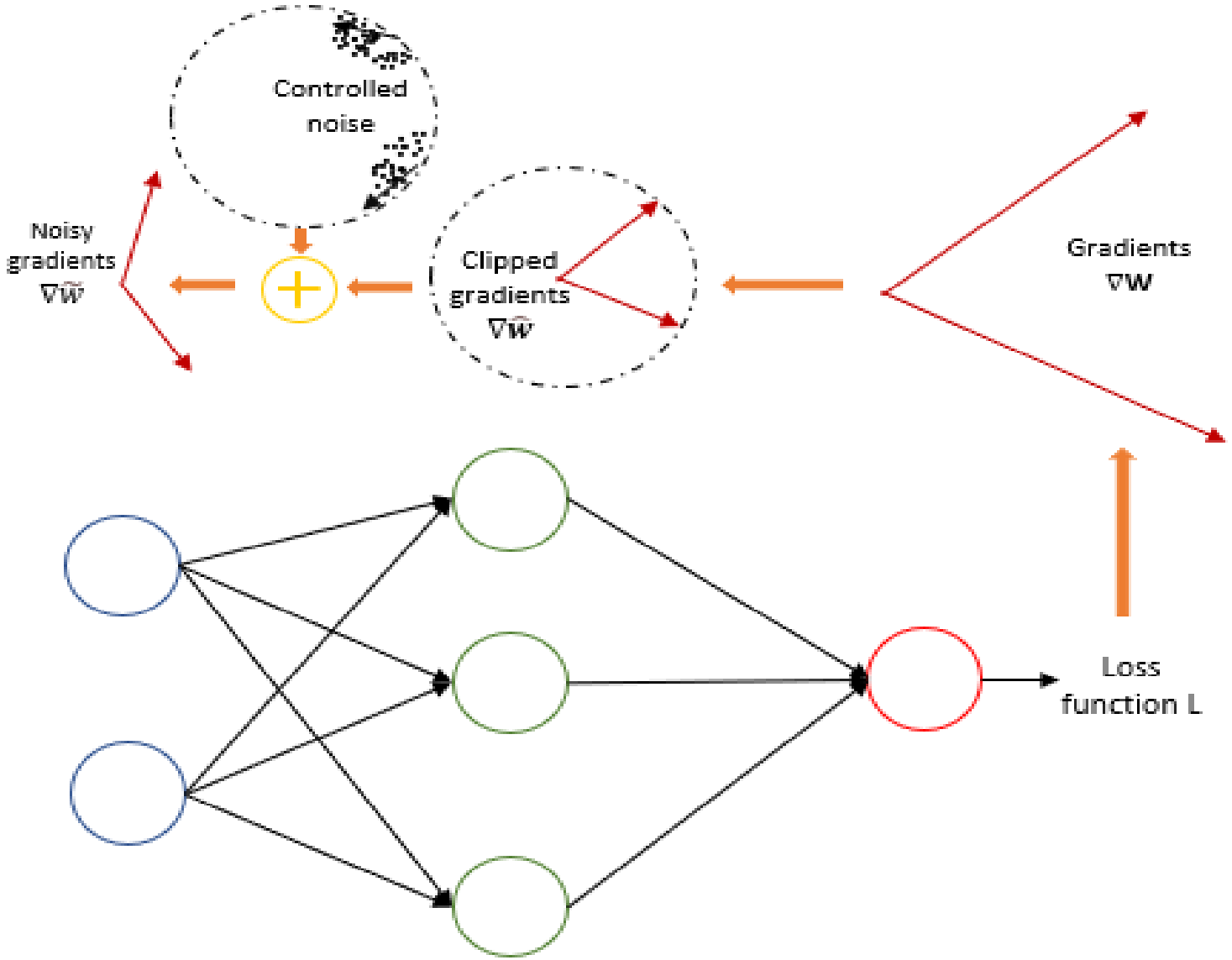
# Differentially Private Stochastic Gradient Descent (DP-SGD)



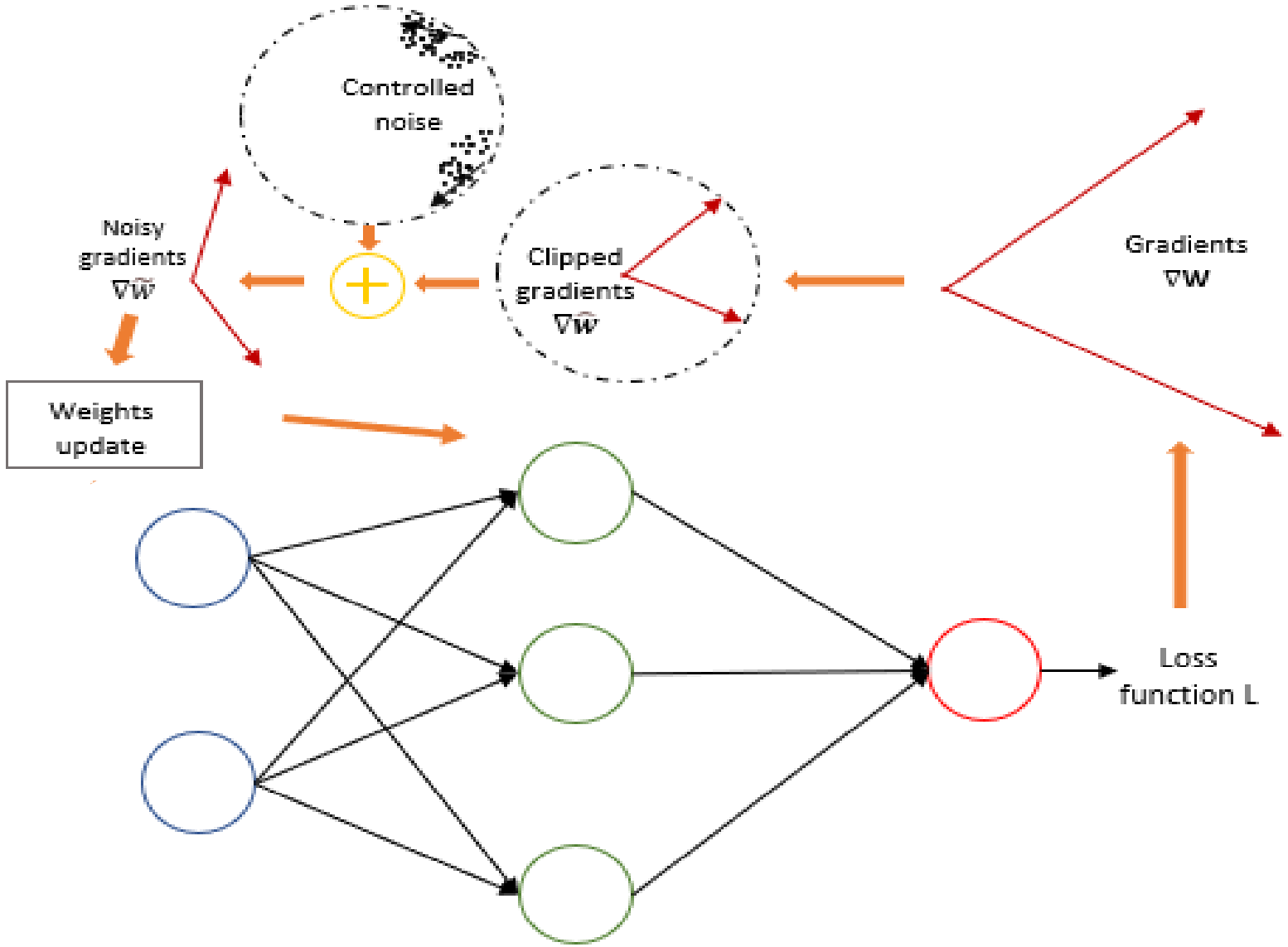
# Differentially Private Stochastic Gradient Descent (DP-SGD)



# Differentially Private Stochastic Gradient Descent (DP-SGD)



# Differentially Private Stochastic Gradient Descent (DP-SGD)



# Demo: Tools for Differentially Private Machine Learning

- TensorFlow Privacy Tutorial

Code Available at: https://colab.research.google.com/drive/1WuxP1ON5291dqlWiUEETUuxaqgWQM4TM?usp=sharing

- PyTorch Opacus Tutorial

Code Available at: https://colab.research.google.com/drive/12kKdAyi2NywLHH-8O_ywFyFwx1xjTgLA?usp=sharing

# Measuring Privacy Using Attacks

We can measure privacy indirectly, by measuring success of attacks.

Types of attack environments:

- White-Box Attack:
  - The attacker has full access to the model's architecture and parameters.
  - Allows precise, tailored attacks due to high-level insight.
- Black-Box Attack:
  - The attacker only has access to the model's inputs and outputs.
  - Relies on testing multiple inputs to infer information.

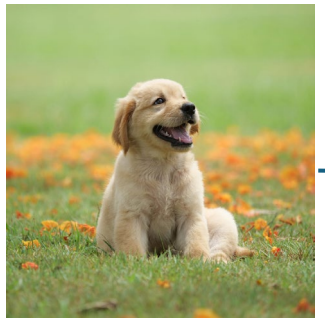
# Black Box Membership Inference Attack

(ML-Leaks, Salem et al.)

Simplest example:

- Test your model on training and test data
- Attempt to identify train samples from model outputs.
- Membership can be identified using a threshold (ex. >95%)

Input:



Black Box Model (API)

| |
|---|
| 0.99 |
| 0.32 |
| 0.12 |
| 0.52 |
| 0.51 |
| 0.21 |
| 0.13 |
| 0.05 |

**Sample is overly confident!**

We can reason that this was a member of the training data, identifying membership!

**Full Procedure:**

Measure attack accuracy on a uniform dataset. (train vs test)

Highly accurate attacks indicate low model privacy.

# Conclusion & Take-Aways

- Why Privacy Matters in AI
  - Safeguards personal and sensitive information
  - Ensures compliance with international regulations
  - Builds transparency and trust in data-sharing practices
- Key Techniques for Privacy-Preserving AI
  - ***Federated Learning*** for decentralized training (e.g., Flower)
  - ***Differential Privacy*** for data anonymization (e.g., TF-Privacy, Opacus)
- Challenges and Opportunities
  - Measuring privacy remains complex but achievable using attack efficacy as a benchmark
  - Private Machine Learning has vast potential for improvements in safety, efficiency, and performance

# Thank you!

# Questions?

Special Thanks to:

Dr. Sen-Ching Cheung, Luna Zhu and Dongjie Chen

# References:

[1]M. Fredrikson, S. Jha, and T. Ristenpart, ‘Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures’, in Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, Colorado, USA, 2015, pp. 1322–1333.

[2]A. Salem, Y. Zhang, M. Humbert, P. Berrang, M. Fritz, and M. Backes, ‘ML- Leaks: Model and Data Independent Membership Inference Attacks and Defenses on Machine Learning Models’, arXiv [cs.CR]. 2018.

Flower https://flower.ai/

TF-Privacy https://github.com/tensorflow/privacy

PyTorch Opacus https://github.com/pytorch/opacus