

Privacy By Design: Differential Privacy For Secure Machine Learning

About Me

Outline

- Motivation For Privacy-Preserving ML
- Data Privacy Problem
- Need for Computational Privacy
- Plausible Deniability (Randomized Response)
- Differential Privacy
- Mechanisms for Achieving Differential Privacy
- Applications and Challenges
- Conclusion and Q&A

With Great Data, Comes Great Responsibility!

The exponential growth of machine learning (ML) hinges on data, but its acquisition often raises privacy concerns due to the potential exposure of sensitive information.

Thus, the challenge lies in balancing the potential benefits of machine learning with the risks of compromising user privacy.

With Great Data, Comes Great Responsibility!

Imagine unlocking this:

- ✨ Genomic secrets that could cure diseases
- ✨ Predictive insights to eliminate supply chain waste
- ✨ Chevrons of untapped energy sources

The potential for AI-driven breakthroughs remain locked away

- Trillion-dollar industries can't leverage valuable data for AI due to privacy concerns and strict regulations.
- To advance energy, healthcare, and collaboration, AI systems must train and infer on data while ensuring end-to-end privacy.

Differential Privacy (DP), a privacy-preserving machine learning (**PPML**) mechanism could help unlock these new possibilities.

Motivation

Why Differential Privacy?

Releasing “too many” statistics that are “too accurate” necessarily makes one vulnerable to:

- **Database Reconstruction:** reconstructing almost the entire underlying dataset [Dinur-Nissim '03,...]
 - Applied in practice to Census releases [Garfinkel-Abowd-Martindale '18] and Diffix [Cohen-Nissim '19].
- **Membership Inference:** determining whether a target individual is in the dataset [Dwork-Smith-Steinke-Ullman-V. '15]
 - Applied in practice to genomic data [Homer et al. '08,...] and ML as a service [Shokri et al. '17,...]

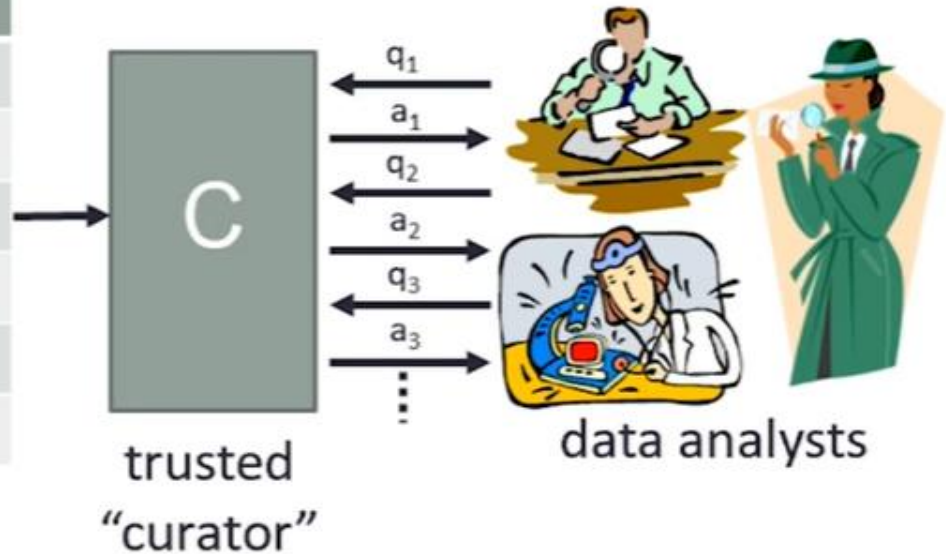
Statistical Releases

Name	Sex	Blood	...	HIV?
Chen	F	B	...	Y
Jones	M	A	...	N
Smith	M	O	...	N
Ross	M	O	...	Y
Lu	F	A	...	N
Shah	M	B	...	Y



Statistical Query Systems

Name	Sex	Blood	...	HIV?
Chen	F	B	...	Y
Jones	M	A	...	N
Smith	M	O	...	N
Ross	M	O	...	Y
Lu	F	A	...	N
Shah	M	B	...	Y




Data Privacy Problem

Data Privacy: The Problem

- Given a dataset with sensitive information, such as:
 - Census data
 - Health records
 - Social network activity
 - Telecommunications data
- How can we:
 - enable desirable uses of the data
 - while protecting the privacy of the data subjects?

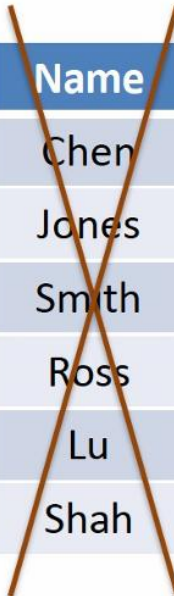
- 
- Informing policy
 - Identifying subjects for drug trial
 - Searching for terrorists
 - Market analysis

Approach 1: Encrypt the Data

Name	Sex	Blood	...	HIV?		Name	Sex	Blood	...	HIV?
Chen	F	B	...	Y		100101	001001	110101	...	110111
Jones	M	A	...	N		101010	111010	111111	...	001001
Smith	M	O	...	N		001010	100100	011001	...	110101
Ross	M	O	...	Y		001110	010010	110101	...	100001
Lu	F	A	...	N		110101	000000	111001	...	010010
Shah	M	B	...	Y		111110	110010	000101	...	110101

Problems: How to search over data or compute statistics? Who has the encryption key?

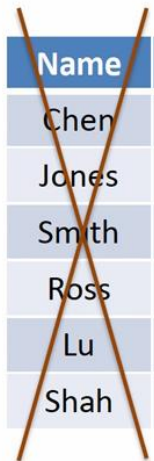
Approach 2: Anonymize the Data



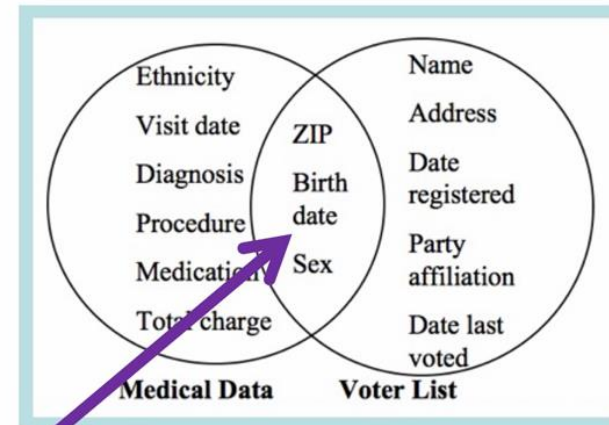
Name	Sex	Blood	...	HIV?
Chen	F	B	...	Y
Jones	M	A	...	N
Smith	M	O	...	N
Ross	M	O	...	Y
Lu	F	A	...	N
Shah	M	B	...	Y

Problems?

Reidentification via Linkage



Name	Sex	Blood	...	HIV?
Chen	F	B	...	Y
Jones	M	A	...	N
Smith	M	O	...	N
Ross	M	O	...	Y
Lu	F	A	...	N
Shah	M	B	...	Y



[Sweeney '97]

Uniquely identify > 60% of the US population [Sweeney '00, Golle '06]

All it takes is a knowledge of a small number of attributes to identify/name the person!

Netflix Challenge Re-Identification

[Narayanan-Shmatikov '08]

thumbs up		thumbs down	thumbs up		
	thumbs up				
thumbs up		thumbs down		thumbs up	thumbs up
thumbs up			thumbs down		
	thumbs up		thumbs down	thumbs down	
		thumbs down	thumbs up		

Anonymized
NetFlix data

+

thumbs up			thumbs up		
	thumbs up				
thumbs up					thumbs up
thumbs up			thumbs down		
				thumbs down	
		thumbs down			

Public, incomplete
IMDB data

Alice
Bob
Charlie
Danielle
Erica
Frank

=

thumbs up		thumbs down	thumbs up		
	thumbs up				
thumbs up		thumbs down		thumbs up	thumbs up
thumbs up			thumbs down		
	thumbs up		thumbs down	thumbs down	
		thumbs down	thumbs up		

Identified NetFlix Data

Alice
Bob
Charlie
Danielle
Erica
Frank

How
many
movies
required
on
average to
uniquely
identify a
user?

Four!

Narayanan-Shmatikov Set-Up

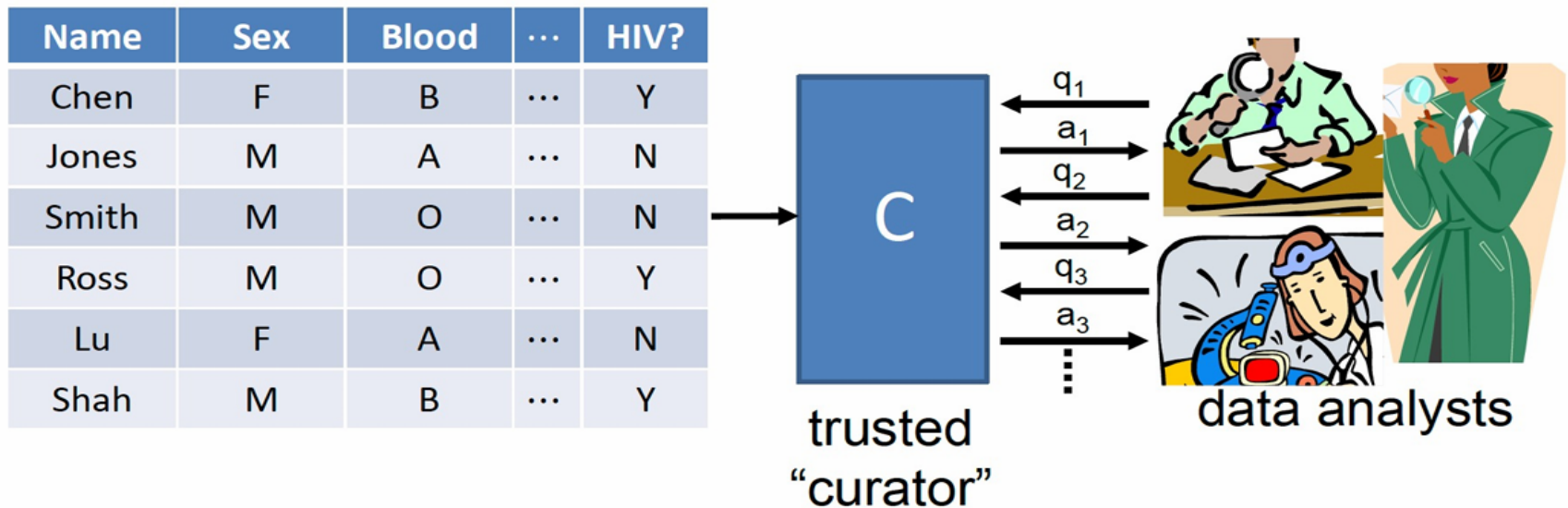
- Dataset: x = set of records (e.g., Netflix ratings)
- Adversary's inputs:
 - x' = subset of records from x , distorted slightly
 - aux = auxiliary information about a record $r \in D$ (e.g., a particular identifiable user's IMDB ratings)
- Adversary's goal: output either
 - $r' \in x'$ = record that is “close” to r , or
 - \perp = failed to find a match

Narayanan-Shmatikov Results

- For the \$1m Netflix Challenge, a dataset of 5,00,000 subscribers' ratings (less than 1/10 of all subscribers) was released (total of 100m ratings over 6 years).
- Out of 50 sampled IMDB users, two standouts were found, with eccentricities of 28 and 15.
- Reveals all movies watched from only those publicly rated on IMDB.
- Class action lawsuit, cancelling of Netflix Challenge II.

Message: Any attribute can be a “quasi-identifier”

Approach 3: Mediate Access



Problems: Curator sees all the data. What queries are allowed? How much do they leak?

Why Is Anonymization Hard?

Anonymization Is Hard

Why Is Anonymization Hard?

Some examples of anonymization failures (taken from *The Ethical Algorithm*)

- In the 1990s, a government agency released a database of medical visits, stripped of identifying information (names, addresses, social security numbers)
 - But it did contain zip code, birth date, and gender.
 - Researchers estimated that 87 percent of Americans are uniquely identifiable from this triplet.
- Netflix Challenge (2006), a Kaggle-style competition to improve their movie recommendations, with a \$1 million prize
 - They released a dataset consisting of 100 million movie ratings (by “anonymized” numeric user ID), with dates
 - Researchers found they could identify 99% of users who rated 6 or more movies by cross-referencing with IMDB, where people posted reviews publicly with their real names

Anonymization Is Hard

Why Is Anonymization Hard?

Not sufficient to prevent unique identification of individuals.

Name	Age	Gender	Zip Code	Smoker	Diagnosis
*	60–70	Male	191**	Y	Heart disease
*	60–70	Female	191**	N	Arthritis
*	60–70	Male	191**	Y	Lung cancer
*	60–70	Female	191**	N	Crohn's disease
*	60–70	Male	191**	Y	Lung cancer
*	50–60	<i>Female</i>	191**	N	HIV
*	50–60	Male	191**	Y	Lyme disease
*	50–60	Male	191**	Y	Seasonal allergies
*	50–60	<i>Female</i>	191**	N	Ulcerative colitis

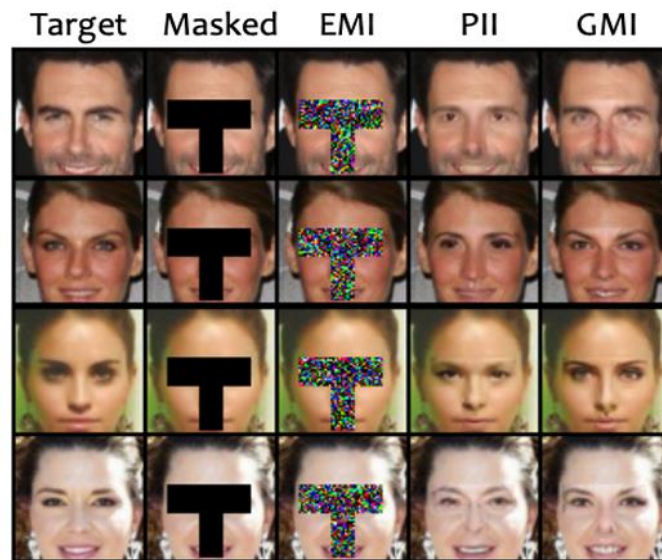
Kearns & Roth, *The Ethical Algorithm*

From this (fictional) hospital database, if we know Rebecca is 55 years old and in this database, then we know she has 1 of 2 diseases.

Anonymization Is Hard

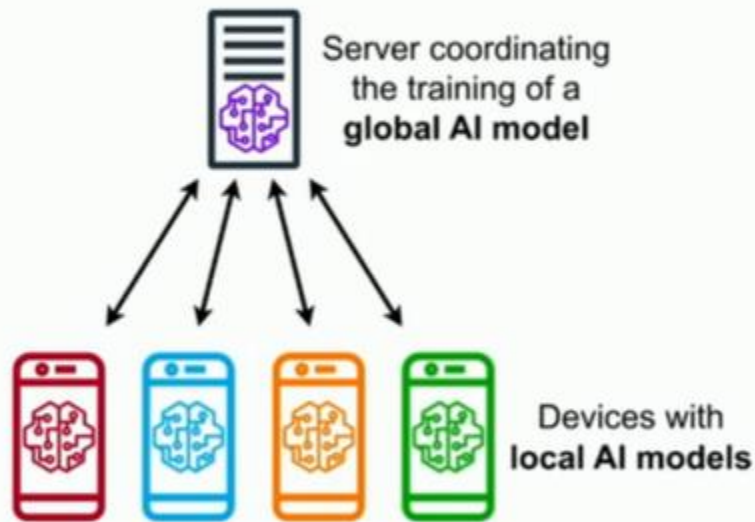
Why Is Anonymization Hard?

- Even if you don't release the raw data, the weights of a trained network might reveal sensitive information.
- **Model inversion** attacks recover information about the training data from the trained model.
- Here's an example of reconstructing individuals from a face recognition dataset, given a classifier trained on this dataset and a generative model trained on an unrelated dataset of publicly available images.
- **Col 1:** training image. **Col 2:** prompt. **Col 4:** best guess from only public data. **Col 5:** reconstruction using classification network.
- **Source:** Zhang et al., "The secret revealer: Generative model-inversion attacks against deep neural networks." <https://arxiv.org/abs/1911.07135>



Anonymization Is Hard

Reconstruction in Federated Learning



https://en.wikipedia.org/wiki/Federated_learning



Figure 1: Reconstruction of an input image x from the gradient $\nabla_{\theta} \mathcal{L}_{\theta}(x, y)$. Left: Image from the validation dataset. Middle: Reconstruction from a trained ResNet-18 trained on ImageNet. Right: Reconstruction from a trained ResNet-152. In both cases, the intended privacy of the image is broken. Note that previous attacks cannot recover either ImageNet-sized data [35] or attack trained models.

Geiping et al. 2020, NeurIPS, "Inverting Gradients - How easy is it to break privacy in federated learning?"

Anonymization Is Hard

Why Is Anonymization Hard?

- A neural net language model trained on Linux source code learned to output the exact text of the GPL license.

```
/*  
 * Copyright (c) 2006-2010, Intel Mobile Communications. All rights reserved.  
 *  
 * This program is free software; you can redistribute it and/or modify it  
 * under the terms of the GNU General Public License version 2 as published by  
 * the Free Software Foundation.  
 *  
 * This program is distributed in the hope that it will be useful,  
 * but WITHOUT ANY WARRANTY; without even the implied warranty of  
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
```

<http://karpathy.github.io/2015/05/21/rnn-effectiveness/>

- Gmail uses language models for email autocompletion. Imagine if the autocomplete feature spits out the entire text of one of your past emails.

Anonymization Is Hard

Memoization

Extracting Training Data from Large Language Models

Nicholas Carlini¹ Florian Tramèr² Eric Wallace³ Matthew Jagielski⁴
Ariel Herbert-Voss^{5,6} Katherine Lee¹ Adam Roberts¹ Tom Brown⁵
Dawn Song³ Úlfar Erlingsson⁷ Alina Oprea⁴ Colin Raffel¹
¹Google ²Stanford ³UC Berkeley ⁴Northeastern University ⁵OpenAI ⁶Harvard ⁷Apple

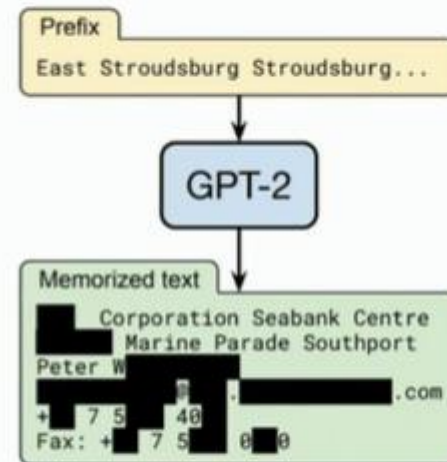


Figure 1: **Our extraction attack.** Given query access to a neural network language model, we extract an individual person's name, email address, phone number, fax number, and physical address. The example in this figure shows information that is all accurate so we redact it to protect privacy.

Anonymization Is Hard

Why Is Anonymization Hard?

- It's hard to guess what capabilities attackers will have, especially decades into the future.
 - **Analogy with crypto:** Cryptosystems today are designed based on what quantum computers might be able to do in 30 years.
 - To defend against unknown capabilities, we need mathematical guarantees.
- **Want to guarantee:** no individual is directly harmed (e.g. through release of sensitive information) by being part of the database, even if the attacker has tons of data and computation.

The Need For Computational Privacy

Need for Computational Privacy

Goal: Privacy-Preserving Machine Learning

Motivating Example: Healthcare Data Analysis

Adversary:

- Membership inference attack
- Data reconstruction attack
- Linkage attack

Intuition: Uncertainty in the process means uncertainty for the attacker.

We need a mathematical guarantee on the "**process**" which helps us quantify and upper bound our loss of privacy.

Example: A statistical analysis which learns that smoking causes cancer

There are 2 level of harms we could associated to each smoker:

Harm 1: Harm caused by smoking – what statistical analysis can help with.

Harm 2: Harm caused by insurance companies becoming aware that person X is a smoker – higher insurance fee.

We want to learn that "**smoking causes cancer**" in order to eradicate **harm 1**, without causing **harm 2** to people in the process of data analysis.

- **Differential Privacy (DP)** will help us achieve that.

Randomized Response

Concept: A survey technique ensuring some level of privacy through randomization.

Example: Coin flip mechanism to provide truthful or random answers.

NB: Randomized response is not the same as differential privacy, but it serves as a foundational concept on which differential privacy is built.

Randomized Response

Example: private surveys

Imagine you are a hospital, and need to count how many smokers there are (helps allocate resources).

People may not want to tell you: you might tell their insurance.

Big Question: How do we still see the forest but somehow, lose some resolution on the trees?

Idea: ask them to flip a coin

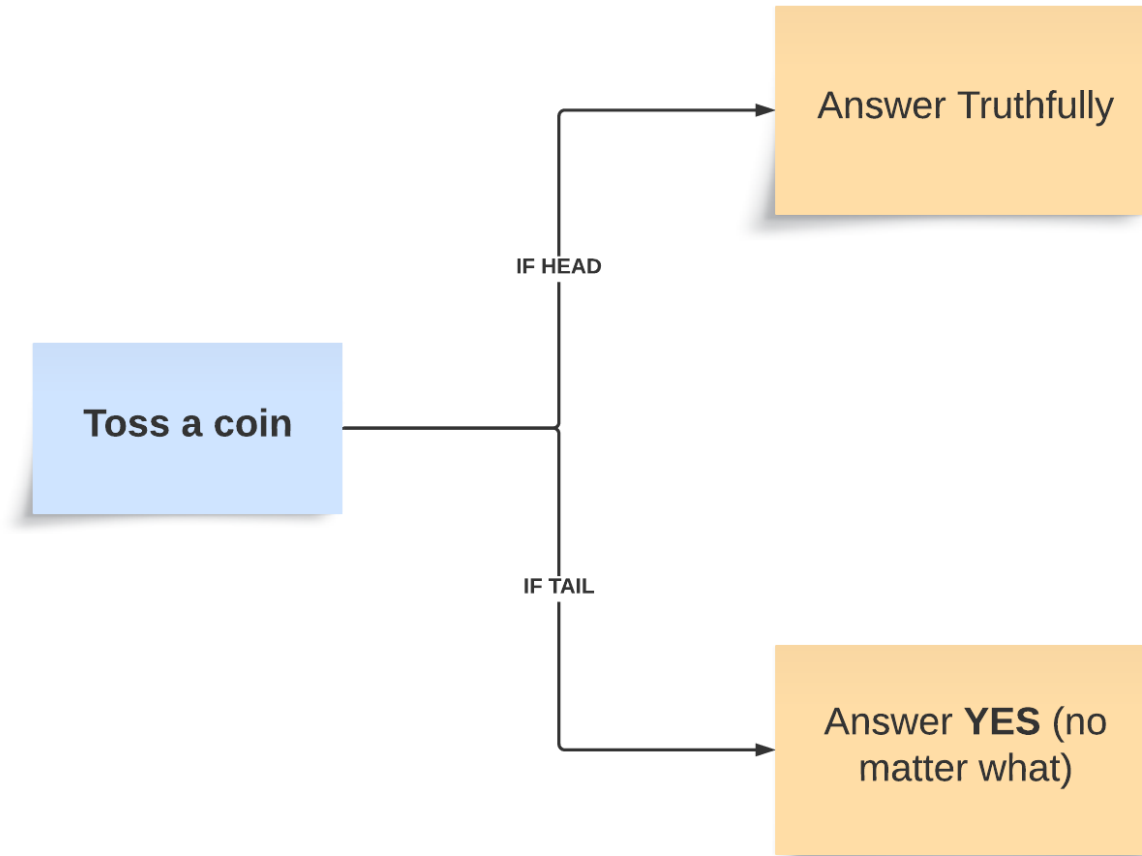


Answer honestly



Say **YES**, you are a smoker (no matter what)

Question: Are you a smoker?



Idea: ask them to flip a coin

YES: 650

1000 participants

NO: 350

Randomized Response

Idea: ask them to flip a coin

YES: 650

- 500



1000 participants

NO: 350

Randomized Response

Idea: ask them to flip a coin

YES: 650

- 500



1000 participants

- 500



NO: 350

Randomized Response

Idea: ask them to flip a coin

YES: 650

- 500



= 150/500 = 30%

1000 participants

- 500



NO: 350

Randomized Response

Idea: ask them to flip a coin

YES: 650

- 500



= 150/500 = 30%

1000 participants

- 500



150/650 = 23% actual smoker

NO: 350

Plausible deniability

- Any person who answers "Yes" has **plausible deniability**.
- This is because, even if the person is a smoker, they can credibly claim that their "Yes" response was due to the coin landing on tails, not because they were admitting to smoking. (**NO TRACE, NO CASE**)

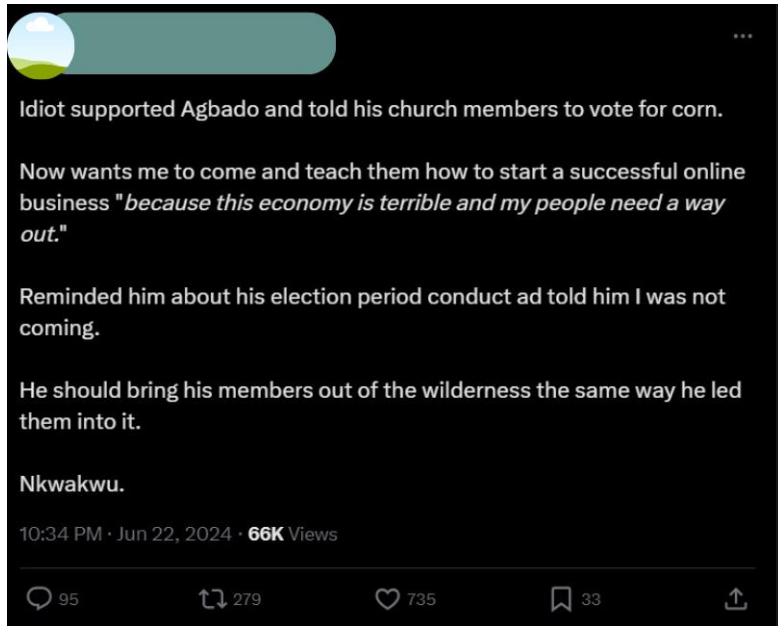
What Is Differential Privacy?

Differential Privacy

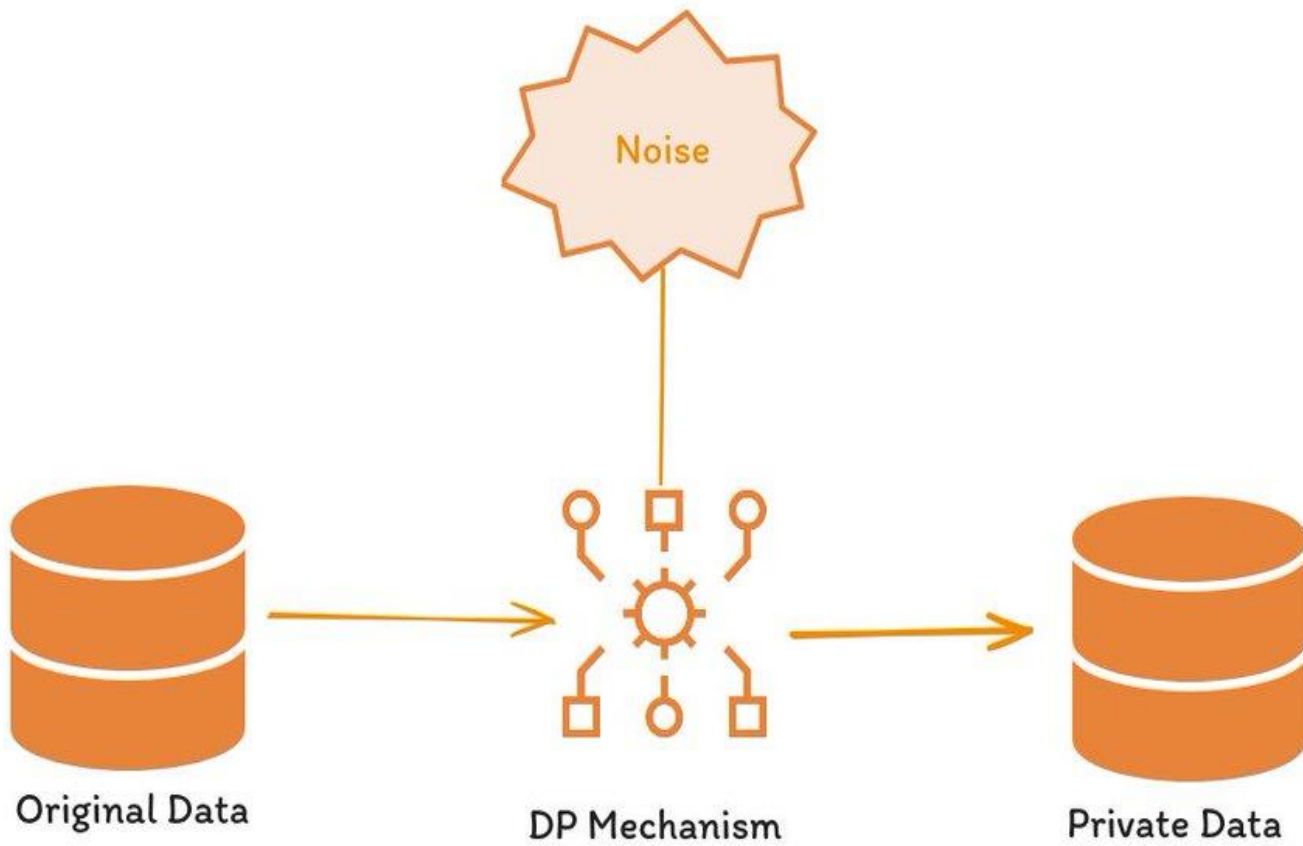
Differential Privacy is a system for publicly sharing information about a dataset which obscures individual contributions while retaining the big picture, by adding controlled random noise.

- Doesn't require attack modeling
- Privacy loss is quantifiable
- Compose multiple queries
- Accessible, minimal utility loss, easy to compute

Imagine a scenario like this

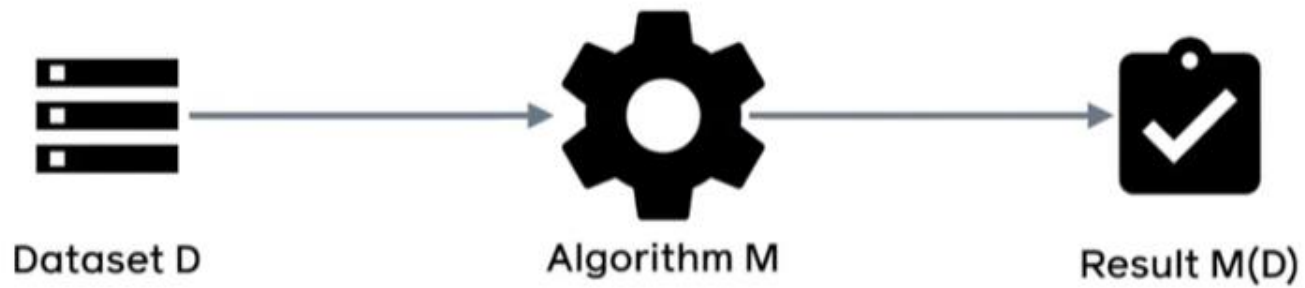


By applying differential privacy, organizations can better manage sensitive data related to individuals' privacy; example, political opinions and behaviors, as seen in the above tweets, ensuring that users' privacy is maintained while still benefiting from the collected data for broader analysis and insights.

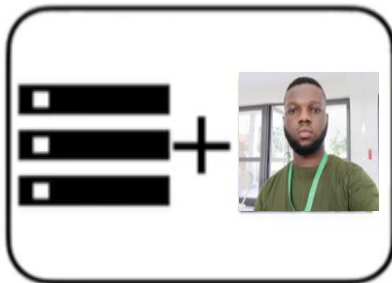
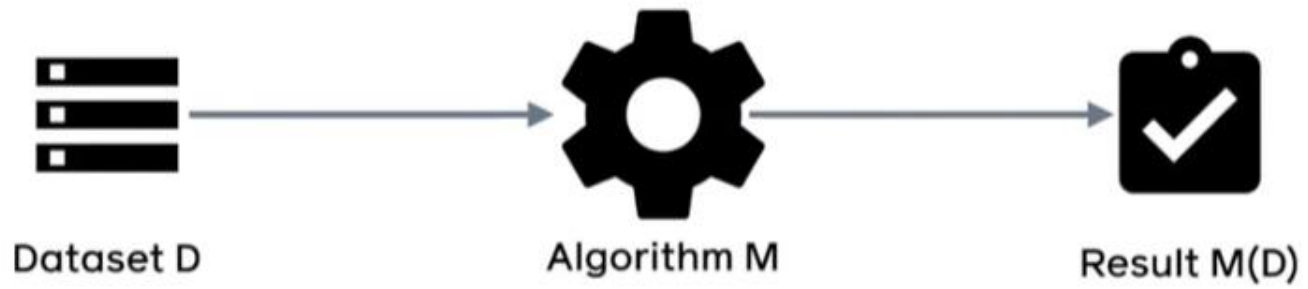


- **DP** ensures that if you have two datasets that are almost identical, except for one record, the probability of getting the same outcome from an algorithm applied to each dataset remains nearly unchanged.
- This means that even small changes in the data won't drastically affect the results.

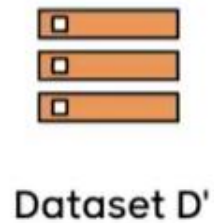
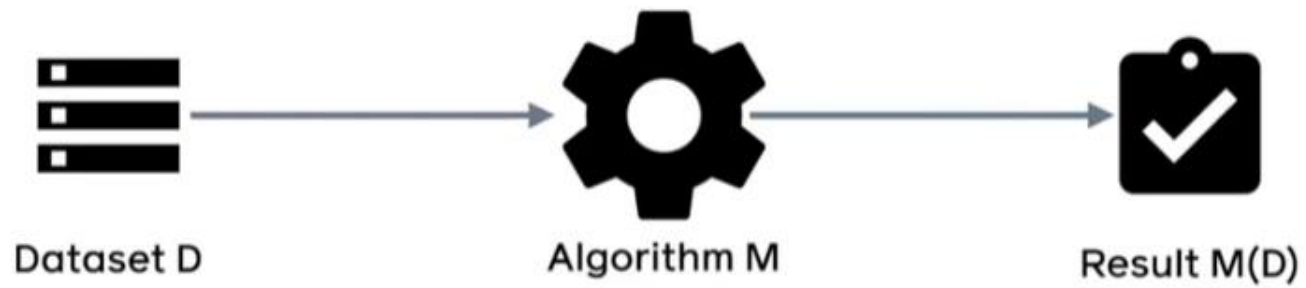
DP definition



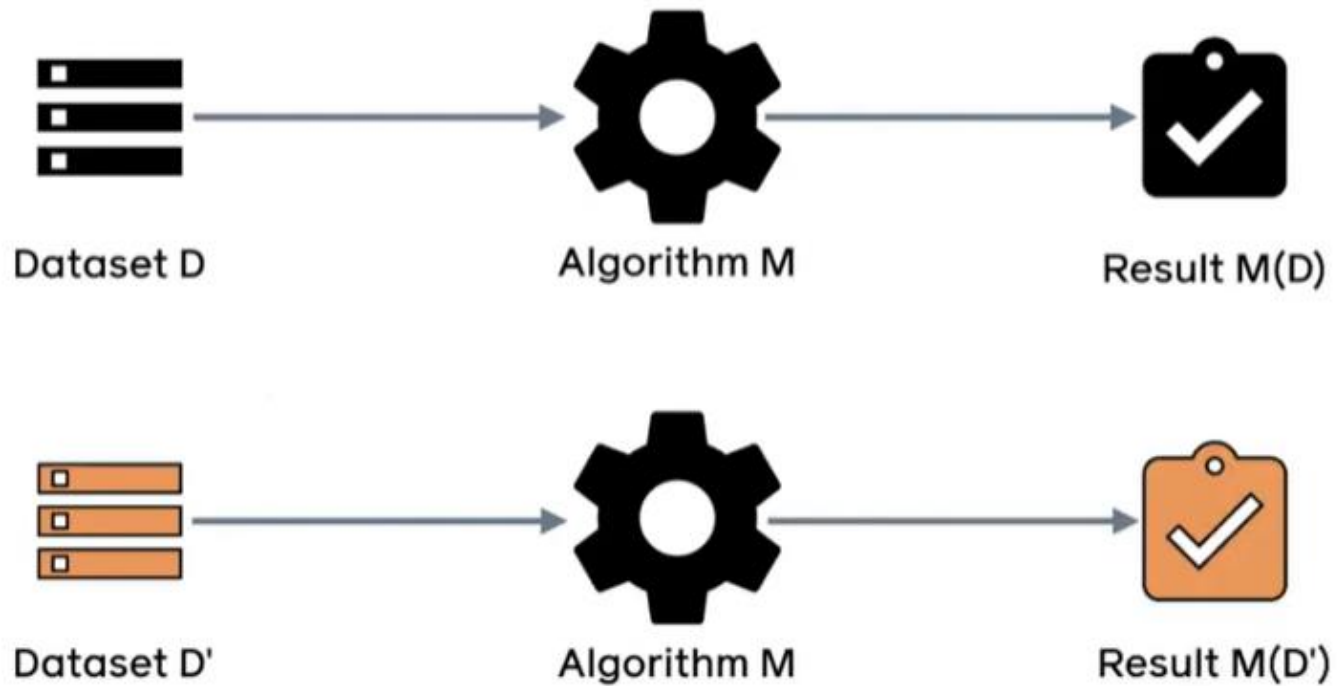
DP definition



DP definition



DP definition



DP definition



Almost
the same



DP definition

$$\forall \text{ adjacent } D, D' \forall x: \Pr[M(D) = x] \leq \exp(\epsilon) \cdot \Pr[M(D') = x] + \delta$$



Dataset D



Dataset D'



Algorithm M



Result M(D)



Result M(D')

DP definition

$$\forall \text{ adjacent } D, D' \forall x: \Pr[M(D) = x] \leq \exp(\epsilon) \cdot \Pr[M(D') = x] + \delta$$

Privacy (loss) budget



Dataset D



Dataset D'



Algorithm M



Result M(D)



Result M(D')

Epsilon (ϵ): is a parameter that measures the privacy guarantee.

- ϵ parameter quantifies the amount by which **Result M(D)** & **Result M(D')** differs
- A smaller ϵ = better privacy protection but more noise added to the data
- A larger ϵ = less noise added to the data but weaker privacy protection.

Delta (δ): a parameter representing the probability of the privacy guarantee failing.

Properties of DP

Two awesome properties

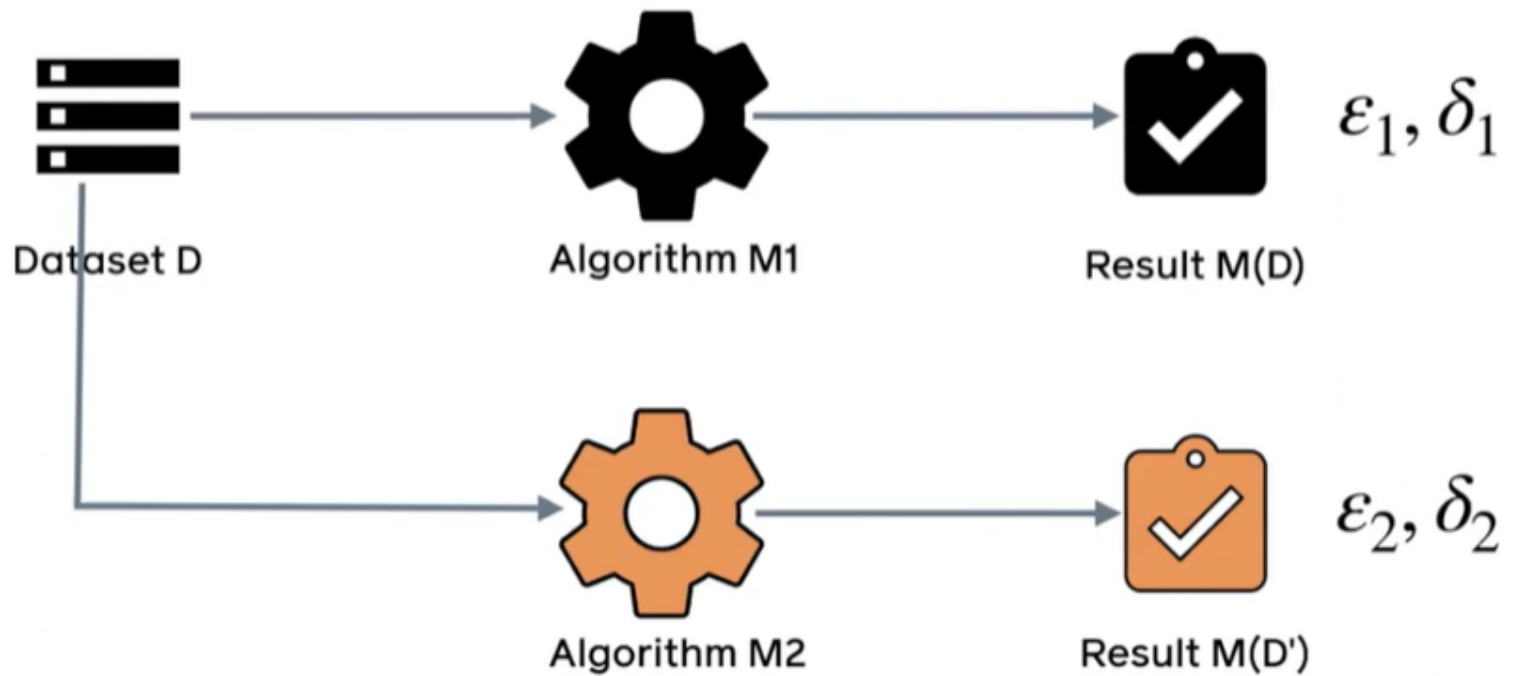
1. **Composition rule**
2. Robustness to **post-processing**

Properties of DP

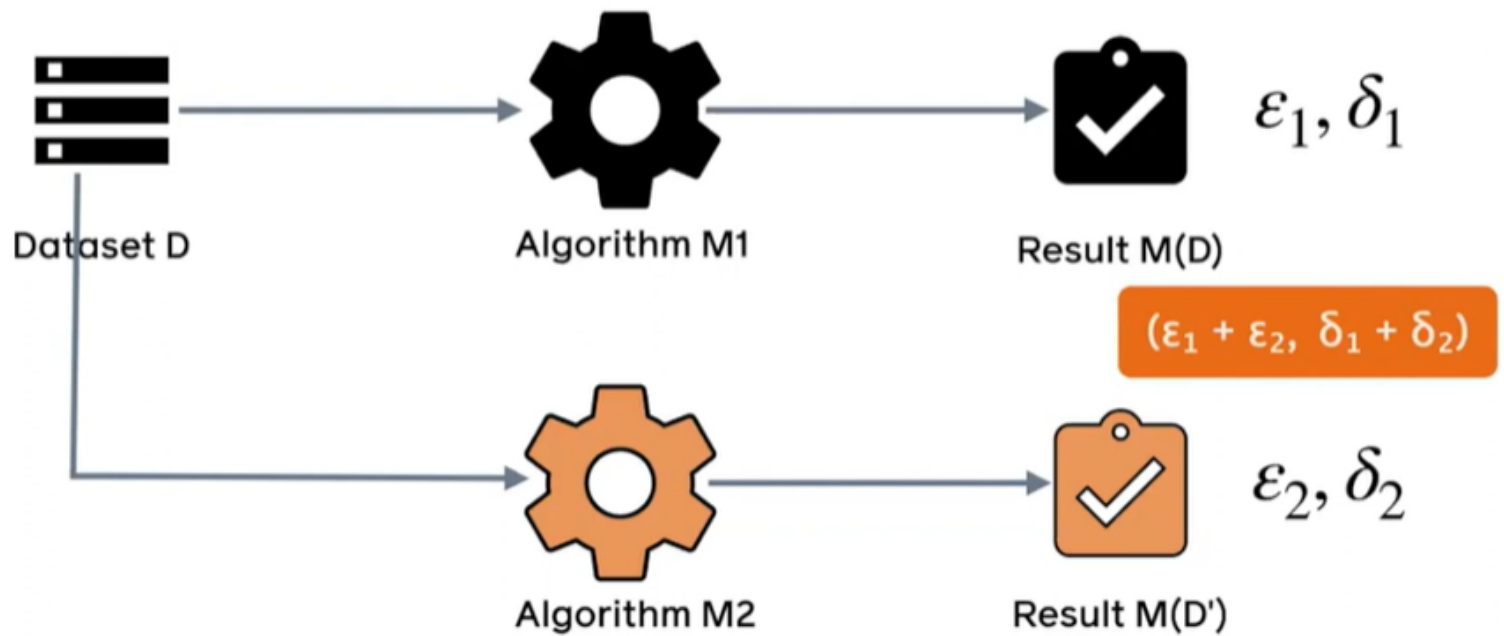
Composition rule



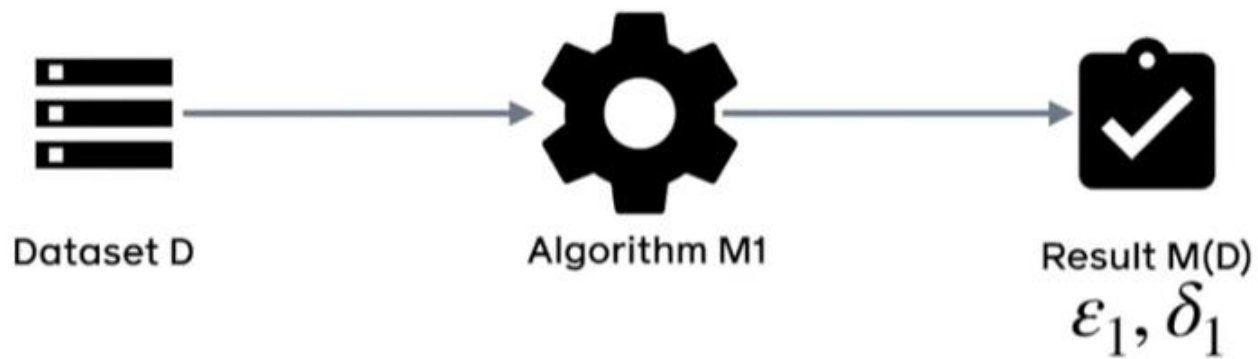
Composition rule



Composition rule



Post-processing



No matter the post processing

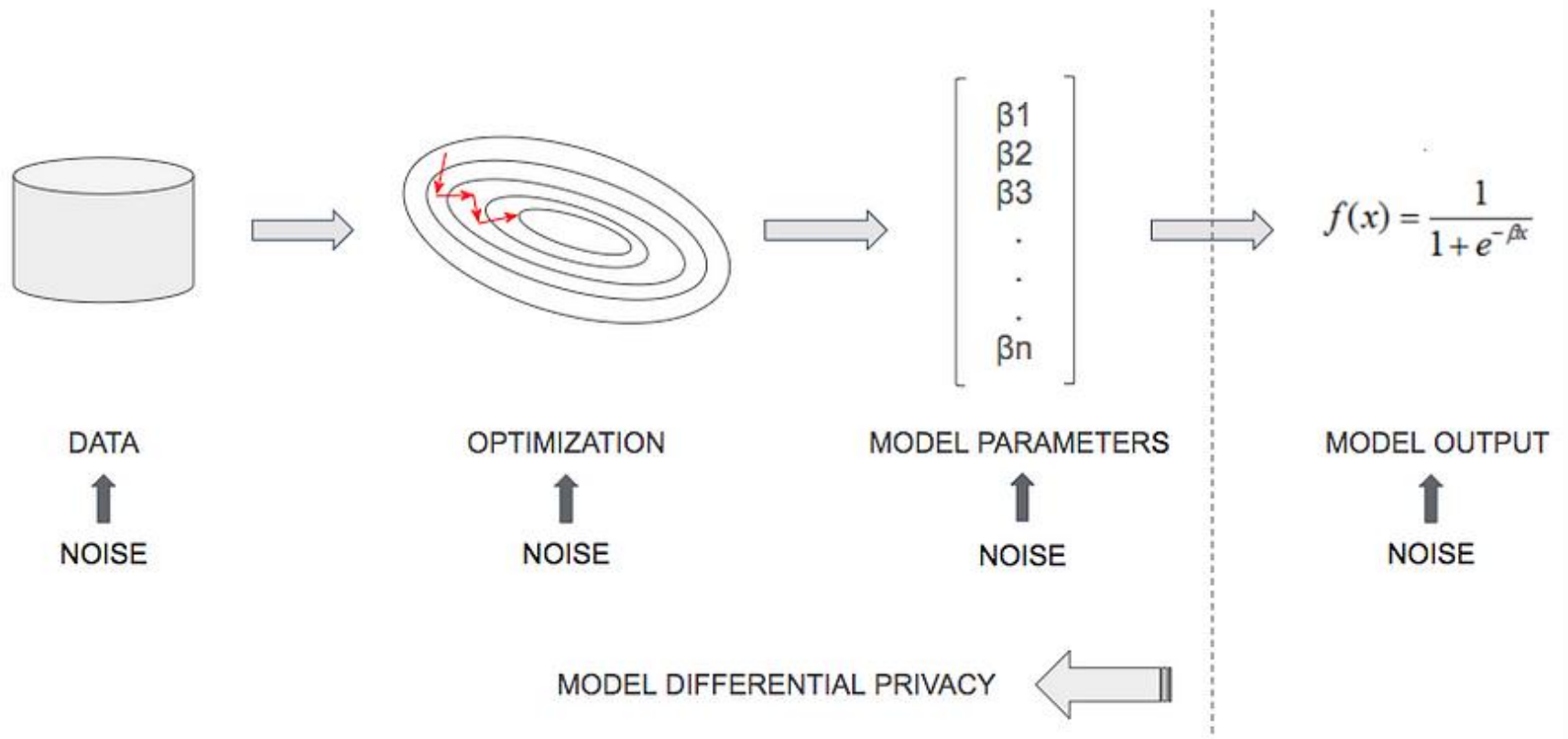
Differential Privacy in Machine Learning

DP in Machine Learning

Differential Privacy (DP)

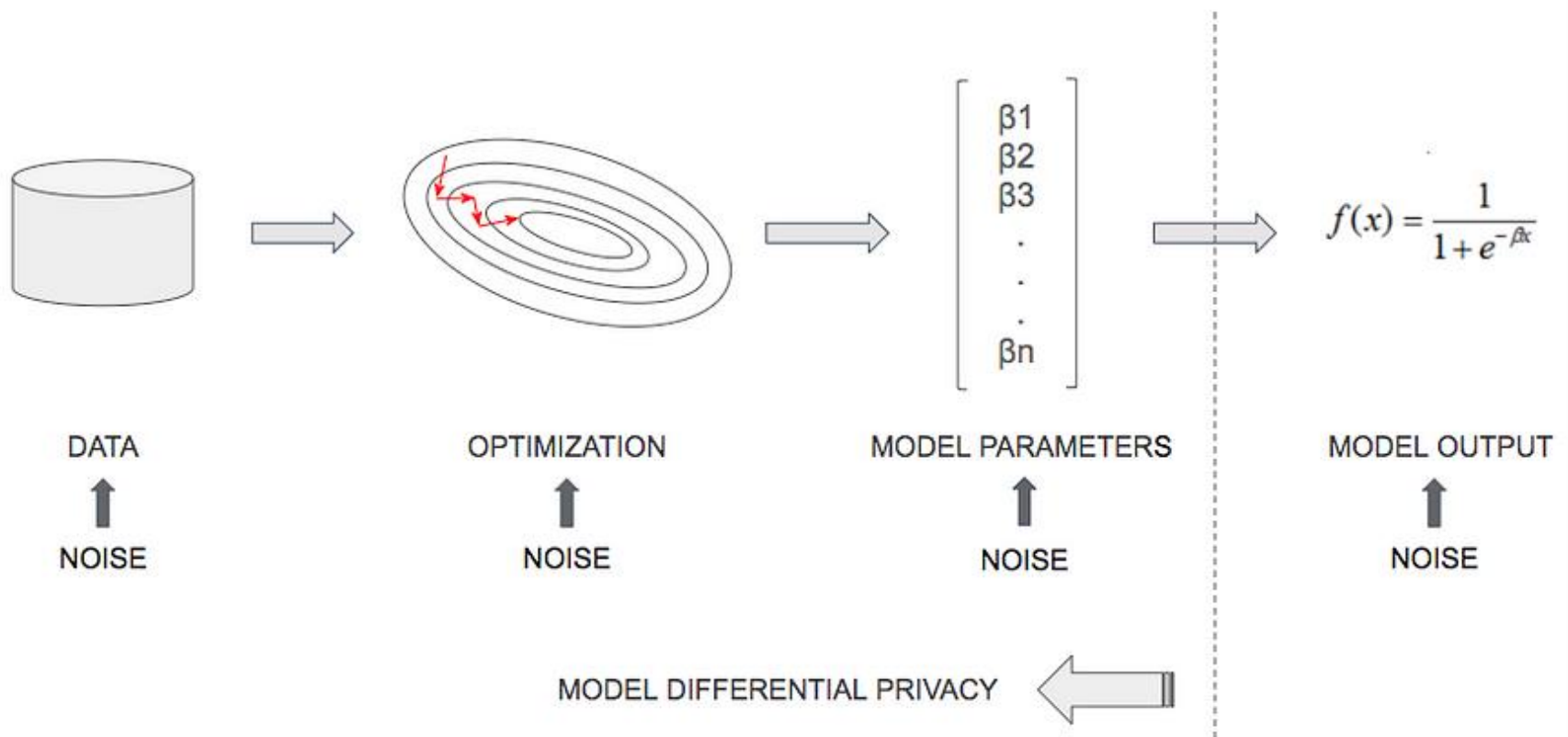
- DP formally defines anonymity of a data operation, such as:
 - Computing a statistic
 - Training an ML model
 - Generating synthetic data
- Introduces ϵ : privacy loss
 - If we observe the same processed data twice the ϵ sums up
- Anonymity is achieved by adding random noise

DP in Machine Learning



Noise Injection Methods:

- In the training data itself
- During optimization
- In the parameters output by the model,
- At query time (each time the model is used)



The model in the last method is not differentially private, but the querying mechanism is: this means that each query to the model leaks some small amount of information about the original data. Therefore we can only use that type of model a limited amount of times before the **privacy budget** becomes too great, thus risking exposing individual records.

DP in Machine Learning

Deep Learning with DP

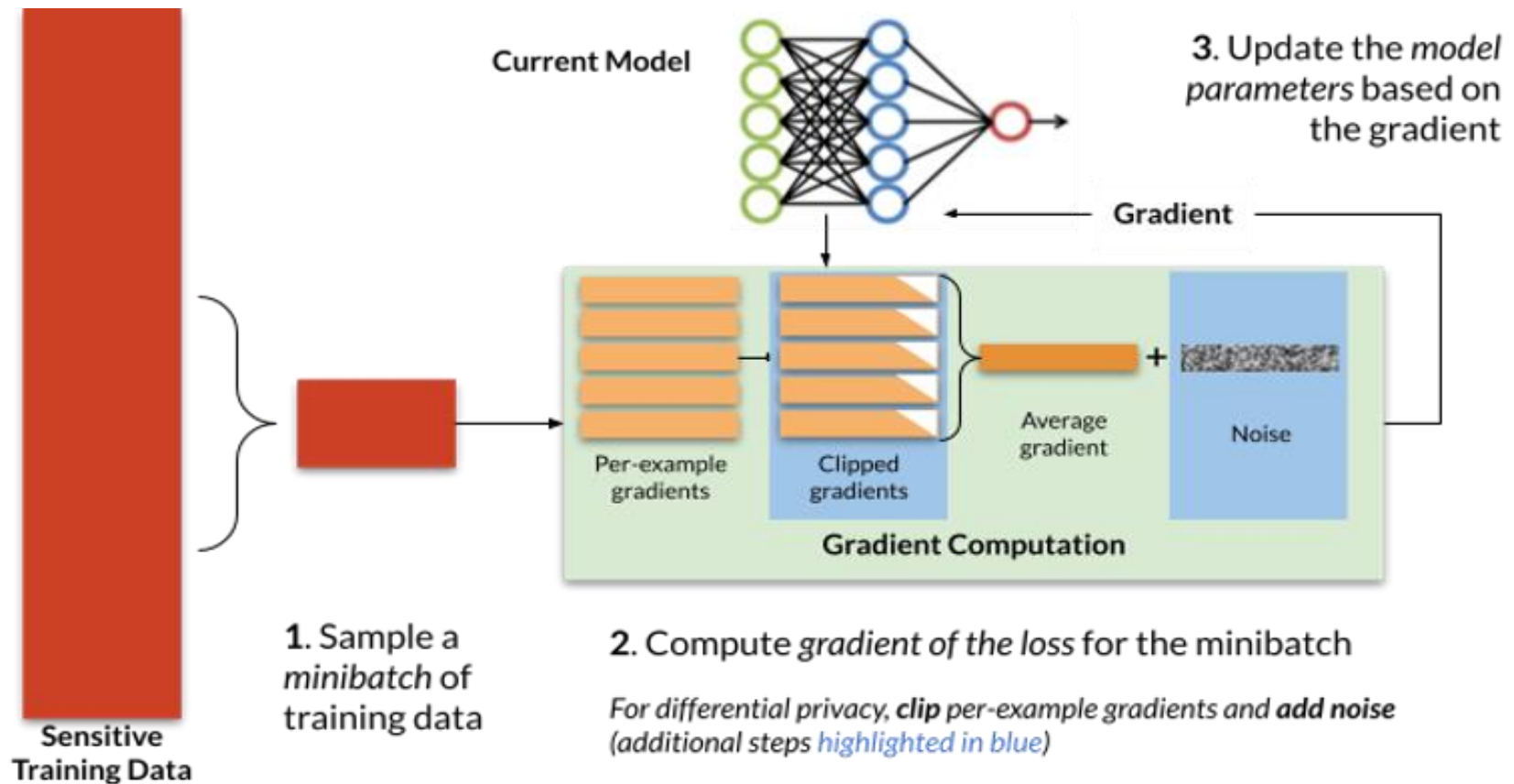
Applying DP to the weights of Deep Learning models

- Publish a model
- Keep an internal model clean of personal information
- Prevent leakage of personal information during inference
- Federated Learning

How it works: DP-SGD

- Adding random noise to gradients while training

DP in Machine Learning



DP in Machine Learning

Frameworks To Implement Differential Privacy



Opacus



PySyft

Applications of Differential Privacy



Limitations of DP

1) Privacy-Utility Trade-off

- **Data Accuracy:** Noise added to ensure privacy can degrade data accuracy and utility.
- **Balance:** Achieving the right balance between privacy and utility can be challenging.

2) Computational Overhead

- **Performance:** Adding noise introduces significant computational overhead, making algorithms slower and more resource-intensive.
- **Scalability:** Computational cost can be a limiting factor for large datasets or real-time applications.

3) Parameter Selection

- **Epsilon (ϵ) Value:** Controls the level of privacy. Small ϵ offers better privacy but worse utility, while large ϵ provides better utility but weaker privacy.
- **Lack of Guidelines:** No universally accepted guidelines for selecting ϵ , requiring domain-specific considerations.

Conclusion

Conclusions

- Unexpected vulnerabilities in statistical releases abound, and are very hard to avoid with ad hoc approaches.
- DP and variants are the only principled ways known to avoid these risks.
- There is a rapidly evolving literature to achieve a better privacy vs. utility tradeoff, with some amazing theoretical possibilities.

Code Implementation

- Scan me!!

References

- Cynthia Dwork and Aaron Roth's book on The Algorithmic Foundations of Differential Privacy: <https://www.cis.upenn.edu/~aaroht/Papers/privacybook.pdf>
- Kritika's Introduction to Differential Privacy: https://iiittheorygroup.github.io/Initiatives/Seminar-Saturdays/2021_spring/slides_019.pdf
- University of Toronto CSC 2515 Lecture 11: Differential Privacy: https://www.cs.toronto.edu/~rgrosse/courses/csc2515_2019/slides/lec11-slides.pdf
- Harvard CS208: Applied Privacy for Data Science. <https://6s060.csail.mit.edu/2021/lec/diffpriv.pdf>
- Davide Testuggine's Differentially Private Model Training with Opacus: <https://www.youtube.com/watch?v=MWPwofiQMdE>
- Sophia Collet's blog on Differential Privacy in the Real World: <https://medium.com/bluecore-engineering/differential-privacy-in-the-real-world-f31a5df1398f>

THANK YOU!

Any Question?