



Privacy at your Fingertips

Kenneth Emeka Odoh

<https://kenluck2001.github.io>

13th July 2023



Table of Contents

- Bio
- Overview
- Mathematical Foundations
- Case Studies
- Conclusions

Referenced blog: https://kenluck2001.github.io/blog_post/privacy_at_your_fingertips



Bio

- **Work Experience:**
 - Currently open for work
 - Previously worked at
 - Microsoft Corporation (web dev, backend dev, faster video streaming, infrastructure engineering)
 - Intel Corporation (CI/CD dev, low-level dev on far-memory devices)
 - Multiple startups
- **Open Source Contributions:** <https://kenluck2001.github.io/projects>
- **Blogging:** <https://kenluck2001.github.io/blogs/1>
- **Publications:** <https://kenluck2001.github.io/publications>

Overview

Manual attempt at privacy

- Removing field that may cause privacy leaks.
 - Not suitable when multicollinearity exist

Scope of Privacy

- What is **privacy**? How is privacy defined?
- What is an acceptable privacy level?
- When it comes to privacy, what are the limits?

Differential Privacy is a mathematical framework

- Prevents an analyst from leaking information when querying a data store.
- Adding noise to the data while preserving the database's global statistical properties.
- Obtaining aggregated information without sacrificing individual privacy.






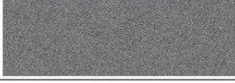
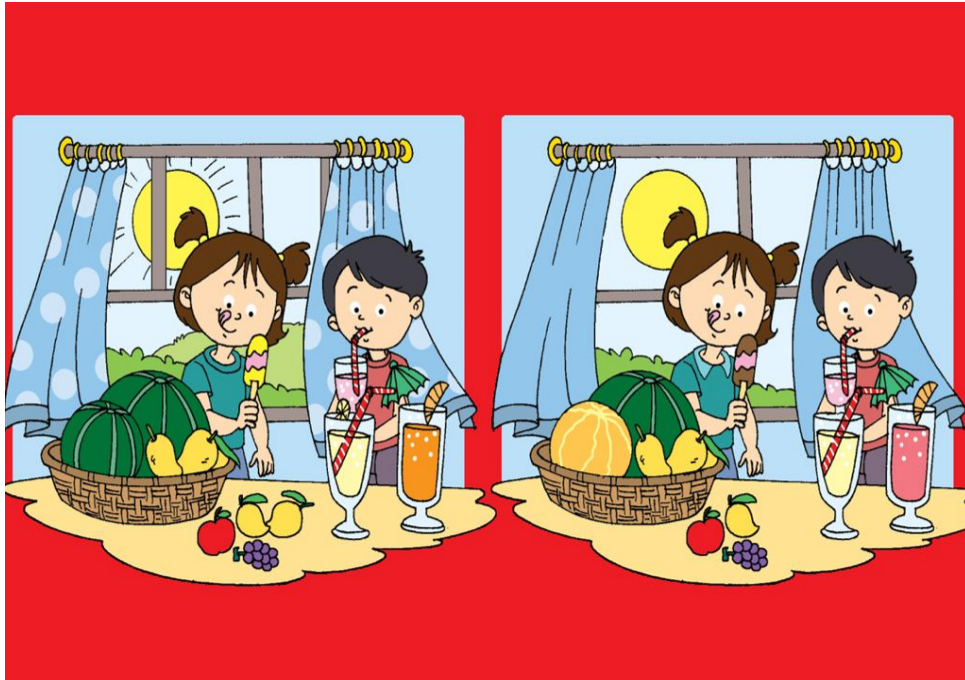
Method	Description	Computing hardware	Transformed result
Differential privacy	Adds noise to the data; minimally affects population-level analysis	Edge computer	
Face blurring	Detects and blurs human faces	Sensor, edge computer	
Dimensionality reduction	Reduces the input size by reducing the number of features	Sensor, edge computer	
Body masking	Replaces people with faceless avatars	Edge computer	
Federated learning	Edge devices learn locally, then sends gradient updates to central server	Edge computer, centralized server	
Homomorphic encryption	Enables predictions to be made from encrypted data	Edge computer, centralized server	

Fig. 4 | Computational methods to protect privacy. There is a trade-off between the level of privacy protection provided by each method and the required computational resources. The methods used to generate the transformed images are described in detail elsewhere: differential privacy, ref.¹⁶⁶; dimensionality reduction, ref.¹⁶⁷; body masking, ref.¹⁶⁸; federated learning, ref.¹⁶⁹; homomorphic encryption, ref.¹⁷⁰. The original image was produced by S. McCoy and has previously been published¹⁷¹. The appearance of US Department of Defence visual information does not imply or constitute endorsement by the US Department of Defence.



Credit: <https://www.champak.in/spot-the-difference/puzzles-for-kids-spot-the-difference>

Differential privacy can be configured:

- **Local:** Data is randomized on the **client** before sending off to the server.
 - Provides stronger privacy guarantees.
- **Central:** Data is collected and randomized on the **server**.
 - Easier to abuse.
 - Trusting that the server is impartial.
 - Provides weaker privacy guarantees.



Mathematical Foundations

- $A : D \rightarrow Y$ (randomizer)
- $(d, \hat{d}) \in D$ (data) and $y \in Y$ (label)
- Pr : probability

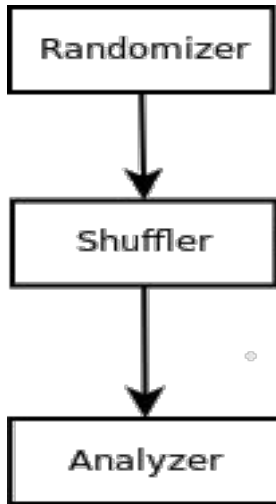
$$-\xi \leq \ln\left(\frac{Pr(A(d)=y)}{Pr(A(\hat{d})=y)}\right) \leq \xi$$

$$(Pr_{post}(\hat{X}) - Pr_{pre}(\hat{X})) \leq \delta$$

ξ is the measure that any two pairs of elements (d, \hat{d}) in the data store are similar.

δ is described in terms of the **attacker's advantage**.

Basic Components of a Differential Privacy Scheme



- **Randomizer (encoder):** takes input from the client device and obtains an intermediary transformed output by adding noise.
- **Shuffler:** However, it takes transformed input from the randomizer and does permutation (breaking order). [optional]
- **Analyzer (decoder):** Ensure that you remove some noise and make meaningful computations needed by the application.

Figure 1: Architecture of Differential Privacy



Case Study: Negative Databases

Save complements of a records by formulating entries as satisfiability of a 3-SAT problem.

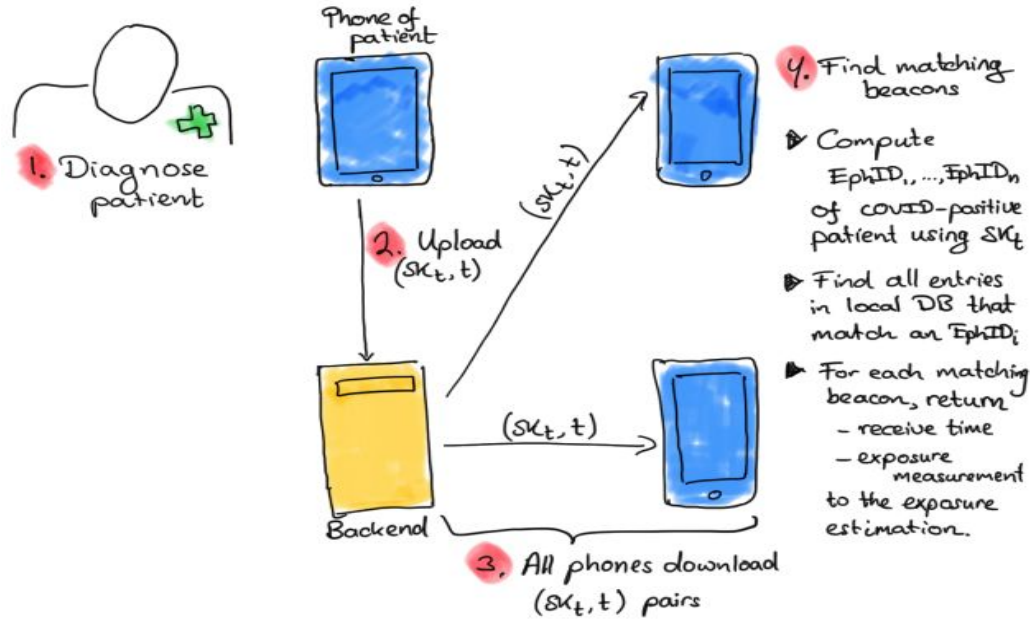
- Source code link: <https://github.com/kenluck2001/diffprivacy/tree/main/NegativeDB>
- Paper: <https://crypto.stanford.edu/portia/papers/HardNDBFinal.pdf>

Case Study: Frequency estimation learning at Scale

Locally private scheme to get frequency of a telemetry data stream on a set of dictionary word events.

- Source code link: <https://github.com/kenluck2001/diffprivacy/tree/main/Privacy-Preserving-Telemetry>
- Paper: <https://docs-assets.developer.apple.com/ml-research/papers/learning-with-privacy-at-scale.pdf>

Case Study: Privacy-aware Contact Tracing Scheme

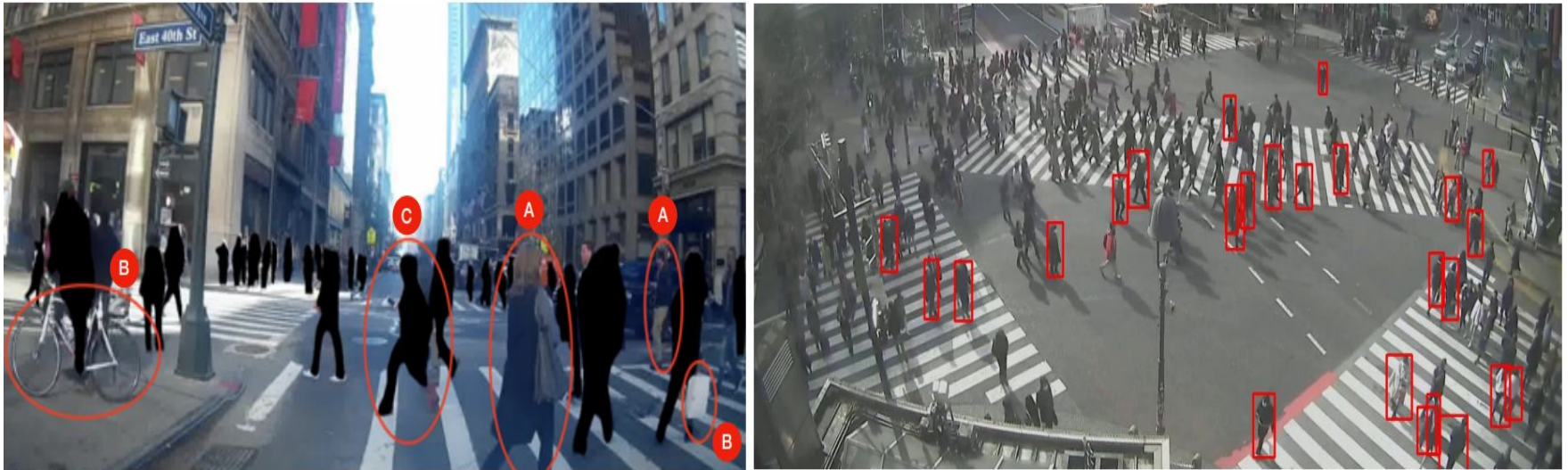


Credit: <https://arxiv.org/pdf/2005.12273.pdf>

Figure 2: structure of contact tracing scheme

Case Study: Privacy-aware Video Surveillance System

Credit: <https://arxiv.org/abs/2106.12083>





Conclusions

- Differential privacy can future-proof applications from evolving governmental regulations.
- Differential privacy provide a way to quantify privacy level.
- Even if you used differential privacy, secondary ancillary information can enable re-identification attacks.



References

- [1] Victor Balcer, Albert Cheu. Separating Local & Shuffled Differential Privacy via Histograms, <http://arxiv.org/abs/1911.06879> , 2019.
- [2] A Graduate Course in Applied Cryptography by Dan Boneh and Victor Shoup. <http://toc.cryptobook.us/>
- [3] <https://docs-assets.developer.apple.com/ml-research/papers/learning-with-privacy-at-scale.pdf>
- [4] Peeter Laud, Alisa Pankova. Interpreting Epsilon of Differential Privacy in Terms of Advantage in Guessing or Approximating Sensitive Attributes, <https://arxiv.org/abs/1911.12777>