Privacy

Márk Jelasity

University of Szeged, Hungary

Stallman on Facebook, etc

Richard Stallman
(Free Software Foundation)

So the companies that wanted to collect data about people could take advantage of this general misguided ideology to get away with whatever they might have wanted to do. Which happened to be collecting data about people. But I think they shouldn't be allowed to collect data about people.

We need a law. Fuck them – there's no reason we should let them exist if the price is knowing everything about us. Let them disappear. They're not important – our human rights are important. No company is so important that its existence justifies setting up a police state. And a police state is what we're heading toward.

New EU regulations

- Users are supposed to
  - Have easier acces to personal data
  - Have easier portability of data
  - Have a right to erasure
- Companies can still collect data, the point is the users are supposed to have complete control over their data
- It is a big question what the effect will be (if any)

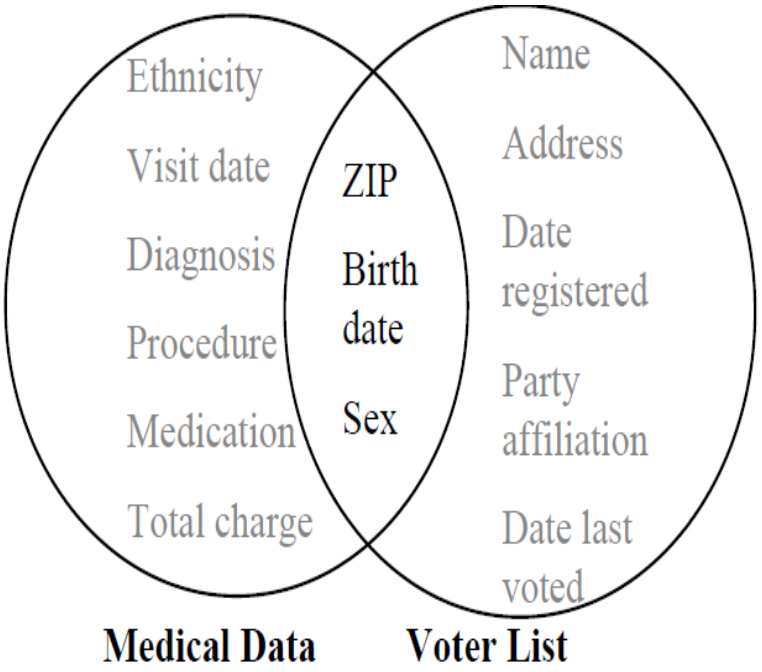
https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=LEGISSUM:310401_2

Aspects of privacy

- We do not want private data to go public
- What does this mean
  - No one can get raw private data in any way
  - Leaking indirect information on private data must be limited
    - **So called “anonymized” data is suspicious**
    - **Distorted (eg noise-added) data is suspicious**
    - **The output of aggregate queries is also suspicious**

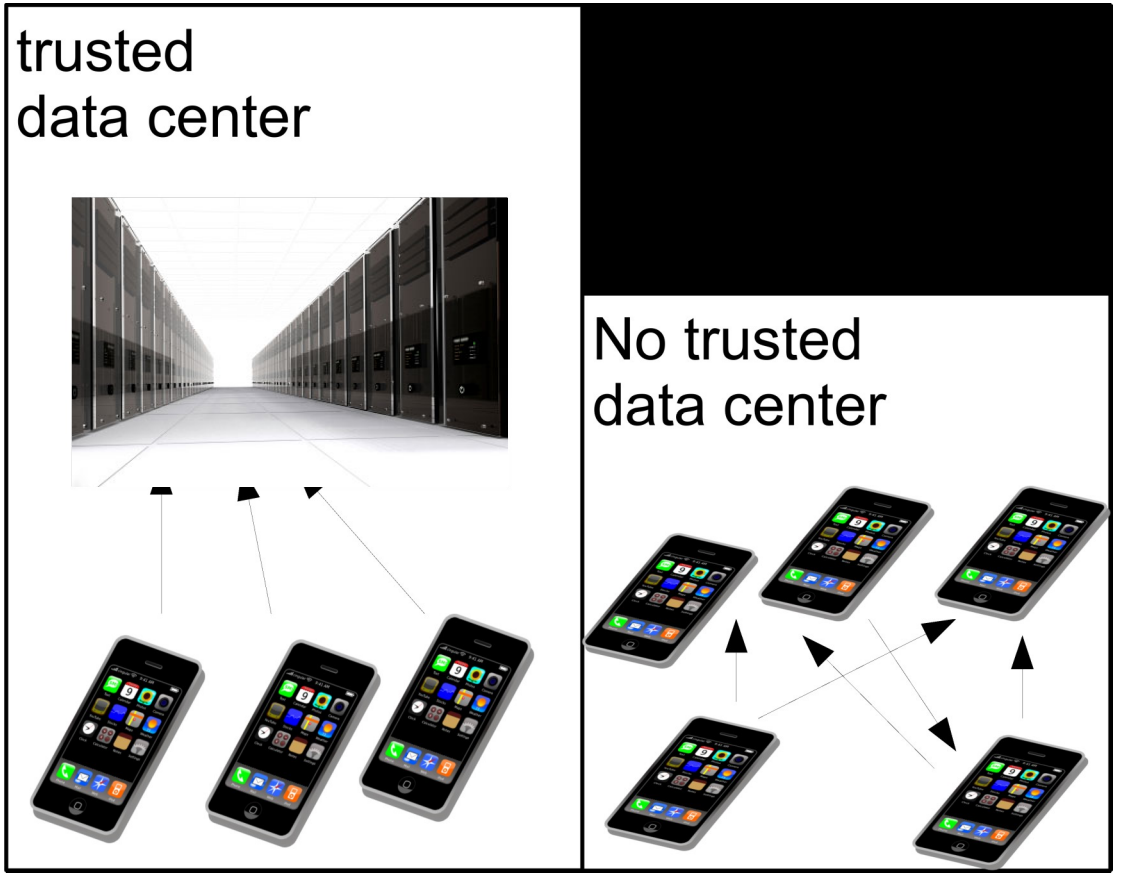
Publishing data

- Anonymized data is not secure due to eg linkage attacks
  - Netflix and IMDB [Narayanan and Smatikov]
  - MGIC and voting register [Latanya Sweeney]
  - etc.



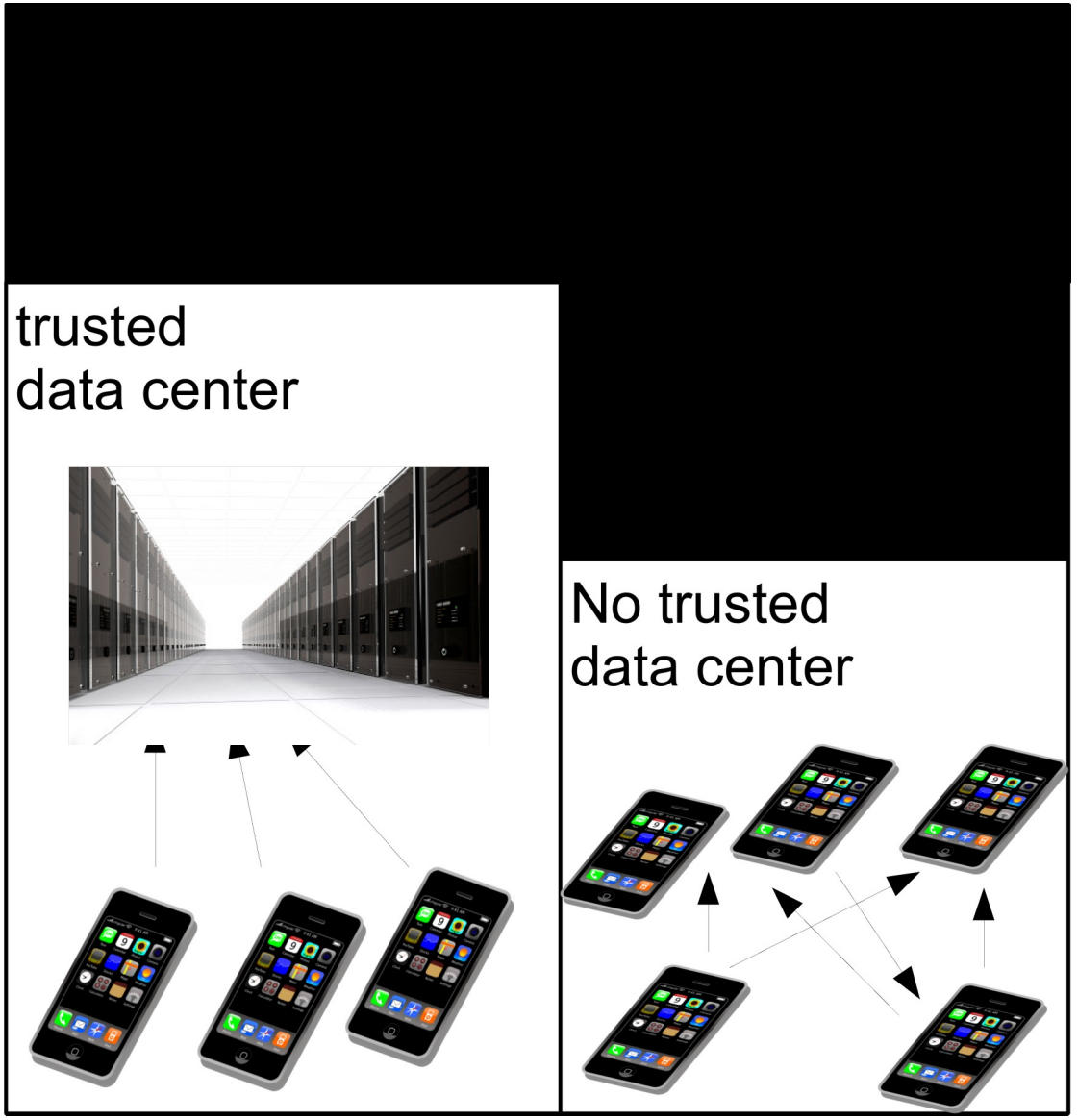
Interactive databases

- Executing queries on hidden data
- If there is no trusted central database, we need secure distributed algorithms
  - secure multiparty computation



Interactive databases

- Publishing the results of the executed queries
- This can also leak private information
  - min, max
  - Many specially designed queries
  - One can audit sets of queries for privacy but this is very hard



What approach to take?

- So is it ok to assume (or make sure) data is not shared at all and only query results are published?
- Not quite
  - Some queries return individual records
    - **Min, max, etc**
  - Sets of queries can be designed that allow inference of values of individual records

Publishing query outputs

- So even aggregate and statistical queries might reveal private information
- Should we add noise to the output?
  - Clearly, we need noise with an expected value of 0, but then many queries can be used to average it out
    - **So we need to add the same noise to the same query**
    - **But it is hard to detect whether it is the same query**
  - Even if we can make sure the same query gets the same noise, we are not safe

Publishing query outputs

- Special case: n records, each record is one bit, sum query, bounded noise
- Can we restore 99% of the records using a number of queries?
  - Even noise less than n/401=O(n) is not enough if we execute all the 2^n queries (all the subsets of the records)
  - With an $O(\sqrt{n})$ bound on noise $O(n \log^2 n)$ queries are enough
- Non-interactive case: if we publish a noisy database then
  - We either have very noisy query results
  - Or (most of) the records will be recovered

What should we protect?

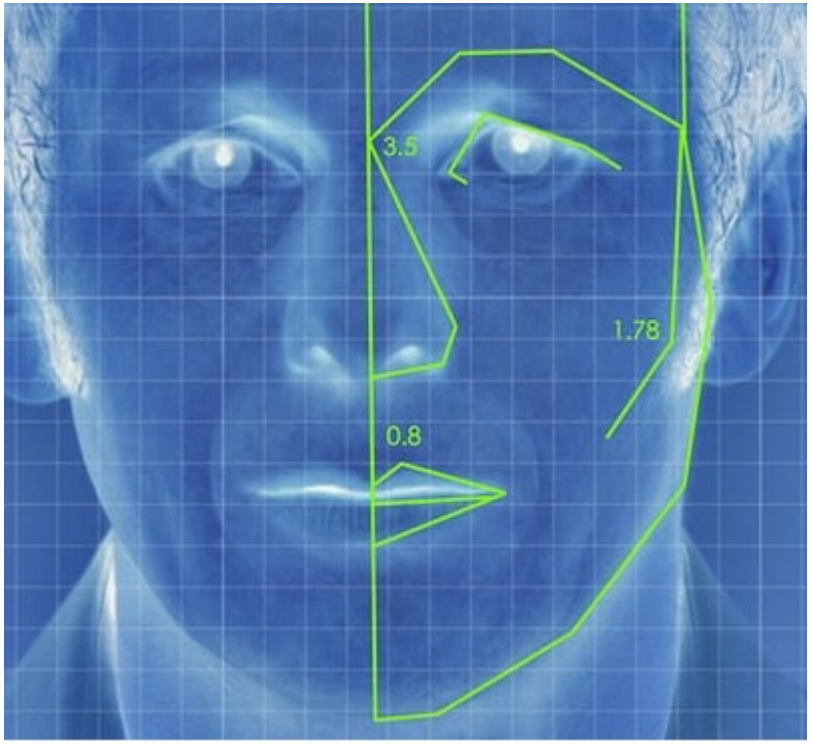
- Dalenius's desideratum (1977)
  - „Anything that can be learned about a respondent in a statistical database should be learnable without access to the database”
- Is this possible?
  - We need to learn *something* otherwise queries make no sense at all!
  - This is the main difference from cryptography: some information must leak

What should we protect?

- So how to formulate the problem then?
  - Same (or very similar) prior and posterior knowledge about an individual?
    - **Doesn't work: if we learn that almost every record has a given property (aliens learn that “humans have two feet”) then any given record will have that with a high probability: the aliens learned a lot about me**
    - **Auxilary information: Turing is two inches taller than the average height: now, if we can query the average height, we learned a lot again about Turing (now using auxilary info)**
- Let's do this then
  - To what extend does my risk grow by being included in the database compared to not being included

A thing to digest

- Differential privacy thus sometimes might not protect privacy at all!
  - Based on the information gained in a database some secrets can be revealed even if one was never in the database in the first place!



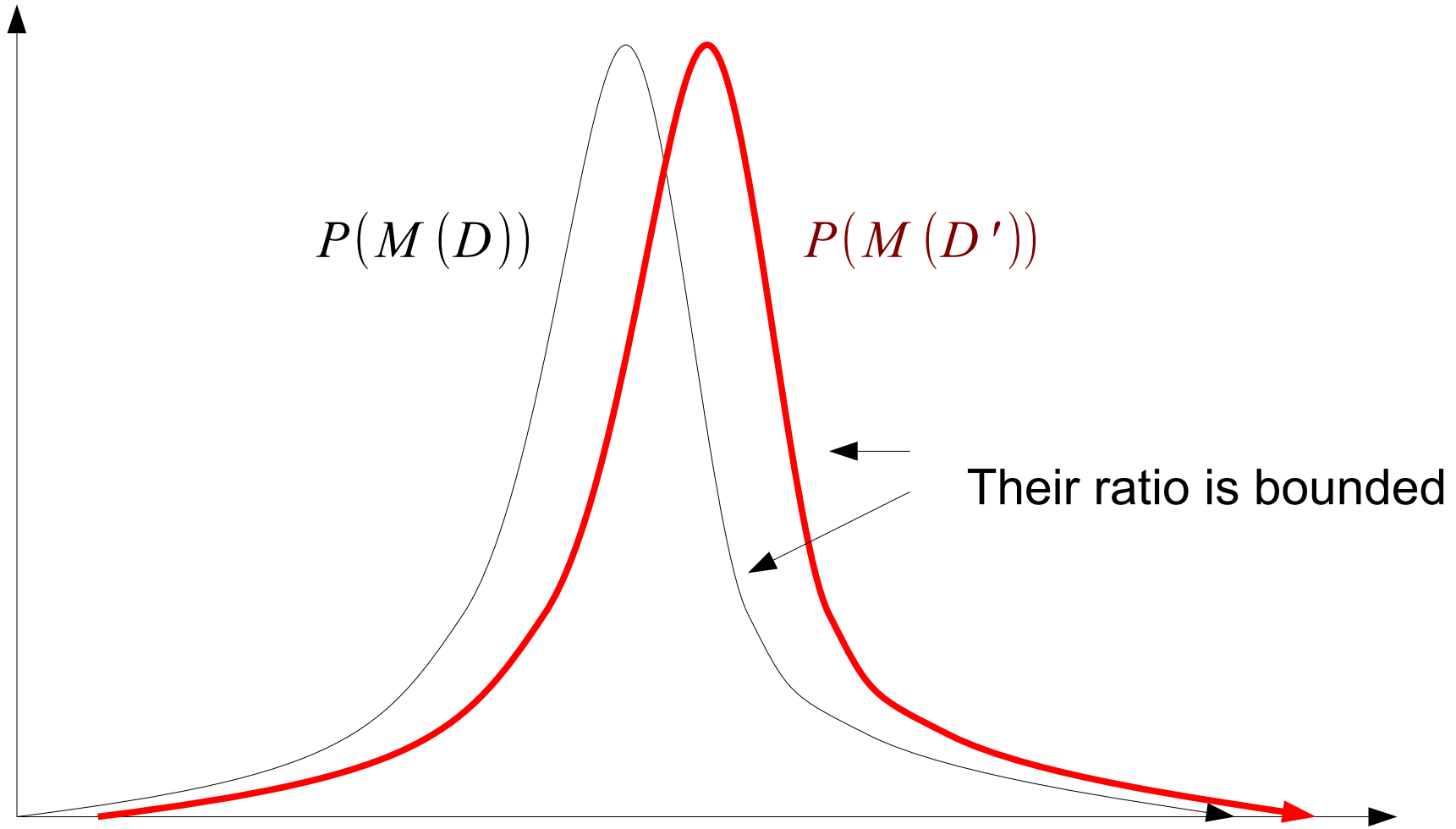
https://www.theguardian.com/technology/2017/sep/07/new-artificial-intelligence-can-tell-whether-youre-gay-or-straight-from-a-photograph

Differential privacy

- Let us take a database D
- Let M(D) be the noisy query output
  - M(D) is a random variable where randomness is due to added noise by function M(), D is constant
- Let D' be a database that differs in only one record from D
- M() is ε-differentially private if

$$P(M(D) \in S) \leqslant \exp(\varepsilon) P(M(D') \in S)$$

Differential privacy



Differential privacy

- Let us take a database D
- Let M(D) be the noisy query output
  - We think of M as a random variable with a distribution that depends on D
- Let D' be a database that differs in only one record from D
- M() is ε-differentially private if

$$P(M|D) \leqslant \exp(\varepsilon) P(M|D')$$

- Which is the same as

$$P(D|M) \leqslant \exp(\varepsilon) P(D'|M)$$

  if the database priors are the same

Compositionality

- If M_1 and M_2 are queries over the same database records, are independent and are ε_1-, and ε_2-differentially private, respectively, then publishing both M_1 and M_2 is $\varepsilon_1+\varepsilon_2$-differentially private
- If M_1 and M_2 are queries over non-overlapping subsets of records (that are defined independently of the actual database) then publishing both of them is $\max(\varepsilon_1,\varepsilon_2)$-differentially private
  - „privacy budget”: a fixed ε parameter that allows a finite number of queries or noise must be increased with time
  - When the budget is over no more queries are allowed at all (the database must be deleted)

A possible implementation of differential privacy

- Global sensitivity:

$$\Delta g = \max_{D, D' \text{ differ in one record}} |g(D) - g(D')|$$

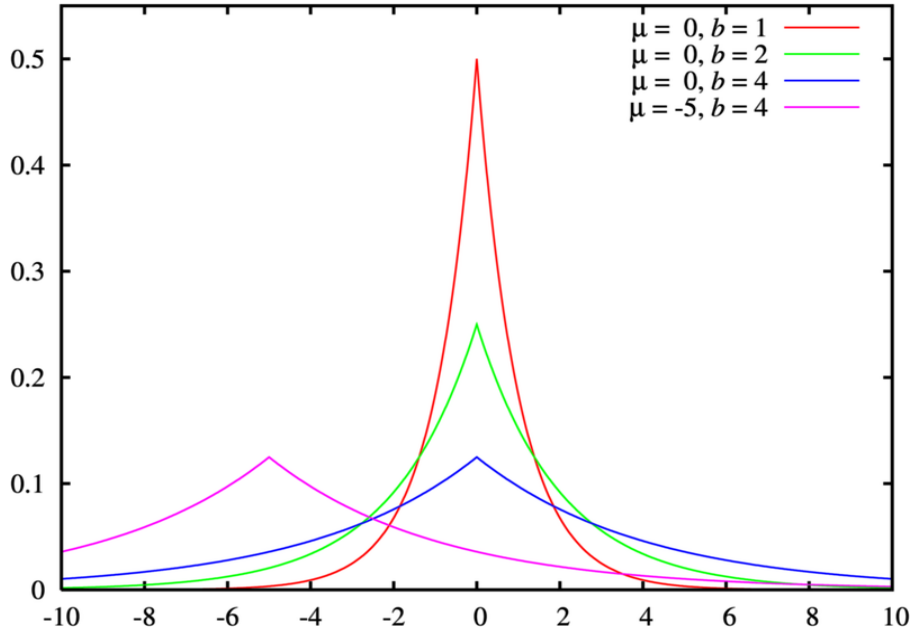
- Laplace distribution:

$$f(x|\mu, \beta) = \frac{1}{2\beta} \exp\left(\frac{-|x-\mu|}{\beta}\right)$$

- Adding noise to deterministic query

$$M(D) = Y + g(D)$$

  - Y ~ Laplace(0,Δg/ε)



Counting, sum, histogram

- Counting
  - Sensitivity is 1
- Sum
  - Sensitivity depends on the range of values: it is the absolute value of the record with maximum absolute value
- Histogram
  - We have to count the number of points in each cell
  - Since changing one record changes only one cell, the overall sensitivity is also 1

k-means

- Let us consider features from [0,1]
- The assignment step is private (requires no output)
- Averaging step
  - Average: sum divided by count
  - Counts in k cells: like a histogram, sensitivity 1
  - Sums of d dimensional vectors in each cell: sensitivity is 1 in all dimensions (ie sensitivity is d in terms of the 1-norm)
- So we "burn" $(d+1)\varepsilon$ in each iteration from our budget
- This is when we publish partial results in each iteration (count and cluster centers)
  - What about publishing just the end result?

Decentralized implementation

- Differential privacy mechanisms add noise to the query output
- If the query is implemented in a decentralized (privacy preserving) manner, noise also has to be added in a decentralized way
- Let us focus on the sum query again
  - One approach is to decompose the noise term and distribute it to the participants

Infinitely divisible distributions

- A probability distribution is infinitely divisible if it can be expressed as the distribution of a sum of any number of i.i.d. random variables of some appropriate distribution
- Clearly, infinitely divisible noise distributions can be added to the sum collectively using a privacy preserving sum query
  - Every node i samples the appropriate distribution locally adding it to its local value: $x_i+\xi_i$
  - The sum of these samples will form the noisy sum query result with the desired (infinitely divisible) noise distribution: $\Sigma_i x_i+\Sigma_i \xi_i$
- The most well known infinitely divisible distribution is the normal distribution

Decomposing the Laplacian noise

- The Laplacian noise is also infinitely divisible

Lemma 1 (Divisibility of Laplace distribution [13]). *Let $\mathcal{L}(\lambda)$ denote a random variable which has a Laplace distribution with PDF $f(x,\lambda) = \frac{1}{2\lambda} e^{\frac{|x|}{\lambda}}$. Then the distribution of $\mathcal{L}(\lambda)$ is infinitely divisible. Furthermore, for every integer $n \geq 1$, $\mathcal{L}(\lambda) = \sum_{i=1}^{n} [\mathcal{G}_1(n,\lambda) - \mathcal{G}_2(n,\lambda)]$, where $\mathcal{G}_1(n,\lambda)$ and $\mathcal{G}_2(n,\lambda)$ are i.i.d. random variables having gamma distribution with PDF $g(x,n,\lambda) = \frac{(1/\lambda)^{1/n}}{\Gamma(1/n)} x^{\frac{1}{n}-1} e^{-x/\lambda}$ where $x \geq 0$.*

- This follows from the facts that
  - The gamma distribution itself is infinitely divisible
  - The exponential distribution is a special case of the gamma distribution (G(1,λ) is exponential)
  - And the Laplace distribution is the difference of two exponential distributions ($L(\lambda)=G_1(1,\lambda)-G_2(1,\lambda)$)

References

- Chris Clifton, Murat Kantarcioglu, Jaideep Vaidya, Xiaodong Lin, and Michael Y. Zhu. Tools for privacy preserving distributed data mining. SIGKDD Explor. Newsl., 4(2):28–34, December 2002. (doi:10.1145/772862.772867)
- Danny Bickson, Danny Dolev, Genia Bezman, and Benny Pinkas. Peer-to-Peer secure multi-party numerical computation. In IEEE International Conference on Peer-to-Peer Computing, pages 257–266. IEEE Computer Society, 2008. (doi:10.1109/P2P.2008.22)
- Cynthia Dwork. A firm foundation for private data analysis. Commun. ACM, 54(1):86–95, January 2011. (doi:10.1145/1866739.1866758)
- Gergely Ács and Claude Castelluccia. I have a dream! (differentially private smart metering). In Tomáš Filler, Tomáš Pevný, Scott Craver, and Andrew Ker, editors, Information Hiding, volume 6958 of Lecture Notes in Computer Science, pages 118–132. Springer Berlin Heidelberg, 2011. (doi:10.1007/978-3-642-24178-9_9)