Towards Formal Verification of Differential Privacy in Isabelle/HOL

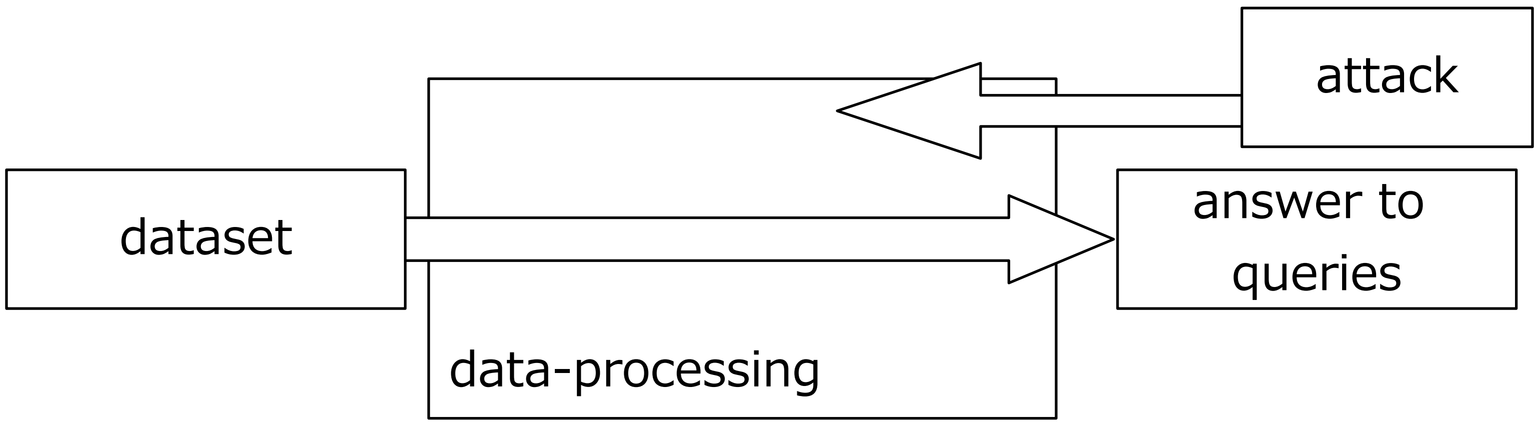
8th Franco-Japanese Cybersecurity Workshop

November 29, 2023

Tetsuya Sato (Tokyo Institute of Technology)

(special thanks: Shin-ya Katsumata・Yasuhiko Minamide)

Background



- Consider a query that processes datasets.
  The dataset contains individuals' private data.
- Even if the query does not show the private data, attacker can steal them from the answers if the attackers have enough background-knowledge.

An Example of Privacy Leakage

- consider a query of average income...

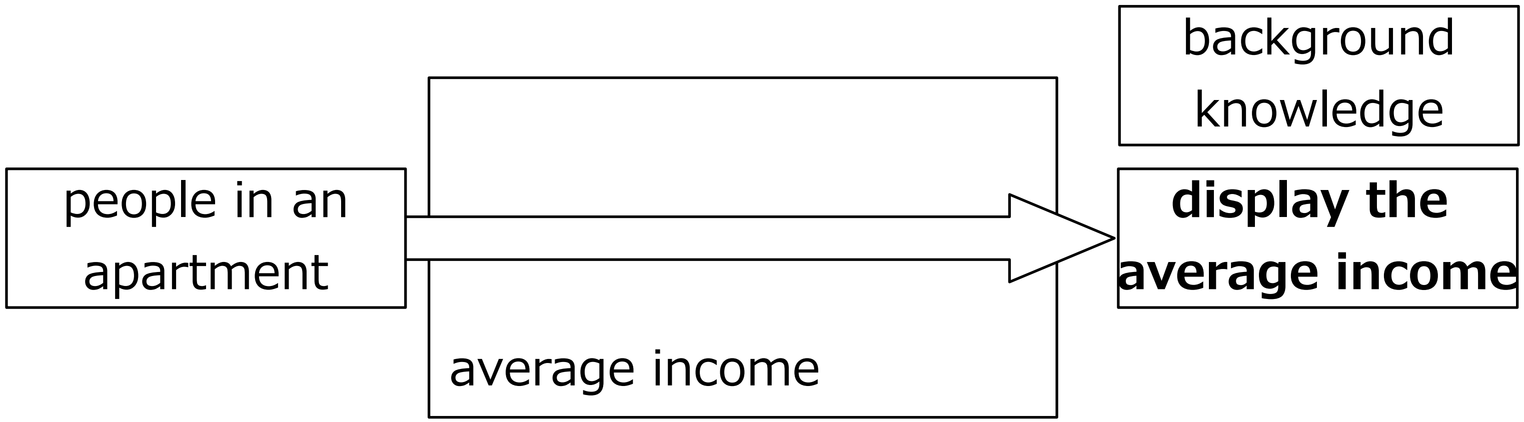
background knowledge

people in an apartment

average income

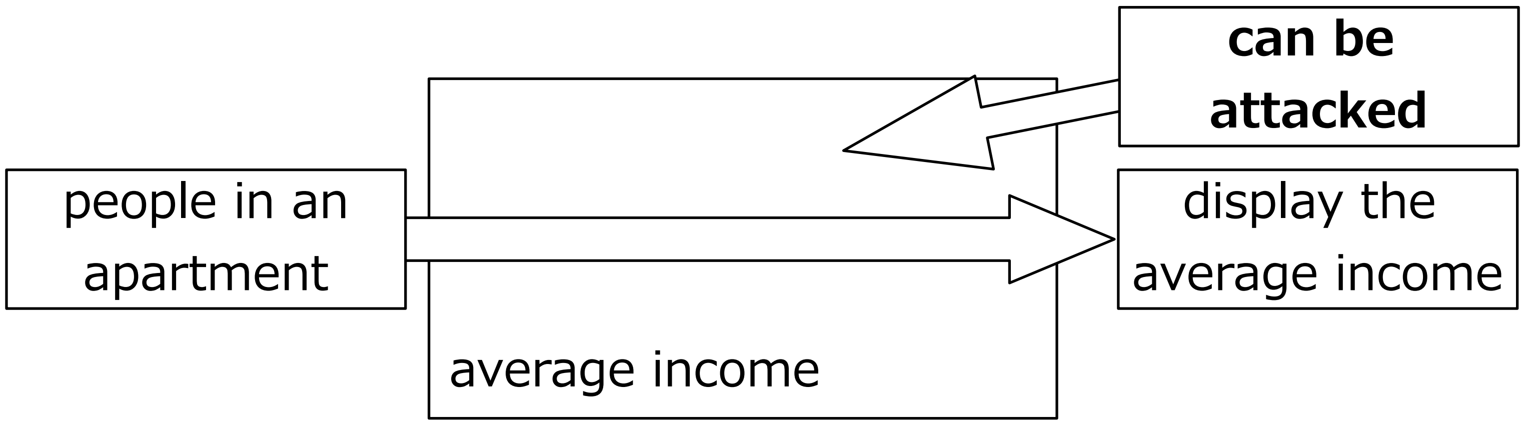
  - 3 people lives here. Their average income is $50,000.

An Example of Privacy Leakage



- 3 people lives here. Their average income is $50,000.
- Now, Bill, the 4th person joins here.
  Then, the average income changes to $150,000.

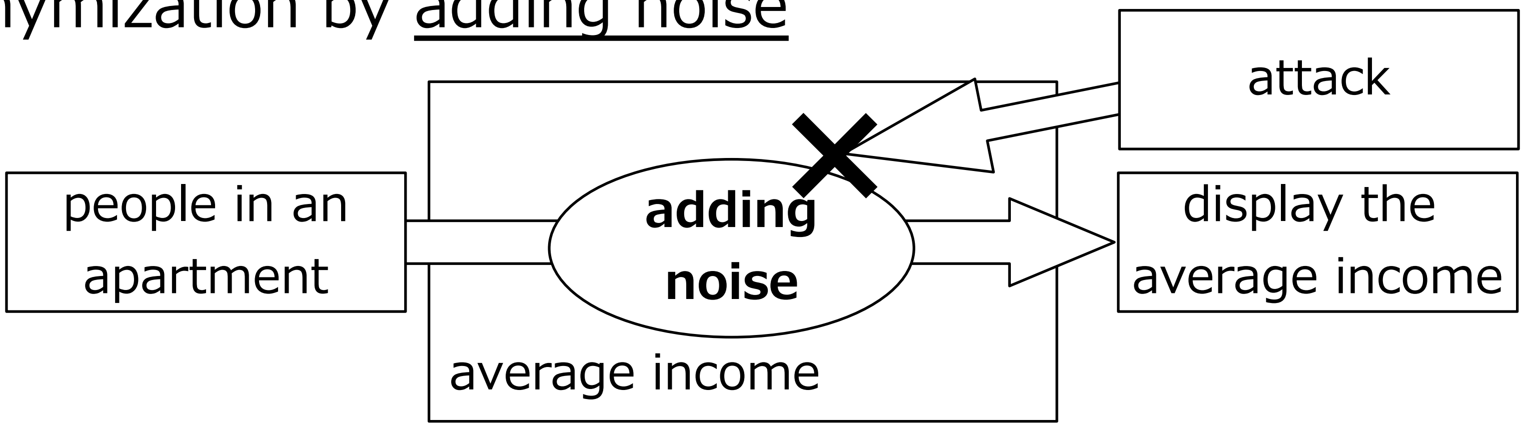
An Example of Privacy Leakage



- 3 people lives here. Their average income is $50,000.
- Now, Bill, the 4th person joins here.
  Then, the average income changes to $150,000.
- While the "average income" query <u>shows only the average, but it **leaks Bill's income**</u>, $450,000.

Differential Privacy

1. Anonymization by <u>adding noise</u>



- By adding noise, we make query hard to leak private data.
- The method is robust against background knowledge attack.

2. **Standards of privacy in such <u>randomized</u> queries**

Definition of Differential Privacy

[Dwork+, TCC 2006]

- A randomized mechanism **M: X → Prob(Y)** is
  (ε,δ)-differentially private(DP)
  if for "adjacent" datasets D1～D2,
  the following inequality holds for any S ⊆ Y :

$$\boxed{\Pr[M(D_1) \in S] \leq \exp(\varepsilon)\Pr[M(D_2) \in S] + \delta}$$

  - intuition:
    The ratio of probability is bounded by **ε**
    except in probability **δ**.

    (If (ε,δ)=(0,0) then, the distributions are equal.)

Reformulating DP via Divergences

[Barthe & Olmedo, ICALP 2013]

- M: X → Prob(Y) satisfied (ε,δ)-DP
- ⇔ for adjacent datasets D1∼D2,

$$\boxed{\Pr[M(D_1) \in S] \leq \exp(\varepsilon)\Pr[M(D_2) \in S] + \delta}$$

⇔ for adjacent datasets D1∼D2,

$$\boxed{\underbrace{\sup_{S \in \Sigma_Y} \left(\Pr[M(D_1) \in S] - \exp(\varepsilon)\Pr[M(D_2) \in S]\right)}_{\Delta^{\varepsilon}(M(D_1)||M(D_2))} \leq \delta}$$

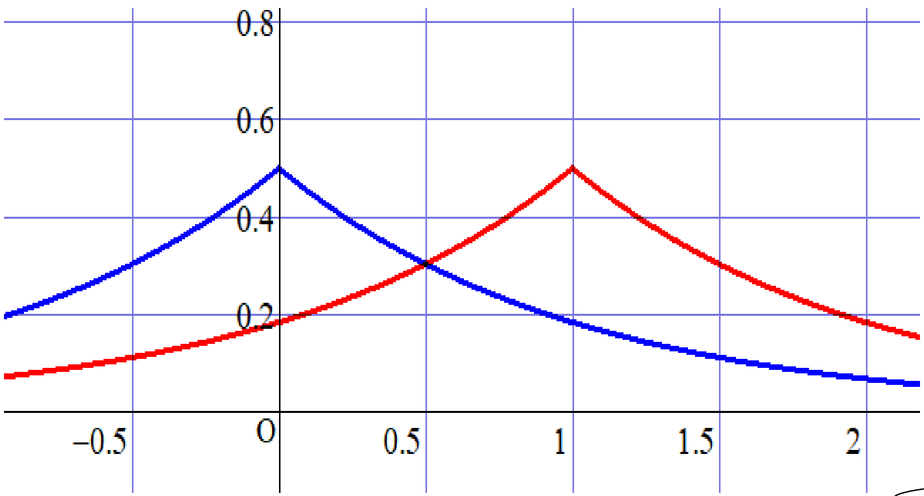
Δ^{ε} … the divergence for (ε,δ)-DP

Example: Laplace Mechanism

- The mechanism adding the noise sampled from Laplacian distribution. It is (ε,0)-DP if the adjacency is |x-y| ≦ 1.

$$\mathtt{Lap}_\varepsilon : \mathbb{R} \to \mathrm{Prob}(\mathbb{R})$$

$\mathtt{Lap}_\varepsilon(x)$ is the Laplacian distribution(avg. x, var. $2\varepsilon^{2}$)

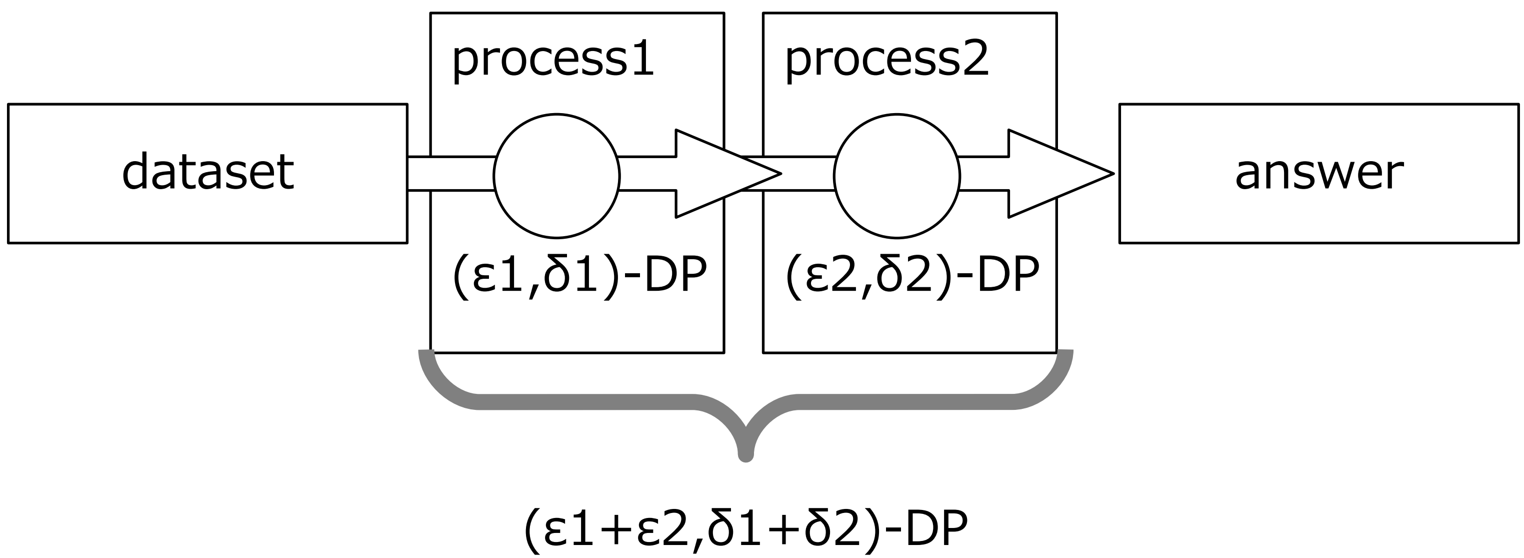


$$\begin{aligned}\Pr[\mathtt{Lap}_\varepsilon(x) = z] &= \frac{1}{2\varepsilon}\exp(-\frac{|x-z|}{\varepsilon})\\ &\le \frac{1}{2\varepsilon}\exp(-\frac{|y-z|-|x-y|}{\varepsilon})\\ &\le \exp(\varepsilon)\frac{1}{2\varepsilon}\exp(-\frac{|y-z|}{\varepsilon})]]\\ &= \exp(\varepsilon)\Pr[\mathtt{Lap}_\varepsilon(y) = z]\end{aligned}$$

The ratio is bounded by exp(ε) everywhere.

(Sequential) Composability of DP

- Differential privacy of a sequential composition of processes can be estimated by ones of its components.



  - DP of a fixed number of loop of private mechanism can be estimated by the DP of the loop body.

Naive Report Noisy-Max (RNM)

- Consider the following simple mechanism:

$$
\begin{array}{l}
\mathtt{RNM}_\varepsilon : \mathtt{list}(\mathbb{R}) \to \mathrm{Prob}(\mathbb{R}) \\
\qquad \text{input} : D = [x_1, \ldots, x_n] \in \mathtt{list}(\mathbb{R}) \\
\qquad\quad 1.\ \text{sample } y_k \leftarrow \mathtt{Lap}_\varepsilon(x_k) \quad (1 \le k \le n) \\
\qquad\quad 2.\ \text{return } \max\{y_k | 1 \le k \le n\}
\end{array}
$$

- Using the composability, we have (ε,0)-DP
  - when the adjacency is defined by

$$D_1 \sim D_2 \iff \underbrace{|D_1| = |D_2|}_{\text{same length}} \wedge \underbrace{||D_1 - D_2||_1 \le 1}_{\text{L1-norm}}$$

Today's topic: we formalize this fact.

Proof Sketch

$$\begin{array}{l}\mathtt{RNM}_\varepsilon : \mathtt{list}(\mathbb{R}) \to \mathrm{Prob}(\mathbb{R}) \\ \qquad \text{input} : D = [x_1, \ldots, x_n] \in \mathtt{list}(\mathbb{R}) \\ \qquad\quad 1.\ \text{sample } y_k \leftarrow \mathtt{Lap}_\varepsilon(x_k) \quad (1 \le k \le n) \\ \qquad\quad 2.\ \text{return } \max\{y_k \mid 1 \le k \le n\}\end{array}$$

- Show a bit stronger statement, by induction on length n:

$$|D_1| = |D_2| = n \wedge ||D_1 - D_2||_1 \le r$$
$$\implies \Delta^{r\varepsilon}(\mathtt{RNM}_\varepsilon(D_1) || \mathtt{RNM}_\varepsilon(D_2)) \le 0$$

  - (case: n = 1) Using the DP of Laplacian mechanism:

$$|x_1 - x_2| \le r \implies \Delta^{r\varepsilon}(\mathtt{Lap}_\varepsilon(x_1) || \mathtt{Lap}_\varepsilon(x_2)) \le 0$$

  - (case: n = k + 1) Use I.H. and the below equation:

$$\mathtt{RNM}_\varepsilon(x :: xs)$$
$$= (\mathtt{Lap}_\varepsilon(x) \otimes \mathtt{RNM}_\varepsilon(xs)) >\!\!>\!\!= (\lambda(x, y).\mathtt{return}\max(x, y))$$

Proof Assistant

- **Proof Assistant:**
  - a tool that assists with writing formal proofs.
    - We can program definitions, theorems and proofs, and certificate their validity.

(Isabelle/HOL example)

```
fun func1::"nat list ⇒ nat" where
  "func1 [] = 0" | "func1(x # xs) = x + func1 (xs)"

lemma
  fixes a xs
  shows "func1 (xs @ [a]) = func1 (a # xs)"
    by (induction xs, auto)
```

Probability Theory in Isabelle/HOL

- Isabelle/HOL's standard library contains:
  - Measure type `'a measure` (X, Σ_X, μ)
    - underlying set `"space" :: "'a measure ⇒ 'a set"` X
    - σ-algebra `"sets" :: "'a measure ⇒ 'a set set"` Σ_X
    - evaluation `"Sigma_Algebra.measure" :: "'a measure ⇒ 'a set ⇒ real"` $\mu(A) \quad A \in \Sigma_X$
    - measurable functions `"(→M)" :: "'a measure ⇒ 'b measure ⇒ ('a ⇒ 'b) set"`
  - Monad for probability `"prob_algebra M" :: "'a measure measure"` $\mathrm{Prob}(X, \Sigma_X)$
    - bind `"(≫=)" :: "'a measure ⇒ ('a ⇒ 'b measure) ⇒ 'b measure"`
    - return `"return" :: "'a measure ⇒ 'a ⇒ 'a measure"`
  - Radon-Nikodym derivative `"RN_deriv M N" :: "'a ⇒ ennreal"` $\dfrac{d\mu_M}{d\mu_N}$
  - Lebesgue measure `"lborel" :: "real measure"`

Formalizing the Divergence for DP in Isabelle/HOL

ereal = [-∞,∞]

(definition)

```
definition  DP_divergence:: "'a measure ⇒ 'a measure ⇒ real ⇒ ereal " where
  "DP_divergence M N ε = (⨆ A ∈ (sets M). ereal( measure M A - (exp ε) * measure N A))"
```

(non-negativity)

```
lemma DP_divergence_nonnegativity:
  assumes M: "M ∈ space (prob_algebra L)" and N: "N ∈ space (prob_algebra L)"
  shows "0 ≤ DP_divergence M N ε "
```

(basic properties (for detail, [Olmedo, Phd thesis 2014]))

```
lemma DP_divergence_monotonicity:
  assumes M: "M ∈ space (prob_algebra L)" and N: "N ∈ space (prob_algebra L)"
    and "ε1 ≤ ε2 "
  shows "DP_divergence M N ε2 ≤ DP_divergence M N ε1 "

lemma DP_reflexivity:
  shows " DP_divergence M M 0 = 0"
```

"locale" structure providing the assumption M, N ∈Prob(N)

```
theorem (in comparable_probability_measures) DP_composability:
  assumes f: "f ∈ measurable L (prob_algebra K)"
    and g: "g ∈ measurable L (prob_algebra K)"
    and div1: "DP_divergence M N ε1 ≤ (δ1::real)"
    and div2: "∀x ∈ (space L). DP_divergence (f x) (g x)  ε2 ≤ (δ2::real)"
    and "0≤ ε1" "0≤ε2"
  shows "DP_divergence (bind M f) (bind N g) (ε1 + ε2) ≤ δ1 + δ2"
```

Proof Sketch of Composability of DP

expanding the bind with densities

linearity of integrals

$$
\begin{aligned}
&\Pr[\mu \gg\!= f \in S] - \exp(\varepsilon_1 + \varepsilon_2)\Pr[\nu \gg\!= g \in S] \\
&= \int f(-)(S) \cdot \frac{d\mu}{d\pi}\ d\pi - \exp(\varepsilon_1 + \varepsilon_2) \int g(-)(S) \cdot \frac{d\nu}{d\pi}\ d\pi \\
&= \int f(-)(S) \cdot \frac{d\mu}{d\pi} - \exp(\varepsilon_1 + \varepsilon_2) g(-)(S) \cdot \frac{d\nu}{d\pi}\ d\pi \\
&\leq \int \left(\max\left(0, f(-)(S) - \delta_2\right) + \delta_2\right) \cdot \frac{d\mu}{d\pi} - \exp(\varepsilon_1) \min\left(1, \exp(\varepsilon_2) g(-)(S)\right) \cdot \frac{d\nu}{d\pi}\ d\pi \\
&= \int \max\left(0, f(-)(S) - \delta_2\right) \cdot \frac{d\mu}{d\pi} - \exp(\varepsilon_1) \min\left(1, \exp(\varepsilon_2) g(-)(S)\right) \cdot \frac{d\nu}{d\pi}\ d\pi + \int \delta_2 \frac{d\mu}{d\pi}\ d\pi \\
&\leq \int_B \left(\frac{d\mu}{d\pi} - \exp(\varepsilon_1)\frac{d\nu}{d\pi}\right) \cdot \min(1, \exp(\varepsilon_2) \cdot g(-)(S))\ d\pi + \int \delta_2 \frac{d\mu}{d\pi}\ d\pi \\
&\leq \delta_1 + \delta_2 \\
&= \Delta^{\varepsilon_1}(\mu || \nu) + \sup_x \Delta^{\varepsilon_2}(f(x) || g(x))
\end{aligned}
$$

taking the positive part

assumptions

Formal Proof of Composability of DP in Isabelle/HOL

- Most of the formal proof can be done according to the sketch.

```
have "(measure (M ≫= f) A) - exp (ε1 + ε2) * (measure (N ≫= g) A) [3 lines]
also have "... = (∫ x. (dM x) * (measure (f x) A) ∂(sum_measure M N)) - (∫ x. (exp (ε1 + ε2)) * (dN x) * (measure (g x) A) ∂(sum_measure M N)) " [1 lines]
also have "... = (∫ x. (dM x) * (measure (f x) A) - (exp (ε1 + ε2)) * (dN x) * (measure (g x) A) ∂(sum_measure M N)) " [1 lines]
also have "... = (∫ x. (dM x) * (measure (f x) A) - (exp ε1) * (exp ε2) * (dN x) * (measure (g x) A) ∂(sum_measure M N)) " [1 lines]
also have "... ≤ (∫ x. (dM x) * (max 0 (measure (f x) A - δ2) + δ2) - (exp ε1) * (dN x) * min 1 ((exp ε2) * (measure (g x) A)) ∂(sum_measure M N)) "  [15 lines]
also have "... = (∫ x. (dM x) * (max 0 (measure (f x) A - δ2)) - (exp ε1) * (dN x) * min 1 ((exp ε2)* (measure (g x) A)) + (dM x) *δ2 ∂(sum_measure M N)) " [1 li
also have "... ≤ (∫ x. (dM x) * (min 1 ((exp ε2)* (measure (g x) A))) - (exp ε1) * (dN x) * min 1 ((exp ε2)* (measure (g x) A)) + dM x *δ2 ∂(sum_measure M
N)) "  [12 lines]
also have "... =(∫ x. ((dM x) - (exp ε1) * (dN x)) * min 1 ((exp ε2) * (measure (g x) A)) + dM x *δ2 ∂(sum_measure M N)) " [1 lines]
also have "... =(∫ x. ((dM x) - (exp ε1) * (dN x)) * min 1 ((exp ε2) * (measure (g x) A))∂(sum_measure M N)) +(∫ x. dM x *δ2 ∂(sum_measure M N)) " [1 lines]
finally have *: "(measure (M ≫= f) A) - exp (ε1 + ε2) * (measure (N ≫= g) A)≤(∫ x. ((dM x) - (exp ε1) * (dN x)) * min 1 ((exp ε2)* (measure (g x) A))∂
(sum_measure M N)) +(∫ x. dM x *δ2 ∂(sum_measure M N))".

have "(∫ x. dM x *δ2 ∂(sum_measure M N))=(∫ x. δ2 ∂(density (sum_measure M N) dM))" [1 lines]
also have "... =(∫ x. δ2 ∂(density (sum_measure M N)(ennreal o dM)))" [1 lines]
also have "... =(∫ x. δ2 ∂M)" [1 lines]
also have " ... = δ2 * measure M (space M)" [1 lines]
also have " ... ≤ δ2" [1 lines]
finally have **: "(∫ x. dM x *δ2 ∂(sum_measure M N)) ≤ δ2".

let ?B = "{x∈ space (sum_measure M N). 0 ≤ ((dM x) - (exp ε1) * (dN x)) }"

have mble10: " ?B ∈ sets (sum_measure M N)" [1 lines]

have "(∫ x. ((dM x) - (exp ε1) * (dN x)) * min 1 ((exp ε2)* (measure (g x) A))∂(sum_measure M N)) ≤ (∫ x∈ ?B. ( ((dM x) - (exp ε1) * (dN x)) * min 1 ((exp
ε2)* (measure (g x) A)) )∂(sum_measure M N))"
proof(rule integral_drop_negative_part2) [17 lines]
qed
also have "... ≤ (∫ x∈ ?B. ((dM x) - (exp ε1) * (dN x)) ∂(sum_measure M N))" [11 lines]
also have "... = (∫ x∈ ?B. (dM x) ∂(sum_measure M N)) - (∫ x∈ ?B. ((exp ε1) * (dN x)) ∂(sum_measure M N))" [8 lines]
also have "... = (∫ x∈ ?B. (dM x) ∂(sum_measure M N)) - (exp ε1) * (∫ x∈ ?B. (dN x) ∂(sum_measure M N))" [1 lines]
also have "... = measure M ?B - (exp ε1) * (measure N ?B)" [42 lines]
also have "... ≤ δ1" [1 lines]
finally have ***:"(∫ x. ((dM x) - (exp ε1) * (dN x)) * min 1 ((exp ε2)* (measure (g x) A))∂(sum_measure M N)) ≤ δ1".

show "measure (M ≫= f) A - exp (ε1 + ε2) * measure (N ≫= g) A ≤ δ1 + δ2"
  using * ** *** by auto
qed
```

Formalizing the Laplace Mechanism in Isabelle/HOL

- (definition)

```
definition laplace_density :: "real ⇒ real ⇒ real ⇒ real" where
  "laplace_density l m x = (if l > 0 then (exp(-¦ x - m¦ / l) / (2* l)) else 0)"
```

```
definition Lap_mechanism ::"real ⇒ real ⇒ real measure"
  where "Lap_mechanism ε x = (if ε ≤ 0 then return lborel x
    else (density lborel (laplace_density (1/ε) x)))"
```

- (measurability)

```
lemma measurable_Lap_mechanism[measurable]:
  shows "Lap_mechanism ε ∈ lborel →M prob_algebra lborel"
```

- (differential privacy(formalized via divergence DP))

```
proposition DP_Lap_mechanism':
  fixes x y ε ::real
  assumes "ε > 0" and "¦ x - y ¦ ≤ r"
  shows "DP_divergence (Lap_mechanism ε x) (Lap_mechanism ε y)  (r * ε) ≤ (0::real)"
```

Formalizing the Naive RNM in Isabelle/HOL

- (definition)

```
fun RNM :: " real ⇒ real list ⇒ real measure "
  where
    "RNM ε [] = (return lborel 0)" | (* empty case, it is dummy *)
    "RNM ε [x] = (Lap_mechanism ε x)" |
    "RNM ε (x # xs) = do { x1 ← (Lap_mechanism ε x);
      x2 ← (RNM ε xs); (return lborel (max x1 x2)) }"
```

- (measurability)

```
lemma measurable_RNM:
  shows "(RNM ε) ∈ (listM lborel) →M  (prob_algebra lborel)"
```

- (differential privacy)

```
theorem DP_RNM:
  fixes xs ys::"real list" and ε::real and n::nat and r::real
  assumes pose [arith]: "ε > (0::real)"
    and adj: "length xs = n ∧ length ys = n
      ∧ ( ∑ i∈{1..n}. ¦ nth xs (i-1) - nth ys (i-1) ¦) ≤ r"
    and posr [arith]:"r ≥ 0"
  shows " DP_divergence (RNM ε xs) (RNM ε ys) (r * ε) ≤ 0"
```

Concluding Remark

- Formal verification of DP in the discrete setting is already implemented in Coq[Barthe+, TOPLAS2013].
- We now aim to develop an Isabelle/HOL library for formal verification of DP in the continuous setting.
  - Today, we have formalized DP of naive report noisy-max in the continuous setting. It is the first formalization example.
  - Next,
    - We are formalizing DP of (true) report noisy-max.
    - We need to optimize what we have implemented.
    - We want to formalize relaxation of DP, such as RDP[Mironov, CSF2017], zCDP[Bun+, TCC2016].

Thank you!