Vol.2022-CSEC-98 No.12
Vol.2022-SPT-48 No.12
2022/7/19

IoTデバイス上の差分プライバシー強化手法の提案と評価

田口 魁人[1,a)]　櫻井 幸一[2,b)]　飯田 全広[3,c)]

概要：IoT デバイス等で得られたデータの活用は我々の生活に利便性をもたらす一方で, 使用される個人データのプライバシー保護が課題である. 局所差分プライバシーは解決策の一つである. 計算資源の限られたハードウェアにおいて, 局所差分プライバシーの単純な実装を行うことは固定小数点や低解像度が原因となり, 無限大のプライバシー損失が引き起こされる. この問題に対し先行研究により出力分布の幅を制限することで, 無限大のプライバシー損失を抑えることが可能であることが示された. 本研究では, 出力分布の幅の制限を行っても低解像度が原因で無限大のプライバシー損失が起こる場合について説明し, 計算資源の限られたハードウェア上で実装を想定した整数値出力の局所差分プライバシー回路において低解像度によるプライバシー損失を回避し, プライバシー保護を強化する手法を提案する. また, ソフトウェアシミュレーションを行い提案手法を適用した際の有用性を示す.

Proposal and Evaluation of a Method for Enhancing Local Differential Privacy in IoT Devices

1. 序文

IoT デバイス等で得られたデータの活用は我々の生活に利便性をもたらす一方で, 使用される個人データのプライバシー保護が課題である. これに対してセンサ側で得られたデータに確率的な処理を加える局所差分プライバシーは解決策の一つである.

またIoT デバイス等からのデータは次々に入力がなされるストリーム形式であり, ストリームに流れてくるデータを処理する際 CPU を使用するよりも FPGA などのハードウェアを用いたほうが処理速度が安定することがわかっている [2][3][4]. この利点を生かしてハードウェア実装の局所差分プライバシーを実現したい.

しかしエッジコンピューティングなどで用いられる小面積で計算資源の限られたハードウェアにおいて, 局所差分プライバシーの単純な実装を行うことは固定小数点や低解像度が原因となり, 無限大のプライバシー損失が引き起こされる. この問題に対し先行研究により出力分布の幅を制限することで, 無限大のプライバシー損失を抑えることが可能であることが示されている [6].

本研究では, 出力分布の幅の制限を行っても低解像度が原因で無限大のプライバシー損失が起こる場合について説明し, 計算資源の限られたハードウェア上で実装を想定した整数値出力の局所差分プライバシー回路のプライバシー保護を強化する手法を提案する. またシミュレーションによる評価としてプライバシー損失を起こさず高い有用性を持つことを示す. 本論文の貢献は次の通りである。

- 低解像度が原因でラプラス分布の裾において発生する値抜けが先行研究における閾値を設定しても発生することを示す.
- 整数値出力において値抜けの起こらない範囲の探索法を示す.
- 提案した探索法で得られた閾値において, 使用可能な ε の範囲を示す.

本論文の構成は次のとおりである. 第二章において, 局所差分プライバシーについて説明を行う. 第三章ではハードウェアの局所差分プライバシーを実装する上で必要な逆関数法と先行研究における取り組みを紹介し, そのうえで無限大のプライバシー損失が起こる状況について説明を行う. 第四章は無限大のプライバシー損失の原因である値抜けとその回避法について提案する. 本論文は, 参考文献 [1] に基づいたものである.

[1] 九州大学大学院システム情報科学府
[2] 九州大学大学院システム情報科学研究院
[3] 熊本大学大学院先端科学研究部
[a)] taguchi.kaito.834@s.kyushu-u.ac.jp
[b)] sakurai@inf.kyushu-u.ac.jp
[c)] iida@cs.kumamoto-u.ac.jp

2. 背景

2.1 局所差分プライバシー

局所差分プライバシー [8][9] はセンサ等で得られたデータをデータベースに送信する前にデータを変化させ, これを保存することで仮にデータが漏洩してもそのデータから元データが推測されにくいという技術である.

2.1.1 定義

センサから受け取ったデータを x_1, x_2 とし, ランダムメカニズム $\mathcal{M}$ を適用して得られた出力値 y の出現確率を $Pr[\mathcal{M}(x_1) = y]$, $Pr[\mathcal{M}(x_2) = y]$ とする. このとき式 (1) を満たすことを, ε–LDP であるという.

$$Pr[\mathcal{M}(x_1) = y] \leq e^{\varepsilon} Pr[\mathcal{M}(x_2) = y] \tag{1}$$

また, プライバシー損失関数は式 (2) で表せる.

$$loss_{x_1,x_2} = \log \frac{Pr[\mathcal{M}(x_1) = y]}{Pr[\mathcal{M}(x_2) = y]} \tag{2}$$

式 (2) の値が, 小さいほどプライバシー損失が少ないことを示し, どちらか一方の出現確率が 0 となったときに無限大のプライバシーの損失が起こることを示している.

2.1.2 ラプラスメカニズム

ラプラスメカニズム [7] は局所差分プライバシーで用いられるランダムメカニズムの一つである [10]. 式 (3) ゼロ平均ラプラス分布に従ったノイズを元データに対して付与することで ε–LDP を実現する.

$$f(x) = \frac{1}{2b} e^{-\frac{|x|}{b}} \tag{3}$$

ラプラス分布の幅を決める変数である b は式 (4) で決定される.

$$b = \frac{GS}{\varepsilon} \tag{4}$$

GS とは敏感度 (global sensitivity) であり $f : D^n \to \mathbb{R}^d$ としたとき, 式 (5) によって表される.

$$GS = \max_{x,y:d(x,y)\leq 1} \|f(x) - f(y)\|_1 \tag{5}$$

2.1.3 プライバシー保護と統計的有用性のトレードオフ

前述の通り, 式 (3) であらわされるラプラス分布の幅 b は, 式 (4) で定義された. 式 (4) と式 (5) より, ε を小さくすると, 分布の幅が広がり, 異なる値から同じ出力値が得られる確率が高くなるため, 式 (2) が 0 に近づき, より強くプライバシーが守られていると判断できる. しかし, 同時に元のデータが異なるデータになる確率が高いことを意味し, これにより元のデータが保有していた統計的な性質を失う.

局所差分プライバシーにおいては, ノイズ付きのデータを用いて統計処理を行うことも想定しており, プライバシーを守るために ε を小さくしてしまえば, そもそも統計処理を目的としたデータとして用いることが困難となる. 逆に ε を大きくすると分布の幅が狭くなり, 異なる値から同じ出力値が得られる確率が低くなるため, 式 (2) が大きくなり元データが推測しやすくなる一方で, 元データから変化しない確率が高くなるため統計的な処理を行う上で精度のよい結果が得られることになる.

2.2 局所差分プライバシーの実装

2.2.1 逆関数法

固定小数点においてある確率分布に沿った乱数を得る方法として, 逆関数法があげられる. 逆関数法は, 使用したい確率分布の累積分布関数の逆関数である逆累積分布関数に $0 < u < 1$ となる一様乱数 u を入力することで, その確率における確率分布に沿った乱数が得られる. 例えば, ゼロ平均ラプラス分布の累積度数分布は, 式 (6) であらわせる.

$$u = \begin{cases} \frac{1}{2} e^{\frac{x}{b}} & (x < 0) \\ 1 - \frac{1}{2} e^{-\frac{x}{b}} & (x \geq 0) \end{cases} \tag{6}$$

これを変形して, 式 (7) を得る.

$$x = \begin{cases} b \ln (2u) & (u < 0.5) \\ -b \ln (2 - 2u) & (u \geq 0.5) \end{cases} \tag{7}$$

すなわち式 (7) を計算することで, ラプラス分布に従ったノイズを生成することが可能となる.

3. 関連研究

3.1 HW 実装の先行研究

局所差分プライバシーは理想的に, $[-\infty, \infty]$ の範囲で定義される確率分布に基づいているため出力値がどんな値であっても常に ε-LDP であることを保証できる. しかし, 実装するにあたって理想的な実数を用いることは不可能であるため, 必ず分布に上限値が存在する. ある 2 入力が与えられたとき片方の出力分布では生成できるが, もう一方の出力分布では出力できない値が存在し, 片方の出現確率が 0 となることでで式 (2) が発散し, 無限大のプライバシー損失がおこる.

例として $\varepsilon = 1, GS = 255$, 0, 255 の 2 値を入力したラプラス分布を図 1 に示す. 横軸は入力にノイズを加算したノイズ付き出力の大きさを示しており, 縦軸は各ノイズ付き出力の出現確率を表す. この例では 0 のときに約± 2828, 255 のときに約 3083, -2573 が分布の最大値となる. そのため [-2828, -2573), (2828, 3083] においてプライバシー損失が発生する.

また, 図 2 は, 図 1 の分布の右側の裾の部分を拡大したものであるが, ここで見られるように出力値に抜けが発生しており, 分布の上限と同様に片方の確率が 0 となるためプライバシー損失が起こる.

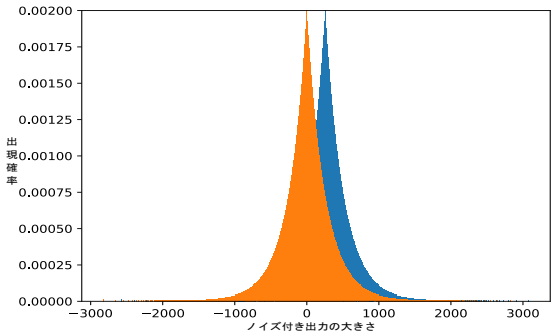

図 1 入力 0, 255, $\varepsilon = 1, GS = 255$ のラプラス分布

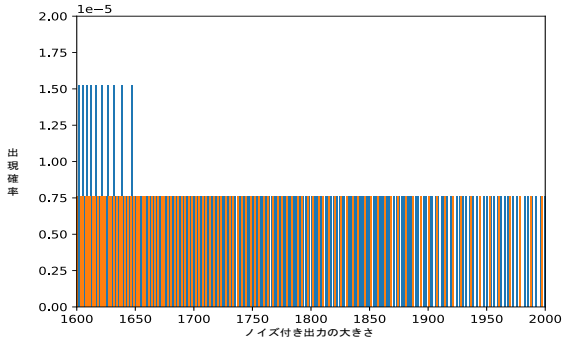

図 2 図 1 のラプラス分布の裾で起こる値抜け

先行研究において, 文献 [5] では, ラプラスメカニズムで得られたノイズ値を丸めることで細かい小数値の値抜けに対応し, また上限値を超える値に対してはある値に丸め込むことで DP が保証可能であることを示した. 文献 [6] では, 上限値を超える値に対して適切な閾値を設定して超えた場合に閾値で丸め込む閾値処理 (thresholding), 上限値を超えた場合に範囲内で収まるまでノイズ生成を繰りかえす再抽出処理 (resampling) の 2 つの手法により DP を保証した固定小数点演算のハードウェア実装 DP-Box を提案している.

3.2 先行研究における課題

文献 [6] 内で示されている適切な閾値 n_{th2} を設定すると分布の上限が原因となるプライバシー損失を抑えることは可能である. しかし, 我々は図 2 に見られるように, 依然として値抜けによるプライバシー損失が発生することを確認した.

また, 式 (7) では u がどちらの場合でも 2 倍されていることがわかる. これにより出力の際ビット数が実質 1 つ減るにもかかわらず, 以下に示す式 (8) 中ではその値の考慮が抜けていることを指摘する. 解決策として, B_x ではなく $B_x - 1$ とすればよい.

文献 [6] で示された閾値処理における適切な閾値 n_{th2} は, 敏感度を表す d, ノイズの量子化ステップ幅である Δ, 一様

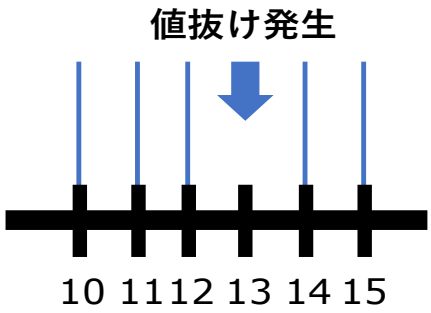

図 3 ノイズ分布の出力 13 で値抜けがおこる例

乱数のビット数である B_x, ノイズの値 n, プライバシー損失の上限を式 $n\varepsilon$ として (8) で表される.

$$n_{th2} = d + \frac{\Delta}{2} + \frac{d}{\varepsilon}(B_x \log 2 + \log(\exp(-\varepsilon) - \exp(-n\varepsilon))) \tag{8}$$

n_{th2} を設定することでプライバシー損失が $n\varepsilon$ で抑えられるというのが彼らの主張である. これをもとに python(3.10.1) にて $B_x = 16, d = 255, \delta = 1/2^{64}$ とし閾値の計算を行った. n を十分大きくとれば, ラプラス分布の出力上限である約 2828 に収束するのでこの値を右側の閾値とすれば分布に上限が存在することによるプライバシー損失を抑えることが可能である. しかし先ほど見たように 1600 付近で値抜けが起こっているため無限大のプライバシー損失は存在している.

逆に 1 より大きく小さな値をとれば, 1200 付近の値を得ることもできるがこの式だけではどこで値抜けによるプライバシー損失が起こっているかわからないという課題がある. なお, 図 1 は先に述べたように出力時に-1 ビットされることから $B_x = 17$ として出力している.

4. 提案

本研究では整数値出力の局所差分プライバシーにおいて値抜けを考慮した閾値を設定したプライバシー保護強化手法を提案する.

4.1 値抜けの発生条件

整数値出力の局所差分プライバシーにおいて, ノイズの分布のうち隣接するノイズの差分が 1 より大きいときに値抜けがあることを意味する. 例えば, 図 3 に示すように 13 で値抜けが発生した場合, 隣接したノイズの差分は 1 になるはずである. しかしノイズ 12 とノイズ 14 の差分は 2 となるためノイズ 13 が出力されておらず値抜けが発生したことを検知できる.

値抜けの主な原因は量子化された一様乱数を対数値に変換することである. 一様乱数は各値の幅が一定であるのに対し, 各一様乱数の値に対応した対数値は非線形に増加するため幅が一定ではない. 特に 0 に近い一様乱数を入力した場合対数値は大きな値になり値抜けが発生しやすい. そ

のため分布の最大値から内側に向かうようにして探索していくと最も小さな値抜けを発見できる．さらにラプラス分布は左右対称の分布であるから分布右側だけ考えればよい．

以上の条件を用いて中心の 0 から値抜けが起こらない最大の整数値 hl を見つける．hl を特定するには隣接するノイズの差分が 1 より大きいノイズのペアのうち中心に近いノイズ値の最小値を得ればよく，i 番目のノイズの値を $noise_i$ とすれば，

$$i_{min} = \min_{(noise_{i+1} - noise_i) > 1; i} i \tag{9}$$

$$hl = noise_{i_{min}} \tag{10}$$

となる．すなわち隣接するノイズの差分が 1 より大きいという条件を満たす i の最小値を式 (9) により求め，このときのノイズ値が式 (10) に示す hl となる．

4.2 値抜けなし範囲の調査

前節の方法を用いて hl をもとめ，これを値抜けなし範囲として調査する．すなわち値抜けなし範囲とはラプラス分布の右側の中心から値抜けが発生するまで長さと考えることができる．調査項目は一様乱数 u のビット数と ε の大きさである．u は 8-17 ビット，ε は 0.5-8 まで調査を行った．結果を表 1 に示す．縦軸が u，横軸が ε を示している．

表 1 ε，一様乱数 u を変化させたときの hl[1]

	0.5	0.75	1	2	3	4	5	6	7	8
8 ビット	0	0	1	9	41	54	48	54	52	50
9 ビット	1	3	20	101	102	97	90	79	77	72
10 ビット	1	3	192	187	160	140	123	113	100	94
11 ビット	1	3	359	272	220	182	156	139	123	115
12 ビット	1	3	546	365	280	224	195	172	154	137
13 ビット	1	3	716	450	338	267	229	199	177	159
14 ビット	1	3	902	535	399	319	261	223	201	180
15 ビット	1	3	1071	628	447	361	301	259	224	201
16 ビット	1	3	1257	713	510	404	334	282	255	223
17 ビット	1	3	1427	810	566	447	366	318	278	242

4.3 考察

ε が 1 より小さい場合に hl が小さくなることがわかった．原因は内部処理で用いている固定小数点のビット数にある．例えば今回は固定小数点を 8 ビットとして用いており，また入力も 8 ビット符号なし整数であるため $\varepsilon = 1$ の時点で b との乗算が凡そ 8 回分の左シフトに相当し，小数点以下のビットすべてが 0 で埋まっていることを意味する．これを踏まえ $\varepsilon = 0.5$ とするとさらに 1 回左シフトをすることと等価であるので整数部最下位の 1 ビットが 0 となり 1 を出力できなくなる．そのため 8 ビット整数を入力する場合，$\varepsilon = 0.1$ を使用するのに小数部を 12 ビット用意すればよいことになる．

表 2 FPGA のロジック数

LB	8 ビット MUL
448	16

表 3 データセット一覧

種類 [単位]
男性の身長 [cm]
女性の身長 [cm]
男性の体重 [kg]
女性の体重 [kg]
男性の最高血圧 [mmHg]
女性の最高血圧 [mmHg]
男性の最低血圧 [mmHg]
女性の最低血圧 [mmHg]
男性の BMI
女性の BMI
ランダム

5. 評価

前章の結果を踏まえ，値抜けの起こらない hl を閾値として設定し閾値処理を行った場合の局所差分プライバシーについて統計的有意性の指標である有用性の評価を行う．

5.1 評価環境

評価は iverilog を用いたシミュレーションにより行った．その際に表 2 に示した計算資源を持つ FPGA デバイスへの実装を想定し，テクノロジマッピングを行った上で搭載が可能であるものをシミュレーションに用いた [1]．

回路は入力に 8 ビット符号なし整数，出力に符号あり 12 ビット整数をとり，一様乱数 u のビット数は 17 ビットである．回路内部処理の小数点は 8 ビットで，一様乱数生成器には 43 ビット線形帰還シフトレジスタと 37 ビットセルラーオートマタシフトレジスタの複合乱数生成器を用いた [11]．

使用したデータセットは日本政府の統計サイトである e-Stat のデータをもとに作成した 10 個のデータセットとランダムな符号無 8 ビット整数値を保有するデータセットの計 11 個である．データセットの内容について表 3 に示す．各データのデータ数は 5000 である．

5.2 評価方法

局所差分プライバシーを適用したデータセットに統計処理を加えた結果の *ldp-out* と元のデータセットに対し統計処理を加えた結果の *original-out*，その 100 回分の平均絶対誤差 (MAE) をとることにより有用性を評価する (式 (11))．

$$MAE = \frac{1}{100} \sum_{i=1}^{100} |original\text{-}out_i - ldp\text{-}out_i| \tag{11}$$

また, 入力が 8 ビットである場合, 入力が 0 の時に 255 まで出力でき, 入力が 255 の時に 0 まで出力できなければならない. つまり hl が 255 以上である必要があるため乱数の解像度 17 ビットで条件を満たす $\varepsilon = 1 \sim \varepsilon = 7$ までの結果を掲載する.

なお, MAE が小さくなることは元データに対して誤差が小さくなることを意味し, このとき有用性が高いと表現する.

5.3 結果

平均処理における各データセットで得られた MAE を表 4 に示す. 種類はデータセットの種類を表し, 真値は元データセット統計処理を行った場合の値を示す.

また, 表の各値は $MAE \pm std$ である. ここで std はデータセットの真値からの誤差である MAE の標準偏差を表しおり, 式 (12) により求めた.

$$std = \sqrt{\frac{1}{100}\sum_{i=1}^{100}(|original\text{-}out_i - ldp\text{-}out_i| - MAE)^2} \tag{12}$$

平均値において高い有用性が示せた. また, 男性の BMI, 女性の BMI といった一部のデータセットでは, ε が大きくなると誤差が大きくなることがわかる.

5.4 考察

平均値は値の大きさに依存した指標であるため左右の異なる位置で丸め込むと、分布の裾が長いほうが値が大きくなるため平均値が偏り有用性が低くなる. ε を大きくすることでラプラス分布の幅が狭まり, 度数の多い中心に近い箇所で値を丸めた場合にその影響が大きくなる. そのため提案手法により丸め込み処理をより近くで行う真値が 0 や 255 に近いデータにおいてこの傾向が強く見られると考えられる.

5.5 議論

本提案により無限大のプライバシー損失を防ぐことができるが, 事前の計算が必要である点および乱数の解像度が上昇した場合, 素朴に実行すると指数時間かかることが問題点である. また, 有用性の計測においては入出力が限定的であるためより一般化した状況についても検討できれば好ましい.

さらに, 一般的に局所差分プライバシーでは $\varepsilon = 0.5$ 等の小さな値で行われることが多い. そのため特に一般化した場合に $\varepsilon < 1$ を設定可能な一様乱数の解像度および内部の固定小数点のビット数を探索する必要がある.

謝辞 本研究を進めるにあたり御助言御鞭撻を頂いた熊本大学大学院先端科学研究部久我守弘准教授, 相談に乗っていただき回路規模の検証にも協力いただいた熊本大学大学院自然科学教育部中里優弥さんに深く感謝する. 本研究は, JST, CREST, JPMJCR19K1 の支援を受けたものである.

参考文献

[1] 田口魁人, *FPGA* 向け *Local Differential Privacy* 回路の研究, 熊本大学卒業論文, 2022.
[2] S. Biookaghazadeh, Zhao, M. , and Ren, F. , *Are FPGAs Suitable for Edge Computing?.* The USENIX Workshop on Hot Topics in Edge Computing (HotEdge '18). BOSTON, MA, 2018.
[3] Rene Mueller, Jens Teubner, and Gustavo Alonso. 2009. *Data processing on FPGAs.* Proc. VLDB Endow. 2, 1 (August 2009), 910–921. DOI:https://doi.org/10.14778/1687627.1687730
[4] S. Wu et al., *When FPGA-Accelerator Meets Stream Data Processing in the Edge.* 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS), 2019, pp. 1818-1829, doi: 10.1109/ICDCS.2019.00180.
[5] Ivan Gazeau, Dale Miller, and Catuscia Palamidessi.: *Preserving differential privacy under finite-precision semantics.* Theor. Comput. Sci. 655, PB, 92–108. DOI:https://doi.org/10.1016/j.tcs.2016.01.015 (2016)

表 4 平均値: 各データセットにおける MAE[1]

種類	真値	ε=1	ε=2	ε=3	ε=5	ε=7
男性の身長	168.22	3.85 ± 3.32	1.99 ± 1.70	1.41 ± 1.18	1.10 ± 0.79	1.30 ± 0.68
女性の身長	154.28	3.84 ± 3.31	1.97 ± 1.69	1.38 ± 1.16	1.00 ± 0.75	1.01 ± 0.63
男性の体重	67.25	3.82 ± 3.25	1.92 ± 1.63	1.27 ± 1.08	0.77 ± 0.64	1.10 ± 0.58
女性の体重	53.64	3.82 ± 3.25	1.91 ± 1.63	1.27 ± 1.08	0.88 ± 0.64	1.78 ± 0.64
男性の最高血圧	128.7	3.83 ± 3.29	1.95 ± 1.67	1.33 ± 1.14	0.86 ± 0.72	0.69 ± 0.53
男性の最低血圧	76.53	3.82 ± 3.26	1.92 ± 1.64	1.28 ± 1.09	0.75 ± 0.64	0.82 ± 0.49
女性の最高血圧	122.97	3.83 ± 3.29	1.95 ± 1.67	1.33 ± 1.13	0.84 ± 0.70	0.64 ± 0.51
女性の最低血圧	72.86	3.82 ± 3.26	1.92 ± 1.64	1.28 ± 1.09	0.76 ± 0.64	0.92 ± 0.52
男性の BMI	23.74	3.82 ± 3.24	1.91 ± 1.62	1.29 ± 1.08	1.40 ± 0.77	4.62 ± 0.58
女性の BMI	22.39	3.82 ± 3.24	1.91 ± 1.62	1.29 ± 1.08	1.45 ± 0.78	4.82 ± 0.58
ランダム	125.59	3.83 ± 3.29	1.94 ± 1.67	1.32 ± 1.13	0.82 ± 0.69	0.56 ± 0.47

[6] Woo-Seok Choi, Matthew Tomei, Jose Rodrigo Sanchez Vicarte, Pavan Kumar Hanumolu, and Rakesh Kumar.: *Guaranteeing local differential privacy on ultra-low-power systems* In Proceedings of the 45th Annual International Symposium on Computer Architecture (ISCA ' 18). IEEE Press, 561–574. DOI:https://doi.org/10.1109/ISCA.2018.00053 (2018)

[7] Dwork, C., McSherry, F., Nissim, K., Smith, A. (2006). *Calibrating Noise to Sensitivity in Private Data Analysis*. In: Halevi, S., Rabin, T. (eds) Theory of Cryptography. TCC 2006. Lecture Notes in Computer Science, vol 3876. Springer, Berlin, Heidelberg. https://doi.org/10.1007/11681878_14

[8] J. C. Duchi, M. I. Jordan and M. J. Wainwright, *Local Privacy and Statistical Minimax Rates,* 2013 IEEE 54th Annual Symposium on Foundations of Computer Science, 2013, pp. 429-438, doi: 10.1109/FOCS.2013.53.

[9] S. P. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova and A. Smith, *What Can We Learn Privately?,* 2008 49th Annual IEEE Symposium on Foundations of Computer Science, 2008, pp. 531-540, doi: 10.1109/FOCS.2008.27.

[10] X. Xiong, S. Liu, D. Li, Z. Cai, and X. Niu, *A Comprehensive Survey on Local Differential Privacy,* Security and Communication Networks, vol. 2020, Article ID 8829523, 29 pages, 2020.

[11] Tkacik, T.E. (2003). *A Hardware Random Number Generator*. In: Kaliski, B.S., Koç, ç.K., Paar, C. (eds) Cryptographic Hardware and Embedded Systems - CHES 2002. CHES 2002. Lecture Notes in Computer Science, vol 2523. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-36400-5_32