局所差分プライバシー制約下における逐次heavy hitters検知

小野　元† 福地　一斗†† 佐久間　淳††,†††,††††

† 筑波大学情報学群　〒305–8577 茨城県つくば市天王台 1–1–1
†† 筑波大学システム情報工学研究科　〒305–8577 茨城県つくば市天王台 1–1–1
††† 理化学研究所 革新知能統合研究センター　〒103-0027 東京都中央区日本橋 1-4-1 日本橋一丁目三井ビルディング 15 階
†††† JST CREST　〒102–0076 102-0076 東京都千代田区五番町 7　K 's 五番町
E-mail: †{hajime,kazuto}@mdl.cs.tsukuba.ac.jp, ††jun@cs.tsukuba.ac.jp

あらまし　我々は局所差分プライバシー制約下における逐次 heavy hitters 検知というものを考える. 時系列でそれぞれのデータ提供者が 0 または 1 の状態をもつとする. 逐次 heavy hitters 検知は各時刻 $t = 1, ..., T$ において状態が 1 である提供者の割合を計算し, それがあらかじめ与えられた閾値 θ を超えた時に検出する, という問題である. ローカルなプライバシー設定においては, 提供者はプライバシーメカニズムを通してデータ収集者に状態を送信することで, プライバシーを保護する. 本研究では逐次 heavy hitters 検知のための局所差分プライバシメカニズムとして m-shot reporting を提案し, その有用性を解析する. m-shot reporting においては, 提供者は全時刻ではなく, $m(< T)$ 個の時刻でのみ状態を送信する. 我々は m が適切に選択された時に, 1-shot reporting や m-shot reporting よりも優れたトレードオフを達成することを示す. 加えて, プライバシーバジェットが時刻数 T に対して劣線形 (e.g., $O(\log T)$, $O(T^b)$ ただし $b \in (0, 1)$) のときに $O(\sqrt{T^{1-b} \log(T/N)})$ の誤差を達成することを示した. これは同条件での 1-shot reporting や T-shot reporting の誤差のオーダーより小さい.

キーワード　局所差分プライバシー, 時系列データ処理, ヘビーヒッター

1. ま え が き

オンラインサービスのサービスプロバイダーにとって, ユーザーがどのように自分たちのサービスを利用しているか知るのはサービス向上のために重要である. ユーザーの端末がオンラインのとき, サービスプロバイダーは継続的に情報を収集することができ, ユーザーの行動をよりよく知ることができる. しかしながら, このような継続的な情報収集はユーザーのプライバシーを損ないかねない [6], [11]. ユーザーについての理解とプライバシー保護の間には深刻な対立がある.

統計量開示のプライバシー保護のためのアプローチとして広く用いられているものの一つに差分プライバシー [5](differential privacy, DP) がある. これは, 公開された統計量からはデータベース中の一つのレコードのみが違うような 2 つのデータベース (隣接データベース) を確率的にしか区別できないことを保証するものである. しかしながら, DP は収集者が信頼できる状況ではデータ提供者のプライバシーを保護するが, 収集者に悪意がある場合や, クラッキング等でデータベース自体が流失した場合には提供者のプライバシーは保護されない. そこで, 収集者が信頼できない場合においても提供者のプライバシーを保護するアプローチとして局所差分プライバシー [4](local differential privacy, LDP) が提案された. DP は隣接データベースの識別不可能性の着目したアプローチであったが, それを拡張してデータ提供者が収集者にデータを送信する際に, 送信したデータの識別不可能性を保障するものである.

LDP フレームワークにおけるデータ収集は次のように特徴付けられる. (i) データ提供者はランダム化されたデータを送信する. (ii) データ収集者は提供者の真のデータを知ることができない. (iii) 収集者はメカニズムを通じて収集したデータを集約し, その統計量を得ることができる. そのため, 仮に収集者の持つデータベースが流出したとしてもユーザーのプライバシーは守られる. また, LDP を保証することはユーザーのプライバシーを保護するだけでなく, データ流出に伴い発生する潜在的なリスク (e.g., 社会的信用の喪失) から収集者を保護するはたらきもある.

本研究では, 局所差分プライバシーにおける逐次 heavy hittres 検知問題を中心に考える. これは次のような問題である. N 人のデータ提供者がおり, 時刻 $1, 2, ..., T$ の中で, 彼らが各時刻において $0, 1$ で表される状態をそれぞれ持っているという状況を考える. このとき, 各時刻において情報収集者が状態が 1 である提供者の割合があらかじめ定められた閾値よりも大きい時刻を逐次的に検知するという問題である.

この問題の具体例として次のようなものが考えられる.

- 例 1: あるアプリケーションが起動しているかどうかを運営会社がプライバシーを侵害しないようにモニタリングし, 閾値よりも多くのユーザが利用している時間はサーバの起動数増やす.
- 例 2: ある地域の人が在宅しているかいないかを電力会社がプライバシーを侵害しないようにモニタリングし, 閾値よりも多くの人が在宅している時間には電力の供給を増やす.

これらの例においては，一定期間経過後にイベントを検知しても役に立たない．イベント発生時にリアルタイムでイベントを検知したいという需要がある．すなわち，ある時刻 t における検知結果を次の時刻 $t+1$ までに出力する必要がある (逐次制約)．LDP 制約下における逐次 heavy hitters 検知問題は 3. 章で定式化する．

Related Work.

0 または 1 を LDP 制約下で送信するとき，2 つの異なる離散分布からサンプリングした値を送信することになる．階段メカニズムはこの 2 つの分布の距離を最大化するメカニズムであることが知られている [7]．そのため, 本研究でも階段メカニズムを 2 値の状態に特殊化した Binary mechanism を利用する．

LDP 制約下での heavy hitters 問題 [2], [3], [10] という問題がある．N 人の情報提供者がおり, それぞれ d 種類のアイテムのいずれかを持っている．提供者はそのアイテムをプライバシーが保護された状態でデータ収集者に送信する．収集者はプライバシー保護のためにノイズが加えられたデータを集計して, 出現頻度の高い上位 $k(<d)$ 個のアイテムを特定し, かつそれらの真の頻度を推測するという問題である．

LDP 制約下における逐次 heavy hitters 検知問題は逐次制約がなければ，T 種類のアイテムに関する heavy hitters 問題と同じ問題である．しかし，逐次制約があるという点でこの 2 つの問題は明確に区別される．

Our Contiribution.

逐次 heavy hitters 検知問題の最も単純な解法はユーザーが binary mechanism [7] を通じて毎時刻に自身の状態を報告するという方法である．我々が提案する m-shot reporting ではそれぞれのユーザーは全時刻ではなく $m(<T)$ 個の時刻でのみ報告を行う．残りの時刻ではそれぞれのユーザーは自身の状態に関係のないダミー情報を送信する．こうすることで m 個の時刻にプライバシーバジェットを集中させることができ, 有用性の向上が期待される．ユーザーが binary mechanism を使う m 個の時刻は一様ランダムに選択される．

我々は m-shot reporting の誤差の確率的上界を導出した．また, この誤差を小さくするように m を選べることも示した．最適な m を用いる m-shot reoporting を m^*-shot reporting と呼ぶ．m^*-shot reporting の誤差の上界を 1-shot reporting, T-shot reporting のそれと比較し, 以下の結果を得た．解析は定数のプライバシーバジェット ($\epsilon \in O(1)$) が与えられたとき 1-shot reporting が $O(\sqrt{T\log T/N})$ の誤差を達成すること, プイラバシーバジェットが T に対して線形に増やしても良いとき ($\epsilon \in O(T)$) に $O(\sqrt{\log T/N})$ の誤差を達成することを示した．m^*-shot reporting がそれらのいいとこ取りであることも示した．加えて, ϵ が $O(\log T)$ や $(T^{1-b})(b \in (0,1))$ といったオーダーで増加させてもいいときに, m^*-shot reporting が $O(\sqrt{T^{1-b}\log T/N})$ の誤差を達成することも示した．

(局所ではない) 差分プライバシーにおいては, 一般的には T とは無関係に設定される．しかし, 局所差分プイラバシー制約下での継続的な調査においては, 意味のある結果を得るためにはこの条件が厳しすぎる時がある．例えば, ユーザーの状況を調べる (e.g., iPhone で人気のある絵文字を探す, Safari において電力とメモリを多く消費する要因を特定する) 際に, Apple 社は局所差分プライバシーを保証し, 日毎にプライバシーバジェットを割り当てている [1]．これはユーザーから情報を集めるために時間に対して線形にプライバシーバジェットを増やしていることを意味する．m^*-reporting はプライバシーバジェットが時間に対して劣線形であるときに特に優れた性能を発揮するという意味で新しいプライバシーバジェットの配分戦略を与えている．

この論文の構成は以下の通りである．2. 章では前提知識を導入する．3. 章では局所差分プライバシー制約下における逐次 heavy hitters 問題の定式化をする．4. 章では m-shot reporting を提案する．5. 章では m-shot reporting の有用性の解析を行い, m^*-shot reporting を導く．6. 章では m^*-shot reporting に関する数値的な実験を行う．7. 章では結論を述べる．

2. 準　　備

この章では local differential privacy と binary mechanims という 2 つの必要な事前知識について説明する．

2.1 Local Differential Privacy.

ユーザー (データ提供者) がデータ収集者にデータを提供し, 収集者は集めたデータからなんらかの統計量を得ることを考える．ユーザーはデータ集合 $\mathcal{V}$ に含まれるデータ ν を持ち，メカニズムを通して収集者にデータを送信する．メカニズムは $\nu \in \mathcal{V}$ を受け取って，$z \in \mathcal{Z}$ を返す．収集者が受け取るデータはメカニズムの出力集合 $\mathcal{Z}$ の要素であるとする．このとき, ユーザーのプライバシーを保護するために，ユーザーが実際に持っているデータが収集者に決定的に推測されないことを保証したい．

Local differential privacy (LDP) [4] は，ユーザーがデータ収集者にデータを送信する前にデータをランダム化メカニズムを通じて送信することで収集者がユーザーの持つ真のデータを決定的に知ることをできなくしてユーザーのプライバシーを守るというアプローチである．形式的には以下のように定義される．

Definition 2.1. *[4]* $\mathcal{Q}$ を, 集合 $\mathcal{V}$ の要素 ν を受け取って，集合 $\mathcal{Z}$ の要素 z を出力する確率的メカニズムとする．$\epsilon > 0$ において任意のペア $\nu, \nu' \in \mathcal{V}$ と任意の部分集合 $\mathcal{S} \subset \mathcal{Z}$ に対して $\mathcal{Q}$ が次の条件を満たす時, メカニズム $\mathcal{Q}$ は ϵ-*LDP* であるという．

$$Pr[\mathcal{Q}(\nu) \in \mathcal{S}] \leq e^{\epsilon} Pr[\mathcal{Q}(\nu') \in \mathcal{S}]$$

また，ϵ をプライバシーバジェットと呼称する．プライバシーバジェットは小さいほどプライバシーが強固に保護されている．$\mathcal{Q}$ はランダム化したデータを送信するために収集者はメカニズム出力からユーザーが持つ真のデータを決定的に確定できないことに注意されたい．

プライバシーバジェットは直列合成性により複数のメカニズムに配分できることが知られている．これはよく知られた使い勝手のいい DP の性質である．

Theorem 1 (直列合成定理 [8])**.** $\mathcal{Q}_1, ..., \mathcal{Q}_K$ は同じ入力ドメイ

ンを持つプライバシーメカニズムで, それぞれのメカニズムは ϵ_i-*differential privacy* だとする．このとき, メカニズム列 $\mathcal{Q}_i$ は $(\sum_i \epsilon_i)$-*differential privacy* である．

直列合成定理より, 我々はプライバシーバジェット ϵ を分割して複数のプライバシーメカニズム配分することができる．LDP も同様に直列合成性を持つ．何故ならば LDP はユーザーが一人しかいない状況での DP を満たしていると言えるからである．

2.2 Binary Mechanism.

入力 $\mathcal{V} = \{0,1\}$, 出力 $\mathcal{Z} = \{0,1\}$ のときに ϵ-LDP を達成する確率的メカニズムの一つとして，Binary mechanism [7] が知られている．

データ提供者が 0 または 1 で表されるような情報をデータ収集者に送信する場面を考える．提供者は確率 $e^\epsilon/(e^\epsilon+1)$ で真の値を収集者に送信し，$1/(e^\epsilon+1)$ の確率で真の値と逆の値を送信する．形式的には $\nu \in \{0,1\}$ を入力とする binary mechanism Q_{Bin} は次のように表される．

$$Q_{\mathrm{Bin}}(\nu;\epsilon) = \begin{cases} \nu \text{ w.p. } \frac{e^\epsilon}{e^\epsilon+1}, \\ 1-\nu \text{ w.p. } \frac{1}{e^\epsilon+1}. \end{cases}$$

binary mechanism のプライバシーは以下の定理によって保証される．

Theorem 2. *[7] binary mechanism $Q_{Bin}(\cdot;\epsilon)$ は ϵ-LDP である．*

3. 問題の定式化

3.1 逐次 heavy hitters 検知問題

ユーザーが N 人いて, それらのユーザー $n \in [N]$ が時刻 1 から T までのそれぞれの時刻 t において状態 $v_{n,t} \in \{0,1\}$ を持っているとし, 状態が 1 であることをアクティブ, 0 であることをインアクティブと表現する．ユーザーは自分の状態を毎時刻データ収集者に送信する．データ収集者は集約関数

$$\Phi_0(V_{:,t}) = \frac{1}{N}\sum_{n=1}^{N} v_{n,t}$$

を用いてアクティブユーザーの比を観測する．データ収集者の目的は時刻 t におけるアクティブユーザーの比とあらかじめ定められた閾値 θ を比較して，不等式 $\Phi_0(V_{:,t}) \geqq \theta$ の成立を検知することである．この不等式が成立するような時刻を「heavy hitter」と呼称する．また，時刻 t が heavy hitter かどうかは時刻 $t+1$ の情報を見る前に判断し出力するものとする．この問題を逐次 heavy hitters 検知問題と呼ぶ．

3.2 LDP 制約下における逐次 heavu hitters 検知問題

本研究における問題設定は LDP 制約下における逐次 heavy hitters 検知問題である．N 人のデータ提供者と時刻 $t \in [T]$ を考える．各ユーザーは ϵ-LDP メカニズムを保証した確率的メカニズム Q を通じて自身の状態 $v_{n,t}$ をランダム化した $y_{n,t}$ を収集者に各時刻において送信する．収集者は Q のランダムネスを考慮した集約関数 $\Phi(Y_{:,t})$ を用いて heavy hitter を各時刻において検知し, 出力する．アルゴリズムは時間の長さ T をあらかじめ知っているものとする．また，各ユーザーに割り当てられたプライバシーバジェットは全てのアルゴリズムにおいて等しく ϵ であるとする．

表 1　Notation table

表記	意味
$N \in \mathbb{N}$	ユーザー (データ提供者) の数
$T \in \mathbb{N}$	期間の長さ
$[T]$	$\{1,...,T\}$
$v_{n,t} \in \{0,1\}$	時刻 t におけるユーザー n の状態
$V \in \{0,1\}^{N\times T}$	$V = \{v_{n,t}\}_{n\in N, t\in T}$
$y_{n,t} \in \{0,1\}$	ランダム化された $v_{n,t}$
$Y \in \{0,1\}^{N\times T}$	$Y = \{y_{n,t}\}_{n\in N, t\in T}$
$\Phi_0(V_{:,t}) \in [0,1]$	時刻 t における真のアクティブユーザーの比: $\Phi_0(V_{:,t}) = \lvert\{n \lvert t, v_{n,t}=1\}\rvert/N$

検知の正確性は以下で定義される (α,β)-正確性で評価する．

Definition 3.1. 集約関数 Φ が少なくとも確率 $1-\beta$ で任意の時刻 $t \in [T]$ において以下の条件を満たすならば，集約関数 Φ は *(α,β)*-正確である．

$$|\Phi(Y_{:,t}) - \Phi_0(V_{:,t})| < \alpha$$

この定義において α はアクティブユーザーの比の推定値 $\Phi(Y_{:,t})$ と真の比 $\Phi_0(V_{:,t})$ の差の上界を意味し, β は確信度を示す．本研究における目的は β, ϵ, T, N が与えれらたときに α を最小化する ϵ-LDP メカニズムを確立することである．

必要な記法を表 1 に示す．

4. m-Shot Reporting

LDP 制約下における逐次 heavy hitters 検知問題の解法として，全時刻に均等にプライバシーバジェットを配分して，ユーザーが全時刻で binary mechanism を通じて情報を送信するというナイーブな方法が考えられる．しかし，この方法はプライバシーバジェット ϵ が非常に大きい場合を除いて，α が $O(T\sqrt{\frac{\log(T)}{N}})$ になり，期間の長さ T を大きくするとき，ユーザー数 N を T の 2 乗以上の速さで増加させなければ，正確さを保てない．

これを避けるために，我々は m-shot reporting を導入する．このメカニズムにおいて, 各ユーザーは T 個の時刻の中から一様ランダムに m 個の時刻をあらかじめ選択し, プライバシーバジェットをその m 個の時刻にのみ配分しする．現在の時刻がその m 個に含まれていればユーザーは binary mechanism を用いて自身の状態を送信する．他方, 含まれていなければ状態に依存しないダミー情報を送信する．

4.1 節では LDP 制約下における逐次 heavy hitters 問題の解法として，m-shot reporting を提案し，その具体的なアルゴリズムを紹介する．4.2 節では m-shot reporting においてユーザーのプライバシーが保護されていることを保障する．ユーザーのデータ収集者に対するデータの送信が LDP を満たしていることを証明する．4.3 節では有用性とプライバシーのより

よいトレードオフを達成する m とダミー情報の選択戦略について議論する.

4.1 Algorithm

ここでは LDP 制約下における逐次 heavy hitters 検知問題の解法として m-shot reporting を提案する. まずはアルゴリズムを設計する.

アルゴリズムは逐次制約を満たしている必要があるので, ユーザーは毎時刻収集者に情報を送信する. さらに, LDP 制約を満たしている必要があるため, ユーザーの状態を収集者に送信する際には binary mechanism 通じて送信する.

また, 正確さの向上のために, 真の状態に紐付いた情報の送信は全ての時刻では行わず, 一部の時刻ではバジェットを消費しないダミー情報の送信を行う. 多くの場合において, このようにしてプライバシーバジェットを一部の時刻に集中させることで全時刻でバジェットを均一に消費するよりも正確性が高いことを 5. 節で示す.

アクティブユーザーの比が閾値を超えているかどうかを知るもっとも単純な方法の一つは比の不偏推定量を構成し, その推定量と閾値を比較することである. そのため, 集約関数 $\Phi(Y_{:,t})$ が真の比 $\Phi_0(V_{:,t})$ の不偏推定量になるように設計する.

Algorithm 1 が m-shot reporting mechanism の疑似コードである. まず 2 行目で各ユーザー n は m 個の時刻の集合 τ_n を一様ランダムに選択する. 時刻集合 τ_n に含まれる各時刻においてユーザー n は binary mechanism を通じて情報の送信を行う. τ_n に含まれない時刻においてはダミー情報の送信を行う. このサンプリングはバジェットの割り当てを決定している. m の決め方は 5. 節で議論する. 7 行目は τ_n に含まれる時刻 t におけるユーザー n の行動を表す. $t \in \tau_n$ ならばプライバシーバジェット ϵ/m だけ消費して binary mechanism で自分の状態 $v_{n,t}$ を収集者に送信する. それ以外の時刻では 9 行目のように自分の状態 $v_{n,t}$ によらず, 確率 r で 1 を送信する. これはダミー情報であり, プライバシーバジェットを消費しない. r の決め方は 5. 節で議論する. 12 行目では時刻 t における収集者の集約関数の値 $\Phi(Y_{:,t})$ と閾値 θ を比較しており, アルゴリズムはその比較結果を時刻 t における出力とする.

ここで, $\Phi_0(V_{:,t})$ の普遍推定量になるように $\Phi(Y_{:,t})$ を設計する. binary mechanism の性質より, 時刻 t に $v_{n,t}=1$ であるユーザー n が収集者に $y_{n,t}=1$ と送信する確率, $v_{n,t}=0$ であるユーザー n が収集者に $y_{n,t}=1$ と送信する確率はそれぞれ

$$Pr[y_{n,t}=1|v_{n,t}=1] = \frac{m}{T}\frac{e^{\epsilon/m}}{e^{\epsilon/m}+1} + (1-\frac{m}{T})r,$$
$$Pr[y_{n,t}=1|v_{n,t}=0] = \frac{m}{T}\frac{1}{e^{\epsilon/m}+1} + (1-\frac{m}{T})r$$

である. 簡単のため, $p = Pr[y_{n,t}=1|v_{n,t}=1], q = Pr[y_{n,t}=1|v_{n,t}=0]$ とおく. このとき, 集約関数 $\Phi(Y_{:,t})$ は以下のように与えられる:

$$\Phi(Y_{:,t}) = \frac{1}{N}\frac{1}{p-q}\left(\sum_{n=1}^{N} y_{n,t} - Nq\right)$$

集約関数 $\Phi(Y_{:,t})$ が時刻 t におけるアクティブユーザーの比

Algorithm 1: m-Shot Reporting

Input: V, θ, ϵ, m, r

1 **for** $n = 1$ *to* N **do**
2 　$\tau_n \sim$ uniform random from $\{s | s \in 2^{\{1,\dots,T\}}, |s| = m\}$;
3 **end**
4 **for** $t = 1$ *to* T **do**
5 　**for** $n = 1$ *to* N **do**
6 　　**if** $t \in \tau_n$ **then**
7 　　　$y_{n,t} = Q_{\mathrm{Bin}}(v_{n,t}, \epsilon/m)$
8 　　**else**
9 　　　$y_{n,t} = \begin{cases} 1 & \text{w.p. } r \\ 0 & \text{w.p. } 1-r \end{cases}$
10 　　**end**
11 　**end**
12 　**if** $\Phi(Y_{:,t}) \geqq \theta$ **then**
　　　Output: $\top$
13 　**else**
　　　Output: $\bot$
14 　**end**
15 **end**

$\Phi_0(V_{:,t})$ の不偏推定量であることは以下の定理によって示される.

Theorem 3. 集約関数 $\Phi(Y_{:,t})$ は $\Phi_0(V_{:,t})$ の不偏推定量である. すなわち,

$$\mathbb{E}[\Phi(Y_{:,t})] = \Phi_0(V_{:,t}).$$

証明は省略する.

4.2 Privacy Analysis

ここでは m-shot reporting が ϵ-LDP であることを示す.

Theorem 4. *m-shot reporting は ϵ-LDP である.*

証明は省略する.

4.3 m と r の選択

ここでは m-shot reporting の (α, β)-正確性について議論する. T, N 及び ϵ は問題によって決定されるパラメータであるので, m, r の決め方が議論の対象となる. まず, 収集者の集約関数 $\Phi(Y_{:,t})$ とアクティブユーザーの真の比 $\Phi_0(V_{:,t})$ の誤差についての確率的上界について議論する.

Theorem 5. $m \in \{1, 2, ..., T\}$ とする. もし,

$$\alpha < \frac{1 + (e^{\epsilon/m}+1)(\frac{T}{m}-1)r}{e^{\epsilon/m}-1} \tag{1}$$

であるならば, *Algorithm 1* で得られる $\Phi(Y_{:,t})$ は

$$Pr[\exists t \in [T]\ |\Phi(Y_{:,t}) - \Phi_0(V_{:,t})| \geqq \alpha]$$
$$\leqq 2T \exp\left(-f(m,r)\frac{\epsilon N \alpha^2}{3T}\right) \tag{2}$$

を満たす. ただし,

$$f(m,r) = \frac{m}{\epsilon}\frac{(e^{\epsilon/m}-1)^2}{e^{\epsilon/m}(e^{\epsilon/m}+1) + (e^{\epsilon/m}+1)^2(\frac{T}{m}-1)r}.$$

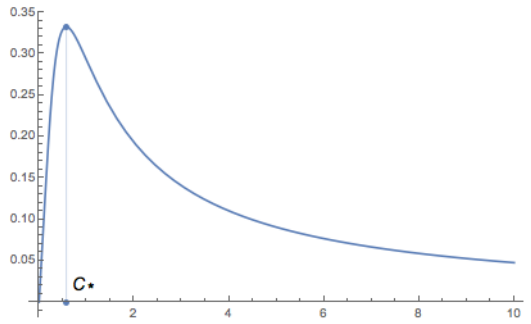

図 1　$m/\epsilon > 0$ に対する関数 $g(m/\epsilon)$ のプロット．横軸は m/ϵ を表し，縦軸は $g(m/\epsilon)$ の値を表す．

証明は付録 1. に記載した．この定理は $m = 1, T$ を含む任意の $m \in [T]$ について成り立つことに注意する．

ここから上記の上界の最小化に基づいた誤差の最小化を通じた m-shot reporting の最適化についての議論に入る．すなわち，最適な m, r を m^*, r^* として，以下で議論する問題は次のように表現される．

$$(m^*, r^*) = \underset{m\in[T], r\in[0,1]]}{\operatorname{argmax}} f(m, r) \tag{3}$$

f は $r \in [0, 1]$ について単調減少であるので，m によらず f を最大化する r は r^*=0 である．これは f の関数形より明らか．

この結果から目的関数は以下のように単純化された:

$$\max_{m,r} f(m, r) = \max_{m} f(m, 0). \tag{4}$$

$f(m, 0)$ は m/ϵ の関数として書き直せる．そこでさらなる単純化のために m/ϵ を変数として扱い，$f(m, 0) = g(m/\epsilon)$ と再定義する．ただし，$g(x) = \frac{x(e^{1/x}-1)^2}{e^{1/x}(e^{1/x}+1)}$ である．図 1 は $m/\epsilon > 0$ において $g(m/\epsilon)$ をプロットしたものである．この図より $g(m/\epsilon)$ は極値をただ一つ持つことがわかる．

$m^* = \underset{m\in\{1,\ldots,T\}}{\operatorname{argmax}} g(m/\epsilon)$ とすると，式 (2) の右辺を最小化する m が決定される．式 (3) で定義された最適な m を用いる m-shot reporting を m^*-shot reporting と呼ぶ．m^* についてのより詳しい議論は 5. 章で行う．

5. Utility Analysis

Theorem 5 で議論した関数 g が単純な形でなかったために，誤差が T と N について明らかな形で表現されていなかった．この章では m を $1, m^*, T$ にそれぞれ固定して誤差についてさらなる解析を行う．

5.1 Case 1. $m = 1$

まずはじめに 1-shot reporting の誤差を解析する．1-shot reporting では各ユーザーが全ての時刻の中で 1 度だけメカニズムを通じた情報の送信を行う．以下の定理は 1-shot reporting の誤差についての解析を与える．

Theorem 6. $\alpha_1 < \frac{1}{e^\epsilon+1}$ と仮定すると，$\epsilon > \gamma$ である普遍定数 $\gamma > 0$ が存在して，*1-shot reporting* は (α_1, β)-正確である．ただし，

$$\alpha_1 = O\left(\sqrt{\frac{T\log(T/\beta)}{N}}\right). \tag{5}$$

証明は付録 2. に記載した．

5.2 Case 2. $m = T$

次に T-shot reporting について考える．T-shot reporting においては各ユーザーが全ての時刻において ϵ/T ずつバジェットを消費してメカニズムを通じた情報の送信を行う．以下の定理は T-shot reporting についての解析を与える．

Theorem 7. $\alpha_T < \frac{1}{e^{\epsilon/T}+1}$, $\epsilon/T < 1$ と仮定すると，*T-shot reporting* は (α_T, β)-正確である．ただし，

$$\alpha_T = O\left(\sqrt{\frac{T^2\log(T/\beta)}{\epsilon^2 N}}\right). \tag{6}$$

証明は付録 3. に記載した．

Theorem 6 と Theorem 7 の比較から次のような直感が得られる．十分に大きな ϵ が使えるときには，T-shot reporting は 1-shot reporting よりの小さい誤差を与える．なぜならば，1-shot reporting の誤差は ϵ と独立と見なせるからである．α と ϵ の関係についてのより詳しい議論は 5.4 節で行う．

5.3 Case 3. $m = m^*$

最後に m^*-shot reporting を考える．4.3 節で議論したように，$g(m/\epsilon)$ を最大化する m^* を用いることによって α が小さくなることが期待される．ここで

$$c^* = \underset{x>0}{\operatorname{argmin}}\ g(x)$$

とする．残念ながら c^* の解析解は得ることができないが，図 1 より，c^* が一意に決定されることと $c^* \approx 0.574$ であることがわかる．m が各ユーザーがメカニズムを用いた状態の送信を表していることから，m は 1 以上 T 以下の整数でなければならない．したがって，$g(m/\epsilon)$ を最大化する m は以下のように 3 つの場合にわけて与えらえる:

$$m^* = \begin{cases} 1 & (\epsilon \leqq 1/c^*), \\ \underset{m'\in\{\lfloor \epsilon c^* \rfloor, \lceil \epsilon c^* \rceil\}}{\operatorname{argmax}} g(m'/\epsilon) & (1/c^* < \epsilon < T/c^*), \\ T & (\epsilon \geqq T/c^*). \end{cases} \tag{7}$$

直感的には，式 (7) は次のようなことを示唆する:

（1） $\epsilon < 1/0.575$ のとき，1-shot reporting を使うべきである．

（2） $1/0.575 < \epsilon < T/0.574$ のとき，m^*-shot reporting を使うべきである．

（3） $\epsilon > T/0.574$ のとき，T-shot reporting を使うべきである．

このように，m は ϵ と T に依存して決定される．

m^*-shot reporting の誤差は以下の定理によって与えられる．

Theorem 8. $\alpha_{m^*} < \frac{1}{e^{\epsilon/m^*}-1}$ と仮定すると *m^*-shot reporting* は (α_{m^*}, β)-正確である．ただし，

$$\alpha_{m^*} = O\left(\sqrt{\frac{T\log(T/\beta)}{\epsilon N}}\right).$$

証明は付録 4. に記載した.

5.4 Time Dependency of ϵ

Theorem 6, Theorem 7, Theorem 8 の 3 つの定理は ϵ の時間依存性を明示的に考えることにより統合することができる.

Theorem 9. プライバシーバジェットの時間依存性が $\epsilon = \Theta(T^b)$ と与えられたとする, ただし $0 \leqq b \leqq 1$. このとき, *Theorem 6* における α_1, *Theorem 8* における α^*_m, *Theorem 7* における α_T は以下のようになる

$$\alpha_1 = O\left(\sqrt{\frac{T\log(T/\beta)}{N}}\right), \tag{8}$$

$$\alpha_{m^*} = O\left(\sqrt{\frac{T^{(1-b)}\log(T/\beta)}{N}}\right), \tag{9}$$

$$\alpha_T = O\left(\sqrt{\frac{T^{(2-2b)}\log(T/\beta)}{N}}\right). \tag{10}$$

この定理はは Theorem 6, Theorem 7, Theorem 8 からただちに導かれるため証明は省く.

この定理は, プライバシーバジェットが定数の時には 1-shot reporting を, プライバシーバジェットが T に対して線形に増やせるときには T-shot reporting を使うべきだ, ということを示唆している.

m^*-shot reporting は両者のいいとこ取りである. すなわち, プライバシーバジェットが定数のときには 1-shot reporting としてふるまい, 線形に増加する時には T-shot reporting として振舞う. さらに, $\epsilon = O(\log(T)), O(T^{1/2})$ のような中間の場合でも, 1-shot reporting, T-shot reporting より小さい誤差を達成する.

6. 実　　験

議論の正しさを確かめるために以下の 2 つの実験を行った. 第一に, 式 (7) に基づく m の選択の妥当性を確かめるための実験を行った. すなわち, 式 (7) にある ϵ の条件ごとに 3 種類の設定で実験を行い, m^* が実際に式 (7) にある通りになるかどうかを確認する. 第二に, m^*-shot reporting がナイーブな方法よりも正確であることを確かめるための実験を行った. すなわち, $m^* \ll T$ のときに T を増加させて T-shot reporting と m^*-reporting の正確さを比較した.

6.1 評　　価

以下で定義される F-measure を性能評価のために用いる.

$$\text{F-measure} = \frac{2 \times \text{適合率} \times \text{再現率}}{\text{適合率} + \text{再現率}} \tag{11}$$

ただし,

$$\text{適合率} = \frac{|\{t|\Phi_0(V_{:,t}) \geqq \theta, \phi(Y_{:,t}) \geqq \theta\}|}{|\{t|\phi(Y_{:,t}) \geqq \theta\}|},$$

$$\text{再現率} = \frac{|\{t|\Phi_0(V_{:,t}) \geqq \theta, \phi(Y_{:,t}) \geqq \theta\}|}{|\{t|\Phi_0(V_{:,t}) \geqq \theta\}|}.$$

適合率はアルゴリズムが heavy hitter であるとして検知した時刻のうち, 実際に heavy hitter だった時刻の割合をあらわしており, 再現率は真の heavy hitters のうち, アルゴリズムが検知できた時刻の割合を表している. F-measure は適合率と再現率の調和平均である. F-measure は最大で 1 であり, 1 に近いほどアルゴリズムが最適解に近い結果を出力できていることを示す.

6.2 実　験　1

実験設定

実験には次のような人工データを用いた. $\Phi_0(V_{:,t}) = t/T$ となるようなデータを作成し, 列の順番をシャッフルした.

Algorithm 1 の m を 1 から T に設定してそれぞれ 100 回ずつ F-measure を測定しその平均値をプロットする. ただし $r = 0$ とする. $T = 100, N = 10000, \theta = 0.8$ は固定し, $\epsilon = 1, 10, 200$ でそれぞれ実験を行う. 式 (7) と $1/2 < c^* < 2/3$ に基づくと, $\epsilon = 1$ のとき $m^* = 1$ は $\epsilon = T$ のとき $m^* = T$. また, $\epsilon = 10$ のときは, 正確さを最大化にする m は $10 \times 1/2 < 10 \times c^* < 10 \times 2/3$ より, 6 前後と予想される.

結果

$\epsilon = 1, 10, 200$ のときの結果はそれぞれ図 2(a),2(b),2(c) である. 横軸は m を表し, 縦軸は F-measure の平均値を表している.

$\epsilon = 1$ の場合, $m = 1$ で F-measure が最大になり, その値は 0.71 であった. $\epsilon = 10$ の場合, $m = 6$ で F-measure が最大になり, その値は 0.91 であった. $\epsilon = 200$ の場合, $m = 100$ で F-measure が最大になり, その値は 0.99 であった. 式 3 により, 高い F-measure を与える m が選択できていることがわかる. また, 比較的 ϵ が大きい時には, $1 < m^* < T$ における m^*-shot reporting が有用であることもわかる.

6.3 実　験　2

実験設定

実験 1 と同じ方法で作った人工データを用いて実験を行う.

$m^* \ll T$ の場合に関して, m^*-shot reporting と T-shot reporting の F-measure の比較を行う. ただし, いずれも $r = 0$ とする. $N = 100000, \theta = 0.8$ は共通で, $T = 100, 200, ..., 1000$ のときの F-measure をそれぞれ測定する.

また, これを $\epsilon = 1, 10, 200$ の場合でそれぞれプロットする. $\epsilon = 1, 10$ のときは, $\epsilon c^* \leqq 1, 1 < \epsilon c^* < T$ の場合であり, いずれの場合も T の増加に伴って, T-shot reporting の性能は m^*-shot reporting よりも早いオーダーで悪化すると予想される. $\epsilon = 200$ のときは $m^* = T$ であるので, 2 つの F-measure はほとんど一致するはずである.

結果

図 2(d),2(e),2(f) はそれぞれ $\epsilon = 1, 10, 200$ のときの結果である. いずれも横軸は T, 縦軸は F-measure を表している. 図 2(d) においては, T-shot reporting が常に大きく引き離されていることが見て取れる. また, 図 2(e) においては, T-shot reporting は m^*-reporting よりも明らかに F-measure の低下のスピードが早い. 図 2(f) においては, 2 つのアルゴリズムの F-measure はほぼ一致している.

6.4 まとめ

2 つの実験によって次の 2 点が確かめられた.

（1） 式 (7) に基づく m の選択は妥当である.

（2） プライバシーバジェットが極端に大きい場合を除くと, T-shot reporting は F-measure において m^*-shot reporting に大きく劣る.

7. 結　論

本論文において, LDP 制約下における逐次 heavy hitters 検知問題を定式化し, それに対するアルゴリズムとして m-shot reporting を提案した. m-shot reporting において, 各ユーザーは全時刻ではなく $m(< T)$ 個の時刻でのみ自身の状態を送信する. m-shot reporting を用いた時の誤差の確率的上界を導出し, 誤差を最小化するように m を決定できることを示した. m^*-shot reporting の誤差の確率的上界と 1-shot reporting, T-shot reporting のそれを比較し, m^*-reporting がより小さい誤差を達成することを明らかにした.

8. 謝　辞

本研究は JST CREST および科学研究費 16H02864 の助成を受けました.

文　献

[1] Apple Inc. Learning with privacy at scale. 2017.

[2] Raef Bassily and Adam Smith. Local, private, efficient protocols for succinct histograms. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 127–135. ACM, 2015.

[3] Raef Bassily, Uri Stemmer, Abhradeep Guha Thakurta, et al. Practical locally private heavy hitters. In *Advances in Neural Information Processing Systems*, pages 2285–2293, 2017.

[4] John C Duchi, Michael I Jordan, and Martin J Wainwright. Local privacy and statistical minimax rates. In *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*, pages 429–438. IEEE, 2013.

[5] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *TCC*, volume 3876, pages 265–284. Springer, 2006.

[6] Saul Hansell. Aol removes search data on vast group of web users. *New York Times*, 8:C4, 2006.

[7] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. Extremal mechanisms for local differential privacy. In *Advances in neural information processing systems*, pages 2879–2887, 2014.

[8] Frank D McSherry. Privacy integrated queries: an extensible platform for privacy-preserving data analysis. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*, pages 19–30. ACM, 2009.

[9] Michael Mitzenmacher and Eli Upfal. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, New York, NY, USA, 2005.

[10] Zhan Qin, Yin Yang, Ting Yu, Issa Khalil, Xiaokui Xiao, and Kui Ren. Heavy hitter estimation over set-valued data with local differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 192–203. ACM, 2016.

[11] Yunhong Zhou, Dennis Wilkinson, Robert Schreiber, and Rong Pan. Large-scale parallel collaborative filtering for the netflix prize. In *International Conference on Algorithmic Applications in Management*, pages 337–348. Springer, 2008.

付　録

1. Proof of Theorem 5

Proof. 式 (2) の左辺の上界より,

$$Pr[\exists t|\Phi(Y_{:,t}) - \Phi_0(V_{:,t})| \geqq \alpha] \tag{A·1}$$

$$\leqq \sum_{t=1}^{T} Pr[|\Phi(Y_{:,t}) - \Phi_0(V_{:,t})| \geqq \alpha] \tag{A·2}$$

ここで確率 $Pr[|\Phi(Y_{:,t}) - \Phi_0(V_{:,t})| \geqq \alpha]$ の上界を求めるために次の定理を導入する.

Theorem 10. *[9](系 4.6)* $X_1, X_2, ..., X_n$ を独立な *Poisson* 試行とし, $Pr[X_i = 1] = p_i$ を満たすものとする. $X = \sum_{i=1}^{n} X_i$ 及び, $\mu = \mathbb{E}[X]$ とすると, 任意の $0 < \delta < 1$ に対して以下が成立する.

$$Pr[|X - \mu| \geqq \delta\mu] \leqq 2e^{-\mu\delta^2/3} \tag{A·3}$$

仮定より

$$\frac{e^{\epsilon/m} - 1}{(e^{\epsilon/m} - 1)\Phi_0(V_{:,t}) + 1 + (e^{\epsilon/m} + 1)(\frac{T}{m} - 1)r}$$

$$\leqq \frac{e^{\epsilon/m} - 1}{1 + (e^{\epsilon/m} + 1)(\frac{T}{m} - 1)r} < 1$$

であるので, Theorem 10 を用いて,

$$Pr[|\Phi(Y_{:,t}) - \Phi_0(V_{:,t})| \geqq \alpha] \tag{A·4}$$

$$\leqq 2\exp\left(-\frac{N(p-q)^2\alpha^2}{3((p-q)\Phi_0(V_{:,t}) + q)}\right) \tag{A·5}$$

$$\leqq 2\exp\left(-\frac{N(p-q)^2\alpha^2}{3p}\right) \tag{A·6}$$

と上界が求まる. ただし, 式 (A·5) から式 (A·6) への変形は任意の p, q に対して成り立つことに注意する. これと式 (A·2) より

$$Pr[\exists t, |\Phi(Y_{:,t}) - \Phi_0(V_{:,t})| \geqq \alpha]$$

$$\leqq \sum_{t=1}^{T} 2\exp\left(-\frac{N(p-q)^2\alpha^2}{3p}\right)$$

$$= 2T\exp\left(-\frac{N(p-q)^2\alpha^2}{3p}\right)$$

$$= 2T\exp\left(-\frac{m}{T}\frac{(e^{\epsilon/m} - 1)^2}{e^{\epsilon/m}(e^{\epsilon/m} + 1) + (\frac{T}{m} - 1)(e^{\epsilon/m} + 1)^2 r}\frac{N\alpha^2}{3}\right).$$

□

2. Proof of Theorem 6

Proof. Theorem 5 より,

$$\Pr[\exists t\ |\Phi(Y_{:,t}) - \Phi_0(V_{:,t})| \geqq \alpha]$$

$$\leqq 2T \cdot \exp\left(-\frac{(e^\epsilon - 1)^2}{e^\epsilon(e^\epsilon + 1)}\frac{N\alpha^2}{3T}\right). \tag{A·7}$$

ϵ は普遍定数により下からバウンドされており, かつ

$$\lim_{\epsilon\to\infty}\frac{(e^\epsilon - 1)^2}{e^\epsilon(e^\epsilon + 1)} = 1$$

であるので, $\frac{(e^\epsilon - 1)^2}{e^\epsilon(e^\epsilon + 1)}$ は T に関して定数とみなせる. $\Phi(Y_{:,t})$ が (α, β)-正確であるとすると,

$$\Pr[\exists t\ |\Phi(Y_{:,t}) - \Phi_0(V_{:,t})| \geqq \alpha] < \beta.$$

そのため, 不等式 (A·7) の右辺を β で抑えると,

$$\alpha > O\left(\sqrt{\frac{T\log(T/\beta)}{N}}\right).$$

□

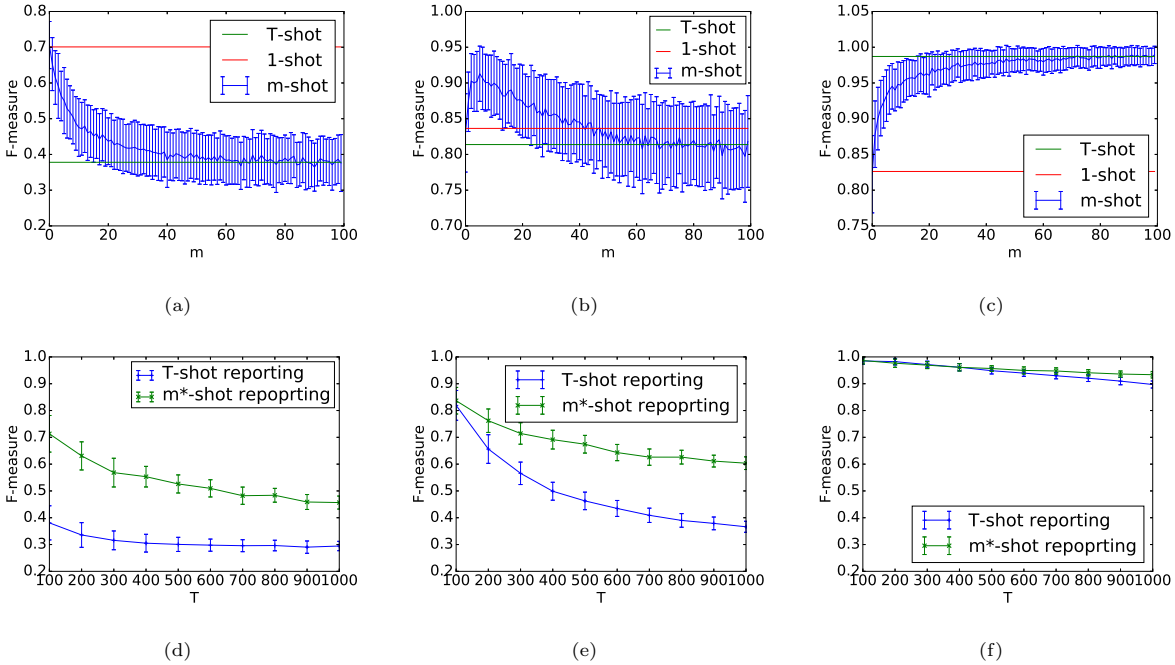

図 2　(a) $\epsilon = 1$ で m を変化させた. 縦軸は F-measure, 横軸は m. (b) $\epsilon = 10$ で m を変化させた. 縦軸は F-measure, 横軸は m. (c) $\epsilon = 200$ で m を変化させた. 縦軸は F-measure, 横軸は m. (d) $\epsilon = 1$ で T を変化させた. 縦軸は F-measure, 横軸は T. (e) $\epsilon = 10$ で T を変化させた. 縦軸は F-measure, 横軸は T. (f) $\epsilon = 200$ で T を変化させた. 縦軸は F-measure, 横軸は T.

3. Proof of Theorem 7

Proof. Theorem 5 より,

$$\Pr[\exists t, |\Phi(Y_{:,t}) - \Phi_0(V_{:,t})| \geqq \alpha]$$
$$\leqq 2T \cdot \exp\left(-\frac{(e^{\epsilon/T}-1)^2}{e^{\epsilon/T}(e^{\epsilon/T}+1)}\frac{N\alpha^2}{3}\right).$$

ここで

$$1 + x \leqq e^x \leqq 1 + x + \frac{e}{2}x^2.$$

が任意の $0 \leqq x \leqq 1$ に対して成り立ち, 仮定より $\epsilon/T < 1$ なので,

$$-\frac{(e^{\epsilon/T}-1)^2}{e^{\epsilon/T}(e^{\epsilon/T}+1)}$$
$$\leqq -\frac{\left(\frac{\epsilon}{T}\right)^2}{(1+\frac{\epsilon}{T}+\frac{e}{2}\frac{\epsilon^2}{T^2})(2+\frac{\epsilon}{T}+\frac{e}{2}\frac{\epsilon^2}{T^2})}$$
$$= -\frac{1}{(T/\epsilon+1+\frac{e}{2}\frac{\epsilon}{T})(2\frac{T}{\epsilon}+1+\frac{e}{2}\frac{\epsilon}{T})}.$$

したがって,

$$\Pr[\exists t, |\Phi(Y_{:,t}) - \Phi_0(V_{:,t})| \geqq \alpha]$$
$$\leqq 2T \cdot \exp\left(-\frac{1}{(\frac{T}{\epsilon}+1+\frac{e}{2}\frac{\epsilon}{T})(2\frac{T}{\epsilon}+1+\frac{e}{2}\frac{\epsilon}{T})}\frac{N\alpha^2}{3}\right). \quad \text{(A·8)}$$

$\Phi(Y_{:,t})$ が (α, β)-正確のとき,

$$\Pr[\exists t | \Phi(Y_{:,t}) - \Phi_0(V_{:,t})| \geqq \alpha] < \beta.$$

不等式 (A·8) の右辺を β で抑えて,

$$\alpha > \sqrt{\frac{3(\frac{T}{\epsilon}+1+\frac{e}{2}\frac{\epsilon}{T})(2\frac{T}{\epsilon}+1+\frac{e}{2}\frac{\epsilon}{T})}{N}\log\left(\frac{2T}{\beta}\right)}.$$

よって,

$$\alpha = O\left(\sqrt{\frac{T^2 \cdot \log(T/\beta)}{\epsilon^2 N}}\right).$$

□

4. Proof of Theorem 8

Proof. $m^* = 1$ または T のとき, Theorem 7, 6 からただちに導かれる.

$1/c^* < \epsilon < T/c^*$ のとき, $g(m^*/\epsilon) = g(c^*) + O(1)$. 何故ならば $g(c^*) \in O(1)$, $g(m^*/\epsilon) \in O(1)$ が成り立つからである. Theorem 5 より,

$$\Pr[\exists t\ |\Phi(Y_{:,t}) - \Phi_0(V_{:,t})| \geqq \alpha]$$
$$\leqq 2T \cdot \exp\left(-\frac{\epsilon}{T}g(m^*/\epsilon)\frac{N\alpha^2}{3}\right)$$
$$\leqq 2T \cdot \exp\left(-K\frac{\epsilon}{T}N\alpha^2\right). \quad \text{(A·9)}$$

ただし K はある定数. $\Phi(Y_{:,t})$ が (α, β)-正確とすると,

$$\Pr[\exists t, |\Phi(Y_{:,t}) - \Phi_0(V_{:,t})| \geqq \alpha] < \beta.$$

このとき, 不等式 (A·9) の右辺を β で抑えて,

$$\alpha > O\left(\sqrt{\frac{T \cdot \log(T/\beta)}{\epsilon N}}\right).$$

□