研究論文

不完全データに対する最尤推定値の標準誤差計算のための ブートストラップ法の高速化

黒田正博*（岡山理科大学経営学部）

要約：

本研究では，不完全データに対するブートストラップ法の高速化について検討する．ブートストラップ法では，リサンプリングで生成される各標本に対し EM アルゴリズムを用いた推定を行うが，その収束が遅く，リサンプリング回数の多さと相まって計算コストが大きくなる．そこで本研究では，EM アルゴリズムの収束を加速する εR-accelerated EM アルゴリズムを導入した高速化ブートストラップ法を提案し，数値実験によりその優れた推定性能および加速性能を示した．

キーワード：

ブートストラップ法，欠測データ，EM アルゴリズム，収束の加速

1. はじめに

EM アルゴリズムは，不完全データから最尤推定値を求めるための反復法であり，対数尤度関数を単調に増加させるという性質から安定した収束性を保証している (Dempster et al., 1977)．また，仮定する統計モデルの完全データに対する十分統計量が既知であれば，アルゴリズムを容易に記述できるという利点があり，高い汎用性を有している．一方で，Newton-Raphson 法とは異なり，EM アルゴリズムでは反復過程において不完全データの対数尤度関数のヘッセ行列を計算しないため，最尤推定値の漸近分散共分散行列を直接求めることができない．しかしながら，ヘッセ行列の計算が困難な場合においても適用可能であること，ならびに各反復でこの行列を計算しないことが，EM アルゴリズムの数値解法としての安定性とプログラムの容易さという優れた特性の要因となっているという事実もある．

EM アルゴリズムの枠組みにおいて最尤推定値の漸近分散共分散行列を求める方法の一つに，ブートストラップ法がある (Efron, 1979)．不完全データに対するブートストラップ法では，リサンプリングにより生成された各標本ごとに EM アルゴリズムによる反復計算を行う必要があるため，計算全体として反復回数および計算時間が極めて大きくなるという問題がある．そこで本研究では，EM アルゴリズムの収束を加速することにより，ブートストラップ計算全体の高速化を図り，反復回数および計算時間の大幅な削減を目的とする．この目的のための加速法として，Kuroda et al. (2015) により提案された εR-accelerated EM アルゴリズムを用いる．

本論文の構成は以下のとおりである．第 2 節では，不完全データに対するブートストラップ法について述べる．第 3 節では，EM アルゴリズムの収束を加速する εR-accelerated EM アルゴ

*責任著者：kuroda@ous.ac.jp

リズムを紹介する．第 4 節では，εR-accelerated EM アルゴリズムを用いた高速化ブートストラップ法を提案する．さらに，最尤推定値の標準誤差およびパラメータの信頼区間の計算法を示す．第 5 節では，数値実験により，高速化ブートスラップ法の推定性能および加速性能を評価する．第 6 節では，本論文のまとめと今後の課題について述べる．

2. 不完全データに対するブートストラップ法

観測データ $\mathbf{x}$ に欠測部分があり，その発生メカニズムに MAR (Missing At Random) を仮定する．$\mathbf{x}$ の観測部分を $\mathbf{y}$，欠測部分を $\mathbf{z}$ で表すとき，$\mathbf{z}$ に何らかの値を代入することで $\mathbf{x} = [\mathbf{y}, \mathbf{z}]^\top$ は完全データになる．また，$\mathbf{x}$，$\mathbf{y}$，$\mathbf{z}$ に対応する確率変数 $\mathbf{X}$，$\mathbf{Y}$，$\mathbf{Z}$ の標本空間をそれぞれ $\Omega_{\mathbf{X}}$，$\Omega_{\mathbf{Y}}$，$\Omega_{\mathbf{Z}}$ と書き，$\mathbf{X}$ および $\mathbf{Y}$ の従う分布の確率分布の密度関数をそれぞれ $f(\mathbf{x}|\boldsymbol{\theta})$ および $f(\mathbf{y}|\boldsymbol{\theta})$ とする．このとき，

$$f(\mathbf{y}|\boldsymbol{\theta}) = \int_{\Omega_{\mathbf{Z}}} f(\mathbf{x}|\boldsymbol{\theta}) d\mathbf{z} = \int_{\Omega_{\mathbf{Z}}} f(\mathbf{y}, \mathbf{z}|\boldsymbol{\theta}) d\mathbf{z}$$

となる．ここで，$\boldsymbol{\theta} = [\theta_1, \ldots, \theta_d]^\top$ はパラメータ空間 $\Omega_{\boldsymbol{\theta}} (\subset R^d)$ 上の未知パラメータである．不完全データ $\mathbf{y}$ から $\boldsymbol{\theta}$ の最尤推定値 $\hat{\boldsymbol{\theta}}$ を求める反復法として，EM アルゴリズムがある．

EM アルゴリズムでは，$\mathbf{y}$ に対する対数尤度関数 $\ell_o(\boldsymbol{\theta}) = \ln f(\mathbf{y}|\boldsymbol{\theta})$ の尤度方程式を解く代わりに，$\mathbf{x}$ に対する関数 $\ell_c(\boldsymbol{\theta}) = \ln f(\mathbf{x}|\boldsymbol{\theta})$ を最大化する $\hat{\boldsymbol{\theta}}$ を求める．このアルゴリズムは，$\mathbf{z}$ を推定する Expectation-step (E-step) と，$\boldsymbol{\theta}$ を求める Maximization step (M-step) からなる．第 t 回目の反復における $\mathbf{z}$ および $\boldsymbol{\theta}$ の推定値をそれぞれ $\mathbf{z}^{(t)}$ および $\boldsymbol{\theta}^{(t)}$ と表すとき，EM アルゴリズムは以下の手順で与えられる：

- **E-step:** 期待値

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) = \mathrm{E}[\ell_c(\boldsymbol{\theta})|\mathbf{y}, \boldsymbol{\theta}^{(t)}] = \int_{\Omega_{\mathbf{Z}}} \ell_c(\boldsymbol{\theta}) f(\mathbf{z}|\mathbf{y}, \boldsymbol{\theta}^{(t)}) d\mathbf{z}$$

  から $\mathbf{z}^{(t+1)}$ を計算し，$\mathbf{x}^{(t+1)} = [\mathbf{y}, \mathbf{z}^{(t+1)}]$ を得る．ここで，$f(\mathbf{z}|\mathbf{y}, \boldsymbol{\theta})$ は，$\mathbf{y}$ が与えられたもとでの $\mathbf{z}$ の予測分布であり，$f(\mathbf{z}|\mathbf{y}, \boldsymbol{\theta}) = f(\mathbf{x}|\boldsymbol{\theta}) / f(\mathbf{y}|\boldsymbol{\theta})$ である．

- **M-step:** $\mathbf{x}^{(t+1)}$ が与えられたもとで，

$$\boldsymbol{\theta}^{(t+1)} = \underset{\boldsymbol{\theta} \in \Omega_{\boldsymbol{\theta}}}{\arg\max}\, Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$$

  により $\boldsymbol{\theta}^{(t+1)}$ を推定する．

以降，EM アルゴリズムによる $\boldsymbol{\theta}^{(t)}$ の更新を

$$\boldsymbol{\theta}^{(t+1)} = M(\boldsymbol{\theta}^{(t)}) \tag{1}$$

で表すことにする．

一般に，$\hat{\boldsymbol{\theta}}$ の漸近分散共分散行列 $\mathrm{V}_o[\hat{\boldsymbol{\theta}}]$ は，最尤推定量の漸近正規性に基づき，$-\ell_o(\boldsymbol{\theta})$ のヘッセ行列に $\hat{\boldsymbol{\theta}}$ を代入した観測情報量 $\mathcal{I}_o(\hat{\boldsymbol{\theta}})$ の逆行列から求めることができる (Cox & Hinkley, 1974)．しかし，EM アルゴリズムの反復においてこのヘッセ行列を計算しないため，漸近分散共分散行列を直接求めることができない．また，EM アルゴリズムの適用場面の多くは $\ell_o(\boldsymbol{\theta})$ の尤度方程式を直接解くことが困難な場合であり，$\mathcal{I}_o(\boldsymbol{\theta})$ の計算も容易ではないことが多い．そこで，EM

アルゴリズムの枠組みの中で $\mathcal{I}_o(\hat{\boldsymbol{\theta}})$ を計算する方法が提案されている．その 1 つとして，Efron (1979) によるブートストラップ法の適用がある．ブートストラップ法では，観測データからリサンプリングにより生成された疑似観測データを用いて，$\mathrm{V}_o[\hat{\boldsymbol{\theta}}]$ を計算することができる．また，解析的な計算が困難な統計的推測問題に対してもこの方法は有効である．ブートストラップ法には，パラメトリックなモデルを仮定しないノンパラメトリックブートストラップ法と，その仮定を置くパラメトリックブートストラップ法がある．ここでは，Efron (1994) によるノンパラメトリックブートストラップ法を考える．

ノンパラメトリックブートストラップ法では，$\mathbf{y}$ から経験分布関数 $\hat{G}(\mathbf{y})$ を構成し，リサンプリングにより疑似観測データを生成する．ここで，$\hat{G}(\mathbf{y})$ は $\mathbf{y} = [\mathbf{y}_1, \ldots, \mathbf{y}_n]$ の n 個の標本点 $\mathbf{y}_1, \ldots, \mathbf{y}_n$ に等確率 $1/n$ を与えることによって求めることができる．これにより得られたブートストラップ標本の値を $\mathbf{y}^* = [\mathbf{y}_1^*, \ldots, \mathbf{y}_n^*]$ と表すことにする．リサンプリングの回数を B とし，その第 b 回目に得られたブートストラップ標本 $\mathbf{y}^{*(b)}$ に基づく $\boldsymbol{\theta}$ の最尤推定値を $\hat{\boldsymbol{\theta}}^{*(b)}$ と表す．このとき，$\hat{\boldsymbol{\theta}}$ の漸近分散共分散行列 $\mathrm{V}_o[\hat{\boldsymbol{\theta}}]$ のブートストラップ法による推定値は

$$\mathrm{V}_{\mathrm{boot}}[\hat{\boldsymbol{\theta}}] = \frac{1}{B-1}\sum_{b=1}^{B}(\hat{\boldsymbol{\theta}}^{*(b)} - \bar{\hat{\boldsymbol{\theta}}}^*)(\hat{\boldsymbol{\theta}}^{*(b)} - \bar{\hat{\boldsymbol{\theta}}}^*)^\top \tag{2}$$

により与えられる．ここで，

$$\bar{\hat{\boldsymbol{\theta}}}^* = \frac{1}{B}\sum_{b=1}^{B}\hat{\boldsymbol{\theta}}^{*(b)}$$

である．また，$\hat{\theta}_i$ $(i = 1, \ldots, d)$ の標準誤差は

$$\mathrm{SE}_{\mathrm{boot}}[\hat{\theta}_i] = \sqrt{\mathrm{V}_{\mathrm{boot}}[\hat{\theta}_i]}$$

から求めることができる．

次に，EM アルゴリズムを用いた不完全データに対するブートストラップ法による $\mathrm{V}_{\mathrm{boot}}[\hat{\boldsymbol{\theta}}]$ の計算アルゴリズムを示す．

Step 1: $\mathbf{y}$ から n 回の復元抽出

$$\mathbf{y}_1^{*(b)}, \ldots, \mathbf{y}_n^{*(b)} \sim \hat{G}(\mathbf{y})$$

をおこない，$\mathbf{y}^{*(b)} = [\mathbf{y}_1^{*(b)}, \ldots, \mathbf{y}_n^{*(b)}]$ を得る．

Step 2: $\mathbf{y}^{*(b)}$ が与えられたとき，EM アルゴリズムにより

$$\boldsymbol{\theta}^{*(b,t+1)} = M(\boldsymbol{\theta}^{*(b,t)})$$

から最尤推定値

$$\hat{\boldsymbol{\theta}}^{*(b)} = [\hat{\theta}_1^{*(b)}, \ldots, \hat{\theta}_d^{*(b)}]^\top$$

を求める．

Step 1 と Step 2 のブートストラップ反復を B 回おこない，$\{\hat{\boldsymbol{\theta}}^{*(b)}\}_{b=1,\ldots,B}$ を求め，式 (2) により $\mathrm{V}_{\mathrm{boot}}[\hat{\boldsymbol{\theta}}]$ を計算する．

一般に，漸近分散共分散行列の推定おいて，B は 50 から 200 で十分であることが多い．一方，$\boldsymbol{\theta}$ の信頼区間の推定では，B を 1000 から 2000 とすることが必要であると言われている．このとき，$\mathbf{y}^{*(b)}$ $(b = 1, \ldots, B)$ に対して EM アルゴリズムの反復計算が必要となり，その総反復回数は膨大になり，それ伴い計算時間も多大になる．そこで，EM アルゴリズムの収束を加速することでブートストラップ法による計算時間を短縮することを考える．

3. EM アルゴリズムの加速

EM アルゴリズムは線形収束するため，適用する統計モデルの複雑さや観測データに占める欠測データの割合に依存して，収束が遅くなることが知られている (Meng & Rubin, 1994)．そこで，EM アルゴリズムの収束を加速するために，Kuroda et al. (2015) による εR-accelerated EM アルゴリズムを用いる．このアルゴリズムは，Wynn (1962) の提案した vector ε（vε）アルゴリズムにより EM アルゴリズムの収束を加速する ε-accelerated EM アルゴリズム (Kuroda & Sakakihara, 2006; Wang et al., 2008) に，re-starting step を組み込み収束速度を改良したものである．vε アルゴリズムは，補外法の 1 つであり，線形収束する反復法から生成されるベクトル列の収束が遅い場合に有効である．

vε アルゴリズムを簡単に説明する．ある反復法から生成されるベクトル列を $\{\boldsymbol{\theta}^{(t)}\}_{t\geq 0}$ で表し，$\boldsymbol{\theta}$ の逆行列を $\boldsymbol{\theta}^{-1} = \boldsymbol{\theta}/\boldsymbol{\theta}^{\top}\boldsymbol{\theta}$ で定義する．vε アルゴリズムは以下の式から変換列 $\{\boldsymbol{\psi}^{(t)}\}_{t\geq 0}$ を生成する：

$$\boldsymbol{\psi}^{(t-1)} = \boldsymbol{\theta}^{(t)} + \left[\left[\Delta\boldsymbol{\theta}^{(t)}\right]^{-1} - \left[\Delta\boldsymbol{\theta}^{(t-1)}\right]^{-1}\right]^{-1} \tag{3}$$

ここで，$\Delta\boldsymbol{\theta}^{(t-1)} = \boldsymbol{\theta}^{(t)} - \boldsymbol{\theta}^{(t-1)}$ である．このとき，$\{\boldsymbol{\theta}^{(t)}\}_{t\geq 0}$ が停留点 $\hat{\boldsymbol{\theta}}$ に収束するならば，$\{\boldsymbol{\psi}^{(t)}\}_{t\geq 0}$ は $\{\boldsymbol{\theta}\}_{t\geq 0}$ より速く $\hat{\boldsymbol{\theta}}$ に収束する (Wynn, 1962)．

vε アルゴリズムを ε-acceleration step として EM アルゴリズムに組み込んだのが ε-accelerated EM アルゴリズムである．さらに，収束速度の向上を目的としてリスタート条件を導入したものが，εR-accelerated EM アルゴリズムである．EM アルゴリズム (EM step) をリスタートすることは，ε-acceleration step で生成される列 $\{\boldsymbol{\psi}^{(t)}\}_{t\geq 0}$ が EM 列 $\{\boldsymbol{\theta}^{(t)}\}_{t\geq 0}$ より速く $\hat{\boldsymbol{\theta}}$ に近づくことに注目したものである．εR-accelerated EM アルゴリズムは以下の手順を繰り返す：

- **EM step:** EM 列 $\{\boldsymbol{\theta}^{(t)}\}_{t\geq 0}$ を

  $$\boldsymbol{\theta}^{(t+1)} = M(\boldsymbol{\theta}^{(t)})$$

  により生成する．

- **ε-acceleration step:** $\{\boldsymbol{\theta}^{(t-1)}, \boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}^{(t+1)}\}$ から，式 (3) により $\{\boldsymbol{\psi}^{(t)}\}_{t\geq 0}$ を生成する．
  - **re-starting step:** 条件 $||\Delta\boldsymbol{\psi}^{(t-2)}||^2 < \delta_{Re}$ かつ $\ell_o(\boldsymbol{\psi}^{(t-1)}) > \ell_o(\boldsymbol{\theta}^{(t+1)})$ を満足するとき，

    $$\boldsymbol{\theta}^{(t+1)} = M(\boldsymbol{\psi}^{(t-1)})$$

    により $\boldsymbol{\theta}^{(t+1)}$ を更新し，$\boldsymbol{\theta}^{(t)}$ と δ_{Re} を再設定する：

    $$\boldsymbol{\theta}^{(t)} := \boldsymbol{\psi}^{(t-1)}, \qquad \delta_{Re} := \delta_{Re} \times 10^{-k}$$

収束判定を $||\Delta\boldsymbol{\psi}^{(t-2)}||^2 < \delta$ によりおこなう．

εR-accelerated EM アルゴリズムが T 回で収束したとき，$\hat{\boldsymbol{\theta}} = \boldsymbol{\psi}^{(T)}$ となる．また，re-starting step における条件

- **条件 1**：$||\Delta\boldsymbol{\psi}^{(t-2)}||^2 < \delta_{Re}$
- **条件 2**：$\ell_o(\boldsymbol{\psi}^{(t-1)}) > \ell_o(\boldsymbol{\theta}^{(t+1)})$

において，条件 1 は EM step をリスタートするための候補となる値を見つけるためのものであり，条件 2 はその候補値が適切であるかどうかの妥当性をチェックする．この条件を満たすことで，リスタート後においても EM step の $\{\ell_o(\boldsymbol{\theta}^{(t)})\}_{t\geq 0}$ が単調増加列であることを保証する．

ここで，リスタート条件を $\delta_{Re} = 1(= 10^0)$，$k = 1$ と設定し，εR-accelerated EM アルゴリズムの収束判定基準を $\delta = 10^{-12}$ とする．このとき，$\delta_{Re} = 10^0, 10^{-1}, \ldots, 10^{-11}$ において，高々 12 回の EM step のリスタートが実行されることになる．この re-starting step の特徴は，リスタートの回数を δ_{Re} の初期値と k でコントロールができる点である．

4. 不完全データに対する高速化ブートストラップ法

第 2 節で示した不完全データに対するブートストラップ法に εR-accelerated EM アルゴリズムを適用する．この方法を高速化ブートストラップ法と呼び，以下の手順で与える：

Step 1: $\mathbf{y}$ から n 回の復元抽出

$$\mathbf{y}_1^{*(b)}, \ldots, \mathbf{y}_n^{*(b)} \sim \hat{G}(\mathbf{y})$$

をおこない，$\mathbf{y}^{*(b)} = [\mathbf{y}_1^{*(b)}, \ldots, \mathbf{y}_n^{*(b)}]$ を得る．

Step 2: $\mathbf{y}^{*(b)}$ が与えられたとき，εR-accelerated EM アルゴリズムにより最尤推定値 $\hat{\boldsymbol{\theta}}^{*(b)}$ を求める．

この反復を B 回おこなうことで得られた $\{\hat{\boldsymbol{\theta}}^{*(b)}\}_{b=1,\ldots,B}$ を用いて，式 (2) により $\mathrm{V}_{\mathrm{boot}}[\hat{\boldsymbol{\theta}}]$ を計算する．

さらに，$\mathrm{V}_{\mathrm{boot}}[\hat{\boldsymbol{\theta}}]$ から得られる $\mathrm{SE}_{\mathrm{boot}}[\hat{\theta}_i]$ $(i = 1, \ldots, d)$ を用いて，$\boldsymbol{\theta}$ の信頼区間を構成することができる．$\hat{\theta}_i$ が平均 θ_i，分散 σ_i^2 の正規分布に近似的に従うと仮定する．このとき，θ_i に対する $(1-2\alpha)\times 100\%$ 信頼区間は

$$\left[\hat{\theta}_i - z_\alpha \sigma_i,\ \hat{\theta}_i + z_\alpha \sigma_i\right]$$

で近似することができる．ここで，標準正規分布の分布関数を Φ で表すとき，$z_\alpha = \Phi^{-1}(\alpha)$ である．また，σ_i が未知のとき，その推定値として $\mathrm{SE}_{\mathrm{boot}}[\hat{\theta}_i]$ を用いることで，正規近似による $(1-2\alpha)\times 100\%$ 信頼区間は

$$\left[\hat{\theta}_i - z_\alpha \mathrm{SE}_{\mathrm{boot}}[\hat{\theta}_i],\ \hat{\theta}_i + z_\alpha \mathrm{SE}_{\mathrm{boot}}[\hat{\theta}_i]\right]$$

で与えられる．正規分布の仮定をせずに信頼区間を求める方法として，パーセンタイル法がある．パーセンタイル法では，ブートストラップ反復により得られた $\{\hat{\boldsymbol{\theta}}^{*(b)}\}_{1\leq b\leq B}$ から $\{\hat{\theta}_i^{*(b)}\}_{1\leq b\leq B}$ を昇順

$$\hat{\theta}_i^{*1} \leq \hat{\theta}_i^{*2} \leq \cdots \leq \hat{\theta}_i^{*B}$$

に並び替え，ブートストラップ分布のパーセント点から θ_i に対する $(1-2\alpha)\times 100\%$ 信頼区間

$$\left[\hat{\theta}_i^{*\alpha B},\ \hat{\theta}_i^{*(1-\alpha)B}\right]$$

を求めればよい．ここで，$\hat{\theta}_i^{*b}$ は昇順に並べた B 個の中の第 b 番目の値である．

5. 数値実験

本数値実験では，εR-accelerated EM アルゴリズムを用いた高速化ブートストラップ法の推定性能，および反復回数と CPU 時間 (秒) による加速性能を評価する．ブートストラップ標本のリサンプリング回数を $B=2000$ とする．各 $\mathbf{y}^{*(b)}$ からの $\hat{\boldsymbol{\theta}}^{*(b)}$ を得るための EM アルゴルズムおよび εR-accelerated EM アルゴリズムの収束判定条件を $\delta=10^{-12}$ とし，εR-accelerated EM アルゴリズムのリスタート条件を $\delta_{Re}=10^{-k}$ $(k=0,1,2,\ldots,11)$ とする．

5.1. 分割表におけるブートストラップ標本の生成

2 値変数 Y_1 および Y_2 について，表 1 の分割表の形式で観測データ $\mathbf{y}_0=[y_{11},y_{21},y_{12},y_{22}]$, $\mathbf{y}_1=[r_1,r_2]$ および $\mathbf{y}_2=[c_1,c_2]$ が得られた場合を考える．ここで，$\mathbf{y}_1$ は Y_2 の観測が欠測し，$\mathbf{y}_2$ は Y_1 の観測が欠測している．なお，欠測を “$*$” で表す．

ブートストラップ標本を生成するため，表 1 に示した分割表を表 2 による個票データの形式へ書き換える．ここで，$y_{++}=\sum_{i,j=1,2}y_{ij}$, $r_+=\sum_{i=1,2}r_i$ および $c_+=\sum_{j=1,2}c_j$ であり，また $n_1=y_{++}$, $n_2=n_1+r_+$ および $n=n_2+c_+=y_{++}+r_++c_+$ である．

表 1: 観測データ $\mathbf{y}=[\mathbf{y}_0,\mathbf{y}_1,\mathbf{y}_2]$

$\mathbf{y}_0=[y_{11},y_{21},y_{12},y_{22}]$		
	$Y_2=1$	$Y_2=2$
$Y_1=1$	y_{11}	y_{12}
$Y_1=2$	y_{21}	y_{22}

$\mathbf{y}_1=[r_1,r_2]$	
	$Y_2=*$
$Y_1=1$	r_1
$Y_1=2$	r_2

$\mathbf{y}_2=[c_1,c_2]$		
	$Y_2=1$	$Y_2=2$
$Y_1=*$	c_1	c_2

表 2: 個票形式による観測データ $\mathbf{y}=[\mathbf{y}_0,\mathbf{y}_1,\mathbf{y}_2]$

s	1	$\cdots$	y_{11}	$y_{11}+1$	$\cdots$	$y_{11}+y_{21}$	$y_{11}+y_{21}+1$	$\cdots$	$y_{11}+y_{21}+y_{12}$
Y_1	1	$\cdots$	1	2	$\cdots$	2	1	$\cdots$	1
Y_2	1	$\cdots$	1	1	$\cdots$	1	2	$\cdots$	2

s	$y_{11}+y_{21}+y_{12}+1$	$\cdots$	n_1	n_1+1	$\cdots$	n_1+r_1	n_1+r_1+1	$\cdots$	n_2
Y_1	2	$\cdots$	2	1	$\cdots$	1	2	$\cdots$	2
Y_2	2	$\cdots$	2	$*$	$\cdots$	$*$	$*$	$\cdots$	$*$

s	n_2+1	$\cdots$	n_2+c_1	n_2+c_1+1	$\cdots$	n
Y_1	$*$	$\cdots$	$*$	$*$	$\cdots$	$*$
Y_2	1	$\cdots$	1	2	$\cdots$	2

$\mathbf{y}=[\mathbf{y}_0,\mathbf{y}_1,\mathbf{y}_2]$ に対して，パラメータに $\boldsymbol{\theta}$ をもつ多項分布を仮定する．この多項分布モデルのもとで，観測データ $\mathbf{y}=[\mathbf{y}_0,\mathbf{y}_1,\mathbf{y}_2]$ に基づく $\hat{\boldsymbol{\theta}}$ を EM アルゴリズムにより求める．このモデルに対する EM アルゴリズムは，付録 A を参照のこと．

このとき，$\mathbf{y}$ に対する高速化ブートスラップ法は以下の手順を繰り返す：

Step 1. 表 2 の個票データから n 回の復元抽出

$$S^* = \{s_1^*, \ldots, s_n^*\} \sim \mathrm{unif}(n, (1/n, \ldots, 1/n))$$

をおこない，$\mathbf{y}^{*(b)} = [\mathbf{y}_1^{*(b)}, \mathbf{y}_2^{*(b)}, \mathbf{y}_3^{*(b)}]$ を生成する．

Step 2. εR-accelerated EM アルゴリズムを用いて $\mathbf{y}^{*(b)}$ から $\hat{\boldsymbol{\theta}}^{*(b)}$ を求める．

これを B 回おこない $\{\hat{\boldsymbol{\theta}}^{*(b)}\}_{1 \le b \le B}$ を得ることで，ブートストラップ分散共分散行列 $\mathrm{V}_{\mathrm{boot}}[\hat{\boldsymbol{\theta}}]$ を式 (2) から計算し，$\hat{\theta}_{ij}$ の標準誤差 $\mathrm{SE}_{\mathrm{boot}}[\hat{\theta}_{ij}]$ $(i, j = 1, 2)$ を求める．

5.2. 高速化ブートストラップ法による $\boldsymbol{\theta}^*$ の標準誤差および $\boldsymbol{\theta}$ の信頼区間による推定性能

観測データ $\mathbf{y} = [\mathbf{y}_0, \mathbf{y}_1, \mathbf{y}_2]$ として，表 3 の分割表を考える．多項分布モデルのもとで，$\mathbf{y}$ に基づく $\hat{\boldsymbol{\theta}}$ の標準誤差および $\boldsymbol{\theta}$ の信頼区間を，ブートストラップ法と高速化ブートストラップ法により求める．

表 3: 観測データ $\mathbf{y} = [\mathbf{y}_0, \mathbf{y}_1, \mathbf{y}_2]$

(a) $\mathbf{y}_0 = [y_{11}, y_{21}, y_{12}, y_{22}]$

	$Y_2 = 1$	$Y_2 = 2$
$Y_1 = 1$	5	3
$Y_1 = 2$	4	6

(b) $\mathbf{y}_1 = [r_1, r_2]$

	$Y_2 = *$
$Y_1 = 1$	100
$Y_1 = 2$	300

(c) $\mathbf{y}_2 = [c_1, c_2]$

	$Y_2 = 1$	$Y_2 = 2$
$Y_1 = *$	250	150

表 4 は，EM アルゴルズムによるブートストラップ法および εR-accelerated EM アルゴリズムを用いた高速化ブートストラップ法の計算に要した反復回数と CPU 時間を示している．この表より，従来のブートストラップ法と比較したとき，高速化ブートストラップ法は，反復回数および CPU 時間のいずれにおいても大幅な短縮を実現していることがわかる．

表 5 および表 6 は，ブートストラップ法ならびに高速化ブートストラップ法によって得られた $\hat{\boldsymbol{\theta}}$ の標準誤差と，$\boldsymbol{\theta}$ の 95% 信頼区間をそれぞれ示したものである．これらの結果から，高速化ブートストラップ法は，ブートストラップ法と同等の推定精度を保持していることがわかる．

表 4: ブートストラップ法 (EM) および高速化ブートストラップ法 (εR) の反復回数と CPU 時間 (秒)

	反復回数	CPU 時間
EM	367,610	13.22
εR	45,770	4.19

この数値実験において，高速化ブートストラップ法は，従来のブートストラップ法と同等の推定性能を有しつつ，反復回数および CPU 時間の両面において大幅な改善を実現していることが

表 5: ブートストラップ法 (EM) および高速化ブートストラップ法 (εR) による $\hat{\boldsymbol{\theta}}$ の標準誤差

	EM	εR
$\mathrm{SE}_{\mathrm{boot}}[\hat{\theta}_{11}]$	0.047875	0.047876
$\mathrm{SE}_{\mathrm{boot}}[\hat{\theta}_{21}]$	0.062430	0.062431
$\mathrm{SE}_{\mathrm{boot}}[\hat{\theta}_{12}]$	0.038329	0.038330
$\mathrm{SE}_{\mathrm{boot}}[\hat{\theta}_{22}]$	0.049349	0.049350

表 6: パーセンタイル法による $\boldsymbol{\theta}$ の 95% 信頼区間 (ブートストラップ法：EM，高速化ブートストラップ法：εR)

	EM		εR	
	下限	上限	下限	上限
θ_{11}	0.099164	0.264999	0.099159	0.264982
θ_{21}	0.338822	0.526223	0.338833	0.526241
θ_{12}	0.000000	0.157545	0.000000	0.157551
θ_{22}	0.220446	0.390745	0.220433	0.390743

確認された．

5.3. 高速化ブートストラップ法の加速性能

乱数データを用いて，高速化ブートストラップ法の加速性能を評価する．$y_{++} = 18$ および $r_+ = c_+ = 400$ とし，多項分布に従う乱数を用いて $\mathbf{y} = [\mathbf{y}_0, \mathbf{y}_1, \mathbf{y}_2]$ を生成する．この操作を 100 回繰り返し，各乱数データ $\mathbf{y}$ に対してブートストラップ法および高速化ブートストラップ法による $\hat{\boldsymbol{\theta}}$ の標準誤差を推定する．

高速化ブートストラップ法の加速性能を，反復回数および CPU 時間，ならびに以下で定義する Speed-up の観点から評価する．

$$\text{Speed-up} = \frac{\text{ブートストラップ法の反復回数 (CPU 時間)}}{\text{高速化ブートストラップ法の反復回数 (CPU 時間)}}$$

表 7 は，ブートストラップ法および高速化ブートストラップ法における反復回数と CPU 時間を要約統計量としてまとめたものである．また，図 1 は，これらの結果を箱ひげ図で示したものである．高速化ブートストラップ法における反復回数の最大値は従来のブートストラップ法の最小値を下回っており，CPU 時間についても高速化ブートストラップ法の値は従来法の第 1 四分位数より小さい．ただし，高速化ブートストラップ法では反復回数が少ない場合に，従来のブートストラップ法よりも CPU 時間が長くなることがある．これは，高速化ブートストラップ法では各反復において vε アルゴリズムを適用するため，1 反復あたりの計算時間が従来のブートストラップ法よりも増加することに起因する．その結果，高速化ブートストラップ法における CPU 時間の最小値が，従来法のそれを上回る場合が生じている．しかしながら，表 7 の結果から，従来のブートストラップ法と比較して，高速化ブートストラップ法は反復回数および CPU 時間の

両面において，計算コストを大幅に削減できていることがわかる．

表 7: ブートストラップ法 (EM) および高速化ブートストラップ法 (εR) による反復回数および CPU 時間 (秒) の要約統計量

	反復回数		CPU 時間	
	EM	εR	EM	εR
最小値	53,660	19,300	2.480	2.630
第 1 四分位数	129,200	32,340	5.378	3.650
中央値	239,300	38,920	8.970	4.040
平均	246,500	38,240	9.474	3.959
第 3 四分位数	360,400	45,900	13.240	4.290
最大値	472,300	47,650	17.850	4.570

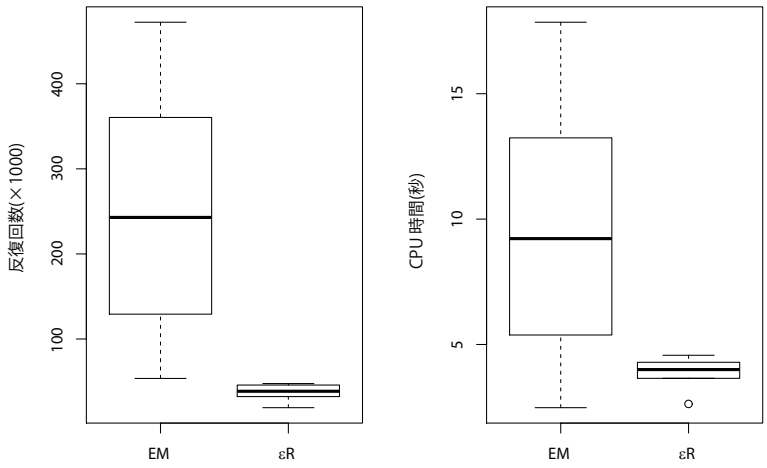

図 1: ブートストラップ法 (EM) および高速化ブートストラップ法 (εR) による反復回数 (左) および CPU 時間 (右) の箱ひげ図

高速化ブートストラップ法の加速性能を評価するため，Speed-up の結果を表 8 に示す．この表より，平均的には反復回数で約 6 倍，CPU 時間で約 2.3 倍の高速化が達成されていることがわかる．反復回数に比べて CPU 時間における Speed-up が相対的に小さい理由として，本数値実験で用いた分割表が二元表であり，モデル構造が単純な飽和モデルであったことが挙げられる．二元表の飽和モデルではパラメータ数が 3 と少なく，最尤推定値を求める式が明示的に与えられるため M-step の計算が容易であり，1 反復あたりの計算量が小さい．高次元分割表に対してより複雑なモデルを仮定した場合には，計算時間においても大幅な削減が期待できる．

次に，ブートストラップ法における CPU 時間と Speed-up の関係を検討する．このため，図 2 には，従来のブートストラップ法の CPU 時間に対する高速化ブートストラップ法の CPU 時間の Speed-up を示した散布図を示す．この図から，ブートストラップ法の CPU 時間と Speed-up の間に正の相関関係が確認できる．これは，従来のブートストラップ法における計算時間が長い

表 8: 反復回数および CPU 時間 (秒) の Speed-up の要約統計量

	反復回数	CPU 時間
最小値	2.640	0.7671
第 1 四分位数	4.048	1.5030
中央値	6.185	2.2240
平均	6.064	2.3120
第 3 四分位数	7.845	3.1060
最大値	10.300	4.1470

場合ほど，高速化ブートストラップ法による計算速度の向上が顕著になることを意味している．実際に，ブートストラップ法における CPU 時間の第 3 四分位数（13.24）以上の領域では，3 倍以上の高速化が達成されている．ブートストラップ法の高速化は計算コストが高い場面での適用が特に重要であるため，これらの結果は，本提案手法の有効性と優位性を示すものといえる．

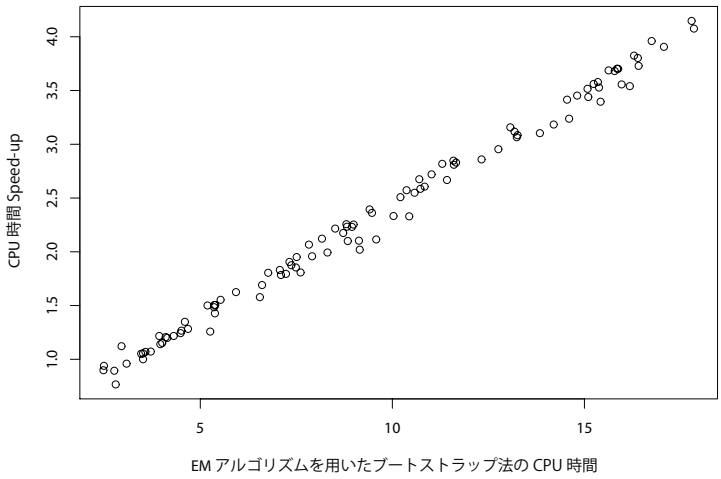

図 2: ブートストラップ法の CPU 時間に対する高速化ブートストラップ法の CPU 時間 Speed-up の散布図

6. おわりに

本研究では，不完全データに対する高速化ブートストラップ法を提案した．不完全データからのパラメータ推定では反復計算が必要であり，収束の安定性と汎用性の観点から EM アルゴリズムが広く用いられている．しかし，EM アルゴリズムは線形収束であるため，ブートストラップ法ではリサンプリング回数に比例して反復計算を要し，反復回数および計算時間が著しく増大するという問題がある．そこで本研究では，εR-accelerated EM アルゴリズムをブートストラップ法に導入し，これらの計算コストの削減を図った．

数値実験の結果，最尤推定値の標準誤差およびパラメータの信頼区間の推定において，高速化ブートストラップ法は EM アルゴリズムを用いた従来のブートストラップ法と同程度の推定性能

を有していることが示された．さらに，高速化ブートストラップ法は，反復回数および CPU 時間の両面において，計算コストを大幅に削減できることが確認された．特に，従来のブートストラップ法で計算コストが増大する状況では，本高速化法の有効性が明確に示された．

Kuroda et al. (2015) では，混合正規分布モデルの最尤推定において，εR-accelerated EM アルゴリズムが計算効率の改善に大きく寄与することが示されている．この結果を踏まえると，最尤推定値の標準誤差およびパラメータの信頼区間の推定においても，本アルゴリズムが有効な計算手法となることが期待できる．今後の課題としては，より複雑な欠測構造をもつ統計モデルおよび高次元パラメータ空間への適用，加えて他の EM アルゴリズムの加速法との比較検討が挙げられる．これらを通じて，不完全データ解析におけるブートストラップ推論のさらなる効率化が期待できる．

謝辞

本研究は JSPS 科研費 JP21K11800 の助成を受けたものである．

参考文献

Dempster A.P., Laird, N.M., Rubin D.B. (1977). Maximum likelihood from incomplete data via the EM algorithm, *Journal of the Royal Statistical Society, Series B*, **39**, 1–22.

Efron, B. (1979). Bootstrap methods: Another look at the jackknife. *Annals of Statistics*, **7**, 1–26.

Efron, B. (1994). Missing data, imputation, and the bootstrap. *Journal of the American Statistical Association*, **89**, 463–475.

Cox, D.R. and Hinkley, D.V. (1974). *Theoretical statistics*. Chapman & Hall, London.

Kuroda, M. and Sakakihara, M. (2006). Accelerating the convergence of the EM algorithm using the vector epsilon algorithm. *Computational Statistics & Data Analysis*, **51**, 1549–1561.

Kuroda, M., Geng, Z. and Sakakihara, M. (2015). Improving the vector ε acceleration for the EM algorithm using a re-starting procedure. *Computational Statistics*, **30** pp. 1051–1077.

Meng, X.L. and Rubin, D.B. (1994). On the global and componentwise rates of convergence of the EM algorithm. *Linear Algebra and its Applications*, **199**, pp. 413–425.

Wang, M., Kuroda, M., Sakakihara, M. and Geng, Z. (2008). Acceleration of the EM algorithm using the vector epsilon algorithm. *Computational Statistics*, **23**, 469–486.

Wynn, P. (1962). Acceleration techniques for iterated vector and matrix problems. *Mathematics of Computation*, **16**, 301–322.

付録 A. EM アルゴリズムの導出

第 5 節の多項分布モデルにおける EM アルゴリズムを示す．表 1 で与えられる分割表において，$\mathbf{y}_1$ および $\mathbf{y}_2$ の欠測部分を補完 (impute) した拡大データをそれぞれ $\tilde{\mathbf{y}}_1 = [\tilde{r}_{11}, \tilde{r}_{21}, \tilde{r}_{12}, \tilde{r}_{22}]$ および $\tilde{\mathbf{y}}_2 = [\tilde{c}_{11}, \tilde{c}_{21}, \tilde{c}_{12}, \tilde{c}_{22}]$ で表す．ただし，

$$\sum_{j=1,2} \tilde{r}_{ij} = r_i, \qquad \sum_{i=1,2} \tilde{c}_{ij} = c_j$$

である．このとき，$\tilde{\mathbf{y}} = [\mathbf{y}_0, \tilde{\mathbf{y}}_1, \tilde{\mathbf{y}}_2]$ はパラメータに $\boldsymbol{\theta}$ をもつ多項分布モデルからの観測データとする．$\mathbf{y}$ に対する $\boldsymbol{\theta}$ の対数尤度関数は

$$\ell_o(\boldsymbol{\theta}) \propto \sum_{i=1,2} \sum_{j=0,1} y_{ij} \ln \theta_{ij} + \sum_{i=1,2} r_i \ln \theta_{i+} + \sum_{j=1,2} c_j \ln \theta_{+j}$$

である．この尤度関数は εR-ccelerated EM アルゴリズムの re-starting step で用いる．また $\tilde{\mathbf{y}}$ に対する $\boldsymbol{\theta}$ の対数尤度関数は

$$\ell_c(\boldsymbol{\theta}) \propto \sum_{i=1,2} \sum_{j=1,2} \Big(y_{ij} + \tilde{r}_{ij} + \tilde{c}_{ij} \Big) \ln \theta_{ij}$$

である．したがって，$\mathbf{y}$ および $\boldsymbol{\theta}^{(t)}$ が与えられたもとでの $\ell_c(\boldsymbol{\theta})$ の条件付き期待値である Q 関数は

$$\begin{aligned} Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) &= \mathrm{E}[\ell_c(\boldsymbol{\theta})|\mathbf{y}, \boldsymbol{\theta}^{(t)}] \\ &\propto \sum_{i=1,2} \sum_{j=1,2} \Big(y_{ij} + r_i \theta^{(t)}_{i(j)} + c_j \theta^{(t)}_{(i)j} \Big) \ln \theta_{ij} \end{aligned}$$

で与えられる．ここで，$\theta^{(t)}_{i(j)} = \theta^{(t)}_{ij} / \sum_{j=1,2} \theta^{(t)}_{ij}$ であり，$\theta^{(t)}_{(i)j} = \theta^{(t)}_{ij} / \sum_{i=1,2} \theta^{(t)}_{ij}$ である．

これより，EM アルゴリズムは以下の手順で与えられる：

E-step: $\mathbf{y}$ および $\boldsymbol{\theta}^{(t)}$ が与えられたもとで，

$$\tilde{r}^{(t+1)}_{ij} = r_i \theta^{(t)}_{i(j)}, \qquad \tilde{c}^{(t+1)}_{ij} = c_j \theta^{(t)}_{(i)j}$$

により，$\tilde{\mathbf{y}}^{(t+1)}_1 = [\tilde{r}^{(t+1)}_{11}, \tilde{r}^{(t+1)}_{21}, \tilde{r}^{(t+1)}_{12}, \tilde{r}^{(t+1)}_{22}]$ および $\tilde{\mathbf{y}}^{(t+1)}_2 = [\tilde{c}^{(t+1)}_{11}, \tilde{c}^{(t+1)}_{21}, \tilde{c}^{(t+1)}_{12}, \tilde{c}^{(t+1)}_{22}]$ をそれぞれ求め，

$$\tilde{\mathbf{y}}^{(t+1)} = [\mathbf{y}_0, \tilde{\mathbf{y}}^{(t+1)}_1, \tilde{\mathbf{y}}^{(t+1)}_2]$$

を得る．

M-step: $\tilde{\mathbf{y}}^{(t+1)}$ が与えられたもとで，

$$\theta^{(t+1)}_{ij} = \frac{y_{ij} + \tilde{r}^{(t+1)}_{ij} + \tilde{c}^{(t+1)}_{ij}}{y_{++} + r_+ + c_+}$$

により $\theta^{(t+1)}$ を得る．

したがって，この多項分布モデルのもとでの式 (1) は以下で与えられる．

$$\begin{aligned} \boldsymbol{\theta}^{(t+1)} &= \begin{bmatrix} \theta^{(t+1)}_{11} \\ \theta^{(t+1)}_{12} \\ \theta^{(t+1)}_{21} \\ \theta^{(t+1)}_{22} \end{bmatrix} = \frac{1}{y_{++} + r_+ + c_+} \begin{bmatrix} y_{11} + r_1 \theta^{(t)}_{1(1)} + c_1 \theta^{(t)}_{(1)1} \\ y_{12} + r_1 \theta^{(t)}_{1(2)} + c_2 \theta^{(t)}_{(1)2} \\ y_{21} + r_2 \theta^{(t)}_{2(1)} + c_1 \theta^{(t)}_{(2)1} \\ y_{22} + r_2 \theta^{(t)}_{2(2)} + c_2 \theta^{(t)}_{(2)2} \end{bmatrix} \\ &= M(\boldsymbol{\theta}^{(t)}) \end{aligned}$$