数理解析研究所講究録
第1908巻 2014年 57-77

Lassoタイプの正則化法に基づくスパース推定法を用いた超高次元データ解析

廣瀬 慧

大阪大学 大学院基礎工学研究科 システム創成専攻 数理科学領域
〒560-8531 大阪府豊中市待兼山町 1-3

E-mail: hirose@sigmath.es.osaka-u.ac.jp.

概要

超高次元・大規模データに対するモデルのスパース推定を効率的に行うことのできるL_1型正則化法は，これまで生命科学，機械学習などの分野で用いられてきたが，最近は，量子トモグラフィ，ネットワーク推定への応用が試みられている．本稿では，L_1型正則化法の中でも広く使われている回帰モデリングにおいて，これまで用いられてきた様々な推定アルゴリズムと調整パラメータの選択法について解説する．また，多変量解析の一つである因子分析に対するL_1型正則化法を紹介する．

1 はじめに

最近，生命科学，量子統計学，システム工学など，諸科学の様々な分野で超高次元・大規模データが取得されるようになっている．このようなデータから重要な情報を高効率に抽出するためには，大規模なモデルが必然となり，従来の最小二乗法や最尤法などは機能しなくなった．この問題に対処する有効な手法のひとつがL_1型正則化法である．L_1型正則化法は，目的関数にパラメータの絶対値を含む正則化項を加えて推定する方法であり，その正則化項の特徴から，パラメータのいくつかを正確に0と推定することができる．そのため，パラメータの数が膨大で，そのほとんどが0である場合，L_1型正則化法が有効に機能することが多い．

L_1型正則化法は，様々な統計モデルに適用できるが，その中でも特によく用いられているモデルが線形回帰モデルである．線形回帰モデルにL_1型正則化法を適用すると，変数の数が膨大な場合においても効率的に変数選択ができる．とくに，Lasso (Least absolute shrinkage and selection operator, Tibshirani 1996) と呼ばれる正則化法は広く用いられており，効率的なアルゴリズム (Friedman et al. 2010) や推定量の漸近的性質 (Zhao and Yu 2007) なども研究されている．いま，p次元説明変数$(x_1,\ldots,x_p)^T$と目的変数yに関するN組のデータ$\{(y_i,x_{i1},\ldots,x_{ip});i=1,\ldots,N\}$が得られたとする．このとき，次の線形回帰モデルを考える．

$$y_i=\beta_0+\sum_{j=1}^{p}\beta_j x_{ij}+\varepsilon_i,\quad i=1,\ldots,N. \tag{1.1}$$

以下，説明変数ベクトル $\boldsymbol{x}_j = (x_{1j}, \ldots, x_{Nj})^T$ は平均 0，大きさが 1 に基準化され，目的変数は中心化されているとする．

$$\sum_{i=1}^{N} y_i = 0, \quad \sum_{i=1}^{N} x_{ij} = 0, \quad \sum_{i=1}^{N} x_{ij}^2 = 1, \qquad j = 1, \ldots, p.$$

こうすることにより，一般性を失うことなく，$\beta_0 = 0$ とすることができる (小西 2010)．

(1.1) 式はベクトルと行列を用いて次のように表すことができる．

$$\boldsymbol{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}.$$

ただし，$\boldsymbol{y} = (y_1, \ldots, y_N)^T$ は N 次元目的変数ベクトル，$X = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_p)$ は $N \times p$ 計画行列，$\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^T$ は p 次元係数ベクトル，$\boldsymbol{\varepsilon}$ は N 次元誤差ベクトルとする．Lasso 推定値は，次の正則化最小二乗法

$$\hat{\boldsymbol{\beta}}(t) = \arg\min_{\boldsymbol{\beta}} \frac{1}{N}(\boldsymbol{y} - X\boldsymbol{\beta})^T(\boldsymbol{y} - X\boldsymbol{\beta}), \text{ subject to } \sum_{j=1}^{p} |\beta_j| \leq t$$

によって求められる．ただし，t は調整パラメータとよばれ，罰則 ($\sum |\beta_j|$) の強さを調節する役割を果たす．

Lasso の最大の特徴は，係数ベクトル $\boldsymbol{\beta}$ のいくつかの成分が正確に 0 と推定される点にある．調整パラメータ t の値が小さければ，$\boldsymbol{\beta}$ の成分のうち 0 になる個数が多くなり，スパース（疎）な解が得られる．逆に，もし t の値が十分に大きければ，Lasso 推定値は最小二乗推定値に近い．

L_1 型正則化法は，正則化項に原点で微分不可能な絶対値の項が含まれるため，解析的に解を求めることが困難となる．それゆえ，このモデリングの過程において，

- 解を求める高速なアルゴリズム，
- 調整パラメータ t の選択

が重要となる．解を求める高速なアルゴリズムについては，調整パラメータ t を固定した場合の解を求めるというより，(ほとんど) 全ての調整パラメータ t に対する解の “パス” を求めるアルゴリズムが求められている．これまで，解のパスを求めるアルゴリズムとして，Least Angle Regression (LARS, Efron et al. 2004)，局所線形近似 (Zou and Li 2008)，Coordinate Descent アルゴリズム, (CDA, Friedman et al. 2010; Breheny and Huang 2011; Mazumder et al. 2011)，Generalized Path Seeking アルゴリズム (GPS, Friedman 2012) など，様々なアルゴリズムが提案されている．特に，CDA は，パラメータの次元が超高次元になった場合においても高速である．さらに，CDA は，回帰モデルのみならず，一

般化線形モデル，Coxの比例ハザードモデル，グラフィカルモデリングなど，様々なモデルに拡張されている (Friedman et al. 2008; Simon et al. 2011)．また，ソフトウェアも充実しており，Rのパッケージ`glmnet`や`grpreg`は，驚くほど高速に解のパスを推定することができる．

一方，調整パラメータの選択は，モデル評価・モデル選択の問題と捉えることができる (Wang et al. 2007; Zou et al. 2007; Wang et al. 2009; Mazumder et al. 2011; Hirose et al. 2012)．これまで広く用いられてきたモデル評価基準 AIC, BIC は，最大対数尤度にモデルの自由度に基づくペナルティを加えるというものであった．モデルの自由度は，最尤法や最小二乗法ではパラメータ数で与えられる．しかしながら，Lasso タイプの正則化法におけるモデルの自由度を計算するためには，一般化自由度 (Ye 1998; Efron 2004) を導入する必要がある．L_1 型正則化法では，この一般化自由度を解析的に求めることが難しい．そこで，数理的アプローチに計算アルゴリズムを融合した新しいモデル評価基準が必要とされる．本稿では，これまで用いられてきた様々な L_1 型正則化法の推定アルゴリズムとともに，一般化自由度に基づく調整パラメータの選択法について解説する．

これまで，L_1 型正則化法は，線形回帰モデルのみならず，一般化線形モデル，サポートベクターマシン，グラフィカルモデリング，主成分分析，因子分析など，様々なモデルへ応用されている (たとえば，Hastie et al. 2004; Yuan and Lin 2007; Friedman et al. 2010 等を参照されたい)．このような一般のモデルに対する L_1 型正則化法は，目的関数 $R(\boldsymbol{\beta})$ にパラメータの絶対値に基づく制約 $P(\boldsymbol{\beta})$ を加味した形で，以下のように定式化される．

$$\hat{\boldsymbol{\beta}}(t) = \arg\min_{\boldsymbol{\beta}} R(\boldsymbol{\beta}) \quad \text{s.t.} \quad P(\boldsymbol{\beta}) \leq t.$$

あるいは，ラグランジュ未定乗数法を用いて

$$\hat{\boldsymbol{\beta}}(t) = \arg\min_{\boldsymbol{\beta}} \{R(\boldsymbol{\beta}) + \lambda P(\boldsymbol{\beta})\} \tag{1.2}$$

と表すこともできる．この場合，目的関数 $R(\boldsymbol{\beta})$，正則化項 $P(\boldsymbol{\beta})$ の形に応じて，適切な解の推定アルゴリズムおよび調整パラメータの選択が必要とされる．本稿では，とくに，多変量解析の一つである因子分析に対する L_1 型正則化法 (Hirose and Yamamoto 2012) とその推定アルゴリズムを紹介する．

本論文の構成は以下の通りである．まず，第2節では，Lasso の代表的なアルゴリズムである LARS と CDA，GPS について説明する．第3節では，Lasso と Lasso を拡張した非凸な正則化項に基づく正則化法に対する一般化自由度の推定法について説明する．第4節では L_1 型正則化法を因子分析モデルに適用した，スパース因子分析を紹介する．

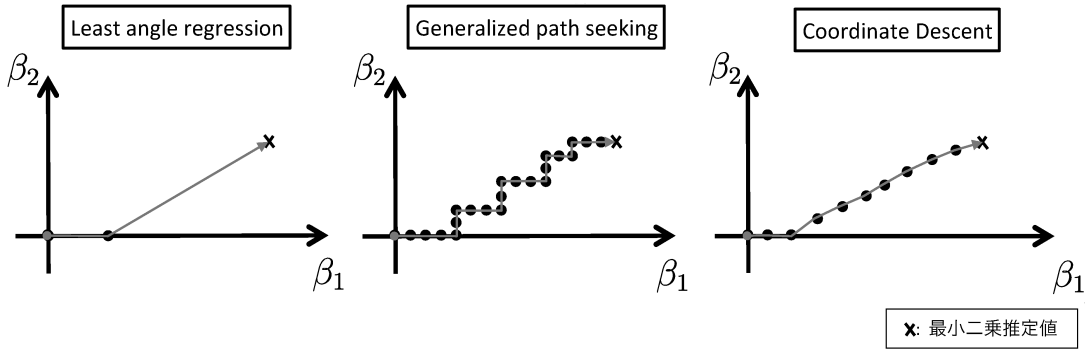

図 2.1: $p = 2$のときの3つのアルゴリズムのイメージ図. LARS (左), GPS(中央), CDA (右).

2 Lassoの推定アルゴリズム

Lassoの推定アルゴリズムは数多く存在するが, ここではLARS, GPS, CDAについて述べる. まず, $p = 2$のときの3つのアルゴリズムのイメージを図2.1に載せた. 図の黒丸 (●) が各調整パラメータに対する解である. 全ての調整パラメータに対する解のパスは赤の矢印で与えられる. LARSでは, 1回のステップで1つの変数を追加または削除して推定値のパスを求める. そのため, ほぼp回で全てのステップを全ての調整パラメータtに対する解を計算することができる. GPSでは, Forward Stagewise法 (たとえばHastie et al. 2008参照) と同様に, 調整パラメータを少しずつ動かして更新するが, 1回の更新に1つのパラメータしか更新しない. そのため, 真のパスを近似するパスを計算する. CDAでは, 調整パラメータの候補をいくつか与えて, 各調整パラメータに対して反復計算を行って更新していく方法である.

図2.1を見ると, 計算スピードはLARSが最もはやく, GPSやCDAは遅いように見える. しかしながら, LARSは1回の計算ステップでの計算量が高次元になると大きくなり, そのような場合はCDAやGPSの方が圧倒的に早くなる. たとえば, $p = 10000, N = 1000$とし, 説明変数を標準正規分布 (変数同士は無相関) から発生させ, 誤差分散9として誤差を発生させてデータを作り, 3つのアルゴリズムを適用すると, CDAが2秒, GPSが13秒, LARSが13分程度かかった*.

以下, 3つのアルゴリズムについて詳しく述べる.

*Intel Core i7, 2.3 GHz, メモリ 16GB, Mac OS X の PC を使った. LARS, GPS, CDA の解のパスは, それぞれ R パッケージ `lars`, `msgps`, `glmnet` を用いて計算した.

2.1 LARS

Efron *et al.* (2004) は，線形回帰モデルの変数選択のアルゴリズムとして LARS アルゴリズムを提唱した．LARS アルゴリズムで得られた解と Lasso 解は厳密には一致しないが，ほぼ一致しており，さらに LARS を少し修正することにより厳密な Lasso 解を計算できる．

Lasso 推定値は，次の正則化二乗和誤差関数

$$R_\lambda(\boldsymbol{\beta}) = \frac{1}{N}(\boldsymbol{y} - X\boldsymbol{\beta})^T(\boldsymbol{y} - X\boldsymbol{\beta}) + \lambda \sum_{j=1}^{p} |\beta_j| \tag{2.1}$$

の最小化によって求めることができるが，λ の値が大きければいくつかの係数の推定値は 0 となる．そこで，推定値が 0 でない添字集合 $\mathcal{A}$ を

$$\mathcal{A} = \{j \in \{1, \ldots, p\} : \ \hat{\beta}_j^{\text{Lasso}} \neq 0\}$$

と定義すると，Lasso 推定値 $\hat{\boldsymbol{\beta}}^{\text{Lasso}}$ は次の性質を満たすことが分かる．

$$\begin{aligned} &\frac{2}{N}\boldsymbol{x}_j^T(\boldsymbol{y} - X\hat{\boldsymbol{\beta}}^{\text{Lasso}}) = \lambda \cdot \text{sign}(\hat{\beta}_j^{\text{Lasso}}), \quad \forall j \in \mathcal{A}, \\ &\frac{2}{N}|\boldsymbol{x}_j^T(\boldsymbol{y} - X\hat{\boldsymbol{\beta}}^{\text{Lasso}})| < \lambda, \qquad\qquad \forall j \notin \mathcal{A}. \end{aligned} \tag{2.2}$$

この式は，任意の $j \in \mathcal{A}$ に対し，説明変数 $\boldsymbol{x}_j$ と残差 $(\boldsymbol{y} - X\hat{\boldsymbol{\beta}}^{\text{Lasso}})$ の相関 (内積) の絶対値が等しいことを意味する．LARS は，説明変数と残差の相関の絶対値が大きい（つまり角度が小さい）方向に解を進めていくアルゴリズムであり，そのため Lasso とほぼ同じ解が得られる．

次に，LARS アルゴリズムを図 2.2 を使って解説する．まず，最小二乗推定値を $\bar{\boldsymbol{y}}_2$ とおくと，任意の $\hat{\boldsymbol{\mu}} = X\hat{\boldsymbol{\beta}}$ と書ける推定値に対し，

$$X^T(\boldsymbol{y} - \hat{\boldsymbol{\mu}}) = X^T(\bar{\boldsymbol{y}}_2 - \hat{\boldsymbol{\mu}})$$

が成り立つことに注目しておく．

まず，$\boldsymbol{y}$ の予測値の初期値を $\hat{\boldsymbol{\mu}} = \boldsymbol{0}$ とおく．次に，p 個の説明変数のうち，残差との相関の絶対値が最大となる説明変数を選ぶ．すなわち，

$$\hat{j} = \arg\max_j |c_j|, \quad c_j = \boldsymbol{x}_j^T(\boldsymbol{y} - \hat{\boldsymbol{\mu}})$$

をみたす $\hat{j}$ を求め，これを j_1 とおく．図 2.2 の場合では，$j_1 = 1$ である．また，$\hat{C} = \max |c_j|$ とおく．ここで，アクティブ集合と呼ばれる添字集合 $\mathcal{A}$ を $\mathcal{A} = \{j_1\}$ と定義する．なお，アクティブ集合は 1 ステップごとに 1 つずつ追加されていく．

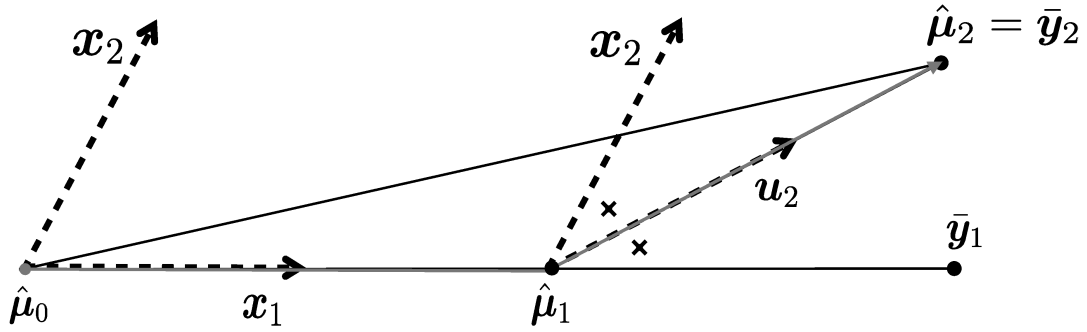

図 2.2: $p = 2$ としたときの LARS アルゴリズムのイメージ図.

まず，予測値を $\mathrm{sign}(c_{j_1})\boldsymbol{x}_{j_1}$ 方向に動かす．予測値は，ξ (≥ 0) に関する関数 $\hat{\boldsymbol{\mu}}(\xi) = \hat{\boldsymbol{\mu}} + \xi s_{j_1}\boldsymbol{x}_{j_1}$ でかける．ただし，$s_j = \mathrm{sign}(c_j)$ とする．ξ はどれだけ予測値を $\boldsymbol{x}_{j_1}$ 方向に動かすかを表す．このように動かした推定値 $\hat{\boldsymbol{\mu}}(\xi)$ に関する残差と $\boldsymbol{x}_j$ との相関を $c_j(\xi) = \boldsymbol{x}_j^T(\boldsymbol{y} - \hat{\boldsymbol{\mu}}(\xi))$ とおくと，

$$|c_{j_1}(\xi)| = s_{j_1}c_{j_1}(\xi) = s_{j_1}\boldsymbol{x}_{j_1}^T\boldsymbol{y} - \xi, \tag{2.3}$$

$$c_j(\xi) = \boldsymbol{x}_j^T\boldsymbol{y} - \xi s_{j_1}\boldsymbol{x}_j^T\boldsymbol{x}_{j_1}, \qquad j \in \mathcal{A}^C \tag{2.4}$$

となるので，$|c_{j_1}(\xi)|$ は ξ に関して単調減少関数となる．よって，ξ の値を連続的に大きくしていくと相関の絶対値が徐々に小さくなり，$\xi = \hat{\xi}$ のとき，$|c_{j_1}(\hat{\xi})| = |c_{j_2}(\hat{\xi})|$ となるような $j_2 \notin \mathcal{A}$ が出現する．このとき，(2.3)，(2.4) 式から，$\hat{\xi}$ は次で与えられる．

$$\hat{\xi} = \min_{j \in \mathcal{A}^C}{}^{+} \left\{ \frac{|c_{j_1}| - c_j}{1 - s_{j_1}\boldsymbol{x}_j^T\boldsymbol{x}_{j_1}}, \frac{|c_{j_1}| + c_j}{1 + s_{j_1}\boldsymbol{x}_j^T\boldsymbol{x}_{j_1}} \right\}.$$

ただし，“$\min^+$”は，正の値のみから選ぶことを意味する（両方とも 0 以下になることはあり得ない）．この $\hat{\xi}$ を用いて予測値ベクトルを $\hat{\boldsymbol{\mu}}^{\mathrm{new}} \leftarrow \hat{\boldsymbol{\mu}} + \hat{\xi}s_{j_1}\boldsymbol{x}_{j_1}$ と更新する．これが LARS アルゴリズムの最初の 1 ステップである．

次のステップではまず，アクティブ集合を $\mathcal{A} = \{j_1, j_2\}$ とし，更新された予測値 $\hat{\boldsymbol{\mu}} = \hat{\boldsymbol{\mu}}^{\mathrm{new}}$ を使って $c_j, \hat{C}, s_j$ を更新する．次に，先ほどと同様に予測値を動かす関数 $\hat{\boldsymbol{\mu}}(\xi) = \hat{\boldsymbol{\mu}} + \xi\boldsymbol{u}_{\mathcal{A}}$ を作る．ただし $\boldsymbol{u}_{\mathcal{A}}$ は，図 2.2 のような角度が等しくなるような長さ 1 のベクトルとする．このとき $\boldsymbol{u}_{\mathcal{A}}$ は，$s_{j_1}\boldsymbol{x}_{j_1}^T\boldsymbol{u}_{\mathcal{A}} = s_{j_2}\boldsymbol{x}_{j_2}^T\boldsymbol{u}_{\mathcal{A}}$, $|\boldsymbol{u}_{\mathcal{A}}| = 1$ をみたすため，次で与えられる．

$$\boldsymbol{u}_{\mathcal{A}} = A_{\mathcal{A}}\boldsymbol{X}_{\mathcal{A}}(\boldsymbol{X}_{\mathcal{A}}^T\boldsymbol{X}_{\mathcal{A}})^{-1}\mathbf{1}_{\mathcal{A}}, \quad A_{\mathcal{A}} = (\mathbf{1}_{\mathcal{A}}^T(\boldsymbol{X}_{\mathcal{A}}^T\boldsymbol{X}_{\mathcal{A}})^{-1}\mathbf{1}_{\mathcal{A}})^{-1/2}. \tag{2.5}$$

ただし，$\boldsymbol{X}_{\mathcal{A}} = (s_{j_1}\boldsymbol{x}_{j_1}, s_{j_2}\boldsymbol{x}_{j_2})$，$\mathbf{1}_{\mathcal{A}}$ は各成分が 1 の $|\mathcal{A}|$ 次元ベクトルとする．ここで，$c_j(\xi) = \boldsymbol{x}_j^T(\boldsymbol{y} - \hat{\boldsymbol{\mu}}(\xi))$ と定義すると，

$$|c_j(\xi)| = \hat{C} - \xi A_{\mathcal{A}}, \quad \forall j \in \mathcal{A}$$

となるので，ξ に関して単調減少関数となる．また，この式から，任意の $j \in \mathcal{A}$ に対し，$|c_j(\xi)|$ は全て同じ値をとるので，すべてのアクティブ集合に対する相関の絶対値は常に同じ値をとることが分かる．次に，ξ を少しずつ大きくする．$p = 2$ のときは，$c_j(\xi) = 0$ となるまで ξ を大きくする．このとき最小二乗推定値にたどり着いているのでアルゴリズムはストップする．$p > 2$ の場合は，ξ を少しずつ大きくしたとき，$\xi = \hat{\xi}'$ のとき，$|c_{j_1}(\hat{\xi}')| = |c_{j_2}(\hat{\xi}')| = |c_{j_3}(\hat{\xi}')|$ となるような $j_3 \notin \mathcal{A}$ が出現する．このとき $\hat{\xi}'$ は次で与えられる．

$$\hat{\xi}' = \min_{j \in \mathcal{A}^C}{}^{+} \left\{ \frac{\hat{C} - c_j}{A_{\mathcal{A}} - \boldsymbol{x}_j^T \boldsymbol{u}_{\mathcal{A}}}, \frac{\hat{C} + c_j}{A_{\mathcal{A}} + \boldsymbol{x}_j^T \boldsymbol{u}_{\mathcal{A}}} \right\}.$$

以下，同様の操作を行う．この操作を p 回繰り返すと，最小二乗推定値にたどり着く．

LARS で得られた推定値と Lasso 推定値はかなり似ているが，必ずしも完全に一致するとは限らない．なぜなら，Lasso 推定においては，0 でない係数 β_j に対し，λ を連続的に小さくする過程で $\hat{\beta}_j^{\text{Lasso}}$ が 0 に達したとき，添字 j に対して (2.2) 式が成り立たなくなるので予測値ベクトルの方向が変わることがあるが，LARS においてはこのような現象が生じたとしても予測値ベクトルが同じ方向のまま進むためである．しかし，0 でなかった係数が λ を連続的に小さくする過程において再び 0 に達したとき，その対応する変数を取り除いた上で予測値の進むべき方向を再計算するように LARS アルゴリズムを修正することにより，Lasso 推定値を求めることができる．

LARS は，1 回のステップで 1 つの変数を増減することのできる効率的なアルゴリズムではあるものの，(2.5) 式にあるように，アクティブセットに基づくグラム行列の逆行列を計算しなければならない．そのため，データが高次元になると LARS は劇的に遅くなる．

2.2 GPS

いま，(1.2) 式の解を求める問題を考える．GPS アルゴリズムは，下記を満たす正則化項に適用することのできるアルゴリズムである (Friedman 2012)．

$$\left\{ \frac{\partial P(\boldsymbol{\beta})}{\partial |\beta_j|} > 0 \right\}_{j=1}^{p}. \tag{2.6}$$

たとえば，Lasso は $P(\boldsymbol{\beta}) = \sum_{j=1}^{p} |\beta_j|$ であるが，微分すると $\partial P(\boldsymbol{\beta})/\partial |\beta_j| = 1 > 0$ となるため条件をみたす．同様に Elastic Net (Zou and Hastie 2005), Adaptive Lasso (Zou 2006),

Smoothly Clipped Absolute Deviation (SCAD; Fan and Li 2001), Generalized Elastic Net (Friedman 2012) など幅広いクラスの凸・非凸な正則化項がこの条件を満たす.

いま, $\hat{\boldsymbol{\beta}}(t)$ を調整パラメータ t に対する解とする. まず, 初期値を $\hat{\boldsymbol{\beta}}(0) = \mathbf{0}$ と定義し, t での推定値 $\hat{\boldsymbol{\beta}}(t)$ から $t + \Delta t$ での推定値 $\hat{\boldsymbol{\beta}}(t + \Delta t)$ を

$$\hat{\boldsymbol{\beta}}(t + \Delta t) = \hat{\boldsymbol{\beta}}(t) + \boldsymbol{d}(t),$$

で推定する. ただし, $\boldsymbol{d}(t) = (d_1(t), \cdots, d_p(t))^T$ は方向ベクトル, $\Delta t > 0$ は微小な正数とする. このようにGPSアルゴリズムは逐次更新アルゴリズムであるが, 以下のように, 1回のステップで, 1つの変数しか更新しない.

$$\hat{\beta}_k(t + \Delta t) = \hat{\beta}_k(t) + d_k(t),$$
$$\{\hat{\beta}_j(t + \Delta t) = \hat{\beta}_j(t)\}_{j \neq k}.$$

GPSアルゴリズムは, 多くのステップを踏むことにより, 解の経路を近似的に求めることができる.

GPSアルゴリズムを導出するため, まず, 次の量を定義する.

$$g_j(t) = -\left[\frac{\partial R(\boldsymbol{\beta})}{\partial \beta_j}\right]_{\boldsymbol{\beta}=\hat{\boldsymbol{\beta}}(t)},$$
$$p_j(t) = \left[\frac{\partial P(\boldsymbol{\beta})}{\partial |\beta_j|}\right]_{\boldsymbol{\beta}=\hat{\boldsymbol{\beta}}(t)}.$$

このとき, 次の2つの系が成り立つ.

系 2.1. 以下の問題

$$\Delta\hat{\boldsymbol{\beta}}(t) = \arg\min_{\Delta\boldsymbol{\beta}}[R(\hat{\boldsymbol{\beta}}(t) + \Delta\boldsymbol{\beta}) - R(\hat{\boldsymbol{\beta}}(t))] \quad \text{s.t.} \quad P(\hat{\boldsymbol{\beta}}(t) + \Delta\boldsymbol{\beta}) - P(\hat{\boldsymbol{\beta}}(t)) \leq \Delta t \quad (2.7)$$

の解は $\Delta\hat{\boldsymbol{\beta}}(t) = \hat{\boldsymbol{\beta}}(t + \Delta t) - \hat{\boldsymbol{\beta}}(t)$ で与えられる.

系 2.2. 解 $\hat{\boldsymbol{\beta}}(t)$ が t に関して連続かつ単調増加とする, すなわち,

$$\{|\hat{\beta}_j(t + \Delta t)| \geq |\hat{\beta}_j(t)|\}_1^p$$

とする. このとき, *(2.6)* 式を仮定すると, *(2.7)* 式は近似的に次で与えられる.

$$\Delta\hat{\boldsymbol{\beta}}(t) = \arg\max_{\{\Delta\beta_j\}_1^p} \sum_{j=1}^{p} g_j(t) \cdot \Delta\beta_j \quad \text{s.t.} \quad \sum_{j=1}^{p} p_j(t) \cdot |\Delta\beta_j| \leq \Delta t. \quad (2.8)$$

系 2.1，系 2.2 により，(2.8) 式の解は近似的に

$$k = \arg\max_{1\le j\le p} |g_j(t)|/p_j(t),$$
$$\Delta\hat{\beta}_k(t) = g_k(t)/p_k(t)\cdot\Delta t, \tag{2.9}$$
$$\Delta\hat{\beta}_j(t) = 0 \quad (j\neq k) \tag{2.10}$$

で与えられる．それゆえ，(2.9) 式より係数は，

$$\hat{\beta}_k(t+\Delta t) = \hat{\beta}_k(t) + \Delta t\cdot\lambda_k(t) \tag{2.11}$$

により更新される．ただし，$\lambda_j(t) = g_j(t)/p_j(t)$ とする．もし $p_j(t) = 1$ (Lasso) とすると，(2.11) 式は

$$\hat{\beta}_k(t+\Delta t) = \hat{\beta}_k(t) + \Delta t\cdot g_k(t) \tag{2.12}$$

となる．ここで，たとえ Lasso を適用しなかったとしても，(2.12) 式の更新式を使うことができる．その理由は，条件式 (2.6) より $\mathrm{sign}(g_k(t)) = \mathrm{sign}(\lambda_k(t))$ が成り立ち，解の進むべき方向（すなわち符号）が (2.11) 式と (2.12) が一致するため，$\Delta t\to 0$ のとき，これらの 2 つの式で与えられる更新式による解経路は同一のものとなるためである．それゆえ，今後は (2.12) 式を適用する．なお，解の更新に (2.11) 式でなく，(2.12) 式を用いる理由は，3.2 節で後述のように，一般化自由度を導出することが容易であるためである．

(2.10) 式と (2.12) 式より，回帰モデルにおける $t+\Delta t$ での予測値は，ロス関数が $R(\boldsymbol{\beta}) = (\boldsymbol{y}-X\boldsymbol{\beta})^T(\boldsymbol{y}-X\boldsymbol{\beta})/N$ で与えられるため，$g_j(t) = 2\boldsymbol{x}_j^T(\boldsymbol{y}-\hat{\boldsymbol{\mu}}(t))/N$ となり，

$$\hat{\boldsymbol{\mu}}(t+\Delta t) = \hat{\boldsymbol{\mu}}(t) + \frac{2}{N}\Delta t\ \boldsymbol{x}_k\boldsymbol{x}_k^T\cdot(\boldsymbol{y}-\hat{\boldsymbol{\mu}}(t)) \tag{2.13}$$

で与えられる．

例 2.1. X が直交，すなわち $X^TX = I$ のときを考える．このとき，g_j は次で更新される．

$$g_j(t+\Delta t) = \begin{cases} (1-2\Delta t/N)\, g_j(t) & j = k, \\ g_j(t) & j\neq k. \end{cases}$$

ゆえに，

$$g_j(t) = (1-2\Delta t/N)^{t_j}\cdot 2\boldsymbol{x}_j^T\boldsymbol{y}/N$$

で与えられる．ただし，t_j は調整パラメータが 0 から t までのステップで，第 j 変数が選択された回数とする．ここで，$|g_j(t)|$ は単調減少関数であり，$t_j\to\infty$ のときに $g_j(t)\to 0$ となる．これは，無限回のステップを踏むと *GPS* アルゴリズムが最小二乗推定値に収束することを意味する．

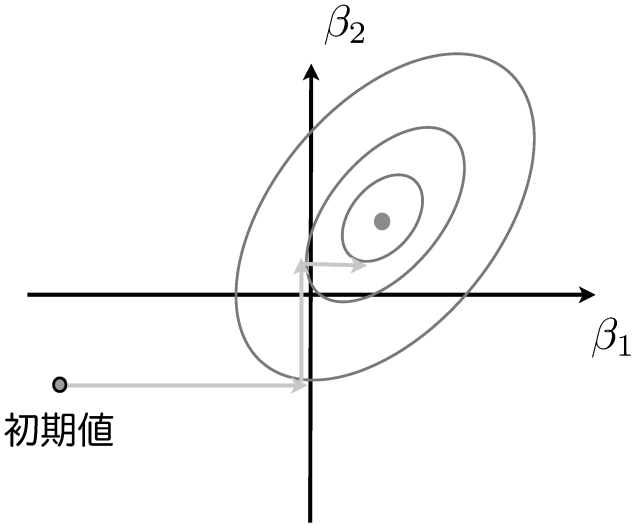

図 2.3: パラメータが 2 次元のときの CDA のイメージ図．目的関数の最小値が赤丸で表されており，青の楕円は目的関数の値が同じ値を取る等高線である．CDA では，2 つのパラメータのうち 1 つを固定して最小化するということを収束するまで反復する．

2.3 CDA

CDA は，Lasso 解を計算する極めて高速なアルゴリズムとして知られている．いま，(2.1) 式の最小化問題を考える．CDA では，まず，係数ベクトルの j 番目の要素 β_j を除いた $(p-1)$ 次元ベクトル $\tilde{\boldsymbol{\beta}}^{(-j)} = (\tilde{\beta}_1, \tilde{\beta}_2, \ldots, \tilde{\beta}_{j-1}, \tilde{\beta}_{j+1}, \ldots, \tilde{\beta}_p)^T$ が与えられたとき，(2.1) を β_j に関して最小化するプロセスをすべての j に対して反復して行う（図 2.3 参照）．

$\beta_j \neq 0$ のとき，(2.1) 式の正則化二乗和誤差関数 $R_\lambda(\boldsymbol{\beta})$ を β_j に関して微分すると，

$$\frac{\partial R_\lambda(\boldsymbol{\beta})}{\partial \beta_j} = -\frac{2}{N}\sum_{i=1}^{N} x_{ij}(y_i - x_{ij}\beta_j - \langle \boldsymbol{x}_i^{(-j)}, \tilde{\boldsymbol{\beta}}^{(-j)} \rangle) + \lambda \mathrm{sign}(\beta_j) \tag{2.14}$$

となる．ただし，$\boldsymbol{x}_i^{(-j)}$ は，$\boldsymbol{x}_i$ の j 番目の要素を取り除いた $(p-1)$ 次元ベクトル，$\langle \cdot, \cdot \rangle$ は 2 つのベクトルの内積とする．(2.14) 式より，正則化二乗和誤差関数 $R_\lambda(\boldsymbol{\beta})$ を β_j に関して最小化する解は次で与えられる．

$$\tilde{\beta}_j \leftarrow S\left(\sum_{i=1}^{N} x_{ij}(y_i - \langle \boldsymbol{x}_i^{(-j)}, \tilde{\boldsymbol{\beta}}^{(-j)} \rangle), \frac{N\lambda}{2}\right).$$

ただし，$S(z, \gamma) = \mathrm{sign}(z)(|z| - \gamma)_+$ とする．

CDA では，β_j を更新するために内積計算 $\langle \boldsymbol{x}_i^{(-j)}, \tilde{\boldsymbol{\beta}}^{(-j)} \rangle$ をすべての $i = 1, \ldots, N$ に対して行う必要があり，さらに，その更新を各 j に対して反復する必要があるため，次元

が高いときに非効率のように見える．しかしながら，実は，毎回のステップで内積計算 $\langle \boldsymbol{x}_i^{(-j)}, \tilde{\boldsymbol{\beta}}^{(-j)} \rangle$ を行う必要がないことが知られている．実際，β_j の更新式で出現する内積計算は以下のように与えられる．

$$\begin{aligned}\sum_{i=1}^{N} x_{ij}(y_i - \langle \boldsymbol{x}_i^{(-j)}, \tilde{\boldsymbol{\beta}}^{(-j)} \rangle) &= \sum_{i=1}^{N} x_{ij}(y_i - \langle \boldsymbol{x}_i, \tilde{\boldsymbol{\beta}} \rangle + x_{ij}\tilde{\beta}_j) \\ &= \langle \boldsymbol{x}_j, \boldsymbol{y} \rangle - \sum_{k:\tilde{\beta}_k \neq 0} \langle \boldsymbol{x}_j, \boldsymbol{x}_k \rangle \tilde{\beta}_k + \tilde{\beta}_j.\end{aligned}$$

よって，内積計算 $\langle \boldsymbol{x}_j, \boldsymbol{y} \rangle$ と $\langle \boldsymbol{x}_j, \boldsymbol{x}_k \rangle$ については，最初に計算してその結果をメモリに保存しておけばよい．

3　一般化自由度に基づく調整パラメータの選択法

一般化自由度 (Ye 1998) は，モデルの複雑さをはかる尺度として用いられる．まず，目的変数ベクトル $\boldsymbol{y}$ の真の平均ベクトルと分散共分散行列が次で与えられると仮定する．

$$E[\boldsymbol{y}] = \boldsymbol{\mu}, \quad V(\boldsymbol{y}) = E[(\boldsymbol{y} - \boldsymbol{\mu})(\boldsymbol{y} - \boldsymbol{\mu})^T] = \tau^2 I.$$

ここで，$\boldsymbol{\mu}, \tau^2$ は，それぞれ目的変数ベクトル $\boldsymbol{y}$ の真の平均と分散である．いま，真の平均構造 $\boldsymbol{\mu}$ を，モデル $\hat{\boldsymbol{\mu}} = \hat{\boldsymbol{\mu}}(\boldsymbol{y})$ で近似することを考える．このとき，一般化自由度 (Efron 1986; Ye 1998; Efron 2004) は

$$df = \sum_{i=1}^{N} \frac{\mathrm{cov}(\hat{\mu}_i, y_i)}{\tau^2}$$

と定義される．

Mallow の C_p 基準におけるモデルの複雑さは，一般化自由度を用いて表すことができる (Efron, 2004)．実際，次の予測二乗誤差

$$\mathrm{Err} = E_{\boldsymbol{y}} E_{\boldsymbol{y}^{\mathrm{new}}}[(\hat{\boldsymbol{\mu}} - \boldsymbol{y}^{\mathrm{new}})^T(\hat{\boldsymbol{\mu}} - \boldsymbol{y}^{\mathrm{new}})] \tag{3.1}$$

を考え，この値が小さい程良いモデルと見なすとする．ただし，期待値 "$E_{\boldsymbol{y}^{\mathrm{new}}}$" は現在得られたデータ $\boldsymbol{y}$ とは独立に新たに $\boldsymbol{y}^{\mathrm{new}} \sim (\boldsymbol{\mu}, \tau^2 I)$ から得られたとする．このとき，次が成り立つ．

系 3.1. *(3.1)* 式で与えられる予測二乗誤差は次で与えられる．

$$\mathrm{Err} = E_{\boldsymbol{y}}\left[\|\boldsymbol{y} - \hat{\boldsymbol{\mu}}\|^2 + 2\tau^2 df\right]. \tag{3.2}$$

Proof. まず，$\|\boldsymbol{y}^{\text{new}} - \hat{\boldsymbol{\mu}}\|^2$ を次の3つの項に分ける．

$$\|\boldsymbol{y}^{\text{new}} - \hat{\boldsymbol{\mu}}\|^2 = \|\boldsymbol{y}^{\text{new}} - \boldsymbol{\mu}\|^2 + \|\boldsymbol{\mu} - \hat{\boldsymbol{\mu}}\|^2 + 2(\boldsymbol{y}^{\text{new}} - \boldsymbol{\mu})^T(\boldsymbol{\mu} - \hat{\boldsymbol{\mu}}). \tag{3.3}$$

第2項目の $\|\boldsymbol{\mu} - \hat{\boldsymbol{\mu}}\|^2$ は次のように表現される．

$$\begin{aligned}\|\boldsymbol{\mu} - \hat{\boldsymbol{\mu}}\|^2 &= \|\boldsymbol{y} - \hat{\boldsymbol{\mu}}\|^2 + \|\boldsymbol{y} - \boldsymbol{\mu}\|^2 - 2(\boldsymbol{y} - \boldsymbol{\mu})^T(\boldsymbol{y} - \hat{\boldsymbol{\mu}}) \\ &= \|\boldsymbol{y} - \hat{\boldsymbol{\mu}}\|^2 - \|\boldsymbol{y} - \boldsymbol{\mu}\|^2 + 2(\boldsymbol{y} - \boldsymbol{\mu})^T(\hat{\boldsymbol{\mu}} - \boldsymbol{\mu}) \\ &= \|\boldsymbol{y} - \hat{\boldsymbol{\mu}}\|^2 - \|\boldsymbol{y} - \boldsymbol{\mu}\|^2 + 2(\boldsymbol{y} - \boldsymbol{\mu})^T(\hat{\boldsymbol{\mu}} - E_{\boldsymbol{y}}[\hat{\boldsymbol{\mu}}]) + 2(\boldsymbol{y} - \boldsymbol{\mu})^T(E_{\boldsymbol{y}}[\hat{\boldsymbol{\mu}}] - \boldsymbol{\mu}).\end{aligned} \tag{3.4}$$

ゆえに，(3.4) 式を (3.3) 式に代入することにより，$\|\boldsymbol{y}^{\text{new}} - \hat{\boldsymbol{\mu}}\|^2$ は以下のように分解される．

$$\begin{aligned}\|\boldsymbol{y}^{\text{new}} - \hat{\boldsymbol{\mu}}\|^2 = \|\boldsymbol{y}^{\text{new}} - \boldsymbol{\mu}\|^2 + \|\boldsymbol{y} - \hat{\boldsymbol{\mu}}\|^2 - \|\boldsymbol{y} - \boldsymbol{\mu}\|^2 + 2(\boldsymbol{y} - \boldsymbol{\mu})^T(\hat{\boldsymbol{\mu}} - E_{\boldsymbol{y}}[\hat{\boldsymbol{\mu}}]) \\ +2(\boldsymbol{y} - \boldsymbol{\mu})^T(E_{\boldsymbol{y}}[\hat{\boldsymbol{\mu}}] - \boldsymbol{\mu}) + 2(\boldsymbol{y}^{\text{new}} - \boldsymbol{\mu})^T(\boldsymbol{\mu} - \hat{\boldsymbol{\mu}}).\end{aligned}$$

期待値 $E_{\boldsymbol{y}^{\text{new}}}$ をとることにより

$$\begin{aligned}E_{\boldsymbol{y}^{\text{new}}}[\|\boldsymbol{y}^{\text{new}} - \hat{\boldsymbol{\mu}}\|^2] = E_{\boldsymbol{y}^{\text{new}}}[\|\boldsymbol{y}^{\text{new}} - \boldsymbol{\mu}\|^2] + \|\boldsymbol{y} - \hat{\boldsymbol{\mu}}\|^2 - \|\boldsymbol{y} - \boldsymbol{\mu}\|^2 \\ +2(\boldsymbol{y} - \boldsymbol{\mu})^T(\hat{\boldsymbol{\mu}} - E_{\boldsymbol{y}}[\hat{\boldsymbol{\mu}}]) + 2(\boldsymbol{y} - \boldsymbol{\mu})^T(E_{\boldsymbol{y}}[\hat{\boldsymbol{\mu}}] - \boldsymbol{\mu})\end{aligned}$$

が得られる．(3.2) 式は上の式に期待値 $E_{\boldsymbol{y}}$ を取ることにより導かれる． □

(3.2) 式の右辺から，次の C_p 基準を導くことができる (たとえば Efron 2004)．

$$C_p = \|\boldsymbol{y} - \hat{\boldsymbol{\mu}}\|^2 + 2\tau^2 df.$$

これは，(3.1) 式の予測二乗誤差の不偏推定量となる．C_p 基準が最小になるモデルを最適なモデルとして選択する．

なお，真の分散 τ^2 は最も複雑なモデル（フルモデル）の誤差分散の不偏推定量によって推定される．$N < p$ の場合，フルモデルの誤差分散は 0 となるため，C_p 基準を計算することができない．そのような場合，たとえば，まずクロスバリデーションで最適なモデルを選択し，選択されたモデルに対する誤差分散を求める．

最小二乗推定値など，推定値が解析的に計算できる場合，一般化自由度は比較的容易に計算できる．たとえば，最小二乗推定値 $\hat{\boldsymbol{\beta}} = (X^TX)^{-1}X^T\boldsymbol{y}$ に対する一般化自由度は，

$$df = \frac{\text{tr}\{\text{cov}(\hat{\boldsymbol{\mu}}, \boldsymbol{y})\}}{\tau^2} = \frac{\text{tr}\{\text{cov}(X(X^TX)^{-1}X^T\boldsymbol{y}, \boldsymbol{y})\}}{\tau^2} = p$$

となり，パラメータ数で与えられる．また，$\hat{\boldsymbol{\mu}} = H\boldsymbol{y}$（$H$ は $\boldsymbol{y}$ に依存しない）と仮定すると一般化自由度は $\text{tr}(H)$ で与えられ，Hastie and Tibshirani (1990) で述べられている有効自由度と一致する．

3.1 Lassoの一般化自由度

Lassoの自由度の計算は容易ではない．なぜなら，正則化項に原点で微分不可能な絶対値を含み，解析的な解を求めることが困難であるためである．しかしながら，$\boldsymbol{y}$に正規性を仮定すると，Lassoの自由度の不偏推定量は，ゼロでないパラメータの数で与えられる (Zou et al. 2007)．すなわち，自由度は

$$df(\hat{\boldsymbol{\beta}}) = E[|\mathcal{A}_\lambda|]$$

で与えられる．ただし，$\mathcal{A}_\lambda$は$\hat{\boldsymbol{\beta}}$のうち，ゼロでない成分に対応する添字の集合とする．

一般に，Lassoの自由度の計算は容易ではないが，λが変化点$\lambda_{\text{transition}}$でない場合は比較的容易に計算できる (ここで，$\lambda$を連続的に小さくしていった時，$\hat{\boldsymbol{\beta}}$のいずれかの成分の符号が変化する$\lambda = \lambda_{\text{transition}}$を変化点という)．いま，$\hat{\boldsymbol{\beta}}$の成分のうち，ゼロでない部分のみを取り出したベクトル$\hat{\boldsymbol{\beta}}_{\mathcal{A}_\lambda}$は，次で与えられる．

$$\hat{\boldsymbol{\beta}}_{\mathcal{A}_\lambda} = (X_{\mathcal{A}_\lambda}^T X_{\mathcal{A}_\lambda})^{-1}\left(X_{\mathcal{A}_\lambda}^T \boldsymbol{y} - \frac{N\lambda}{2}\text{sign}(\hat{\boldsymbol{\beta}}_{\mathcal{A}_\lambda})\right)$$

ゆえに，推定量は

$$\hat{\boldsymbol{\mu}} = X_{\mathcal{A}_\lambda}(X_{\mathcal{A}_\lambda}^T X_{\mathcal{A}_\lambda})^{-1} X_{\mathcal{A}_\lambda}^T \boldsymbol{y} - \frac{N\lambda}{2} X_{\mathcal{A}_\lambda}(X_{\mathcal{A}_\lambda}^T X_{\mathcal{A}_\lambda})^{-1}\text{sign}(\hat{\boldsymbol{\beta}}_{\mathcal{A}_\lambda})$$

で与えられる．$\boldsymbol{y}$に正規性を仮定しているため，Stein's unbiased risk estimate (SURE; Stein 1981) を用いることにより，自由度が

$$df = E\left[\sum_{i=1}^{N} \frac{\partial \hat{\mu}_i}{\partial y_i}\right]$$

で与えられる．いま，$\lambda \neq \lambda_{\text{transition}}$とすると，$X_{\mathcal{A}_\lambda}(X_{\mathcal{A}_\lambda}^T X_{\mathcal{A}_\lambda})^{-1} X_{\mathcal{A}_\lambda}$と$\text{sign}(\hat{\boldsymbol{\beta}}_{\mathcal{A}_\lambda})$の値は，微小な$\boldsymbol{y}$の変化に対して定数となる (Zou et al. 2007, Lemma 5)．それゆえ，

$$\sum_{i=1}^{N} \frac{\partial \hat{\mu}_i}{\partial y_i} = \text{tr}\left[\frac{\partial \hat{\boldsymbol{\mu}}}{\partial \boldsymbol{y}^T}\right] = \text{tr}(X_{\mathcal{A}_\lambda}(X_{\mathcal{A}_\lambda}^T X_{\mathcal{A}_\lambda})^{-1} X_{\mathcal{A}_\lambda}^T) = |\mathcal{A}_\lambda|$$

が得られる．

Lassoの自由度はゼロでないパラメータ数であるが縮小推定している．一方，Subset selectionは縮小推定していないにも関わらずパラメータ数で与えられる．なぜ同じ自由度なのだろうか．それは，Lassoが，どのパラメータがゼロであるかを「探索」しており，そのプロセスも自由度の大きさに含まれるからであると考えられる．それゆえ，Subset selectionでは，縮小推定をしていないが，どのパラメータがゼロであるかを探索するプロセスを含めれば，もう少し自由度が大きくなると考えられる (Mazumder et al. 2011)．

3.2 GPSアルゴリズムに基づく一般化自由度の計算

SUREに基づく一般化自由度は，Generalized Elastic Net (Friedman 2012) をはじめとする非凸な正則化項に対して導出することは困難である．Hirose et al. (2012) は，Lassoより広いクラスの正則化項に対する自由度の導出を行うために，2.2節で述べたGPSアルゴリズムを拡張し，一般化自由度を効率的に計算するアルゴリズムを提案した．

いま，GPSアルゴリズムの更新式 (2.13) 式から，一般化自由度は次の更新式を用いて求めることができる．

$$M(t+\Delta t) = M(t) + \frac{2}{N}\Delta t\ \boldsymbol{x}_k \boldsymbol{x}_k^T \{I - M(t)\}. \tag{3.5}$$

ただし，$M(t)$ は推定されたモデルと現在得られたデータとの共分散行列

$$M(t) = \frac{\mathrm{cov}(\hat{\boldsymbol{\mu}}(t), \boldsymbol{y})}{\tau^2}$$

とする．ゆえに，(3.5) 式を用いて，一般化自由度を逐次的に計算することができる．

ここで，(3.5) 式の更新式について考察する．$k(t)$ を，調整パラメータが t のときに選ばれた変数の番号とすると，(3.5) 式は，

$$I - M(t+\Delta t) = (I - \alpha \boldsymbol{x}_{k(t)} \boldsymbol{x}_{k(t)}^T)(I - M(t)) \tag{3.6}$$

で与えられる．ここで，$\alpha = 2\Delta t/N$ である．ゆえに，(3.6) を解くことにより，調整パラメータが t のときの一般化自由度を次のように求めることができる．

$$M(t) = I - (I - \alpha \boldsymbol{x}_{k(t-1)} \boldsymbol{x}_{k(t-1)}^T)(I - \alpha \boldsymbol{x}_{k(t-2)} \boldsymbol{x}_{k(t-2)}^T) \cdots (I - \alpha \boldsymbol{x}_{k(1)} \boldsymbol{x}_{k(1)}^T). \tag{3.7}$$

なお，上記のアルゴリズムに基づく自由度の計算を行い，様々なモデル評価基準によって選択された解を出力するRパッケージmsgpsは，CRAN(The Comprehensive R Archive Network) から入手可能である (`http://cran.r-project.org/web/packages/msgps/index.html`)．

例 3.1. X が直交のときを考える．このとき，*(3.7)* 式で与えられる共分散行列は

$$\begin{aligned} M(t) &= I - (I - \alpha \boldsymbol{x}_1 \boldsymbol{x}_1^T)^{t_1} (I - \alpha \boldsymbol{x}_2 \boldsymbol{x}_2^T)^{t_2} \cdots (I - \alpha \boldsymbol{x}_p \boldsymbol{x}_p^T)^{t_p} \\ &= \sum_{j=1}^{p} \{1 - (1-\alpha)^{t_j}\} \boldsymbol{x}_j \boldsymbol{x}_j^T \end{aligned}$$

で与えられる．ただし，t_j は例 *2.1* で定義したものである．このとき，一般化自由度は

$$\mathrm{tr}\{M(t)\} = \sum_{j=1}^{p} \{1 - (1-\alpha)^{t_j}\}$$

で与えられる．$1-(1-\alpha)^{t_j} < 1-(1-\alpha)^{t_j+1}$ より，t_j が大きくなるに従い，自由度も大きくなる．一般化自由度は0以上で，ゼロでないパラメータ数より大きくなることもない．もしすべての j に対して $t_j \to \infty$ とすると，一般化自由度はパラメータ数となる．これは，最小二乗推定値に対する一般化自由度と一致することを意味する．

4 因子分析モデルにおけるスパース推定

L_1 型正則化法は，グラフィカルモデリング，サポートベクターマシン，主成分分析など，様々なモデルに応用されているが，本稿では因子分析モデルへの応用について紹介する．因子分析モデルとは，多変量データの相関構造から，背後にある共通因子を見出すモデルで，心理学，社会科学，生命科学をはじめとする諸科学の様々な分野で用いられている．観測データと共通因子を結ぶ因子負荷量は，通常，次の2段階推定によって求められる: (i) 因子分析モデルを最尤法によって推定する，(ii) バリマックス回転 (Kaiser 1958) などの因子回転を用いて，スパース推定する．しかしながら，上記の2段階推定にはいくつか問題点がある．まず，因子分析モデルは，観測変数の次元が高くなるにつれてパラメータ数が膨大となり，最尤法ではしばしば推定が不安定となる (Akaike 1987)．特に，変数の数がサンプルサイズより大きい場合，最尤推定値を求めることができない．また，たとえ最尤推定値を計算することができたとしても，因子回転では十分にスパースな推定を行うことができないことが多い．これらの問題に対処するために，L_1 型正則化法によって因子分析モデルを推定する．

4.1 古典的な因子負荷行列の推定：最尤法と因子回転

まず古典的な因子分析モデルの2段階推定法について述べる．

4.1.1 最尤法

p 次元観測変数を $\boldsymbol{X} = (X_1, \ldots, X_p)^T$ とし，その平均ベクトルと分散共分散行列をそれぞれ $\boldsymbol{\mu}, \boldsymbol{\Sigma}$ とする．このとき，因子分析モデルは次で与えられる．

$$\boldsymbol{X} = \boldsymbol{\mu} + \boldsymbol{\Lambda}\boldsymbol{F} + \boldsymbol{\varepsilon}.$$

ただし，$\boldsymbol{\Lambda} = (\lambda_{ij})$ は $p \times m$ 因子負荷行列，$\boldsymbol{F} = (F_1, \cdots, F_m)^T$ は m 次元共通因子ベクトル，$\boldsymbol{\varepsilon} = (\varepsilon_1, \cdots, \varepsilon_p)^T$ は p 次元独自因子ベクトルとする．共通因子ベクトル $\boldsymbol{F}$ と独自因子ベクトル $\boldsymbol{\varepsilon}$ はそれぞれ独立に多変量正規分布に従うと仮定し，それらの平均ベクトルと分

散共分散行列は $E(\boldsymbol{F})=\mathbf{0}$, $E(\boldsymbol{\varepsilon})=\mathbf{0}$, $E(\boldsymbol{F}\boldsymbol{F}^T)=\boldsymbol{I}_m$, $E(\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^T)=\boldsymbol{\Psi}$ で与えられるとする．ここで，$\boldsymbol{I}_m$ は $m\times m$ 単位行列，$\boldsymbol{\Psi}$ は $p\times p$ 対角行列であり，その第 i 対角成分は ψ_i で与えられる．また，ψ_i は独自分散と呼ばれる．

これらの仮定により，観測変数ベクトル $\boldsymbol{X}$ は，平均ベクトル $\boldsymbol{\mu}$, 分散共分散行列 $\boldsymbol{\Sigma}=\boldsymbol{\Lambda}\boldsymbol{\Lambda}^T+\boldsymbol{\Psi}$ を持つ多変量正規分布に従う．なお，因子間に相関を仮定した斜交モデル（すなわち，$E(\boldsymbol{F}\boldsymbol{F}^T)=\boldsymbol{\Phi}$，$\boldsymbol{\Phi}$ は単位行列でない）に対しても問題なく議論を進めることができるが (Hirose and Yamamoto 2013)，記号が煩雑となるため，本稿では直交モデルで話を進めることとする．

N 個の p 次元データ $\boldsymbol{x}_1,\cdots,\boldsymbol{x}_N$ が $N_p(\boldsymbol{\mu},\boldsymbol{\Lambda}\boldsymbol{\Lambda}^T+\boldsymbol{\Psi})$ から発生した時，モデルパラメータ $\boldsymbol{\Lambda},\boldsymbol{\Psi}$ を最尤法によって推定する．対数尤度関数は

$$\ell(\boldsymbol{\Lambda},\boldsymbol{\Psi})=-\frac{N}{2}\left\{p\log(2\pi)+\log|\boldsymbol{\Sigma}|+\mathrm{tr}(\boldsymbol{\Sigma}^{-1}\boldsymbol{S})\right\},$$

で与えられる．ただし，$\boldsymbol{\Sigma}=\boldsymbol{\Lambda}\boldsymbol{\Lambda}^T+\boldsymbol{\Psi}$，$\mathbf{S}=(s_{ij})$ は標本分散共分散行列とする．

4.1.2 因子回転

$\boldsymbol{T}$ を任意の直交行列とする．このとき，$\boldsymbol{\Sigma}=\boldsymbol{\Lambda}\boldsymbol{\Lambda}^T+\boldsymbol{\Psi}=(\boldsymbol{\Lambda}\boldsymbol{T})(\boldsymbol{\Lambda}\boldsymbol{T})^T+\boldsymbol{\Psi}$ が成り立つため，2つの因子負荷行列 $\boldsymbol{\Lambda}$ と $\boldsymbol{\Lambda}\boldsymbol{T}$ は同じ共分散構造 $\boldsymbol{\Sigma}$ を持つ．そのため，バリマックス法 (Kaiser 1958) などの因子回転によって解釈しやすい共通因子を見つけ出す．いま，$Q(\boldsymbol{\Lambda})$ を，因子負荷行列 $\boldsymbol{\Lambda}$ に対する直交回転基準とする．この基準を初期値 $\hat{\boldsymbol{\Lambda}}_{\mathrm{ML}}$ としたときに最小化することを考える．すなわち，

$$\min_{\boldsymbol{\Lambda}} Q(\boldsymbol{\Lambda}),\ \text{subject to}\ \boldsymbol{\Lambda}=\hat{\boldsymbol{\Lambda}}_{\mathrm{ML}}\boldsymbol{T}\ \text{and}\ \boldsymbol{T}^T\boldsymbol{T}=\boldsymbol{I}_m \tag{4.1}$$

を解く．本稿では，以後，回転基準 $Q(\boldsymbol{\Lambda})$ は Component loss criterion $Q(\boldsymbol{\Lambda})=\sum_{i=1}^{p}\sum_{j=1}^{m}P(|\lambda_{ij}|)$ (Jennrich 2004, 2006) であると仮定する．このとき，(4.1) の問題は

$$\min_{\boldsymbol{\Lambda}}\sum_{i=1}^{p}\sum_{j=1}^{m}P(|\lambda_{ij}|),\ \text{subject to}\ \boldsymbol{\Lambda}=\hat{\boldsymbol{\Lambda}}_{\mathrm{ML}}\boldsymbol{T}\ \text{and}\ \boldsymbol{T}^T\boldsymbol{T}=\boldsymbol{I}_m. \tag{4.2}$$

と表現することができる．

4.2 非凸な正則化項に基づく正則化最尤法

因子回転を用いた古典的な推定法では，十分にスパースな解が得られないことが多い．さらに，データの次元数がサンプルサイズより大きいとき，最尤推定値を計算することが

できない．そこで，因子回転より疎な解を求め，高次元データを扱うことができるように，正則化最尤法を適用する．

まず，古典的な2段階推定法の自然な拡張が正則化最尤法になることを示す．最尤推定値 $\hat{\mathbf{\Lambda}}_{\mathrm{ML}}$ は回転の不定性を除いて一意に存在すると仮定すると（一意性を満たすための必要条件は，たとえば， Anderson and Rubin (1956) の Theorem 5.1 を参照されたい），(4.2) の問題は次のように表現できる．

$$\min_{\mathbf{\Lambda}} \sum_{i=1}^{p} \sum_{j=1}^{m} P(|\lambda_{ij}|), \text{ subject to } \quad \ell(\mathbf{\Lambda}, \mathbf{\Psi}) = \hat{\ell}. \tag{4.3}$$

ただし，$\hat{\ell} = \ell(\hat{\mathbf{\Lambda}}_{\mathrm{ML}}, \hat{\mathbf{\Psi}}_{\mathrm{ML}})$ とする．

ここで，より疎な解を求めるために，(4.3) を次のように修正する．

$$\min_{\mathbf{\Lambda}} \sum_{i=1}^{p} \sum_{j=1}^{m} P(|\lambda_{ij}|), \text{ subject to } \quad \ell(\mathbf{\Lambda}, \mathbf{\Psi}) \geq \ell^*. \tag{4.4}$$

ただし，ℓ^* $(\ell^* \leq \hat{\ell})$ は定数とする．ℓ^* の値はデータへの当てはまりの良さと因子負荷行列のスパース性の強さのバランスを調整している．実際，$\ell^* = \hat{\ell}$ の時，解は最尤推定値に一致し，$\ell^* \to -\infty$ のとき，$\mathbf{\Lambda} = \boldsymbol{O}$ と推定される．

(4.4) 式の問題の解は，次の正則化対数尤度関数 $\ell_\rho(\mathbf{\Lambda}, \mathbf{\Psi})$ の最大化によって得られる．

$$\ell_\rho(\mathbf{\Lambda}, \mathbf{\Psi}) = \ell(\mathbf{\Lambda}, \mathbf{\Psi}) - N \sum_{i=1}^{p} \sum_{j=1}^{m} \rho P(|\lambda_{ij}|). \tag{4.5}$$

ただし，$\rho > 0$ は正則化パラメータとする．ここで，$P(\cdot)$ は正則化項と見なすことができる．ρ の値は正則化パラメータに対応し，パラメータの縮小の度合いを表す．(4.5) 式より，正則化最尤法は古典的な2段階推定の一般化とみなすことができる．

Lasso はスパースな解が得られるものの，バイアスを持ち，そのため過度に密な推定値が得られることが知られている (e.g., Zou 2006; Zhang 2010)．この問題に対処するために，Hirose and Yamamoto (2012) は，非凸な正則化法を適用し，よりスパースな解を求めた．

4.3 アルゴリズム

2.3節では，Lasso の推定アルゴリズムとして，CDA が効率的に計算できることを述べた．しかしながら，因子分析のロス関数は複雑であり，そのまま CDA を適用することは効率的でない．そこで，EM アルゴリズム (Rubin and Thayer 1982) を適用することを考

える．EMアルゴリズムのMステップでは，ロス関数が二乗ロスに対応し，そのため高速なアルゴリズムが構築可能となる．

いま，$\mathbf{\Lambda}_{\text{old}}, \mathbf{\Psi}_{\text{old}}$ が因子負荷行列と独自分散の現在の値とする．このとき，完全対数尤度関数は次で与えられる．

$$E[l_\rho^C(\mathbf{\Lambda}, \mathbf{\Psi})] = -\frac{N}{2}\sum_{i=1}^{p}\log\psi_i - \frac{N}{2}\sum_{i=1}^{p}\frac{s_{ii} - 2\boldsymbol{\lambda}_i^T\boldsymbol{b}_i + \boldsymbol{\lambda}_i^T\boldsymbol{A}\boldsymbol{\lambda}_i}{\psi_i} - \frac{N\rho}{2}\sum_{i=1}^{p}\sum_{j=1}^{m}P(|\lambda_{ij}|) + \text{const.} \tag{4.6}$$

ただし，$\boldsymbol{b}_i = \boldsymbol{M}^{-1}\mathbf{\Lambda}_{\text{old}}^T\mathbf{\Psi}_{\text{old}}^{-1}\boldsymbol{s}_i$，$\boldsymbol{A} = \boldsymbol{M}^{-1} + \boldsymbol{M}^{-1}\mathbf{\Lambda}_{\text{old}}^T\mathbf{\Psi}_{\text{old}}^{-1}\boldsymbol{S}\mathbf{\Psi}_{\text{old}}^{-1}\mathbf{\Lambda}_{\text{old}}\boldsymbol{M}^{-1}$ とする．ここで，$\boldsymbol{M} = \mathbf{\Lambda}_{\text{old}}^T\mathbf{\Psi}_{\text{old}}^{-1}\mathbf{\Lambda}_{\text{old}} + \boldsymbol{I}_m$，$\boldsymbol{s}_i$ は標本分散共分散行列 $\boldsymbol{S}$ の第 i 列ベクトルである．正則化完全対数尤度関数の導出については，Hirose and Yamamoto (2012) を参照されたい．

ここで，新しいパラメータ $(\mathbf{\Lambda}_{\text{new}}, \mathbf{\Psi}_{\text{new}})$ は，(4.6) 式の正則化完全対数尤度関数の最大化によって更新される．(4.6) 式はパラメータの絶対値を含む関数を含むために，パラメータ $\mathbf{\Lambda}_{\text{new}}$ の更新式を解析的に求めることは困難となる．そこで，CDAを適用する．

因子負荷行列の更新値 $\hat{\mathbf{\Lambda}}_{\text{new}}$ を求めた後，独自分散を次のように更新する．

$$(\psi_i)_{\text{new}} = s_{ii} - 2(\hat{\boldsymbol{\lambda}}_i^T)_{\text{new}}\boldsymbol{b}_i + (\hat{\boldsymbol{\lambda}}_i)_{\text{new}}^T\boldsymbol{A}(\hat{\boldsymbol{\lambda}}_i)_{\text{new}} \quad \text{for } i = 1, \ldots, p.$$

ただし，$(\psi_i)_{\text{new}}$ は $\mathbf{\Psi}_{\text{new}}$ の第 i 対角成分，$(\hat{\boldsymbol{\lambda}}_i)_{\text{new}}$ は $\hat{\mathbf{\Lambda}}_{\text{new}}$ の第 i 行ベクトルとする．

備考 4.1. ここでは調整パラメータ与えられたときのパラメータ推定法を述べたが，実際はあらゆる調整パラメータに対する解のパスを求める必要がある．因子分析モデルでは，初期値を適切に設定することにより高速に解のパスを求めることができる．詳細は *Hirose and Yamamoto (2012)* を参照されたい．

5 おわりに

回帰モデルにおける L_1 型正則化法における解の推定アルゴリズムはすでに成熟しつつある．しかしながら，回帰モデルの枠組みを越えると，LARSなどの従来の推定アルゴリズムをそのまま適用することは困難となる．たとえば因子分析では，対数尤度関数にCDAを直接用いることが困難であった．そのため，EMアルゴリズムとCDAを組み合わせる必要があった．このように，ロス関数と正則化項の形に応じて，できるだけ効率的に推定できるアルゴリズムの提案が不可欠となる．

また，適切な調整パラメータの選択も重要となる．ここでいう「適切」とは，解析の目的によって異なり，たとえば，回帰モデルのように，将来取得されるであろうデータへのあてはまりの良さを重視する場合もあれば，因子分析のように，推定された因子の解釈を

重視する場合もある．そのため，モデルのデータへの当てはまりの良さ，推定されたモデルの解釈，将来得られるデータへの予測など，どのような観点から調整パラメータを選択すべきかを考慮した調整パラメータ選択法が必要とされる．

参考文献

Akaike, H. (1987), "Factor analysis and AIC," *Psychometrika*, 52(3), 317–332.

Anderson, T., and Rubin, H. (1956), Statistical inference in factor analysis,, in *Proceedings of the third Berkeley symposium on mathematical statistics and probability*, Vol. 5, pp. 111–150.

Breheny, P., and Huang, J. (2011), "Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection," *The annals of applied statistics*, 5(1), 232.

Efron, B. (1986), "How Biased is the Apparent Error Rate of a Prediction Rule?," *Journal of the American Statistical Association*, 81, 461–470.

Efron, B. (2004), "The Estimation of Prediction Error: Covariance Penalties and Cross-Validation," *Journal of the American Statistical Association*, 99, 619–642.

Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004), "Least Angle Regression (with discussion)," *The Annals of Statistics*, 32, 407–499.

Fan, J., and Li, R. (2001), "Variable Selection via Nonconcave Penalized Likelihood and its Oracle Properties," *Journal of the American Statistical Association*, 96, 1348–1360.

Friedman, J. H. (2012), "Fast sparse regression and classification," *International Journal of Forecasting*, 28(3), 722–738.

Friedman, J., Hastie, T., and Tibshirani, R. (2008), "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, 9(3), 432–441.

Friedman, J., Hastie, T., and Tibshirani, R. (2010), "Regularization Paths for Generalized Linear Models via Coordinate Descent," *Journal of Statistical Software*, 33(1), 1–22.

Hastie, T., Rosset, S., Tibshirani, R., and Zhu, J. (2004), "The entire regularization path for the support vector machine," *The Journal of Machine Learning Research*, 5, 1391–1415.

Hastie, T., and Tibshirani, R. (1990), *Generalized Additive Models*, London: Chapman and Hall/CRC Monographs on Statistics and Applied Probability.

Hastie, T., Tibshirani, R., and Friedman, J. (2008), *The Elements of Statistical Learning*, 2nd edn, New York: Springer.

Hirose, K., Tateishi, S., and Konishi, S. (2012), "Tuning parameter selection in sparse regression modeling," *Computational Statistics & Data Analysis*, .

Hirose, K., and Yamamoto, M. (2012), "Sparse estimation via nonconcave penalized likelihood in a factor analysis model," *arXiv preprint arXiv:1205.5868*, .

Hirose, K., and Yamamoto, M. (2013), "Estimation of oblique structure via penalized likelihood factor analysis," *arXiv preprint arXiv:1302.5475*, .

Jennrich, R. (2004), "Rotation to simple loadings using component loss functions: The orthogonal case," *Psychometrika*, 69(2), 257–273.

Jennrich, R. (2006), "Rotation to simple loadings using component loss functions: The oblique case," *Psychometrika*, 71(1), 173–191.

Kaiser, H. (1958), "The varimax criterion for analytic rotation in factor analysis," *Psychometrika*, 23(3), 187–200.

小西貞則 (2010). 「多変量解析入門: 線形から非線形へ」. 岩波書店.

Mazumder, R., Friedman, J., and Hastie, T. (2011), "SparseNet: Coordinate Descent with Nonconvex Penalties," *Journal of the American Statistical Association*, 106, 1125–1138.

Rubin, D., and Thayer, D. (1982), "EM algorithms for ML factor analysis," *Psychometrika*, 47(1), 69–76.

Simon, N., Friedman, J., Hastie, T., and Tibshirani, R. (2011), "Regularization paths for Cox ' s proportional hazards model via coordinate descent," *Journal of Statistical Software*, 39(5), 1–13.

Stein, C. (1981), "Estimation of the Mean of a Multivariate Normal Distribution," *The Annals of Statistics*, 9, 1135–1151.

Tibshirani, R. (1996), "Regression Shrinkage and Selection via the Lasso," *Journal of the Royal Statistical Society, Ser. B*, 58, 267–288.

Wang, H., Li, B., and Leng, C. (2009), "Shrinkage tuning parameter selection with a diverging number of parameters," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71(3), 671–683.

Wang, H., Li, R., and Tsai, C. (2007), "Tuning Parameter Selectors for the Smoothly Clipped Absolute Seviation Method," *Biometrika*, 94, 553–568.

Ye, J. (1998), "On Measuring and Correcting the Effects of Data Mining and Model Selection," *Journal of the American Statistical Association*, 93, 120–131.

Yuan, M., and Lin, Y. (2007), "Model selection and estimation in the Gaussian graphical model," *Biometrika*, 94(1), 19–35.

Zhang, C. (2010), "Nearly Unbiased Variable Selection Under Minimax Concave Penalty," *The Annals of Statistics*, 38, 894–942.

Zhao, P., and Yu, B. (2007), "On model selection consistency of Lasso," *Journal of Machine Learning Research*, 7(2), 2541.

Zou, H. (2006), "The Adaptive Lasso and its Oracle Properties," *Journal of the American Statistical Association*, 101, 1418–1429.

Zou, H., and Hastie, T. (2005), "Regularization and Variable Selection via the Elastic Net," *Journal of the Royal Statistical Society, Ser. B*, 67, 301–320.

Zou, H., Hastie, T., and Tibshirani, R. (2007), "On the Degrees of Freedom of the Lasso," *The Annals of Statistics*, 35, 2173–2192.

Zou, H., and Li, R. (2008), "One-step sparse estimates in nonconcave penalized likelihood models," *Annals of Statistics*, 36(4), 1509.