楽しい逆問題 ver 2.5

河原　創

2022 年 11 月 12 日

- 本文章は 2015/6 月 17 日の統計ゼミ用に作成した文書です。2022 年現在、改定中です

目次

第 1 章

イントロダクション

逆問題とは、入力に対し応答を計算する順問題に対し、応答から入力を推定する問題である。コンピュータ・トモグラフィー、地震波トモグラフィー、天文電波イメージング、リモートセンシングなど多岐にわたる応用がある。本書の内容はさまざまな問題に適用できるが、問題設定が特定の分野に偏っていると、その背景を理解するのに時間がかかってしまいそうだ。そこで本書では以下の仮想的な問題設定を考えていく。少し長いがお付き合い願いたい。

1.1 Che mapping

ある特殊な部隊が山中を進んでいる。少人数からなり先に偵察を行う斥候隊と様々な機材を持った本隊が、お互いの位置を目視できる程度の距離を保ちながら進んでいく。斥候隊と本隊の間の通信に用いることができるのは、貧弱な無線機とサーチライトのみである。ある時、先に進んだ斥候隊が大きな壁画を発見した。斥候隊はこの壁画にかかれている人物が誰だかわからないが何か重要そうな情報のように思える。そこで本隊にライトと無線を使って、この壁画の画像情報を送ることを考えている。

ケース 1: 謎の壁画

図1.1のように本隊の位置からではタイル壁画が遠すぎて点にしか見えない。そこで夜までまって、斥候隊に壁画の様々な場所 N 点にライトを当ててもらうよう頼んだ。ライトの大きさはタイルよりかなり大きく、山の向こう側から見ている我々は、ライトの当たっている部分の壁外による反射光の和 (観測データ) が本隊の持つカメラを通じてわかるのみである。また、壁画のどのタイルを中心にライトがどれだけ広がっているかは無線機を通じて知らされる。さて、このような観測データとライトの位置情報から元の壁画に誰が描かれているか推定しよう。のちに理由が明らかになるが、この問題を Che mapping と呼ぶ。Che mapping は線形逆問題というタイプの逆問題である。すなわち壁画と観測データは線形変換を通じて結びつく。線型方程式の解法と異なり、本質的に決定できない成分が存在する。線形逆問題では、特異値分解を通じて、決定できる成分（空間）と決定できない成分（空間）を完全に分離することができる。すなわち我々が逆問題を推定するときに、どの空間をあきらめ、どの空間について推定しているのかを明確にすることができる。これは非線形パラメタが介在するとこのように明確に分けることはできない。第2章では、線形逆問題の解法と構造について考える。この章では主に一つの解だけをもとめる手法（点推定）について議論する。

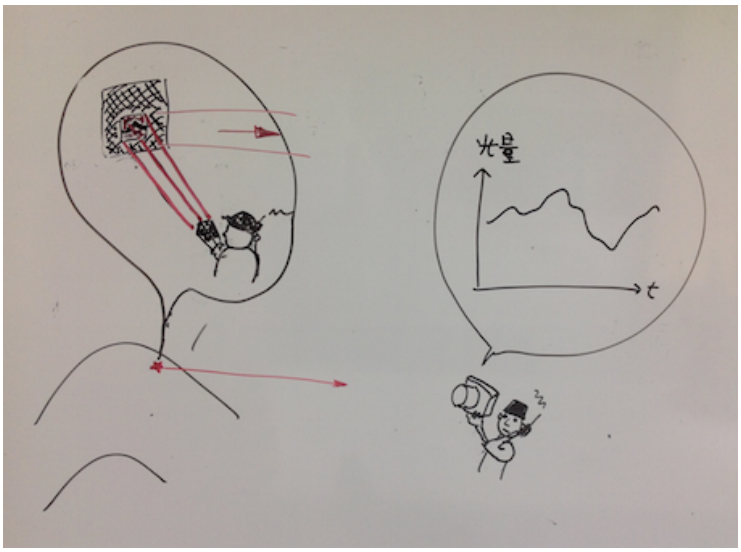

図1.1　Che mapping。

ケース 2: 再生した壁画の不定性を評価する

壁画が無事に再生できたとしよう。しかし、この再生画像はどれくらい確かなのだろうか？さらに斥候隊の知らせてくるサーチライトの座標情報に不定性がある場合、座標情報の推定も同時に行いながら上記の推定はできるのだろうか？第5章では、第2章で議論した逆問題の解法をベイズ統計的に再解釈することで、点推定を推定値の事後分布の形に拡張する。これにより不定性を評価した形で逆問題を解くことができる。

特に、非線形パラメタが介在するとき、また観測ノイズの推定も同時に行いたいとき、ハイパーパラメタの周辺化を行いたいときなどにベイズ逆問題は有効である。

ケース 3: 動く壁画

斥候隊がよくよくみるとこの壁画は徐々にではあるが形を変えているようだ。恐ろしい技術である。この壁画は未来の技術をつかって作られているのだろう。ケース 1 と同様の方法で壁画の動画を本隊に送ることはできるであろうか？この問題は本質的には線形逆問題の拡張で記述できるものの、現実的にはモデルパラメタの数がデータ数に比べて莫大であるため（動画と静止画のファイルサイズの違いを思い出してほしい）、計算量的に工夫が必要である。本稿では、同型写像を利用した線形逆問題の拡大・再収縮法 [**?**] を用いて、コンパクトで計算可能な解を導出することでこの問題を解決する。

ケース 4: カラーの壁画

本隊がカラーのカメラもしくは分光カメラを持っていた場合、壁画に使われているペンキの種類を壁画の絵と同時に知ることはできるだろうか？この問題は行列を分解する問題に帰着する。行列分解は自由度が高く、様々なタイプのものが存在する。7章では、特に [3] に基づき、線形逆問題の拡張としての非負値行列分解を導入してその解法を議論する。

1.2　参考文献等について

本書の内容は、先行の素晴らしい教科書と論文に多くを負っている。線形逆問題は、古くから地球物理学の分野で発展してきた。Menke (1989) [6] は離散線形逆問題の解法とその仕組みついて詳しく解説されている。Hansen (2010) [2] では具体的問題を解くにあたって有用な情報が簡潔な式と共に数多くみられる。また Tarantola (2005) [9] は、逆問題を確率論的推論に発展させた Tarantola による基本的な教科書である。第6章と第7章 (それと第5章の一部) は筆者の研究に基づく内容を多分に含んでいる。それゆえ理論的に不完全な部分も多いかと思われる。読者からのご指摘をお待ちしている。

第 2 章

線形逆問題

2.1 離散線形逆問題

ある種の理論モデルは、モデルパラメタ $\boldsymbol{a}$ をあたえると

$$\boldsymbol{F}(\boldsymbol{d}, \boldsymbol{a}) = \boldsymbol{0} \tag{2.1}$$

のような形式で、観測データ列 $\boldsymbol{d}$ を予言するように問題を構成する事ができる。このような問題系で、観測データ列 $\boldsymbol{d}$ から、モデルパラメタ $\boldsymbol{a}$ を推定する問題を逆問題と呼ぶ。この例は有限個のパラメタを推定するので離散逆問題ともよぶ。

連続モデル関数を推定する逆問題も存在する。代表的な問題の一つに第一種フレドホルム積分型逆問題

$$d(x) = \int W(x, y)a(y)dy \tag{2.2}$$

がある。ここで $W(x, y)$ は積分核でモデルを記述する。

この第一種フレドホルム積分型逆問題を離散化し、式 (2.1) の形にすると

$$\boldsymbol{d} - W\boldsymbol{a} = \boldsymbol{0} \tag{2.3}$$

のように変形することができる。この形式で書かれる逆問題は離散線形逆問題と呼ばれ、逆問題としては最も、基本的な形式となる。ここに W は design matrix とよばれ、モデルをあらわす行列となっている。

実際には、用いる観測データは誤差を含んでいる。観測誤差を含んでいるデータ列を $\boldsymbol{d}^{\mathrm{obs}}$ (observation) で表す事にする。逆問題の手法で何らかのモデル推定が得られたとしよう。この推定されたモデルを $\boldsymbol{a}^{\mathrm{est}}$ (estimation) と表す。するとこの推定されたモデルと式 (2.1) から得られる $\boldsymbol{d}$ にたいする解を $\boldsymbol{d}^{\mathrm{pre}}$ (prediction) としよう。すなわち $\boldsymbol{d}^{\mathrm{pre}}$ は

$$\boldsymbol{d}^{\mathrm{pre}} = W\boldsymbol{a}^{\mathrm{est}} \tag{2.4}$$

をみたす。

観測の予測誤差は

$$\boldsymbol{e} \equiv \boldsymbol{d}^{\mathrm{obs}} - \boldsymbol{d}^{\mathrm{pre}} = \boldsymbol{d}^{\mathrm{obs}} - W\boldsymbol{a}^{\mathrm{est}} \tag{2.5}$$

で定義される量である。予測誤差は $\boldsymbol{a}^{\mathrm{est}}$ の観測データに対する説明力を表していると言えよう。$\boldsymbol{e}$ はベクトルであるので、なんらかのノルムを定義して指標とする。通常 L2 ノルム、すなわち二乗和の平方根

$$|\boldsymbol{e}|_2 \equiv \sqrt{\boldsymbol{e}^T\boldsymbol{e}} \tag{2.6}$$

を指標とする。この予測誤差のノルムが小さいほど $\boldsymbol{a}^{\mathrm{est}}$ は観測データ $\boldsymbol{d}^{\mathrm{obs}}$ を説明していると言える。

2.2 特異値分解

さて、逆問題の推定値 $\boldsymbol{a}^{\rm est}$ を構成する前に、離散線形逆問題で $\boldsymbol{a}$ と $\boldsymbol{d}$ がどのように関係しているかを調べよう。このためには、Design matrix W を特異値分解して考えると都合が良い。

特異値分解とは $N \times N$ 直交行列 U、$M \times M$ 直交行列 V、$N \times M$ の対角固有値行列

$$\Lambda = \begin{pmatrix} \Lambda_p & 0 \\ 0 & 0 \end{pmatrix} \tag{2.7}$$

$$\Lambda_p \equiv diag(\kappa_1, .., \kappa_p) \tag{2.8}$$

をもちいて、Design matrix を

$$W = U\Lambda V^T \tag{2.9}$$

と分解することである (特異値 κ_i は普通大きい順から並べる)。

さてここで直交行列の各列ベクトル

$$U = (\boldsymbol{u}_1, ...\boldsymbol{u}_N) \tag{2.10}$$

$$V = (\boldsymbol{v}_1, ...\boldsymbol{v}_M) \tag{2.11}$$

とかく。特異値分解は固有値の数 p を用いて

$$W = U\Lambda V^T = U_p\Lambda_p V_p^T \tag{2.12}$$

$$U_p = (\boldsymbol{u}_1, ...\boldsymbol{u}_p) \tag{2.13}$$

$$V_p = (\boldsymbol{v}_1, ...\boldsymbol{v}_p) \tag{2.14}$$

と変形できる。すなわち、$\boldsymbol{d} = W\boldsymbol{a}$ は

$$\boldsymbol{d} = U_p\Lambda_p V_p^T \boldsymbol{a} \tag{2.15}$$

とも書き直せる[*1]。

ここで推定したモデル $\boldsymbol{a}^{\rm est}$ を $\boldsymbol{v}$ の線形結合に分解しよう。

$$\boldsymbol{a}^{\rm est} = \sum_{j=1} c_j \boldsymbol{v}_j \tag{2.16}$$

で書いて、式2.15に入れてみると、

$$V_p^T \boldsymbol{a}^{\rm est} = \begin{pmatrix} \boldsymbol{v}_0^T \\ \boldsymbol{v}_1^T \\ ... \\ \boldsymbol{v}_p^T \end{pmatrix} \sum_{j=1} c_j \boldsymbol{v}_j = \begin{pmatrix} c_1 \\ c_2 \\ ... \\ c_p \end{pmatrix} \tag{2.17}$$

となることから (ここで直交行列の性質から $\boldsymbol{v}_i^T \boldsymbol{v}_j = \delta_{ij}$; クロネッカーデルタ、となることを用いている)、

$$\boldsymbol{d}^{\rm pre} = W\boldsymbol{a}^{\rm est} = \sum_{j=0}^{p} \kappa_j c_j \boldsymbol{u}_j \tag{2.18}$$

[*1] V、U は直交行列のため $UU^T = U^TU = VV^T = V^TV = I$ であるが、U_p ($N \times p$ 行列)、V_p ($M \times p$ 行列) は、$p \leq N$、$p \leq M$ であるため、$V_p^TV_p = U_p^TU_p = I$ ($p \times p$ 行列) であるが、$V_pV_p^T$、$U_pU_p^T$ は単位行列とは限らない ($p = N$、もしくは $p = M$ のときのみ成立) ことに注意。

となる。

つまり、$\boldsymbol{d}^{\mathrm{pre}}$ は $j=p+1$ から $j=M$ の項には全く依存しないことが分かる。そこでこの空間を

$$V_\circ = (\boldsymbol{v}_{p+1}, ...\boldsymbol{v}_M) \tag{2.19}$$

と定義する。すなわち $\boldsymbol{a}^{\mathrm{est}}$ は V_p、$V_\circ$ によって張られる直交する二つの空間 (**p** 空間、ヌル空間) 内の (直交する) 二つのベクトル $\boldsymbol{a}_p = \sum_{j=1}^{p} c_j \boldsymbol{v}_j$、$\boldsymbol{a}_\circ = \sum_{j=p+1} c_j \boldsymbol{v}_j$ の和

$$\boldsymbol{a}^{\mathrm{est}} = \boldsymbol{a}_p + \boldsymbol{a}_\circ \tag{2.20}$$

に分けることができ、$\boldsymbol{a}_\circ$ は全く $\boldsymbol{d}^{\mathrm{pre}}$ に影響を与えないことから、モデルからの予測 $\boldsymbol{d}^{\mathrm{pre}}$ と観測データ $\boldsymbol{d}^{\mathrm{obs}}$ の比較から $\boldsymbol{a}_\circ$ を推定する事はできない。$V_\circ$ が空集合、すなわち $p=M$ である時はデータからモデルを完全に決定できるので優問題 (well-posed problem) とよぶ。$p \leq N$ でないとならないので、$N \geq M$、すなわちデータ数がモデルパラメタ数より大きい事は優問題である事の必要条件である。しかし、たとえ $N \geq M$ であっても、$p < M$ で有る場合、もしくは $N \leq M$ であり必然的に $p < M$ である場合は $V_\circ$ が存在するので、推定できない $\boldsymbol{a}_\circ$ が存在する。この場合、を劣問題 (ill-posed problem) もしくは混合問題とよぶ。

問題が ill-posed でも絶望する必要は無い。この場合、逆問題の解法は陽に暗に $\boldsymbol{a}_\circ$ を仮定する事 (事前情報を与えること) によりモデルを推定する事ができる。ベイズ主義者の場合、事前情報を与えて問題を解く形式に違和感は無いだろう。ベイズ主義者でない場合も以下のように考えてもらいたい。例えば、$\boldsymbol{a}_\circ$ の要素にあまり変動の無いベクトルだった場合、つまり対応する離散化前の関数 $m(x)$ がのっぺりした関数だった場合、そういう成分はどうでもよいという事もあるだろう。しかし、モデルのどういった成分が $\boldsymbol{d}^{\mathrm{obs}}$ からは表現できないのかは理解しておく必要がある。

次に、観測データのほうを $\boldsymbol{u}_i$ の線形結合

$$\boldsymbol{d}^{\mathrm{obs}} = \sum_{i=1}^{N} k_i \boldsymbol{u}_i \tag{2.21}$$

で表してみよう。この場合、

$$\boldsymbol{e} = \boldsymbol{d}^{\mathrm{obs}} - W\boldsymbol{a}^{\mathrm{est}} \tag{2.22}$$

を $\boldsymbol{a}^{\mathrm{est}}$ を調整する事でゼロベクトルにすることはできるのだろうか？この場合、$\boldsymbol{d}^{\mathrm{obs}}$ の $\boldsymbol{u}_{p+1}$ から $\boldsymbol{u}_N$ の成分は、どのように $\boldsymbol{a}^{\mathrm{est}}$ を動かしても引き去る事ができない。なぜなら $q > p$ に対して、$\boldsymbol{u}_q$ と $W\boldsymbol{a}^{\mathrm{est}}$ の内積をとると、

$$\boldsymbol{u}_q^T W\boldsymbol{a}^{\mathrm{est}} = \boldsymbol{u}_q^T U_p \Lambda_p V_p^T \boldsymbol{a}^{\mathrm{est}} = 0 \tag{2.23}$$

となり、$W\boldsymbol{a}^{\mathrm{est}}$ の $\boldsymbol{u}_q$ の要素は 0 であるからである。すなわちヌル空間を

$$U_\circ = (\boldsymbol{u}_{p+1}, ...\boldsymbol{u}_N) \tag{2.24}$$

として、$\boldsymbol{d}^{\mathrm{obs}}$ は U_p、$U_\circ$ によって張られる直交する二つの空間 (**p** 空間、ヌル空間) 内の (直交する) 二つのベクトル $\boldsymbol{d}_p = \sum_{i=1}^{p} k_i \boldsymbol{u}_i$、$\boldsymbol{d}_\circ = \sum_{i=p+1}^{N} k_i \boldsymbol{u}_i$ の和

$$\boldsymbol{d}^{\mathrm{obs}} = \boldsymbol{d}_p + \boldsymbol{d}_\circ \tag{2.25}$$

に分けることができ、どんなモデルパラメタ $\boldsymbol{a}^{\mathrm{est}}$ を持ってきても $W\boldsymbol{a}^{\mathrm{est}}$ は $\boldsymbol{d}_\circ$ の成分をもつことはできない。

2.3　自然一般化逆行列

もし観測誤差が存在しない場合、明らかに $\boldsymbol{d}^{\rm obs}$ の $U_\circ$ は空集合となるはずである。しかし、実際には観測誤差のため $\boldsymbol{d}_\circ$ の成分が $\boldsymbol{d}^{\rm obs}$ に存在する。ということは予測誤差のノルム、式 (2.6) を最小にするためには $\boldsymbol{d}^{\rm obs}$ の $\boldsymbol{d}_p$ の成分を $W\boldsymbol{a}^{\rm est}$ で表現できれば良い。このような事を実現する一つの例が自然一般化逆行列である

いま特異値分解により $\boldsymbol{d} = W\boldsymbol{a}$ は

$$\boldsymbol{d} = U_p \Lambda_p V_p^T \boldsymbol{a} \tag{2.26}$$

のように書けている訳だが、これを逆行列のようにして

$$\begin{aligned} \boldsymbol{a}^{\rm est} &= W^{-g}\boldsymbol{d}^{\rm obs} \\ W^{-g} &\equiv V_p \Lambda_p^{-1} U_p^T \end{aligned} \tag{2.27}$$

のようにした W^{-g} を自然一般化逆行列 (natural generalized inverse matrix; NGIM) という。NGIM により得られた解は次の性質を持つ

- A. 予測誤差最小である。

どんな $\boldsymbol{a}$ を持ってきても $\boldsymbol{d}_\circ$ を変えることはできないことから、予測誤差最小とは、予測誤差が $\boldsymbol{d}_p$ の成分を持たないことである。

$$U_p^T(\boldsymbol{d}^{\rm obs} - W\boldsymbol{a}^{\rm est}) = U_p^T(\boldsymbol{d}^{\rm obs} - WW^{-g}\boldsymbol{d}^{\rm obs}) = U_p^T(\boldsymbol{d}^{\rm obs} - U_p U_p^T \boldsymbol{d}^{\rm obs}) = 0 \tag{2.28}$$

であるので、やはり自然一般化逆行列による解 $\boldsymbol{a}^{\rm est}$ は予測誤差最小の解である。

- B. $\boldsymbol{a}^{\rm est}$ は $\boldsymbol{a}_\circ$ の成分を持たない。

これは

$$V_\circ^T \boldsymbol{a}^{\rm est} = 0 \tag{2.29}$$

であることから、$\boldsymbol{a}_\circ = 0$ であることが分かる。このような解を自然解という。

ヌルベクトルは $\boldsymbol{d}^{\rm obs}$ に影響を与えることがないから、NGIM を用いて一般解

$$\boldsymbol{a}^{\rm est} = W^{-g}\boldsymbol{d}^{\rm obs} + V_\circ \alpha \tag{2.30}$$

の形式が得られる。ここに α はヌルベクトルの各成分である。

事前情報として平均ベクトル $\hat{\boldsymbol{a}}$ を与えた場合

$$\boldsymbol{a}^{\rm est} = W^{-g}\boldsymbol{d}^{\rm obs} + (I - W^{-g}W)\hat{\boldsymbol{a}} \tag{2.31}$$

を用いて、解の推定を行うと平均が反映される。ここに

$$V_p^T(I - W^{-g}W)\hat{\boldsymbol{a}} = (V_p^T - V_p^T V_p V_p^T)\hat{\boldsymbol{a}} = 0 \tag{2.32}$$

となっているので、$(I - W^{-g}W)\hat{\boldsymbol{a}}$ の V_p 成分は 0 であり、確かに一般解の一つになっている。また

$$V_\circ^T(I - W^{-g}W)\hat{\boldsymbol{a}} = V_\circ^T\hat{\boldsymbol{a}} = V_\circ^T\hat{\boldsymbol{a}}_0 \tag{2.33}$$

なので、$(I - W^{-g}W)\hat{\boldsymbol{a}}$ の $V_\circ$ 成分は $\hat{\boldsymbol{a}}$ の $V_\circ$ 成分 $\hat{\boldsymbol{a}}_\circ$ になっているので、たしかに事前情報 $\hat{\boldsymbol{a}}$ にもっとも距離が近い一般解を選んでいることになる。

練習：Che mapping

さてイントロダクション Che で導入した Che mapping の例を考えよう。ここではサーチライトは単色であるとする。また壁画は時間変化しない。いま壁画の画素数は $M = 2268$ ある。

附属のコードにて、NGIM で Che mapping をするときは、こんな感じの option を設定する (l=0.0 が NGIM を指定する)。

```
./random_light.py -f che.png -n 1000 -l 0.0 -p 0.7 -w 20.0
```

斥候隊にサーチライトを $N = 1000$ カ所、ランダムに当ててもらった (-n 1000)。またサーチライトの平均の幅は 20 タイル分ある (-w 20.0)。このプログラムでは幅もまちまちにしてあるが、各ショットごとのサーチライトの中心と幅は無線機を通じて正確に伝えられる。観測誤差はないとしている。この場合、NGIM による推定を図2.2に表示した。左が壁画、真ん中が壁画のどこにサーチライトを当てたか、右が NGIM による推定である。斥候隊が壁画の上から 7 割の部分にしかサーチライトの中心を当てなかったため (-p 0.7)、下部分では解像度が悪くなっているが、推定自体はできている。これはサーチライトの幅のおかげで、上 7 割より下も少しだけ情報が含まれているからである。

NGIM でも実際には、数値誤差のためすべての特異値は、なんらかの値をもっているので、実用的には、特異値の列をみて (図2.2)、ギャップのある値以下は 0 であると見なして p を決定する。

特異値分解と一般化逆行列の直交ベクトル表現

線形逆問題では $\boldsymbol{u}$ や $\boldsymbol{v}$ による展開が頻繁に出てくるため、通常の行列表記のみだと、今後の計算が表現しづらい。ここでは Hansen 2010 に従って特異値分解をベクトルで表現し直そう。まず、行列 A、B を

$$A = (\boldsymbol{a}_1, ...\boldsymbol{a}_n) \tag{2.34}$$
$$B = (\boldsymbol{b}_1, ...\boldsymbol{b}_n) \tag{2.35}$$

のように表現する時、

$$AB^T = \sum_{i=1}^{n} \boldsymbol{a}_i\boldsymbol{b}_i^T \tag{2.36}$$

となる。$\boldsymbol{a}\boldsymbol{b}^T$ はダイアド $(\boldsymbol{a} \otimes \boldsymbol{b})$ である。また、

$$AB^T\boldsymbol{c} = \sum_{i=1}^{n}(\boldsymbol{a}_i\boldsymbol{b}_i^T)\boldsymbol{c} = \sum_{i=1}^{n}(\boldsymbol{b}_i^T\boldsymbol{c})\boldsymbol{a}_i \tag{2.37}$$

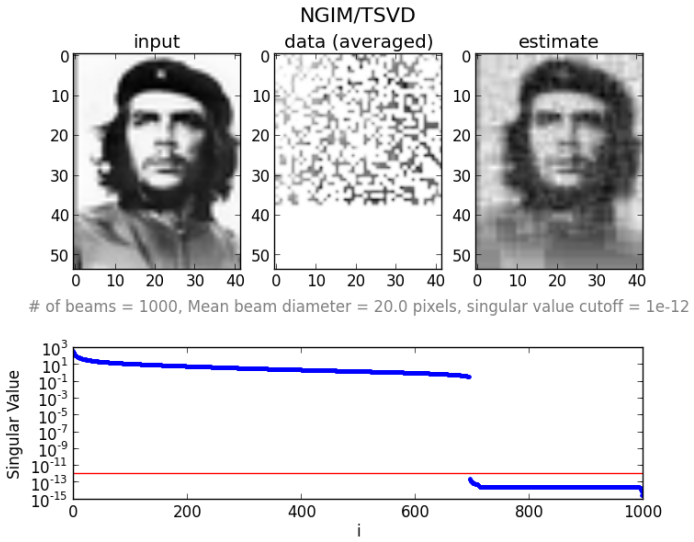

図2.1　NGIM による Che mapping。観測データに（数値誤差を除く）、ノイズがない場合、予測誤差最小は良い推定値を与える。下パネルは特異値を大きい順に並べてある。

である[*2]。$\boldsymbol{b}^T\boldsymbol{c} = \boldsymbol{b}\cdot\boldsymbol{c}$ は内積である。

上記のような表記を用いると特異値分解は

$$U\Lambda V^T = \sum_{i=1}^{\min(N,M)} \kappa_i(\boldsymbol{u}_i\boldsymbol{v}_i^T) \tag{2.38}$$

と表現できる。また一般化逆行列は

$$W^{-g} = \sum_{i=1}^{p} \frac{(\boldsymbol{v}_i\boldsymbol{u}_i^T)}{\kappa_i} \tag{2.39}$$

もしくは特異値に 0 が無いとすると

$$W^{-g} = \sum_{i=1}^{\min(N,M)} \frac{(\boldsymbol{v}_i\boldsymbol{u}_i^T)}{\kappa_i} \tag{2.40}$$

となる。自然解は

$$\boldsymbol{a}^{\text{est}} = W^{-g}\boldsymbol{d} = \sum_{i=1}^{\min(N,M)} \frac{(\boldsymbol{u}_i^T\boldsymbol{d})}{\kappa_i}\boldsymbol{v}_i \tag{2.41}$$

のように、$\boldsymbol{v}_i$ の線形結合で表すことができ、要素は $(\boldsymbol{u}_i^T\boldsymbol{d})/\kappa_i$ となる。

[*2] $(\boldsymbol{a}\otimes\boldsymbol{b})\boldsymbol{c} = \boldsymbol{a}(\boldsymbol{b}\cdot\boldsymbol{c})$

2.4 自然解の不安定性

2.2章ではモデルとデータをそれぞれ

$$\boldsymbol{a}^{\rm est} = \boldsymbol{a}_p + \boldsymbol{a}_\circ \tag{2.42}$$
$$\boldsymbol{d}^{\rm obs} = \boldsymbol{d}_p + \boldsymbol{d}_\circ \tag{2.43}$$

と分けて、お互いに影響を及ぼせない成分を抽出した。しかし、$i \leq p$ の成分についても、特異値が小さい場合はどうなるだろうか？

$$\boldsymbol{e} = \boldsymbol{d}^{\rm obs} - \boldsymbol{d}^{\rm pre} = \boldsymbol{d}^{\rm obs} - W\boldsymbol{a}^{\rm est} = \sum_{j=1}^{p}(k_j - \kappa_j c_j)\boldsymbol{u}_j, \tag{2.44}$$

の $\boldsymbol{u}_i$ 成分 $(i \leq p)$ を $\boldsymbol{a}^{\rm est}$ を調節してゼロにすることを考えよう。この場合、調節するのは c_i ということになる。ここで、もし κ_i が小さいと c_i を大きく変更して $\boldsymbol{d}^{\rm obs}$ の $\boldsymbol{u}_i$ 成分である k_i に合わせないとならない。つまり、特異値が小さい成分に対して、予測誤差最小化をしてしまうと、$\boldsymbol{d}^{\rm obs}$ のちょっとした変化にたいして大幅にモデルが変化してしまう事になる。これを予測誤差最小である自然解の $\boldsymbol{a}^{\rm est}$ の面からも見てみよう。いま自然解は式 (2.41) で表されるように

$$\boldsymbol{a}^{\rm est} = \sum_{i=1}^{\min(N,M)} \nu_i \boldsymbol{v}_i \tag{2.45}$$
$$\nu_i \equiv \frac{\boldsymbol{u}_i^T \boldsymbol{d}^{\rm obs}}{\kappa_i} \tag{2.46}$$

のように、$\boldsymbol{v}_i$ の線形結合で表した時、要素は ν_i となる。ここで、κ_i が小さいと $\boldsymbol{u}_i\boldsymbol{d}^{\rm obs}$ が少し変化しただけで、$\boldsymbol{a}^{\rm est}$ の $\boldsymbol{v}_i$ 成分は大きく変更受けることとなる。これが逆問題の不安定性であり、overfit などと呼ばれている問題である。一般に小さい κ_i に対応する $\boldsymbol{v}_i$ は高周波成分である事が多く、不安定性が起きると、振幅の大きい高周波の不安定成分が $\boldsymbol{a}^{\rm est}$ に現われる

2.5 Tikhonov Regularization

一般に不安定性を抑えるための処方を regularization とよび、様々な種類がある。最も簡単なやり方は p を決定する際に、特異値が厳密な 0 で切るのではなく、特異値をある値以上のものだけ採用して p を選ぶというやり方であり、Trancated SVD などと呼ばれている (そもそも数値的な誤差をかんがえると、実用上は NGIM も Trancated SVD に含まれる)。他に有用な regularization として、**Tikhonov regularization** がある。これは、特異値を

$$1/\kappa_i \to \kappa_i/(\kappa_i^2 + \lambda^2) \tag{2.47}$$

ように dump し、小さい特異値のものを大きいものに入れ替えてから一般化逆行列を構成する。この操作により、推定モデルの不安定性を安定化させる事ができる。すなわち、推定値は

$$\boldsymbol{a}^{\mathrm{est}} = V\Sigma_\lambda U^T \boldsymbol{d}^{\mathrm{obs}} = \sum_{i=1}^{\min(N,M)} \frac{\kappa_i}{\kappa_i^2 + \lambda^2}(\boldsymbol{u}_i^T \boldsymbol{d})\boldsymbol{v}_i \tag{2.48}$$

$$(\Sigma_\lambda)_{ij} \equiv \frac{\kappa_i}{\kappa_i^2 + \lambda^2}\delta_{ij} \tag{2.49}$$

$$\tag{2.50}$$

から求められる。ここに λ は特異値を安定化させるパラメタであり、regularization parameter と呼ばれる。

NGIM と Tikhonov Regularization の比較

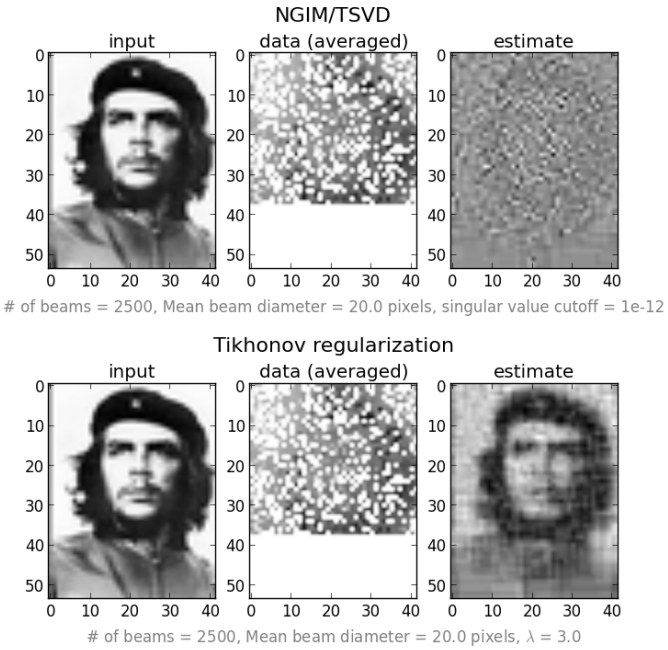

図2.2　データにノイズが有る場合の NGIM(上) と Tikhonov Regularization(下) による Che mapping。NGIM (予測誤差最小) では、モデルの分散が大きすぎ何が書かれているか分からない。これは高周波成分が乗っているためである。そのため、この図をちょっと遠くから見るとなんとなくみえるようになる (この事実は小谷隆行氏指摘による)。Tikhonov Regularization では人だというのはわかるし、チェゲバラな気がする。

図2.2には、観測データにノイズを加えた場合の NGIM と Tikhonov Regularization での Che mapping の結果が示されている。練習コードでは-s でノイズを付加できる。NGIM の場合は、

```
./random_light.py -f che.png -n 2500 -l 0.0 -p 0.7 -s 1.0
```

となる。NGIM の場合、overfit が起きてしまい、モデルの分散が大きくなってしまっている。結果、推定値がランダムノイズ的に見えてしまう。

一方、Tikhonov Regularization で適当な λ を設定して、regularize した場合、それなりに良く推定ができているのがわかる。Tikhonov regularization で Che mapping を行う場合、

```
./random_light.py -f che.png -n 2500 -l 3.0 -p 0.7 -s 1.0
```

のようにすればよい。λ の値はこの例では 3.0 である (-l 3.0)。ノイズは 1.0 である (-s 1.0)

Truncated SVD による regularization も試してみよう。-lim option で 0 とみなす特異値を指定できる。

```
./random_light.py -f che.png -n 2500 -l 0.0 -p 0.7 -s 1.0 -lim 1.0
```

2.5.1 Tiknov regularization とコスト関数

Tikhonov regularization は以下のコスト関数を最小化することと同値である。

$$Q_\lambda = |W\boldsymbol{a} - \boldsymbol{d}^{\rm obs}|^2 + \lambda^2|\boldsymbol{a}|^2 \tag{2.51}$$

これは以下のように示される。式 (2.51) は

$$Q_\lambda = |W'\boldsymbol{a} - \boldsymbol{d}'|^2 \tag{2.52}$$

$$W' \equiv \begin{pmatrix} W \\ \lambda I \end{pmatrix} \tag{2.53}$$

$$\boldsymbol{d}' \equiv \begin{pmatrix} \boldsymbol{d}^{\rm obs} \\ \boldsymbol{0} \end{pmatrix} \tag{2.54}$$

のように変形できる。この関数の最小化は、単純に最小二乗解であるから、解を $\boldsymbol{a}^{\rm est}$ とおくと、正規方程式

$$(W')^T[W'\boldsymbol{a}^{\rm est} - \boldsymbol{d}'] = \boldsymbol{0} \tag{2.55}$$

が成り立つ。すなわち

$$(W^TW + \lambda^2 I)\boldsymbol{a}^{\rm est} - W^T\boldsymbol{d}^{\rm obs} = \boldsymbol{0} \tag{2.56}$$

であり、これを $\boldsymbol{a}^{\rm est}$ について解くと

$$\boldsymbol{a}^{\rm est} = (W^TW + \lambda^2 I)^{-1}W^T\boldsymbol{d}^{\rm obs} \tag{2.57}$$

$$\tag{2.58}$$

となる。さらに W の特異値分解で書き直すと

$$\begin{aligned} \boldsymbol{a}^{\rm est} &= (V\Lambda^T\Lambda V^T + \lambda^2 IVV^T)^{-1}V\Lambda^TU^T\boldsymbol{d}^{\rm obs} && (2.59)\\ &= V(\Lambda^T\Lambda + \lambda^2 I)^{-1}V^TV\Lambda^TU^T\boldsymbol{d}^{\rm obs} && (2.60)\\ &= V(\Lambda^T\Lambda + \lambda^2 I)^{-1}\Lambda^TU^T\boldsymbol{d}^{\rm obs} && (2.61) \end{aligned}$$

となる。$VV^T = I$ を利用している。$(\Lambda^T\Lambda + \lambda^2 I)$ は対角行列であり $\lambda > 0$ である限り逆行列が存在していることに注意。ベクトル形式で書き直すと

$$\begin{aligned} \boldsymbol{a}^{\rm est} &= V\Sigma_\lambda U^T\boldsymbol{d}^{\rm obs} && (2.62)\\ &= \sum_{i=1}^{\min(N,M)} \frac{\kappa_i}{\kappa_i^2 + \lambda^2}(\boldsymbol{v}_i\boldsymbol{u}_i^T)\boldsymbol{d}^{\rm obs} && (2.63)\\ &= \sum_{i=1}^{\min(N,M)} \frac{\kappa_i}{\kappa_i^2 + \lambda^2}(\boldsymbol{u}_i^T\boldsymbol{d}^{\rm obs})\boldsymbol{v}_i && (2.64) \end{aligned}$$

となり、確かに Tikhonov regularization となっている。

2.5.2 Model Prior

モデルの prior $\hat{\boldsymbol{a}}$ を考え、

$$Q_\lambda = |W\boldsymbol{a} - \boldsymbol{d}^{\rm obs}|^2 + \lambda^2|\boldsymbol{a} - \hat{\boldsymbol{a}}|^2 \tag{2.65}$$

を最小化した場合、Tikhonov Regularization はどうなるだろうか？この場合、

$$Q_\lambda = |W'\boldsymbol{a} - \boldsymbol{d}'|^2 \tag{2.66}$$

$$W' \equiv \begin{pmatrix} W \\ \lambda I \end{pmatrix} \tag{2.67}$$

$$\boldsymbol{d}' \equiv \begin{pmatrix} \boldsymbol{d}^{\rm obs} \\ \lambda\hat{\boldsymbol{a}} \end{pmatrix} \tag{2.68}$$

のように変形できるから、正規方程式は

$$(W^TW + \lambda^2)\boldsymbol{a}^{\rm est} - (W^T\boldsymbol{d}^{\rm obs} + \lambda^2\hat{\boldsymbol{a}}) = (W^TW + \lambda^2)(\boldsymbol{a}^{\rm est} - \hat{\boldsymbol{a}}) - W^T(\boldsymbol{d}^{\rm obs} - W\hat{\boldsymbol{a}}) = 0 \tag{2.69}$$

となる。これは式 (2.56) において、$\boldsymbol{a}^{\rm est} \to \boldsymbol{a}^{\rm est} - \hat{\boldsymbol{a}}$、$\boldsymbol{d}^{\rm obs} \to \boldsymbol{d}^{\rm obs} - W\hat{\boldsymbol{a}}$ と置き換えたものと同じなので、結局

$$\boldsymbol{a}^{\rm est} = V\Sigma_\lambda U^T(\boldsymbol{d}^{\rm obs} - W\hat{\boldsymbol{a}}) + \hat{\boldsymbol{a}} \tag{2.70}$$

$$= \sum_{i=1}^{\min(N,M)} \frac{\kappa_i}{\kappa_i^2 + \lambda^2}[\boldsymbol{u}_i^T(\boldsymbol{d}^{\rm obs} - W\hat{\boldsymbol{a}})]\boldsymbol{v}_i + \hat{\boldsymbol{a}} \tag{2.71}$$

となる。

2.5.3 Maximum a Posterior (MAP) と Tikhonov Regularization

最小化 (2.51) は、$\boldsymbol{d}$ の誤差が独立に 1、また $\boldsymbol{a}$ の prior として、平均 0 で variance が λ^{-2} の独立な (covariance matrix が対角行列である)Gaussian を仮定したときの、posterior likelihood の最大化に対応している。ここでもう少し一般的に

$$Q_\lambda = \sum_{i=1}^{N} \frac{|d_i^{\rm obs} - W_{ij}m_j|^2}{\sigma_i^2} + \lambda^2|\boldsymbol{a} - \hat{\boldsymbol{a}}|^2 \tag{2.72}$$

の最小化を考えてみると、このコスト関数の最小化は、$\boldsymbol{d}$ の誤差が独立に σ_i、$\boldsymbol{a}$ の prior として、平均 $\hat{\boldsymbol{a}}$ で variance が λ^{-2} の独立な Gaussian を仮定したときの、posterior likelihood の最大化に対応すしている。

Tarantola 2005 に従って、これを説明しよう。モデルの prior distribution を $p(\boldsymbol{a})$、尤度を $p(\boldsymbol{d}|\boldsymbol{a})$ とすると、事後分布

$$p(\boldsymbol{a}|\boldsymbol{d}) \propto p(\boldsymbol{d}|\boldsymbol{a})p(\boldsymbol{a}) \tag{2.73}$$

のように書ける。ここで、モデルの prior として Gaussian

$$p(\boldsymbol{a}) = \frac{1}{\sqrt{(2\pi)^M \det\Sigma_{\boldsymbol{a}}}} \exp\left[-\frac{1}{2}(\boldsymbol{a} - \hat{\boldsymbol{a}})^T\Sigma_{\boldsymbol{a}}^{-1}(\boldsymbol{a} - \hat{\boldsymbol{a}})\right], \tag{2.74}$$

を仮定する (ここに $\Sigma_{\boldsymbol{a}}$ はモデルの covariance)。またデータの尤度も Gaussian を仮定する。

$$p(\boldsymbol{d}|\boldsymbol{a}) = \frac{1}{\sqrt{(2\pi)^N \det \Sigma_{\boldsymbol{d}}}} \exp\left[-\frac{1}{2}\boldsymbol{\varepsilon}^T \Sigma_{\boldsymbol{d}}^{-1} \boldsymbol{\varepsilon}\right]. \tag{2.75}$$

すると、

$$p(\boldsymbol{a}|\boldsymbol{d}) \propto \exp\left\{-\frac{1}{2}\left[\boldsymbol{\varepsilon}^T \Sigma_{\boldsymbol{d}}^{-1} \boldsymbol{\varepsilon} + (\boldsymbol{a}-\hat{\boldsymbol{a}})^T \Sigma_{\boldsymbol{a}}^{-1} (\boldsymbol{a}-\hat{\boldsymbol{a}})\right]\right\}. \tag{2.76}$$

となるため、事後分布最大値を探すことは、

$$Q = \boldsymbol{\varepsilon}^T \Sigma_{\boldsymbol{d}}^{-1} \boldsymbol{\varepsilon} + (\boldsymbol{a}-\hat{\boldsymbol{a}})^T \Sigma_{\boldsymbol{a}}^{-1} (\boldsymbol{a}-\hat{\boldsymbol{a}}). \tag{2.77}$$

を最小化する事と等しい。ここでデータもモデルも独立ガウシアンを仮定すると、$(\Sigma_{\boldsymbol{d}})_{ij} = \sigma_i^2 \delta_{ij}$ 、$(\Sigma_{\boldsymbol{a}})_{ij} = \lambda^{-2}\delta_{ij}$ となり、すなわち

$$Q_\lambda = \sum \frac{|d_i - W\boldsymbol{a}|^2}{\sigma_i^2} + \lambda^2 |\boldsymbol{a}-\hat{\boldsymbol{a}}|^2. \tag{2.78}$$

を最小化する事が、事後分布最大化に対応する。この話題は5章でより完全な形で議論する。

2.6 L-curve Criterion

さて Tikhonov regularization において、λ をどのように決めるべきであろうか？この一つの基準として有効なのが L-curve criterion と呼ばれるものである [2]。L-curve とは、モデルのノルム $\xi \equiv |\boldsymbol{a}^{\rm est} - \hat{\boldsymbol{a}}|^2$ と予測誤差 $\rho \equiv |W\,\boldsymbol{a}^{\rm est} - \boldsymbol{d}|^2$ を、λ をパラメタとしてプロットしたものである。図2.3(左) は、L curve の一例である。この例では λ を 0.01 から 10.0 まで動かしプロットしている。

小さい λ では、モデルの分散が大きく、すなわちモデルのノルムも大きい値である (図で左上)。λ を大きくしていくとモデルのノルムは下がっていくが、あるところで下がりが鈍り、予測誤差の急激な増大が始まる (右下方向)。この中間点（図で赤点のあるあたり）が選ぶべき λ であろう。

これは言い換えると、対数空間で $\log\rho$-$\log\xi$ での曲率

$$c(\lambda) \equiv -2\frac{(\log\rho)'(\log\xi)'' - (\log\rho)''(\log\xi)'}{[(\log\rho)'^2 + (\log\xi)'^2]^{3/2}}, \tag{2.79}$$

が最大になるところが最適値であるといえる。ここに $'$ は λ による微分を表す。ここで

$$\xi' = -\frac{4}{\lambda}\sum_{i=1}^{M}[1-w_i(\lambda)]w_i(\lambda)^2 \frac{\boldsymbol{u}^T(\boldsymbol{d}-W\hat{\boldsymbol{a}})}{\kappa_i^2} \tag{2.80}$$

$$\rho' = -\lambda^2 \xi' \tag{2.81}$$

$$w_i(\lambda) \equiv \frac{\kappa_i^2}{\kappa_i^2 + \lambda^2}. \tag{2.82}$$

を利用すると、

$$c(\lambda) = -2\frac{\xi\rho}{\xi'}\frac{(\lambda^2\xi'\rho + 2\lambda\xi\rho + \lambda^4\xi\xi')}{(\lambda^4\xi^2 + \rho^2)^{3/2}}, \tag{2.83}$$

となる。$c(\lambda)$ が最大になる λ を探せばよい。図2.3(右) は曲率 $c(\lambda)$ を示している。曲率の最大をとれば、たしかに L-curve の角にあたる場所を検出できることがわかる。

附属プログラムで L-curve criterion を用いるには以下のように行う。

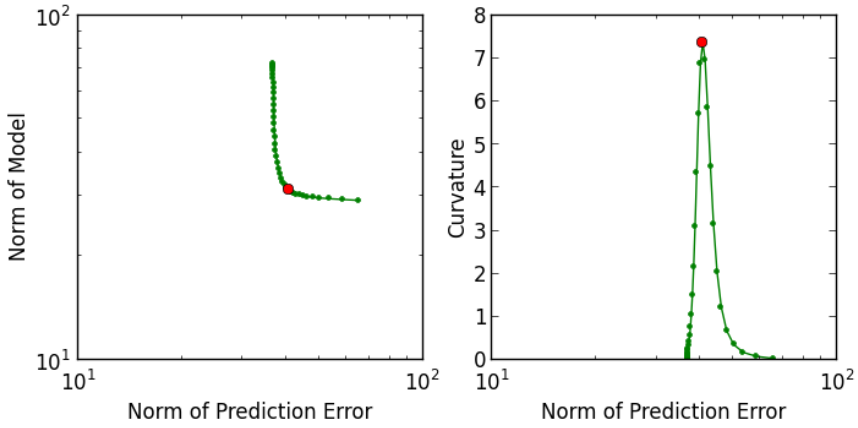

図2.3　L-curve（左）と曲率（右）。曲率が最大の赤い点に対応する λ が L-curve criterion のえらぶ λ である。

```
./random_light.py -f che.png -n 2500 -L 0.01 100.0 -p 0.7 -s 1.0
```

この場合、$\lambda = 0.01$ から $\lambda = 100.0$ まで探索している。

第 3 章

スパース逆問題

to be continued

第 4 章

ヌル空間

第 5 章

ベイズ逆問題

5.1　ベイズ線形逆問題

ここまでは、逆問題を点推定、すなわち一つの解を求めるという考えで解法を構成した。しかし、推定された解がどのくらい信頼できるか評価したいこともある。逆問題は Albert Tarantola により確率論的な観点から再構成された [9]。逆問題を特にここではベイズ統計的に解釈しなおすことで、正則化の意味をより深く理解できるようになり、ガウス過程のようなより一般性のある正則化の導入も容易となる。

線形逆問題

$$\boldsymbol{d} = W\boldsymbol{a} \tag{5.1}$$

をベイズ統計的に考えよう。特に解析的に書けるという点で、尤度や事前分布が多変数正規分布

$$\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \Sigma) = \frac{1}{(2\pi)^{N/2}(\det\Sigma)^{1/2}} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})} \tag{5.2}$$

に従うケースをまず考えたい。

尤度をこの多変数正規分布をもちいて表現すると

$$p(\boldsymbol{d}|\boldsymbol{a}) = \mathcal{N}(\boldsymbol{d}|W\boldsymbol{a}, \Sigma_{\boldsymbol{d}}) \tag{5.3}$$

のようになる。ここに $\Sigma_{\boldsymbol{d}}$ をデータの共分散と呼ぶ。またモデルパラメタ $\boldsymbol{a}$ の事前分布も多変数正規分布でおこう。すなわち

$$p(\boldsymbol{a}) = \mathcal{N}(\boldsymbol{a}|\boldsymbol{0}, \Sigma_{\boldsymbol{a}}), \tag{5.4}$$

となる。ここに $\Sigma_{\boldsymbol{a}}$ はモデルの事前分布の共分散である。共分散の逆行列である精度行列を定義しておこう。

$$\Pi_{\boldsymbol{d}} \equiv \Sigma_{\boldsymbol{d}}^{-1} \tag{5.5}$$

$$\Pi_{\boldsymbol{a}} \equiv \Sigma_{\boldsymbol{a}}^{-1}. \tag{5.6}$$

MAP と点推定

今、尤度とモデルの事前分布ともに多次元正規分布であるので、ベイズの定理

$$p(\boldsymbol{a}|\boldsymbol{d}) = \frac{p(\boldsymbol{d}|\boldsymbol{a})p(\boldsymbol{a})}{p(\boldsymbol{d})} \propto e^{-\frac{1}{2}Q(\boldsymbol{a})} \tag{5.7}$$

より、$\boldsymbol{a}$ の事後分布、$p(\boldsymbol{a}|\boldsymbol{d})$ は $\boldsymbol{a}$ についてやはり多次元正規分布となる。

ここに、コスト関数 $Q(p)$ を計算すると

$$Q(\boldsymbol{a}) = (\boldsymbol{d} - W\boldsymbol{a})^T \Pi_{\boldsymbol{d}} (\boldsymbol{d} - W\boldsymbol{a}) + \boldsymbol{a}^T \Pi_{\boldsymbol{a}} \boldsymbol{a} \tag{5.8}$$

$$= \boldsymbol{a}^T (W^T \Pi_{\boldsymbol{d}} W + \Pi_{\boldsymbol{a}}) \boldsymbol{a} - 2\boldsymbol{a}^T W^T \Pi_{\boldsymbol{d}} \boldsymbol{d} + c, \tag{5.9}$$

となる (c は $\boldsymbol{a}$ に関し定数)。また、モデルの事後分布が最大になる $\boldsymbol{a}$ で定義される A maximum a posterior (MAP) は、$Q(\boldsymbol{a})$ を最小化することで得られる。つまり、式 (5.8) を $\boldsymbol{a}^T$ で微分し 0 とおくことで、

$$\boldsymbol{a}^{\mathrm{MAP}} = (W^T \Pi_{\boldsymbol{d}} W + \Pi_{\boldsymbol{a}})^{-1} W^T \Pi_{\boldsymbol{d}} \boldsymbol{d}. \tag{5.10}$$

となることがわかる。

式 (5.10) と式 (2.57) を比べると、前章の Tikhonov Regularization の解は、$\Sigma_{\boldsymbol{d}} = I$, $\Sigma_{\boldsymbol{a}} = \lambda^{-2} I$ とおくことでベイズ線形逆問題の MAP 解に一致することが確認できる。

事後分布の解析表現

次に $\boldsymbol{a}$ の事後分布、$p(\boldsymbol{a}|\boldsymbol{d})$ を直接求めてみよう。まず、多次元正規分布の負の対数を評価してみよう。式 (5.2) の負の対数を $\boldsymbol{a}$ について展開してみると,

$$-2\log \mathcal{N}(\boldsymbol{a}|\boldsymbol{\mu}, \Sigma) = (\boldsymbol{a} - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{a} - \boldsymbol{\mu}) = \boldsymbol{a}^T \Sigma^{-1} \boldsymbol{a} - 2\boldsymbol{a}^T \Sigma^{-1} \boldsymbol{\mu} + \mathrm{const.} \tag{5.11}$$

が得られる。ここから、もしある多変数正規分布に従うある確率の負の対数が

$$-2\log p(\boldsymbol{a}) = \boldsymbol{a}^T P \boldsymbol{a} - 2\boldsymbol{a}^T \boldsymbol{q} + \mathrm{const}, \tag{5.12}$$

と書けるならば、式 (5.11) と比べることで、その確率が

$$p(\boldsymbol{a}) = \mathcal{N}(\boldsymbol{a}|P^{-1}\boldsymbol{q}, P^{-1}). \tag{5.13}$$

と表される。

さて、事後分布 $p(\boldsymbol{a}|\boldsymbol{d}) \propto p(\boldsymbol{d}|\boldsymbol{a})p(\boldsymbol{a})$ について尤度を式 (5.3) で事前分布を式 (5.4) で与えた場合を考える。この分布の負の対数はコスト関数 (5.8) となるため、式 (5.13) から、事後分布が

$$p(\boldsymbol{a}|\boldsymbol{d}) = \mathcal{N}(\boldsymbol{a}|\boldsymbol{\mu}, \Sigma_{\boldsymbol{a}|\boldsymbol{d}}) \tag{5.14}$$

$$\boldsymbol{\mu} = (W^T \Pi_{\boldsymbol{d}} W + \Pi_{\boldsymbol{a}})^{-1} W^T \Pi_{\boldsymbol{d}} \boldsymbol{d} \tag{5.15}$$

$$\Sigma_{\boldsymbol{a}|\boldsymbol{d}} = (W^T \Pi_{\boldsymbol{d}} W + \Pi_{\boldsymbol{a}})^{-1}. \tag{5.16}$$

となることがわかる。ここから MAP 解が平均解と一致することも確認できる ($\boldsymbol{a}^{\mathrm{MAP}} = \boldsymbol{\mu}$)。

上式では、$N_j \times N_j$ の行列、$\Sigma_{\boldsymbol{a}}$ 及び $(W^T \Pi_{\boldsymbol{d}} W + \Pi_{\boldsymbol{a}})$ の逆行列計算と、$N_i \times N_i$ 行列である $\Sigma_{\boldsymbol{d}}$ の逆行列計算が必要である。逆行列計算は要素数 N の三乗のオーダーの計算量が必要で計算コストが高い。そこで Woodbury matrix identity

$$(Z + UYV)^{-1} = Z^{-1} - Z^{-1}U(Y^{-1} + VZ^{-1}U)^{-1}VZ^{-1}. \tag{5.17}$$

を用いて、

$$(W^T \Pi_{\boldsymbol{d}} W + \Pi_{\boldsymbol{a}})^{-1}$$

$$= \Sigma_{\boldsymbol{a}} - \Sigma_{\boldsymbol{a}} W^T (\Sigma_{\boldsymbol{d}} + W \Sigma_{\boldsymbol{a}} W^T)^{-1} W \Sigma_{\boldsymbol{a}} \tag{5.18}$$

とすることで、$N_i \times N_i$ 行列である $\Sigma_{\boldsymbol{d}}$ 及び $(\Sigma_{\boldsymbol{d}} + W\Sigma_{\boldsymbol{a}}W^T)$ の逆行列計算にのみにすることができる。これはモデルの解像度をデータの数より大きくしたい場合 $(N_j \geq N_i)$ に特に有効である。また、平均は

$$\begin{aligned}\boldsymbol{\mu}_{\boldsymbol{a}|\boldsymbol{d}} &= \Sigma_{\boldsymbol{a}}W^T[I - (\Sigma_{\boldsymbol{d}} + K_W)^{-1}K_W]\Pi_{\boldsymbol{d}}\boldsymbol{d} \\ &= \Sigma_{\boldsymbol{a}}W^T(I + \Pi_{\boldsymbol{d}}W\Sigma_{\boldsymbol{a}}W^T)^{-1}\Pi_{\boldsymbol{d}}\boldsymbol{d} \qquad (5.19) \\ &= \Sigma_{\boldsymbol{a}}W^T\boldsymbol{y} \qquad (5.20)\end{aligned}$$

のようになる。$\boldsymbol{y}$ は、コレスキー分解を用いた線型方程式ソルバーで、

$$\Pi_{\boldsymbol{d}}\boldsymbol{d} = (I + \Pi_{\boldsymbol{d}}W\Sigma_{\boldsymbol{a}}W^T)\boldsymbol{y}, \qquad (5.21)$$

を解くことにより得られる。

5.2 ガウス過程を用いた解の構成

モデル共分散の非対角成分がゼロの時、ベイズ線形逆問題の平均解が、Tikhonov regularization と一致することをみた。共分散行列自体をモデル化することで正則化することができる。これはガウス過程回帰を逆問題に適用しているとみることもできる。つまり以下のように、共分散を $\boldsymbol{a}$ を変数にもつカーネル関数でモデル化する。

$$\Sigma_{\boldsymbol{a}} = K_S(\boldsymbol{a}) \qquad (5.22)$$

カーネル関数としては、たとえば、近いピクセルは同じような値を持つだろうと考え、

$$(K_S)_{jj'} = \alpha k(\eta_{jj'}; \gamma), \qquad (5.23)$$

のように、ピクセル j–j' 間の距離 $\eta_{jj'}$ と距離スケール γ、またカーネルの強度 α を持つ変数でおき、$k(\eta;\gamma)$ としては a radial-basis function (RBF) kernel,

$$k_{\mathrm{RBF}}(\eta;\gamma) = \exp\left(-\frac{\eta^2}{2\gamma^2}\right), \qquad (5.24)$$

を使用することができる。もしくは Matérn -3/2 kernel,

$$k_{\mathrm{M3/2}}(\eta;\gamma) = \left(1 + \frac{\sqrt{3}\eta}{\gamma}\right)e^{-\sqrt{3}\eta/\gamma}. \qquad (5.25)$$

を使用してもよい。ほかにも使用したいモデルに応じて様々なカーネルが提案されている。

5.3 半線形逆問題

半線形逆問題とは、数個のパラメタを固定すると線形逆問題に帰する逆問題である。たとえばモデル共分散カーネルのハイパーパラメタ $\boldsymbol{\theta}_{\boldsymbol{a}}$ も同時に推定したい場合、$\boldsymbol{\theta}_{\boldsymbol{a}}$ については非線形になるので、半線形逆問題となる。逆問題にすべてのパラメタを非線形とみなす非線形逆問題もあるが、一般にパラメタ数が多くなると計算量が増大し、計算不可能となっていく。半線形逆問題では、問題を線形逆問題の部分と非線形逆問題の部分に分け、推定を行っていくことで計算量の問題を回避できる。ここで扱う非線形パラメタとしては $W = W(\boldsymbol{g})$ に含まれるパラメタ $\boldsymbol{g}$、モデルの事前分布内の共分散 $\Sigma_{\boldsymbol{a}} = K_S(\boldsymbol{\theta}_{\boldsymbol{a}})$ 内のハイパーパラメタ $\boldsymbol{\theta}_{\boldsymbol{a}}$、尤度内のデータの共分散のカーネル $\Sigma_{\boldsymbol{d}} = K_D(\boldsymbol{\theta}_{\boldsymbol{d}})$ 内のハイパーパラメタ $\boldsymbol{\theta}_{\boldsymbol{d}}$ が考えられる。

エビデンスと非線形パラメタの周辺事後分布

まず、$\boldsymbol{a}$ について積分した非線形パラメタの周辺尤度もしくはエビデンス $p(\boldsymbol{d}|\boldsymbol{\theta_a}, \boldsymbol{\theta_d}, \boldsymbol{g})$ を計算しよう。事前分布も尤度も多変数正規分布の時エビデンスも多変数正規分布である。ベイズの定理から

$$p(\boldsymbol{a}|\boldsymbol{d}, \boldsymbol{\theta_a}, \boldsymbol{\theta_d}, \boldsymbol{g}) = \frac{p(\boldsymbol{d}|\boldsymbol{a}, \boldsymbol{\theta_a}, \boldsymbol{\theta_d}, \boldsymbol{g})p(\boldsymbol{a}|\boldsymbol{\theta_a}, \boldsymbol{\theta_d}, \boldsymbol{g})}{p(\boldsymbol{d}|\boldsymbol{\theta_a}, \boldsymbol{\theta_d}, \boldsymbol{g})} = \frac{p(\boldsymbol{d}|\boldsymbol{a}, \boldsymbol{\theta_d}, \boldsymbol{g})p(\boldsymbol{a}|\boldsymbol{\theta_a})}{p(\boldsymbol{d}|\boldsymbol{\theta_a}, \boldsymbol{\theta_d}, \boldsymbol{g})}. \tag{5.26}$$

となる。負の対数エビデンスは

$$-2\log p(\boldsymbol{d}|\boldsymbol{\theta_a}, \boldsymbol{\theta_d}, \boldsymbol{g}) = \log p(\boldsymbol{d}|\boldsymbol{a}, \boldsymbol{\theta_d}, \boldsymbol{g}) - \log p(\boldsymbol{a}|\boldsymbol{d}, \boldsymbol{\theta_a}, \boldsymbol{\theta_d}, \boldsymbol{g}) + c \tag{5.27}$$

$$= (\boldsymbol{d} - W\boldsymbol{a})^T \Pi_{\boldsymbol{d}} (\boldsymbol{d} - W\boldsymbol{a})$$

$$-[\boldsymbol{a} - (W^T\Pi_{\boldsymbol{d}}W + \Pi_{\boldsymbol{a}})^{-1}W^T\Pi_{\boldsymbol{d}}\boldsymbol{d}]^T \left(W^T\Pi_{\boldsymbol{d}}W + \Pi_{\boldsymbol{a}}\right)[\boldsymbol{a} - (W^T\Pi_{\boldsymbol{d}}W + \Pi_{\boldsymbol{a}})^{-1}W^T\Pi_{\boldsymbol{d}}\boldsymbol{d}] + c \tag{5.28}$$

$$= \boldsymbol{d}^T \left[\Pi_{\boldsymbol{d}} - \Pi_{\boldsymbol{d}}W(W^T\Pi_{\boldsymbol{d}}W + \Pi_{\boldsymbol{a}})^{-1}W^T\Pi_{\boldsymbol{d}}\right]\boldsymbol{d} + c \tag{5.29}$$

$$= \boldsymbol{d}^T(\Sigma_{\boldsymbol{d}} + W\Sigma_{\boldsymbol{a}}W^T)^{-1}\boldsymbol{d} + c \tag{5.30}$$

となる。ここに Woodbury matrix identity 式（5.17）を用いた。これより

$$p(\boldsymbol{d}|\boldsymbol{\theta_a}, \boldsymbol{\theta_d}, \boldsymbol{g}) = \mathcal{N}(\boldsymbol{d}|\boldsymbol{0}, \Sigma_{\boldsymbol{d}} + W\Sigma_{\boldsymbol{a}}W^T) \tag{5.31}$$

$$= \mathcal{N}(\boldsymbol{d}|\boldsymbol{0}, K_D + K_W) \tag{5.32}$$

となることがわかる。ここにデータ共分散・モデル共分散をそれぞれカーネル $K_D(\boldsymbol{\theta_d})$ と $K_S(\boldsymbol{\theta_a})$ でモデル化したとし、さらに重み付きモデルカーネル

$$K_W(\boldsymbol{\theta_a}, \boldsymbol{g}) \equiv WK_SW^T \tag{5.33}$$

を定義した。

このように非線形パラメタについて解析的に周辺化できるため、非線形パラメタの推定については、一般的に次元数の多い $\boldsymbol{a}$ についてサンプリングする必要がない。すなわち事前分布 $p(\boldsymbol{\theta_a}), p(\boldsymbol{\theta_d}), p(\boldsymbol{g})$ を用いて、周辺事後分布

$$p(\boldsymbol{\theta_a}, \boldsymbol{\theta_d}, \boldsymbol{g}|\boldsymbol{d}) \propto p(\boldsymbol{d}|\boldsymbol{\theta_a}, \boldsymbol{\theta_d}, \boldsymbol{g})p(\boldsymbol{\theta_a})p(\boldsymbol{\theta_d})p(\boldsymbol{g}) \tag{5.34}$$

を MCMC により

$$\boldsymbol{\theta_a}^\dagger, \boldsymbol{\theta_d}^\dagger, \boldsymbol{g}^\dagger \sim p(\boldsymbol{\theta_a}, \boldsymbol{\theta_d}, \boldsymbol{g}|\boldsymbol{d}), \tag{5.35}$$

のようにサンプリングすることができる。

マップの事後分布

式 (5.35) からサンプリングした非線形パラメタをもちいて, $\boldsymbol{a}$ の周辺分布を

$$p(\boldsymbol{a}|\boldsymbol{d}) = \int d\boldsymbol{\theta_a} \int d\boldsymbol{\theta_d} \int d\boldsymbol{g}\, p(\boldsymbol{a}, \boldsymbol{\theta_a}, \boldsymbol{\theta_d}, \boldsymbol{g}|\boldsymbol{d}) \tag{5.36}$$

$$= \int d\boldsymbol{\theta_a} \int d\boldsymbol{\theta_d} \int d\boldsymbol{g}\, p(\boldsymbol{a}|\boldsymbol{d}, \boldsymbol{\theta_a}, \boldsymbol{\theta_d}, \boldsymbol{g})p(\boldsymbol{\theta_a}, \boldsymbol{\theta_d}, \boldsymbol{g}|\boldsymbol{d}) \tag{5.37}$$

$$\approx \frac{1}{N_s}\sum_{n=0}^{N_s-1} p(\boldsymbol{a}|\boldsymbol{d}, \boldsymbol{\theta_a}^\dagger, \boldsymbol{\theta_d}^\dagger, \boldsymbol{g}^\dagger), \tag{5.38}$$

のように近似して求めることができる。ここに N_s はサンプリング数である。

式 (5.38) をもちいると, 任意の $f(\boldsymbol{a})$ の確率 $p(\boldsymbol{a}|\boldsymbol{d})$ の期待値を次のように求めることができる。

$$\langle f(\boldsymbol{a})\rangle \approx \frac{1}{N_s}\sum_{n=0}^{N_s-1}\langle f(\boldsymbol{a})\rangle_{p(\boldsymbol{a}|\boldsymbol{d},\boldsymbol{\theta_a}^\dagger,\boldsymbol{\theta_d}^\dagger,\boldsymbol{g}^\dagger)} \tag{5.39}$$

$$= \frac{1}{N_s}\sum_{n=0}^{N_s-1}\int d\boldsymbol{a} f(\boldsymbol{a}) p(\boldsymbol{a}|\boldsymbol{d},\boldsymbol{\theta_a}^\dagger,\boldsymbol{\theta_d}^\dagger,\boldsymbol{g}^\dagger) \tag{5.40}$$

$$= \frac{1}{N_s}\sum_{n=0}^{N_s-1}\int d\boldsymbol{a} f(\boldsymbol{a}) \mathcal{N}(\boldsymbol{a}|\boldsymbol{\mu}_{\boldsymbol{a}|\boldsymbol{d},\boldsymbol{\theta_a}^\dagger,\boldsymbol{\theta_d}^\dagger,\boldsymbol{g}^\dagger}, \Sigma_{\boldsymbol{a}|\boldsymbol{d},\boldsymbol{\theta_a}^\dagger,\boldsymbol{\theta_d}^\dagger,\boldsymbol{g}^\dagger}) \tag{5.41}$$

ここに $\langle f\rangle_P$ は確率変数 f の確率 P での期待値である. $P = p(\boldsymbol{a}|\boldsymbol{d})$ の場合は下付きを省略しよう。つまり,$\langle\cdot\rangle \equiv \langle\cdot\rangle_{p(\boldsymbol{a}|\boldsymbol{d})}$ である. たとえば、$\boldsymbol{a}$ の周辺分布の平均は

$$\boldsymbol{\mu}_{\boldsymbol{a}|\boldsymbol{d}} = \langle\boldsymbol{a}\rangle \approx \frac{1}{N_s}\sum_{n=0}^{N_s-1}\langle\boldsymbol{a}\rangle_{p(\boldsymbol{a}|\boldsymbol{d},\boldsymbol{\theta_a}^\dagger,\boldsymbol{\theta_d}^\dagger,\boldsymbol{g}^\dagger)} \tag{5.42}$$

$$= \frac{1}{N_s}\sum_{n=0}^{N_s-1}\boldsymbol{\mu}_{\boldsymbol{a}|\boldsymbol{d},\boldsymbol{\theta_a}^\dagger,\boldsymbol{\theta_d}^\dagger,\boldsymbol{g}^\dagger}. \tag{5.43}$$

のように書ける。式 (5.19) を代入して、データ共分散、モデル共分散をそれぞれカーネルでモデル化したし、まとめると

$$\langle\boldsymbol{a}\rangle = \frac{1}{N_s}\sum_{n=0}^{N_s-1} K_S(\boldsymbol{\theta_a}^\dagger) W(\boldsymbol{g}^\dagger)^T \boldsymbol{y}_n \tag{5.44}$$

$$\boldsymbol{y}_n = [I + K_D^{-1}(\boldsymbol{\theta_d}) K_W(\boldsymbol{\theta_a}^\dagger, \boldsymbol{g}^\dagger)]^{-1} K_D^{-1}(\boldsymbol{\theta_d})\boldsymbol{d}$$

のようになる。

第 6 章

次元拡張された線形逆問題

6.1　拡張次元線形問題の同相写像表現

ここまでは線形逆問題として

$$d_i = \sum_j W_{ij} a_j \tag{6.1}$$

というタイプのものを考えてきた。この形式ではモデル $\boldsymbol{a}$ は i 方向の自由度を持っていない。i が時間のインデックス、j が空間方向のインデックスとすると、モデル $\boldsymbol{a}$ は時間方向の自由度を持てない。そこで本章ではモデル $\boldsymbol{a}$ の次元を i 方向にも拡張して、

$$A = \{A_{ij}\}, \tag{6.2}$$

のようなモデルにすることを考える。式 (6.1) にこのような拡張をすると

$$d_i = \sum_j W_{ij} A_{ij} \tag{6.3}$$

となる。ベクトル形式での表記も定義すると

$$\boldsymbol{d} = \boldsymbol{\psi}(W, A) \tag{6.4}$$

となる。ここに $\boldsymbol{\psi} = \boldsymbol{\psi}(W, A)$ は $\psi_i = \sum_j W_{ij} A_{ij}$ という演算子で、言い換えると $\boldsymbol{\psi}(W, A)$ は WA^T の対角成分をベクトルとして取り出す演算子である。

式 (6.4) は通常の線形逆問題の形式ではない。しかし、以下のように W の次元を $\mathbb{R}^{N_i \times N_j}$ から $\mathbb{R}^{N_i \times N_i N_j}$ 拡張し A の同相写像を用いることで、通常の線形逆問題の式となる。拡張された W は

$$\tilde{W} = \begin{pmatrix} \mathcal{D}(\boldsymbol{w}_0) & \mathcal{D}(\boldsymbol{w}_1) & \cdots & \mathcal{D}(\boldsymbol{w}_{N_j-1}) \end{pmatrix} \in \mathbb{R}^{N_i \times N_i N_j} \tag{6.5}$$

である。ここに $\boldsymbol{w}_j$ は W の列を取り出した列ベクトルである。$\mathcal{D}(\boldsymbol{w}_j)$ はベクトル $\boldsymbol{w}_j$ を対角成分にもつ対角行列をつくる演算子、すなわち $\acute{W} = \mathcal{D}(\boldsymbol{w}_j)$ として $\acute{W}_{ij} = \delta_{ij} \boldsymbol{w}_j$ となるような演算子である。ここに δ_{ij} はクロネッカーデルタである。、また A の同相なベクトルを次のように定義する。

$$\boldsymbol{a} = \mathrm{vec}(A) \equiv \begin{pmatrix} \hat{\boldsymbol{a}}_0 \\ \hat{\boldsymbol{a}}_1 \\ \vdots \\ \hat{\boldsymbol{a}}_{N_j-1} \end{pmatrix} \in \mathbb{R}^{N_i N_j}, \tag{6.6}$$

ここに $\mathrm{vec}(A)$ は A をベクトル化したものであり $\hat{\boldsymbol{a}}_j$ は A の列からなる列ベクトルである。つまり

$$\hat{\boldsymbol{a}}_j \equiv (A_{0j}, A_{1j}, \ldots, A_{(N_i-1)j})^T. \tag{6.7}$$

である。次元拡大した $\tilde{W}$ と同相ベクトル $\boldsymbol{a}$ をもちいて、式 (6.4) は

$$\boldsymbol{d} = \tilde{W}\boldsymbol{a}. \tag{6.8}$$

となり、離散線形逆問題の形式となる。式 (6.4) と式 (6.8) の同値関係を添字の形式で表示すると

$$\psi_i = \sum_j W_{ij}A_{ij} = \sum_J \tilde{W}_{iJ}(\mathrm{vec}(A))_J \tag{6.9}$$

となる。ここに J は (i,j) を展開した添字である。

ここで、もう少し一般的に、テンソル A の形状変換を $\mathrm{reshape}^{(p\to q)}(A)$ で記述しておこう。ここに p と q は形状変換前後の形状である。例えば行列 A のベクトル化、及びその（再）行列化は以下のように表すことができる。

$$\mathrm{vec}(A) = \mathrm{reshape}^{(N_i\times N_j \to N_iN_j)}(A) = \boldsymbol{a} \in \mathbb{R}^{N_iN_j} \tag{6.10}$$
$$\mathrm{mat}(\boldsymbol{a}) = \mathrm{reshape}^{(N_iN_j \to N_i\times N_j)}(\boldsymbol{a}) = A \in \mathbb{R}^{N_i\times N_j} \tag{6.11}$$

この reshape の形式を利用することで、式 (6.9) は、例えばテンソル

$$A = \mathrm{reshape}^{(N_i\times N_j\times N_k \to N_iN_j\times N_k)}(\mathcal{A}). \tag{6.12}$$

に対し、

$$\sum_j W_{ij}\mathcal{A}_{ijk} = \sum_J \tilde{W}_{iJ}A_{Jk} \tag{6.13}$$

のように拡張できる。

6.2　グランドカーネルと再収縮公式

さて $\boldsymbol{a}$ の事前分布を

$$p(\boldsymbol{a}|\boldsymbol{\theta}_{\boldsymbol{a}}) = \mathcal{N}(\boldsymbol{a}|\boldsymbol{0}, \Sigma_{\boldsymbol{a}}), \tag{6.14}$$

とおく。ここに $\Sigma_{\boldsymbol{a}}$ は $\boldsymbol{a}$ の共分散である．ここで、モデルの共分散 $\Sigma_{\boldsymbol{a}}$ をグランドカーネル

$$K_{iji'j'} = \alpha k(\eta_{jj'};\gamma)k(|t_i - t_{i'}|;\tau), \tag{6.15}$$

の形でモデル化しよう。ここに α はグランドカーネルの強度、γ は j 方向の相関長（もし j を空間方向とするなら空間スケール)、τ は i 方向の相関長 (i を時間方向とするなら時間スケール) である。式 (6.15) はクロネッカー積 $\otimes$ を用いて

$$K = \alpha K_S \otimes K_T, \tag{6.16}$$

と簡潔に書ける。ここに

$$(K_S)_{jj'} = k(\eta_{jj'};\gamma) \tag{6.17}$$
$$(K_T)_{ii'} = k(|t_i - t_{i'}|;\tau). \tag{6.18}$$

である。グランドカーネル (6.16) のハイパーパラメタは $\boldsymbol{\theta_a} = (\gamma, \alpha, \tau)^T$ である。次元拡張した線形逆問題では、i と j の相関をクロネッカー積でモデル化する場合は、後にみるように次元を再収縮したコンパクトな形で解を構成できるのが特徴である。

線形逆問題 (6.8) より尤度は

$$p(\boldsymbol{d}|\boldsymbol{a}, \boldsymbol{g}) = \mathcal{N}(\boldsymbol{d}|\tilde{W}\boldsymbol{a}, \Sigma_{\boldsymbol{d}}), \tag{6.19}$$

で与えられる。ここに $\Sigma_{\boldsymbol{d}}$　はデータの共分散である。データの共分散もカーネル $K_D(\boldsymbol{\theta_d})$ でモデル化しておく。

前章のベイズ線形逆問題の結果から（式 (5.14) 参照)、$\boldsymbol{d}, \boldsymbol{\theta_a}, \boldsymbol{\theta_d}, \boldsymbol{g}$ の条件付きの $\boldsymbol{a}$ の事後分布は is

$$p(\boldsymbol{a}|\boldsymbol{d}, \boldsymbol{\theta_a}, \boldsymbol{\theta_d}, \boldsymbol{g}) = \mathcal{N}(\boldsymbol{a}|\boldsymbol{\mu}_{\boldsymbol{a}|\boldsymbol{d},\boldsymbol{\theta},\boldsymbol{g}}, \Sigma_{\boldsymbol{a}|\boldsymbol{d},\boldsymbol{\theta},\boldsymbol{g}}) \tag{6.20}$$

$$\boldsymbol{\mu}_{\boldsymbol{a}|\boldsymbol{d},\boldsymbol{\theta},\boldsymbol{g}} = (\tilde{W}^T\Pi_{\boldsymbol{d}}\tilde{W} + K^{-1})^{-1}\tilde{W}^T\Pi_{\boldsymbol{d}}\boldsymbol{d} \tag{6.21}$$

$$\Sigma_{\boldsymbol{a}|\boldsymbol{d},\boldsymbol{\theta},\boldsymbol{g}} = (\tilde{W}^T\Pi_{\boldsymbol{d}}\tilde{W} + K^{-1})^{-1}. \tag{6.22}$$

となる。

式 (6.21) を計算するには、計算量とメモリの両方に問題がある。前者は式 (6.21) 中の逆行列

$$(\tilde{W}^T\Pi_{\boldsymbol{d}}\tilde{W} + K^{-1})^{-1} \in \mathbb{R}^{N_iN_j \times N_iN_j}, \tag{6.23}$$

に $\mathcal{O}(N_i^3N_j^3)$ の計算量を要することである。また式 (6.21) をメモリに格納するには $\mathcal{O}(N_i^2N_j^2)$ のサイズが必要であり、これは $N_i = 10^3$ 及び $N_j = 10^3$ の場合、数十テラバイトが必要である。

まず計算量の問題については、式 (5.18) と同様の変形を用いれば逆行列の計算は $N_i \times N_i$ 行列に対してのものにすることができる。すなわち

$$\begin{aligned}\Sigma_{\boldsymbol{a}|\boldsymbol{d},\boldsymbol{\theta},\boldsymbol{g}} &= (\tilde{W}^T\Pi_{\boldsymbol{d}}\tilde{W} + K^{-1})^{-1} \\ &= K - K\tilde{W}^T(\Sigma_{\boldsymbol{d}} + K_W)^{-1}\tilde{W}K\end{aligned} \tag{6.24}$$

となる。ここに $K_W \in \mathbb{R}^{N_i \times N_i}$ は、重みつきカーネル

$$K_W \equiv \tilde{W}K\tilde{W}^T = \alpha\tilde{W}(K_S \otimes K_T)\tilde{W}^T \tag{6.25}$$

である。

再収縮公式 1

K_W 内の行列 $(K_S \otimes K_T)$ には $\mathcal{O}(N_i^2N_j^2)$ のメモリサイズが必要なため圧縮した表現を得たい。式 (6.25) は、要素積 $\odot$ を用いて

$$K_W = \alpha\tilde{W}(K_S \otimes K_T)\tilde{W}^T = \alpha K_T \odot (WK_SW^T), \tag{6.26}$$

となる。これを証明する。右辺行列の要素 ii' を $Y_{ii'}$ で表す。この量は、要素が $\mathcal{P}_{iji'j'} \equiv (K_S)_{jj'}(K_T)_{ii'}$ で表されるテンソル $\mathcal{P} \in \mathbb{R}^{N_i \times N_j \times N_{i'} \times N_{j'}}$ を用いて

$$Y_{ii'} = (K_T)_{ii'}\sum_{j,j'} W_{ij}(K_S)_{jj'}W_{i'j'} = \sum_j W_{ij}\sum_{j'}\mathcal{P}_{iji'j'}W_{i'j'} = \sum_j W_{ij}\mathcal{Q}_{iji'} \tag{6.27}$$

と表される。ここに

$$\mathcal{Q}_{iji'} \equiv \sum_{j'}\mathcal{P}_{iji'j'}W_{i'j'}. \tag{6.28}$$

である。

さて式 (6.13) を、同相写像表現を式 (6.27) に適用して書き換える。まず $\mathcal{Q}$ に対応する同相写像は

$$Q = \mathrm{reshape}^{(N_i \times N_j \times N_{i'} \to N_i N_j \times N_{i'})}(\mathcal{Q}) = \sum_{j'} \mathcal{P}^*_{Ji'j'} W_{i'j'} = \sum_{j'} \mathcal{P}^*_{Ji'j'} W^T_{j'i'} = \sum_{J'} \mathcal{P}^{**}_{JJ'} \tilde{W}^T_{J'i'} \tag{6.29}$$

である。ここに

$$\mathcal{P}^* = \mathrm{reshape}^{(N_i \times N_j \times N_{i'} \times N_{j'} \to N_i N_j \times N_{i'} \times N_{j'})}(\mathcal{P}) \tag{6.30}$$

$$\mathcal{P}^{**} = \mathrm{reshape}^{(N_i N_j \times N_{i'} \times N_{j'} \to N_i N_j \times N_{i'} N_{j'})}(\mathcal{P}^*) \tag{6.31}$$

$$= \mathrm{reshape}^{(N_i \times N_j \times N_{i'} \times N_{j'} \to N_i N_j \times N_{i'} N_{j'})}(\mathcal{P}) = K_S \otimes K_T, \tag{6.32}$$

である。これより

$$Y_{ii'} = \sum_j W_{ij} \mathcal{Q}_{iji'} = \sum_J \tilde{W}_{iJ} Q_{Ji'} = \sum_J \tilde{W}_{iJ} \sum_{J'} P^{**}_{JJ'} \tilde{W}^T_{J'i'} = \sum_{J,J'} \tilde{W}_{iJ} (S \otimes T)_{JJ'} \tilde{W}^T_{J'i'}. \tag{6.33}$$

が得られ、式 (6.26) が証明される。

再収縮公式 2

次に式 (6.21) の同相写像表現を用いてコンパクトな形式にする。まず式 (5.19) と同様に, 式 (6.21) は

$$\boldsymbol{\mu}_{\boldsymbol{a}|\boldsymbol{d},\boldsymbol{\theta},\boldsymbol{g}} = \alpha (K_S \otimes K_T) \tilde{W}^T (I + \Pi_{\boldsymbol{d}} K_W)^{-1} \Pi_{\boldsymbol{d}} \boldsymbol{d}. \tag{6.34}$$

の形に変形できる。

さて公式

$$(V^T \otimes U)\mathrm{vec}(X) = \mathrm{vec}(UXV), \tag{6.35}$$

に $V = K_S^T$, $U = K_T$, そして $X = \mathrm{mat}(\tilde{W}^T \boldsymbol{y})$ を当てはめることで、

$$(K_S \otimes K_T)\tilde{W}^T \boldsymbol{x} = \mathrm{vec}(K_T \,\mathrm{mat}(\tilde{W}^T \boldsymbol{y})\, K_S^T). \tag{6.36}$$

が得られる。ここで

$$\tilde{W}^T \boldsymbol{x} = \begin{pmatrix} \boldsymbol{w}_0 \odot \boldsymbol{x} \\ \boldsymbol{w}_1 \odot \boldsymbol{x} \\ \vdots \\ \boldsymbol{w}_{N_j - 1} \odot \boldsymbol{x} \end{pmatrix} \in \mathbb{R}^{N_i N_j}, \tag{6.37}$$

となるので、

$$\mathrm{mat}(\tilde{W}^T \boldsymbol{y}) = \begin{pmatrix} \boldsymbol{w}_0 \odot \boldsymbol{y}, & \boldsymbol{w}_1 \odot \boldsymbol{y}, & \cdots, & \boldsymbol{w}_{N_j - 1} \odot \boldsymbol{y} \end{pmatrix} = \mathcal{D}(\boldsymbol{y}) W \tag{6.38}$$

が得られる。式 (6.36) は

$$(K_S \otimes K_T)\tilde{W}^T \boldsymbol{y} = \mathrm{vec}(K_T \mathcal{D}(\boldsymbol{y}) W K_S^T). \tag{6.39}$$

となる。$K_S^T = K_S$ であるので、式 (6.21) の同相写像表現をまとめると

$$\begin{aligned} A^* &= \alpha K_T \mathcal{D}(\boldsymbol{y}) W K_S \\ \boldsymbol{y} &\equiv (I + \Pi_{\boldsymbol{d}} K_W)^{-1} \Pi_{\boldsymbol{d}} \boldsymbol{d} \\ K_W &\equiv \alpha K_T \odot (W K_S W^T), \end{aligned} \tag{6.40}$$

となる。$\mathcal{D}(\boldsymbol{y})W$ を成分表示すると単に

$$(\mathcal{D}(\boldsymbol{y})W)_{ij} = W_{ij}y_i. \tag{6.41}$$

となっている。

再収縮公式 3

事後分布 (6.20) 全部をサンプリングするのは計算量的に大変である。そこで i ごともしくは j ごとに周辺化した分布を求めることを考えよう。i が時間方向の場合、これはスナップショットの周辺事後分布ということになる。A の i 番目の行をとりだした列ベクトルを以下のように定義する。

$$\check{\boldsymbol{a}}_i \equiv (A_{i0}, A_{i1}, \ldots, A_{i(N_j-1)})^T, \tag{6.42}$$

j 方向に取り出す場合は、以前定義した

$$\hat{\boldsymbol{a}}_j \equiv (A_{0j}, A_{1j}, \ldots, A_{(N_i-1)j})^T. \tag{6.43}$$

を考える。つまり目標は式 (6.20) を $\check{\boldsymbol{a}}_i$ もしくは $\hat{\boldsymbol{a}}_j$ に対する周辺事後分布の表現に書き換えることである。

実は、多次元ガウス分布の場合、$\boldsymbol{a}$ の成分のある部分集合からなるベクトル $\boldsymbol{b}$ に対する周辺分布は、$\boldsymbol{b}$ に対応する部分共分散行列と平均を抜きだし構成した多次元ガウス分布である []。そのためにまず行列から i にかかわる成分を抽出する $\mathcal{S}_i$ 抽出演算子と j に関する部分を抽出する $\mathcal{T}_j$ 抽出演算子を定義しよう。$\mathcal{S}_i$ 抽出演算子は $N_iN_j \times N_iN_j$ 行列から i 成分を抽出する演算子である。すなわち、$K_S \otimes K_T \in \mathbb{R}^{N_iN_j \times N_iN_j}$ に対して適用すると

$$\mathcal{S}_i(K_S \otimes K_T) = (K_T)_{ii}K_S. \tag{6.44}$$

となるような演算子である。つまり $\mathcal{S}_i(X)$ は X から添え字からなる行列 $\boldsymbol{J}_i \otimes \boldsymbol{J}_i^T \in \mathbb{R}^{N_j \times N_j}$ の成分を取り出したものである。ここに $\boldsymbol{J}_i = (i, i+N_i, i+2N_i, \cdots, i+(N_j-1)N_i)^T$ である。同様に $\mathcal{T}_j$ 抽出演算子は

$$\mathcal{T}_j(K_S \otimes K_T) = (K_S)_{jj}K_T, \tag{6.45}$$

となる演算子で、すなわち $\mathcal{T}_j(X)$ は、X から添え字行列 $\boldsymbol{I}_j \otimes \boldsymbol{I}_j^T \in \mathbb{R}^{N_i \times N_i}$ を抽出する演算子である。ここに $\boldsymbol{I}_j = (jN_i, 1+jN_i, \cdots, (N_i-1)+jN_i)^T$ である。

さて、$\mathcal{S}_i$ 抽出演算子と $\mathcal{T}_j$ 抽出演算子を共分散 (6.24) に適用することを考える。$\mathcal{S}_i$ 抽出演算子と $\mathcal{T}_j$ 抽出演算子は線形演算子（$\mathcal{S}_i(X+Y) = \mathcal{S}_i(X) + \mathcal{S}_i(Y)$、$\mathcal{S}_i(\alpha X) = \alpha\mathcal{S}_i(X)$ であるので、

$$\mathcal{S}_i(\Sigma_{\boldsymbol{a}|\boldsymbol{d},\boldsymbol{\theta},\boldsymbol{g}}) = \alpha\mathcal{S}_i(K_S \otimes K_T) - \mathcal{S}_i[K\tilde{W}^T(\Sigma_{\boldsymbol{d}} + K_W)^{-1}\tilde{W}K] \tag{6.46}$$

$$= (K_T)_{ii}K_S - \mathcal{S}_i[Z^TPZ] \tag{6.47}$$

となる。ここに $Z = \tilde{W}K$、対称行列 $P = (\Sigma_{\boldsymbol{d}} + K_W)^{-1}$ である。右辺第二項を計算しよう。

Then, let us consider $\mathcal{S}_i(Y)$ and $\mathcal{T}_j(Y)$ for

$$Y = Z^TPZ \tag{6.48}$$

$$Z \equiv \tilde{W}(S \otimes T) \in \mathbb{R}^{N_i \times N_iN_j} \tag{6.49}$$

where P is a square matrix. The element of Y is given by

$$Y_{JJ'} = \hat{\boldsymbol{z}}_J^T P \hat{\boldsymbol{z}}_{J'}, \tag{6.50}$$

where $\hat{\boldsymbol{z}}_J$ is the column vector of the column of Z. Because Z can be expressed as

$$Z = \left(\begin{array}{c} Z'[0]Z'[1]\cdots Z'[N_j-1] \end{array} \right) \tag{6.51}$$

where

$$Z'[j] = \sum_k S_{kj}\mathcal{D}(\boldsymbol{w}_k)T \tag{6.52}$$

we obtain

$$\hat{\boldsymbol{z}}^T_{i+jN_j} P \hat{\boldsymbol{z}}_{i'+j'N_j} = \sum_{ll'} \left[\left(\sum_k S_{kj} W_{lk} T_{li} \right) P_{ll'} \left(\sum_k S_{kj'} W_{l'k} T_{l'i'} \right) \right] \tag{6.53}$$

$$= \sum_{ll'} \left\{ \left[\left(\sum_k W_{lk} S_{kj} \right) T_{li} \right] P_{ll'} \left[\left(\sum_k W_{l'k} S_{kj'} \right) T_{l'i'} \right] \right\} \tag{6.54}$$

Then, extracting $(\boldsymbol{I}_i \times \boldsymbol{I}_i)$ from Y, the S extractor is expressed as follows:

$$\mathcal{S}_i(Y) = [\mathcal{D}(\hat{\boldsymbol{t}}_i)WS]^T P [\mathcal{D}(\hat{\boldsymbol{t}}_i)WS], \tag{6.55}$$

where $\hat{\boldsymbol{t}}_i$ is the column vector of the column of T. Likewise, extracting $(\boldsymbol{J}_j \times \boldsymbol{J}_j)$ from Y, we also obtain

$$\mathcal{T}_j(Y) = (\mathcal{D}(\hat{\boldsymbol{u}}_j)T)^T P (\mathcal{D}(\hat{\boldsymbol{u}}_j)T), \tag{6.56}$$

where $\hat{\boldsymbol{u}}_j$ is the column vector of the column of WS.

Snapshot Given $\boldsymbol{g}$ and $\boldsymbol{\theta}$

$$p(\check{\boldsymbol{a}}_i|\boldsymbol{d},\boldsymbol{\theta},\boldsymbol{g}) = \mathcal{N}(\check{\boldsymbol{a}}_i|\check{\boldsymbol{a}}^*_i, \Sigma_{\check{\boldsymbol{a}}_i|\boldsymbol{d},\boldsymbol{\theta},\boldsymbol{g}}) \tag{6.57}$$

where

$$\check{\boldsymbol{a}}^*_i \equiv (A^*_{i0}, A^*_{i1}, \ldots, A^*_{i(N_j-1)})^T, \tag{6.58}$$

is the snapshot of A^* at time of t_i, and

$$\Sigma_{\check{\boldsymbol{a}}_i|\boldsymbol{d},\boldsymbol{\theta},\boldsymbol{g}} = \alpha(K_T)_{ii}K_S - B_i^T(\Sigma_{\boldsymbol{d}} + K_W)^{-1}B_i, \tag{6.59}$$

is the snapshot covariance derived by adopting the $\mathcal{S}$ extractor in Appendix **??** into Equation (6.24) with

$$B_i = \alpha(WK_S) \bullet \hat{\boldsymbol{t}}_i \tag{6.60}$$

$$\hat{\boldsymbol{t}}_i \equiv ((K_T)_{i0}, \ldots, (K_T)_{i(N_i-1)})^T \tag{6.61}$$

In the above form of a posterior snapshot, we require a memory size of $\mathcal{O}(N_i^2)$ or $\mathcal{O}(N_j^2)$ for each snapshot.

Likewise, we can also consider the posterior for the time-series of the j-th pixel using the $\mathcal{T}$ extractor defined in Appendix **??** as follows:

Pixel-wise Evolution Given $\boldsymbol{g}$ and $\boldsymbol{\theta}$

$$p(\hat{\boldsymbol{a}}_j|\boldsymbol{d},\boldsymbol{\theta},\boldsymbol{g}) = \mathcal{N}(\hat{\boldsymbol{a}}_j|\hat{\boldsymbol{a}}^*_j, \Sigma_{\hat{\boldsymbol{a}}_j|\boldsymbol{d},\boldsymbol{\theta},\boldsymbol{g}}) \tag{6.62}$$

where

$$\hat{\boldsymbol{a}}_j^* \equiv (A_{0j}^*, A_{1j}^*, \ldots, A_{(N_i-1)j}^*)^T, \tag{6.63}$$

is the pixel-wise evolution of A^* at the j-th pixel, and

$$\Sigma_{\hat{\boldsymbol{a}}_j|\boldsymbol{d},\boldsymbol{\theta},\boldsymbol{g}} = \alpha(K_S)_{jj}K_T - C_i^T(\Sigma_{\boldsymbol{d}} + K_W)^{-1}C_i, \tag{6.64}$$

is the pixel-wise covariance derived by adopting the T extractor in Appendix **??** into Equation (6.24) with

$$C_i = \alpha K_T \bullet \hat{\boldsymbol{u}}_j \tag{6.65}$$

$$\hat{\boldsymbol{u}}_j \equiv ((WK_S)_{0j}, \ldots, (WK_S)_{(N_i-1)j})^T \tag{6.66}$$

Equation (6.62) also requires a memory size of $\mathcal{O}(N_i^2)$ or $\mathcal{O}(N_j^2)$ for each pixel. Either Equation (6.57) or (6.62) can be used to compute the posterior of Equation (6.20) depending on the specific purpose. In the following discussion, we use the snapshot posterior without a loss of generality.

The evidence of dynamic mapping is derived through the same procedure for deriving Equations (**??**) and (**??**), namely,

$$p(\boldsymbol{d}|\boldsymbol{\theta},\boldsymbol{g}) = \mathcal{N}(\boldsymbol{d}|\boldsymbol{0}, \Sigma_{\boldsymbol{d}} + \tilde{W}\Sigma_{\boldsymbol{a}}\tilde{W}^T) \tag{6.67}$$

$$= \mathcal{N}(\boldsymbol{d}|\boldsymbol{0}, \Sigma_{\boldsymbol{d}} + K_W). \tag{6.68}$$

Interestingly, the computational cost of Equation (6.68) is almost the same as that for the static mapping because $\Sigma_{\boldsymbol{d}} + K_W \in \mathbb{R}^{N_i \times N_i}$. The analytic form of $p(\boldsymbol{d}|\boldsymbol{\theta},\boldsymbol{g})$ allows us to efficiently sample

$$\boldsymbol{\theta}^\dagger, \boldsymbol{g}^\dagger \sim p(\boldsymbol{\theta},\boldsymbol{g}|\boldsymbol{d}) \propto p(\boldsymbol{d}|\boldsymbol{\theta},\boldsymbol{g})p(\boldsymbol{\theta})p(\boldsymbol{g}), \tag{6.69}$$

using e.g., an MCMC algorithm.

Using the sample of $\boldsymbol{\theta}^\dagger$ and $\boldsymbol{g}^\dagger$, the marginal posterior of the dynamic map can be approximated as follows:

$$p(\check{\boldsymbol{a}}_i|\boldsymbol{d}) \approx \frac{1}{N_s}\sum_{n=0}^{N_s-1} p(\check{\boldsymbol{a}}_i|\boldsymbol{d},\boldsymbol{\theta}^\dagger,\boldsymbol{g}^\dagger). \tag{6.70}$$

In addition, the summary statistics are

$$\langle f(\check{\boldsymbol{a}}_i)\rangle \approx \frac{1}{N_s}\sum_{n=0}^{N_s-1}\int d\boldsymbol{a} f(\check{\boldsymbol{a}}_i)\mathcal{N}(\check{\boldsymbol{a}}_i|(\check{\boldsymbol{a}}_i^*)_n^\dagger, \Sigma_{\check{\boldsymbol{a}}_i|\boldsymbol{d},\boldsymbol{\theta}^\dagger,\boldsymbol{g}^\dagger}), \tag{6.71}$$

where $(\check{\boldsymbol{a}}_i^*)_n^\dagger$ is the snapshot of A^* given $\boldsymbol{\theta} = \boldsymbol{\theta}^\dagger$ and $\boldsymbol{g} = \boldsymbol{g}^\dagger$. The mean of the marginal snapshot for a dynamic geography is given by the following:

$$\langle \check{\boldsymbol{a}}_i\rangle \approx \frac{1}{N_s}\sum_{n=0}^{N_s-1}(\check{\boldsymbol{a}}_i^*)_n^\dagger. \tag{6.72}$$

Reshaping $\check{\boldsymbol{a}}_i$ and $\check{\boldsymbol{a}}_i^*$ in Equation (6.72) to A and A^*, we find the mean geography matrix as

Mean Map Matrix for Dynamic Geography

$$\langle A\rangle \approx \frac{1}{N_s}\sum_{n=0}^{N_s-1}\alpha_n^\dagger(K_T(\tau_n^\dagger)\bullet \boldsymbol{y}_n)W(\boldsymbol{g}^\dagger)K_S(\gamma_n^\dagger), \tag{6.73}$$

where

$$\boldsymbol{y}_n \equiv [I + \Pi_{\boldsymbol{d}}(K_W)_n]^{-1}\Pi_{\boldsymbol{d}}\boldsymbol{d},$$
$$(K_W)_n \equiv \alpha_n^\dagger K_T(\tau_n^\dagger) \odot [W(\boldsymbol{g}^\dagger)K_S(\gamma_n^\dagger)W(\boldsymbol{g}^\dagger)^T],$$

and $\{\boldsymbol{\theta}^\dagger = (\gamma_n^\dagger, \alpha_n^\dagger, \tau_n^\dagger)^T, \boldsymbol{g}_n^\dagger\}$ is the n-th set of hyperparameters sampled from $p(\boldsymbol{\theta}, \boldsymbol{g}|\boldsymbol{d})$ (Equation 6.69).

第 7 章

行列分解形式の逆問題

7.1　行列分解型の逆問題

行列分解とは、行列 D を行列 A と X の行列積に分解する手続きである。すなわち、

$$D = AX \tag{7.1}$$

もしくは要素を用いて

$$D_{ij} = \sum_k A_{ik} X_{kj} \tag{7.2}$$

と分解することである。ほとんどの場合、この分解は一意でない。例えば正則な行列 G を用いて、

$$D = A\,X = A'X' \tag{7.3}$$
$$A' = A\,G \tag{7.4}$$
$$X' = G^{-1}X \tag{7.5}$$

のように新たな分解 A'、G' を簡単に生成できる。そこで、適当な条件を課して分解の自由度を小さくすることが考えられる。このような行列分解の種類として、主成分分析 (PCA)、独立成分分析 (ICA)、そして非負行列分解 (Nonnegative Matrix Factorization; 以下 NMF) などがあげられる。ここでは NMF を用いた行列分解型の逆問題を紹介しよう。NMF の条件は、分解した行列の各要素が非負であること、すなわち

$$D = AX \quad \text{subject to } A_{ik} \geq 0, X_{kj} \geq 0 \tag{7.6}$$

である。通常は入力行列 D の各要素も非負であるとする。

さて前章にならってデータと AX の間にデザイン行列 W が作用している形式を考えよう。つまり

$$D = WAX \tag{7.7}$$

の形式となり、行列分解型の逆問題とは D と W が与えられたときに A と X を求める問題となる。この形式を行列分解型の逆問題と呼ぼう [3]。非負値行列分解型の逆問題もコスト関数の最小化により解を求めることができる。コスト関数としてはカルバック・ライブラーダイバージェンスや二乗ユークリッド距離が利用される。ここでは、これまでの章との接続を考えて、後者を用いることにする。つまり

$$\text{minimize } Q = \frac{1}{2}||D - WAX||_F^2 \tag{7.8}$$
$$\text{subject to } A_{jk} \geq 0, X_{kl} \geq 0. \tag{7.9}$$

の最小化により NMF を実現する。ここに $||\cdot||_F^2$ は二乗したフロベニウスノルム

$$||Y||_F^2 \equiv \sum_j \sum_i Y_{ij}^2. \tag{7.10}$$

である。

Multiplicative Update

NMF を世に広く知らしめた Lee & Seung のアルゴリズム [5] として知られるのが multiplicative update（MU 更新）である。これを W を含む逆問題の形式で構成しよう。まずは正則化項のない場合 $(R(A,X)=0)$ を考えよう. コスト関数 (7.8) の微分を計算すると

$$\nabla_A Q = W^T W A X X^T - W^T D X^T \tag{7.11}$$
$$\nabla_X Q = A^T W^T W A X - A^T W^T D. \tag{7.12}$$

となる。これらの微分値は非負項の $[\nabla Q]_- \geq 0, [\nabla Q]_+ \geq 0$ として

$$\nabla Q = [\nabla Q]_+ - [\nabla Q]_- = 0, \tag{7.13}$$

の形になっていることに注意する。MU 更新は $[\nabla Q]_-/[\nabla Q]_+$ を A や X に掛けていく更新方法である. これならば初期推定値が非負ならば、更新後の値も常に非負が保証される。さてそれはともかくこの MU 更新でどうして収束していくのだろうか？それは MU 更新は勾配降下となっているからである。

MU を勾配降下の形式で書くと以下のように書ける。

$$A \leftarrow A - \eta_A \odot \nabla_A Q \tag{7.14}$$
$$\eta_A = A \oslash [\nabla_A Q]_+ \tag{7.15}$$

また、

$$X \leftarrow X - \eta_X \odot \nabla_X Q \tag{7.16}$$
$$\eta_X = X \oslash [\nabla_X Q]_+ \tag{7.17}$$

である。ここに $\odot$ はアダマール積、つまり要素積である。同様に $\oslash$ は要素商である。以上をまとめると MU 更新は

$$A_{jk} \leftarrow A_{jk} \frac{[W^T D X^T]_{jk} + \epsilon}{[W^T W A X X^T]_{jk} + \epsilon} \tag{7.18}$$

$$X_{kl} \leftarrow X_{kl} \frac{[A^T W^T D]_{kl} + \epsilon}{[A^T W^T W A X]_{kl} + \epsilon}, \tag{7.19}$$

となる。ここに発散を避けるための小さな数 ϵ を導入した。

7.2 正則化のある非負値行列分解型の逆問題

さて非負値条件を入れても、行列分解の自由度はまだかなり高い。そこで正則化項 $R(A,X)$ をいれて解くことを考えよう。すなわち

$$\text{minimize } Q = \frac{1}{2}||D - WAX||_F^2 + R(A,X) \tag{7.20}$$
$$\text{subject to } A_{jk} \geq 0, X_{kl} \geq 0. \tag{7.21}$$

を満たす A, X を求める。

正則化項を入れる場合は、その微分が非負もしくは非正に保障されていれば MU 更新を行うことができる。例えば L2 正則化の場合

$$\nabla_A R(A,X) = \lambda_A A \tag{7.22}$$
$$\nabla_X R(A,X) = \lambda_X X. \tag{7.23}$$

となり、両方とも非負であることから、MU 更新は

$$\mathrm{U(A)}:\ A_{jk} \leftarrow A_{jk} \frac{[W^T D X^T]_{jk} + \epsilon}{[W^T W A X X^T + \lambda_A A]_{jk} + \epsilon} \tag{7.24}$$
$$\mathrm{U(X)}:\ X_{kl} \leftarrow X_{kl} \frac{[A^T W^T D]_{kl} + \epsilon}{[A^T W^T W A X + \lambda_X X]_{kl} + \epsilon}. \tag{7.25}$$

となることがわかる。MU 更新は直感的だが、収束速度が遅く、また微分値の非負（非正）性が保証されない場合は、MU 更新をもちいることはできない。次に示す BCD のほうが有用であることが多い。

Block Coordinate Descent

Block Coordinate Descent (BCD) は以下の手順を繰り返すことで最適化を行う。[4, 10, 1]。

- QP(A): $\boldsymbol{a}_k$ の二次形式最適化 (A の行を取り出した列ベクトル)
- QP(X): $\boldsymbol{x}_k$ の二次形式最適化 (X の列をとりだした列ベクトル)

BCD はこの二つの最適化 (QP(A) and QP(X))、非負最小二乗法 (non-negative least square ;NNLS) を交互に適用する。

まずはフロベニウスノルムの項を $\boldsymbol{a}$ について二次形式に変形してみよう。

$$\frac{1}{2}||D - WAX||_F^2 = \frac{1}{2}\sum_i \sum_l \left(\Delta_{il} - \sum_j W_{ij} A_{jk} X_{kl} \right)^2 \tag{7.26}$$
$$= \frac{1}{2}\sum_i \sum_l X_{kl} X_{lk}^T \left(\sum_j W_{ij} A_{jk} \right)^2 - \sum_i \sum_l \Delta_{li}^T \sum_j W_{ij} A_{jk} X_{kl} + \frac{1}{2}||\Delta||_F^2 \tag{7.27}$$
$$= \frac{1}{2}\sum_l X_{kl}^2 \sum_{j',j} A_{kj'}^T \left(\sum_i W_{j'i}^T W_{ij} \right) A_{jk} - \sum_j \left[\sum_l X_{kl} \left(\sum_i \Delta_{li}^T W_{ij} \right) \right] A_{jk} + \frac{1}{2}||\Delta||_F^2 \tag{7.28}$$
$$= \frac{1}{2}\boldsymbol{a}_k^T \mathcal{L}_A \boldsymbol{a}_k - \boldsymbol{l}_A^T \boldsymbol{a}_k + \mathrm{const.} \tag{7.29}$$

となる。ここに

$$\mathcal{L}_A \equiv \boldsymbol{x}_k^T \boldsymbol{x}_k W^T W \tag{7.30}$$
$$\boldsymbol{l}_A \equiv W^T \Delta\, \boldsymbol{x}_k, \tag{7.31}$$

である、$\Delta = \Delta(k)$ is defined by $\Delta_{il} \equiv D_{il} - \sum_{s \neq k} \sum_j W_{ij} A_{js} X_{sl}$ はデータ行列から k 以外の成分の寄与を除いたものと解釈できる。正則化項の二次形式も求めよう。L2 正則化項は

$$\frac{1}{2}\lambda_A ||A||_F^2 = \frac{1}{2}\boldsymbol{a}_k^T \mathcal{T}_A \boldsymbol{a}_k + \mathrm{const.} \tag{7.32}$$
$$\mathcal{T}_A \equiv \lambda_A I \tag{7.33}$$

となる。ここから、QP(A) は二次形式

$$q_A = \frac{1}{2}\boldsymbol{a}_k^T(\mathcal{L}_A + \mathcal{T}_A)\boldsymbol{a}_k - \boldsymbol{l}_A^T\boldsymbol{a}_k. \tag{7.34}$$

を最小化すればよいことがわかる。

同様に尤度を $\boldsymbol{x}_k$ についての二次形式にすると

$$\frac{1}{2}||D - WAX||_F^2 = \frac{1}{2}\boldsymbol{x}_k^T\mathcal{L}_X\boldsymbol{x}_k - \boldsymbol{l}_X^T\boldsymbol{x}_k + \text{const.} \tag{7.35}$$

$$\mathcal{L}_X \equiv ||W\boldsymbol{a}_k||_2^2 I \tag{7.36}$$

$$\boldsymbol{l}_X \equiv \Delta^T W\boldsymbol{a}_k \tag{7.37}$$

となる。グラム行列式による体積正則化項は

$$\frac{1}{2}\lambda_X \det(XX^T) = \frac{1}{2}\boldsymbol{x}_k^T\mathcal{D}_X\boldsymbol{x}_k \tag{7.38}$$

$$\mathcal{D}_X \equiv \lambda_X \det(\check{X}_k\check{X}_k^T)\left[I - \check{X}_k^T(\check{X}_k\check{X}_k^T)^{-1}\check{X}_k\right], \tag{7.39}$$

と書ける [10]。ここに $\check{X}_k$ は X から k 番目の行を除いた部分行列である。

PG 法

非負条件のついた二次形式

$$q = \boldsymbol{x}^T\mathcal{W}\boldsymbol{x} - \boldsymbol{b}^T\boldsymbol{x}. \tag{7.40}$$

の最適化を解く方法として投影勾配降下（Projected Gradient Descent; PG）法を紹介する。非負条件のもとでの Gradient Descent は投影演算子 $\mathcal{P}[\boldsymbol{x}] = \{\max(x_k, 0)\}$ を用いて

$$\boldsymbol{x}^{(t+1)} = \mathcal{P}[\boldsymbol{x}^{(t)} - \eta\nabla q] = \mathcal{P}[\boldsymbol{x}^{(t)} - \eta(\mathcal{W}\boldsymbol{x}^{(t)} - \boldsymbol{b})], \tag{7.41}$$

のように書くことができる。アルゴリズムとしては以下のようになる。

Projected Gradient Descent (PG)

Minimization of $q = \boldsymbol{x}^T\mathcal{W}\boldsymbol{x} - \boldsymbol{b}^T\boldsymbol{x}$

Initialization: $T = I - \mathcal{W}/L, \boldsymbol{s} = \boldsymbol{b}/L, \boldsymbol{x}_0$

while Condition **do**

$\boldsymbol{x}^{(t+1)} = \mathcal{P}[T\boldsymbol{x}^{(t)} + \boldsymbol{s}]$

end while

PG 法は収束がおそいため、ネステロフ加速 [7] を組み合わせて用いられることも多い。この方法を APG(accelerated projected gradient descent;) 法と呼ぶこともある。

Accelerated Projected Gradient Descent (APG)

Minimization of $q = \boldsymbol{x}^T\mathcal{W}\boldsymbol{x} - \boldsymbol{b}^T\boldsymbol{x}$

Initialization: $T = I - \mathcal{W}/L, \boldsymbol{s} = \boldsymbol{b}/L, \boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)} = \boldsymbol{x}^{(0)}, \alpha_0 = 0.9$

while Condition **do**

$\boldsymbol{x}^{(t+1)} = \mathcal{P}[T\boldsymbol{y}^{(t)} + \boldsymbol{s}]$

$\alpha_{t+1} = (\sqrt{\alpha_t^4 + 4\alpha_t^2} - \alpha_t^2)/2$

$\beta_{t+1} = \alpha_t(1-\alpha_t)/(\alpha_{t+1}+\alpha_t^2)$
$\boldsymbol{y}^{(t+1)} = \boldsymbol{x}^{(t+1)} + \beta_{t+1}(\boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)})$
end while

ネステロフ加速は単調に減少するとは限らないことから、減少をやめた時点でリセットする再加速法を利用することもできる [8].

APG+restart
Minimization of $q = \boldsymbol{x}^T\mathcal{W}\boldsymbol{x} - \boldsymbol{b}^T\boldsymbol{x}$
Initialization: $T = I - \mathcal{W}/L, \boldsymbol{s} = \boldsymbol{b}/L, \boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)} = \boldsymbol{x}^{(0)}, \alpha_0 = 0.9$
while Condition **do**
$\boldsymbol{x}^{(t+1)} = \mathcal{P}[T\boldsymbol{y}^{(t)} + \boldsymbol{s}]$
$q^{(t+1)} = (\boldsymbol{x}^{(t+1)})^T\mathcal{W}\boldsymbol{x}^{(t+1)} - \boldsymbol{b}^T\boldsymbol{x}^{(t+1)}$
$\alpha_{t+1} = (\sqrt{\alpha_t^4 + 4\alpha_t^2} - \alpha_t^2)/2$
$\beta_{t+1} = \alpha_t(1-\alpha_t)/(\alpha_{t+1}+\alpha_t^2)$
$\boldsymbol{y}^{(t+1)} = \boldsymbol{x}^{(t+1)} + \beta_{t+1}(\boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)})$
if $q^{(t+1)} > q^{(t)}$ **then**
$\boldsymbol{x}^{(t+1)} = \mathcal{P}[T\boldsymbol{x}^{(t)} + \boldsymbol{s}]$
$\boldsymbol{y}^{(t+1)} = \boldsymbol{x}^{(t+1)}, \alpha_{t+1} = \alpha_0$
end if
end while

第 8 章

最後に

Che Mapping のような不自然な問題設定は、実は筆者の研究対象である太陽系外惑星の表面推定の問題設定を翻訳したものである。すなわち壁画は、はるか彼方にある惑星の表面であり、サーチライトはその惑星を宿す恒星からの光である。

References

[1] M. S. Ang and Nicolas Gillis. Algorithms and Comparisons of Non-negative Matrix Factorization with Volume Regularization for Hyperspectral Unmixing. *arXiv e-prints*, page arXiv:1903.04362, Mar 2019.

[2] P. C. Hansen. *Discrete Inverse Problems: Insight and Algorithms*. the Society for Industrial and Applied Mathematics, 2010.

[3] Hajime Kawahara. Global Mapping of the Surface Composition on an Exo-Earth Using Color Variability. *The Astrophysical Journal*, 894(1):58, May 2020.

[4] Jingu Kim, Yunlong He, and Haesun Park. Algorithms for nonnegative matrix and tensor factorizations: A unified view based on block coordinate descent framework. *Journal of Global Optimization*, 58(2):285–319, 2014.

[5] Daniel D Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*, pages 556–562, 2001.

[6] W. Menke. *Geophysical data analysis: Discrete inverse theory*. 1989.

[7] Yurii E Nesterov. A method for solving the convex programming problem with convergence rate o (1/k^ 2). In *Dokl. akad. nauk Sssr*, volume 269, pages 543–547, 1983.

[8] Brendan O' donoghue and Emmanuel Candes. Adaptive restart for accelerated gradient schemes. *Foundations of computational mathematics*, 15(3):715–732, 2015.

[9] A. Tarantola. *Inverse Problem Theory and Methods for Model Parameter Estimation*. the Society for Industrial and Applied Mathematics, 2005.

[10] Guoxu Zhou, Shengli Xie, Zuyuan Yang, Jun-Mei Yang, and Zhaoshui He. Minimum-volume-constrained nonnegative matrix factorization: Enhanced ability of learning parts. *IEEE transactions on neural networks*, 22(10):1626–1637, 2011.