

目 次

第1章 確率的生成モデル	1
--------------	---

1.1 確率モデルの定式化	1
-------------------------	---

第2章 反復最適化	3
-----------	---

2.1 コスト関数最小化	3
------------------------	---

2.1.1 最急降下法	4
-----------------------	---

2.1.2 ニュートン法	6
------------------------	---

2.1.3 準ニュートン法	8
-------------------------	---

2.1.4 補助関数法	10
-----------------------	----

2.1.5 乗法更新アルゴリズム	13
----------------------------	----

2.2 確率モデルの最適化	15
-------------------------	----

2.2.1 潜在変数モデル	17
-------------------------	----

2.2.2 最尤推定：EM アルゴリズム	18
--------------------------------	----

2.2.3 ベイズ推定：変分ベイズ法	20
------------------------------	----

2.2.4 ベイズ推定：ギブスサンプリング	24
---------------------------------	----

2.2.5 ベイズ推定：周辺化ギブスサンプリング	27
------------------------------------	----

2.2.6 超パラメータの最適化	28
----------------------------	----

2.3 混合ベイズモデルの学習	30
---------------------------	----

2.3.1 最尤推定：EM アルゴリズム	33
--------------------------------	----

2.3.2 ベイズ推定：変分ベイズ法	33
------------------------------	----

2.4 変分事後分布	33
----------------------	----

2.4.1 VB-E ステップ	34
---------------------------	----

ii 目 次

2.4.2 VB-M ステップ	36
2.4.3 ベイズ推定：ギブスサンプリング	43
2.4.4 ベイズ推定：周辺化ギブスサンプリング	46

第 3 章 因子分解 47

3.1 非負値行列因子分解	47
3.1.1 コスト関数最小化としての定式化	48
3.1.2 乗法更新アルゴリズムに基づく最適化	50
3.1.3 EU-NMF の乗法更新アルゴリズム	51
3.1.4 KL-NMF の乗法更新アルゴリズム	55
3.1.5 IS-NMF の乗法更新アルゴリズム	58
3.1.6 β -NMF の乗法更新アルゴリズム	62
3.1.7 乗法更新アルゴリズム	67
3.2 非負値行列因子分解の確率的な解釈	68
3.2.1 確率モデルの最尤推定としての定式化	69
3.2.2 ノンパラメトリックベイズモデル	69
3.2.3 GaP-KL-NMF のベイズ推定	70
3.2.4 GaP-IS-NMF のベイズ推定	72
3.2.5 BP-KL-NMF のベイズ推定	74
3.3 半正定値テンソル分解	74
3.3.1 コスト関数最小化としての定式化	74
3.3.2 乗法更新アルゴリズムに基づく最適化	75
3.4 確率的潜在成分解析	76
3.4.1 確率モデルの最尤推定としての定式化	76
3.4.2 ノンパラメトリックベイズモデル	76
3.4.3 音源分離への応用	76

第 4 章 音声信号処理 79

4.1 音声分析合成	79
----------------------	----

目 次 iii

4.1.1	線形予測分析	79
4.1.2	Levinson-Durbin アルゴリズム	82
4.1.3	声道スペクトル推定としての解釈	86
4.1.4	無損失系等長音響管による声道モデルとしての解釈	92
4.2	韻律分析合成	95
4.2.1	背景と目的	95
4.2.2	骨格筋の弾性特性	97
4.2.3	基本周波数パターン生成過程モデル ^[?]	98

第 5 章 音楽信号解析 107

5.1	音楽音響信号の構成要素への分解	107
5.1.1	スパース性	109
5.1.2	低ランク性	110
5.1.3	音源分離への応用	111
5.1.4	音源分離への応用	114
5.1.5	確率モデルの最尤推定としての定式化	115
5.2	楽器パートの分離	115
5.2.1	歌声・伴奏音の分離	115
5.2.2	調波音・非調和音の分離	116
5.2.3	音色に基づく分離	116
5.3	おわりに	116

第 6 章 マイクロホンアレイ音響信号解析 121

6.1	音源定位	121
6.2	音源分離	121
6.3	残響除去	121
6.4	音源定位・音源分離・残響除去の統合モデル	121

c h a p t e r

1

確率的生成モデル

[本章では、確率的生成モデルについて説明します。]

1.1 確率モデルの定式化

c h a p t e r

2

反復最適化

本章では、与えられたコスト関数を最小化するパラメータを求める問題に対する反復最適化技法について解説します。現実の多くの問題では、コスト関数を最小化するパラメータを直接計算することができないため、パラメータを反復的に更新することでコスト関数を徐々に小さくする方法をとります。具体的には、コスト関数最小化（あるいは尤度関数最大化）の汎用的なアルゴリズムとして、最急降下法、ニュートン法、準ニュートン法、補助関数法、乗法更新アルゴリズムを紹介します。また、ベイズモデルの事後分布の計算に用いることができる変分ベイズ法とマルコフ連鎖モンテカルロ法についても紹介します。最後に、経験ベイズ法やベイズ最適化についても紹介します。

2.1 コスト関数最小化

最初に、コスト関数を最小化する問題を数学的に定義します。いま、あるパラメータ X に関するコスト関数 $f(X) \in \mathbb{R}$ が与えられたとします。パラメータはスカラ、ベクトル、行列、それらの集合など様々な形式をとりえますが、コスト関数の値は常に実数であるものとします。我々の目的は、

$$X^* = \underset{X}{\operatorname{argmin}} f(X) \quad (2.1)$$

となる最適解 X^* を求めることです。

一般に、関数 $f(X)$ は多峰性を持っているので、 $f(X)$ が最小値をとる最

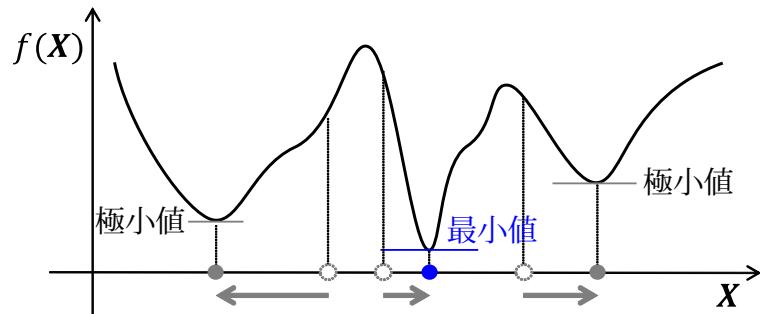


図 2.1 山登り法（山下り法）による最適解の探索。初期値によっては局所解しか見つからない。

適解を求めるとは容易ではなく、 $f(X)$ が極小値を取る局所解を求めることが現実的な目標となります。大域的な最適性が保証されるのは、 $f(X)$ が凸関数である場合がほとんどです。以降で紹介する反復最適化技法は、パラメータ X をなんらかの値に初期化し、その値を少しづつ変化させていく山登り型 (hill climbing) の（コスト関数「最小化」という意味では山下り型の）アルゴリズムです（図 2.1）。そのため、最終的に求まる解が大域的に最適である保証はなく、初期値依存性があることに注意が必要です。したがって、なんらかの事前知識が使える場合は、初期値を適切に設定することがより $f(X)$ を小さくする局所解を探索することにつながります。

2.1.1 最急降下法

最急降下法 (steepest descent method) はもっとも単純な反復最適化技法で、コスト関数の勾配方向の逆方向にパラメータを更新します。本節では、パラメータ X は、 N 次元ベクトル $x = [x_1, \dots, x_N]^T \in \mathbb{R}^N$ であるとします。このとき、関数 $f(x)$ の勾配ベクトル (gradient vector) は

$$\nabla f(x) = \left[\frac{\partial f(x)}{\partial x_1}, \dots, \frac{\partial f(x)}{\partial x_N} \right]^T \quad (2.2)$$

で与えられます。このとき、 $\nabla f(x)$ は $f(x)$ の等高面、すなわち、 $f(x)$ が一定となるような N 次元空間内の曲面に対して、その法線ベクトルを与えます。このベクトルの向きは、関数の値が増加する方向を示しています。

アルゴリズム 2.1： 最急降下法

```

Require: 最小化すべきコスト関数  $f(\mathbf{x}) \in \mathbb{R}$ 
1: パラメータ  $\mathbf{x} \in \mathbb{R}^N$  をランダムに初期化
2: while  $\nabla f(\mathbf{x}) \neq \mathbf{0}$  do
3:   探索ベクトル  $d(\mathbf{x}) = -\nabla f(\mathbf{x})$  を計算
4:   ステップ幅  $\alpha$  を適切に設定
5:    $\mathbf{x} \leftarrow \mathbf{x} + \alpha d(\mathbf{x})$ 
6: end while
7: Return パラメータ  $\mathbf{x}$ 

```

アルゴリズム 2.1 に， \mathbf{x} を更新するアルゴリズムを示します．具体的に， $\mathbf{x} = [x_1, x_2]^T$ として $f(\mathbf{x}) = x_1^2 + x_2^2$ の最小化について考えてみます．このとき，勾配ベクトルは

$$\nabla f(\mathbf{x}) = [2x_1, 2x_2]^T \quad (2.3)$$

となるので，ある \mathbf{x} における $\nabla f(\mathbf{x})$ の向きは，原点と \mathbf{x} とを結ぶ方向になります．この例では， $f(\mathbf{x})$ が一定となる等高線は円であり，確かに $\nabla f(\mathbf{x})$ は等高線に対する法線ベクトルになっています．したがって， $\nabla f(\mathbf{x})$ の逆向きの方向が最も $f(\mathbf{x})$ の値が減少する方向 $d(\mathbf{x})$ であるので， $d(\mathbf{x})$ の方向に沿って \mathbf{x} を「少しづつ」動かせばよいのです．

ここで重要なのは，ステップサイズ α の設定です． α を大きくすると， \mathbf{x} は大きく更新されるので，局所解へ早く収束しそうです．しかし，大きくしすぎると，局所解の周辺をいたりきたりしてしまいます．そのため，実際には，直線探索を用いて適切な α を決定することがよく行われます．一般には， α を少しづつ小さくしていくことが好ましいとされています．

最急降下法は， $f(\mathbf{x})$ が唯一の極小点を持つときには，停留点 ($\nabla f(\mathbf{x}) = \mathbf{0}$ となる \mathbf{x}) への大域的な収束性が保証されています．また，計算量が軽く，実装が簡単ですが，収束が遅いことが欠点です．

2.1.2 ニュートン法

収束が遅いという最急降下法の欠点を克服する方法として、ニュートン法 (Newton's method) が知られています。ニュートン法は、コスト関数 $f(\mathbf{x})$ の一次導関数 $\nabla f(\mathbf{x})$ だけではなく、二次導関数 $\nabla^2 f(\mathbf{x})$ を利用することで、探索ベクトル $d(\mathbf{x})$ の計算に工夫を行います。まず、 $f(\mathbf{x})$ に対して、二次のテイラー展開を行うと

$$\begin{aligned} f(\mathbf{x} + \Delta\mathbf{x}) &\approx f(\mathbf{x}) + \nabla f(\mathbf{x})^T \Delta\mathbf{x} + \frac{1}{2} \Delta\mathbf{x}^T \nabla^2 f(\mathbf{x}) \Delta\mathbf{x} \\ &\stackrel{\text{def}}{=} t(\mathbf{x} + \Delta\mathbf{x}) \end{aligned} \quad (2.4)$$

を得ます。ここで、二次導関数 $\nabla^2 f(\mathbf{x})$ は

$$\nabla^2 f(\mathbf{x}) = \begin{bmatrix} \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_1} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_N} \\ \vdots & & \vdots \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_N \partial x_1} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_N \partial x_N} \end{bmatrix} \quad (2.5)$$

で与えられます。 $\nabla^2 f(\mathbf{x})$ は、関数 $f(\mathbf{x})$ のヘッセ行列 (Hessian matrix) とよばれ、しばしば $H(\mathbf{x})$ と表されます。 $f(\mathbf{x})$ が二階連続微分可能であれば、 $H(\mathbf{x})$ は対称行列となります。式 (2.4) の二次近似の精度が十分によければ、 $t(\mathbf{x} + \Delta\mathbf{x})$ を最小化する $\Delta\mathbf{x}$ が、 $f(\mathbf{x} + \Delta\mathbf{x})$ を最小化する $\Delta\mathbf{x}$ のよい近似になっており、 $\mathbf{x} \leftarrow \mathbf{x} + \Delta\mathbf{x}$ と更新すればよいことになります。

覚えておくべき重要な性質として、 \mathbf{x} が $f(\mathbf{x})$ の極小点をとるには、ヘッセ行列 $H(\mathbf{x})$ は正定値行列である必要があります（必要十分ではありません）。行列の正定値性とは、固有値が全て正であることを意味し、スカラの正値性を拡張した概念です。例えば、 $N = 1$ のとき、 $f(\mathbf{x})$ はスカラを入力とする関数となり、その極小点において、一次導関数の値は負から正に切り替わるので、二次導関数の値は正をとらなくてはなりません。 $N > 1$ のときは、 $f(\mathbf{x})$ はベクトルを入力とする関数であり、二次導関数は行列形式で与えられます。このとき、極小点において、スカラの正値性を多次元拡張した性質である半正定値性が成立することになります。

ヘッセ行列 $H(\mathbf{x})$ が正定値であるとして、式 (2.4) に対して平方完成を行うと、 $\Delta\mathbf{x}$ の二次関数

$$\begin{aligned} t(\mathbf{x} + \Delta\mathbf{x}) &= f(\mathbf{x}) - \frac{1}{2} \nabla f(\mathbf{x})^T H(\mathbf{x})^{-1} \nabla f(\mathbf{x}) \\ &\quad + \frac{1}{2} (\Delta\mathbf{x} + H(\mathbf{x})^{-1} \nabla f(\mathbf{x}))^T H(\mathbf{x}) (\Delta\mathbf{x} + H(\mathbf{x})^{-1} \nabla f(\mathbf{x})) \end{aligned} \quad (2.6)$$

を得ます。この二次関数は、

$$\Delta\mathbf{x} = -H(\mathbf{x})^{-1} \nabla f(\mathbf{x}) \quad (2.7)$$

のとき最小値をとります。したがって、 $\mathbf{x} \leftarrow \mathbf{x} - H(\mathbf{x})^{-1} \nabla f(\mathbf{x})$ とすることで $f(\mathbf{x})$ を効率的に小さくすることができるはずです。

アルゴリズム 2.2： ニュートン法

```

Require: 最小化すべきコスト関数  $f(\mathbf{x}) \in \mathbb{R}$ 
1: パラメータ  $\mathbf{x} \in \mathbb{R}^N$  をランダムに初期化
2: while  $\nabla f(\mathbf{x}) \neq \mathbf{0}$  do
3:   探索ベクトル  $d(\mathbf{x}) = -(\nabla^2 f(\mathbf{x}))^{-1} \nabla f(\mathbf{x})$  を計算
4:   ステップ幅  $\alpha$  を適切に設定
5:    $\mathbf{x} \leftarrow \mathbf{x} + \alpha d(\mathbf{x})$ 
6: end while
7: Return パラメータ  $\mathbf{x}$ 
```

アルゴリズム 2.2 に、 \mathbf{x} を更新するアルゴリズムを示します。 $f(\mathbf{x})$ が二次関数である場合には、式 (2.4) の近似で誤差は生じないため、式 (2.7) のときに $f(\mathbf{x} + \Delta\mathbf{x})$ は最小値をとり、反復は一回で終了します。実際には、 $f(\mathbf{x})$ は二次関数でない場合が普通であり、探索ベクトル $d(\mathbf{x})$ の方向に 1 以外のスケールで動かせるようにステップ幅 α が導入されています。

ニュートン法は収束速度が速いですが、初期値が局所解に十分に近くないと収束性が保証されません。ただし、実用上は、 $\alpha = 1$ としても問題なく収束する場合が多いです。また、特に N が大きい場合に問題となります、数值的に不安定になりやすく、ヘッセ行列 $H(\mathbf{x})$ が正定値性を満たさなくなったり、逆行列 $H(\mathbf{x})^{-1}$ の計算負荷が大きいといった欠点があります。

2.1.3 準ニュートン法

ヘッセ行列 $H(x)$ やその逆行列を直接計算しなければならないというニュートン法の欠点を克服するため、準ニュートン法 (quasi-Newton method) が知られています。準ニュートン法では、最適化の繰り返し計算の過程で得られる勾配ベクトルにより、ヘッセ行列 $H(x)$ の近似を行います。まず、(精度はともかく) 勾配ベクトルは次式で近似できます。

$$\nabla f(x + \Delta x) \approx \nabla f(x) + H(x)\Delta x \quad (2.8)$$

したがって、 $H(x)$ の近似値として、セカント方程式 (Secant equation)

$$\nabla f(x + \Delta x) = \nabla f(x) + B(x)\Delta x \quad (2.9)$$

を満たすような $B(x)$ を求めればよいことになります。

アルゴリズム 2.3： 準ニュートン法

```

Require: 最小化すべきコスト関数  $f(x) \in \mathbb{R}$ 
1: パラメータ  $x \in \mathbb{R}^N$  をランダムに初期化
2: 近似ヘッセ行列  $B(x)$  を初期化 (単位行列など)
3: while  $\nabla f(x) \neq \mathbf{0}$  do
4:   探索ベクトル  $d(x) = -B(x)^{-1}\nabla f(x)$  を計算
5:   ステップ幅  $\alpha$  を適切に設定
6:    $x$  の変化量  $\Delta x = \alpha d(x)$  を計算
7:    $\nabla f(x)$  の変化量  $y = \nabla f(x + \Delta x) - \nabla f(x)$  を計算
8:    $x \leftarrow x + \Delta x$ 
9:   近似ヘッセ行列の逆行列  $B(x)^{-1}$  を更新
10: end while
11: Return パラメータ  $x$ 
```

アルゴリズム 2.3 に、 x を更新するアルゴリズムを示します。準ニュートン法では、パラメータ x だけではなく、近似ヘッセ行列 $B(x)$ も反復的に更新されるため、両者を初期化しておく必要があります。表 2.1 および表 2.2

表 2.1 近似ヘッセ行列 $B(\mathbf{x})$ の更新

手法	更新式
DFP	$B(\mathbf{x}) \leftarrow \left(I - \frac{\mathbf{y} \Delta \mathbf{x}^T}{\mathbf{y}^T \Delta \mathbf{x}} \right) B(\mathbf{x}) \left(I - \frac{\Delta \mathbf{x} \mathbf{y}^T}{\mathbf{y}^T \Delta \mathbf{x}} \right) + \frac{\mathbf{y} \mathbf{y}^T}{\mathbf{y}^T \Delta \mathbf{x}}$
BFGS	$B(\mathbf{x}) \leftarrow B(\mathbf{x}) + \frac{\mathbf{y} \mathbf{y}^T}{\mathbf{y}^T \Delta \mathbf{x}} - \frac{B(\mathbf{x}) \Delta \mathbf{x} (B(\mathbf{x}) \Delta \mathbf{x})^T}{\Delta \mathbf{x}^T B(\mathbf{x}) \Delta \mathbf{x}}$
SR1	$B(\mathbf{x}) \leftarrow B(\mathbf{x}) + \frac{(\mathbf{y} - B(\mathbf{x}) \Delta \mathbf{x})(\mathbf{y} - B(\mathbf{x}) \Delta \mathbf{x})^T}{(\mathbf{y} - B(\mathbf{x}) \Delta \mathbf{x})^T \Delta \mathbf{x}}$
Broyden	$B(\mathbf{x}) \leftarrow B(\mathbf{x}) + \frac{\mathbf{y} - B(\mathbf{x}) \Delta \mathbf{x}}{\Delta \mathbf{x}^T \Delta \mathbf{x}} \Delta \mathbf{x}^T$

表 2.2 近似ヘッセ行列の逆行列 $C(\mathbf{x}) = B(\mathbf{x})^{-1}$ の更新

手法	更新式
DFP	$C(\mathbf{x}) \leftarrow C(\mathbf{x}) + \frac{\Delta \mathbf{x} \Delta \mathbf{x}^T}{\mathbf{y}^T \Delta \mathbf{x}} - \frac{C(\mathbf{x}) \mathbf{y} \mathbf{y}^T C(\mathbf{x})^T}{\mathbf{y}^T C(\mathbf{x}) \mathbf{y}}$
BFGS	$C(\mathbf{x}) \leftarrow \left(I - \frac{\mathbf{y} \Delta \mathbf{x}^T}{\mathbf{y}^T \Delta \mathbf{x}} \right)^T C(\mathbf{x}) \left(I - \frac{\mathbf{y} \Delta \mathbf{x}^T}{\mathbf{y}^T \Delta \mathbf{x}} \right) + \frac{\Delta \mathbf{x} \Delta \mathbf{x}^T}{\mathbf{y}^T \Delta \mathbf{x}}$
SR1	$C(\mathbf{x}) \leftarrow C(\mathbf{x}) + \frac{(\Delta \mathbf{x} - C(\mathbf{x}) \mathbf{y})(\Delta \mathbf{x} - C(\mathbf{x}) \mathbf{y})^T}{(\Delta \mathbf{x} - C(\mathbf{x}) \mathbf{y})^T \mathbf{y}}$
Broyden	$C(\mathbf{x}) \leftarrow C(\mathbf{x}) + \frac{(\Delta \mathbf{x} - C(\mathbf{x}) \mathbf{y}) \Delta \mathbf{x}^T C(\mathbf{x})}{\Delta \mathbf{x}^T C(\mathbf{x}) \mathbf{y}}$

に，近似ヘッセ行列 $B(\mathbf{x})$ あるいはその逆行列 $C(\mathbf{x}) = B(\mathbf{x})^{-1}$ を求めるアルゴリズムを示します。最初のアルゴリズムである DFP 法は，最近はあまり用いられていません。現在最も用いられているアルゴリズムは，BFGS 法（提案者である Broyden, Fletcher, Goldfarb, Shanno の頭文字から）と SR1 法です。準ニュートン法を大規模問題に応用するため，記憶制限準ニュートン法 (limited-memory quasi-Newton method) が発表され，BFGS 法の記憶制限版として L-BFGS 法が盛んに利用されています。SR1 法は，ヘッセ行列の更新時に正定値性が保存されないため，不定値行列に対しても用いることができます。また，Broyden 法は行列が対称行列でなくとも良く，通常の連立方程式の解を求めるのにも使うことができます。

2.1.4 補助関数法

これまで紹介してきた汎用的な最適化技法とは異なり，ある条件下において収束性の保証された更新則を導出できる補助関数法 (auxiliary-function-based method) を紹介します。最急降下法やニュートン法では，通常，最適化が進むにつれて，ステップ幅 α を徐々に小さくしていくことがよく行われますが，収束性を担保しつつ，効率的なスケジューリングを行うことはそれほど簡単ではありません。補助関数法では，このようなステップ幅の設定が不要で，経験的には高速に収束することが知られています。

アルゴリズム 2.4： 補助関数法

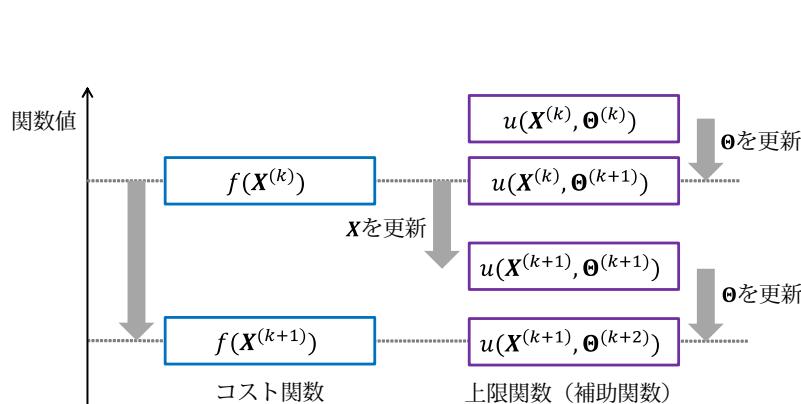
Require: 最小化すべきコスト関数 $f(\mathbf{X}) \in \mathbb{R}$

- 1: 補助変数 Θ を導入して，上限関数 $u(\mathbf{X}, \Theta) \in \mathbb{R}$ を設計
- 2: パラメータ \mathbf{X} をランダムに初期化
- 3: **while** $u(\mathbf{X}, \Theta)$ の減少量が大きい **do**
- 4: $\Theta \leftarrow \operatorname{argmin}_{\Theta} u(\mathbf{X}, \Theta)$
- 5: $\mathbf{X} \leftarrow \operatorname{argmin}_{\mathbf{X}} u(\mathbf{X}, \Theta)$
- 6: **end while**
- 7: Return パラメータ \mathbf{X}

アルゴリズム 2.4 に， \mathbf{X} を更新するアルゴリズムを示します。ここでは， \mathbf{X} はベクトルに限定せず，任意の形式をとるものとします。補助関数法では，コスト関数 $f(\mathbf{X})$ の上限関数 $u(\mathbf{X}, \Theta)$ を設計し， \mathbf{X} と Θ について交互に $u(\mathbf{X}, \Theta)$ を逐次最小化することで，間接的に $f(\mathbf{X})$ を逐次最小化します。ここで， Θ は新たに導入された補助変数で， $u(\mathbf{X}, \Theta)$ を Θ について最小化すると，もとの関数 $f(\mathbf{X})$ と同じ値をとるようにしておきます。

$$f(\mathbf{X}) = \min_{\Theta} u(\mathbf{X}, \Theta) \quad (2.10)$$

このアルゴリズムの収束性についてみてみましょう。いま，あるステップ k における \mathbf{X} および Θ の値を $\mathbf{X}^{(k)}$ および $\Theta^{(k)}$ とすると，上限関数が満たすべき性質から

図 2.2 補助関数法によるパラメータ \mathbf{X} と補助変数 Θ の反復最適化 .

$$\begin{aligned}
 f(\mathbf{X}^{(k)}) &= \min_{\Theta} u(\mathbf{X}^{(k)}, \Theta) = u(\mathbf{X}^{(k)}, \Theta^{(k+1)}) \\
 &\geq u(\mathbf{X}^{(k+1)}, \Theta^{(k+1)}) \\
 &\geq u(\mathbf{X}^{(k+1)}, \Theta^{(k+2)}) = f(\mathbf{X}^{(k+1)})
 \end{aligned} \tag{2.11}$$

となります(図 2.2). したがって, $\{f(\mathbf{X}^{(k)})\}_{k=1}^{\infty}$ は単調非増加 (monotonically non-increasing) となり, 停留点に収束します.

2.2 章で説明する確率モデルの最適化においては, コスト関数を最小化するのではなく, 尤度関数を最大化する問題を解く必要があります. この場合には, 尤度関数の符号を反転させることによりコスト関数とみなせて, 補助関数法が適用できる場合があります.

補助関数法の肝は, $f(\mathbf{X})$ を直接最小化する(例えば \mathbf{X} で偏微分したものをゼロとおいた方程式を解析的に解く)ことが難しい場合に, 一方の変数の値が既知であれば, もう一方の変数について最小化することが容易になるような $u(\mathbf{X}, \Theta)$ をうまく設計することにあります. 例えば, $u(\mathbf{X}, \Theta)$ が \mathbf{X} に関する凸関数となっており, \mathbf{X} で偏微分してゼロとおいた式が解析的に解けるとすると, Θ が与えられたもとでの \mathbf{X} の最適解を得ることができます. このとき, アルゴリズムは高速に収束することができます.

最適化を行いやすい $u(\mathbf{X}, \Theta)$ を設計するうえで有用な基本原理について説明します. まず, f が凸関数(convex function)である場合, イエンセンの不等式(Jensen's inequality)が適用できる可能性があります.

定義 2.1 (イエンセンの不等式)

任意の凸関数 $f : \mathbb{R}^N \mapsto \mathbb{R}$ に対して,

$$f\left(\sum_{k=1}^K \lambda_k \mathbf{x}_k\right) \leq \sum_{k=1}^K \lambda_k f(\mathbf{x}_k) \quad (2.12)$$

が成立します。ただし、 $\{\mathbf{x}_k\}_{k=1}^K$ は任意の N 次元ベクトルで、 $\{\lambda_k\}_{k=1}^K$ は $\lambda_k \geq 0$ かつ $\sum_{k=1}^K \lambda_k = 1$ を満たす非負の実数です。

これが成立することは、凸関数の定義から明らかです。図 2.3(a) に、 $N = 1$ のときの様子を示します。不等式の左辺は、 $\{\mathbf{x}_k\}_{k=1}^K$ の重み付き和の関数値を計算していますが、右辺は、各 x_k における関数値の重み付き和をとっています。したがって、 f が凸関数であるならば、後者の方が大きくなります。例えば、凸関数 $f(x) = -\log(x)$ に対して、次式が成立します。

$$-\log\left(\sum_{k=1}^K \lambda_k x_k\right) \leq -\sum_{k=1}^K \lambda_k \log(x_k) \quad (2.13)$$

一方、 f が凹関数 (concave function) である場合、一次のテイラー展開に基づく接平面を補助関数に用いることができます。

定義 2.2 (接平面に基づく不等式)

任意の凹関数 $f : \mathbb{R}^N \mapsto \mathbb{R}$ に対して,

$$f(\mathbf{x}) \leq f(\boldsymbol{\omega}) + f'(\boldsymbol{\omega})^T(\mathbf{x} - \boldsymbol{\omega}) \quad (2.14)$$

が成立します。ただし、 \mathbf{x} および $\boldsymbol{\omega}$ は任意の N 次元ベクトルです。

図 2.3(b) に、 $N = 1$ のときの不等式の様子を示します。不等式の右辺は、 x の一次式であるので、 $N = 1$ のときは接平面の方程式を表します。例えば、凹関数 $f(x) = \log(x)$ に対して、次式が成立します。

$$\log(x) \leq \log(\omega) + \frac{1}{\omega}(x - \omega) = \frac{x}{\omega} + \log(\omega) - 1 \quad (2.15)$$

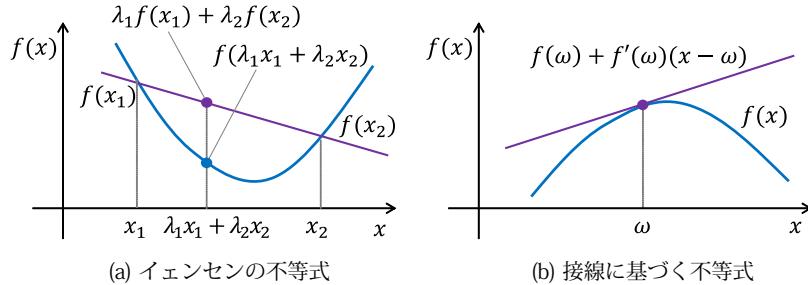


図 2.3 コスト関数の上限関数を導出するうえで有用な基本原理 .

2.1.5 乗法更新アルゴリズム

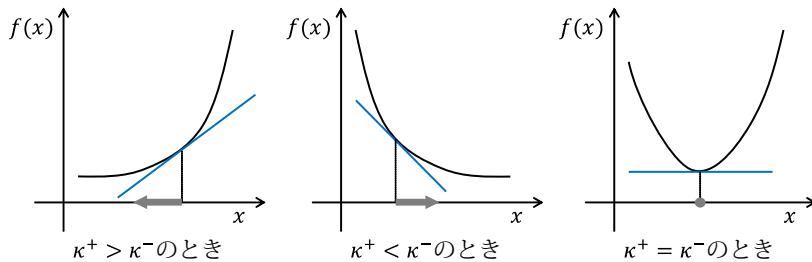
コスト関数 $f(X)$ の入力 X が非負値のスカラ x である場合には、乗法更新アルゴリズムと呼ばれる反復最適化技法が利用できる場合があります。この手法では、特別な制約を導入することなしに、毎回の反復における x の非負値性を自然に保つことができます。

アルゴリズム 2.5： 乗法更新アルゴリズム

Require: 最小化すべきコスト関数 $f(x) \in \mathbb{R}$

- 1: パラメータ x をランダムに初期化
 - 2: **while** $f(x)$ の減少量が大きい **do**
 - 3: $\frac{\partial f(x)}{\partial x} = \kappa^+(x) - \kappa^-(x)$ を計算 . ただし , $\kappa^+(x) > 0$ および
 $\kappa^-(x) > 0$ を満たすものとする .
 - 4: $x \leftarrow \frac{\kappa^-(x)}{\kappa^+(x)}x$
 - 5: **end while**
 - 6: **Return** パラメータ x

アルゴリズム 2.5 に、乗法更新アルゴリズムを示します。いま、ある非負の変数 $x \geq 0$ に関するコスト関数 $f(x)$ が与えられており、これを x について最小化する問題を考えます。このとき、 $f(x)$ の x に関する一次導関数が

図 2.4 $f(x)$ の最小化に乗法更新則を適用した場合の x の変化 .

$$\frac{\partial f(x)}{\partial x} = \kappa^+(x) - \kappa^-(x) \quad (2.16)$$

の形で表現できたとします。ただし、 $\kappa^+(x) > 0$ および $\kappa^-(x) > 0$ は x の関数であり、常に正をとるものとします。このとき、

$$x \leftarrow \frac{\kappa^-(x)}{\kappa^+(x)} x \quad (2.17)$$

とすると、 $f(x)$ が小さくなることが期待できます。収束性は理論的に保証されていますが、実用上は問題がない場合がほとんどです。

このアルゴリズムが $f(x)$ の値をどのように小さくするのかについて、 κ^+ および κ^- の大小関係で場合分けして考察してみましょう（図 2.4）。

- $\kappa(x)^+ > \kappa(x)^-$ のとき：

$\frac{\partial f(x)}{\partial x} > 0$ となり、 $f(x)$ の x に関する傾きは正であるので、 $f(x)$ を小さくするには、 x を小さくする必要があります。式 (2.17) をみると、分子より分母の方が大きくなり、更新によって x が小さくなります。

- $\kappa(x)^+ < \kappa(x)^-$ のとき：

$\frac{\partial f(x)}{\partial x} < 0$ となり、 $f(x)$ の x に関する傾きは負であるので、 $f(x)$ を小さくするには、 x を小さくする必要があります。式 (2.17) をみると、分子より分母の方が小さくなり、更新によって x が大きくなります。

- $\kappa(x)^+ = \kappa(x)^-$ のとき：

$\frac{\partial f(x)}{\partial x} = 0$ となり、 $f(x)$ は x において停留点をとることを示しています。

式 (2.17) をみると、分子と分母が同じになり、 x は更新されません。

2.2 確率モデルの最適化

本節では、確率モデルの学習に必要となる最適化技法について紹介します。いま、確率モデルのパラメータ（の集合）を Θ 、確率モデルから生成された観測データを X とします。観測データ X が与えられた時に、確率モデルのパラメータ Θ を推定するには、主に 3 つのアプローチがあります。

最尤推定 (maximum-likelihood (ML) estimation)

最尤推定では、観測データ X に対して、パラメータ Θ の尤度関数 (likelihood function) $f(\Theta) = p(X|\Theta)$ を最大化するような Θ^* を点推定する（一意に決定する）ことが目標です。

$$\Theta^* = \underset{\Theta}{\operatorname{argmax}} p(X|\Theta) \quad (2.18)$$

ここで、 $p(X|\Theta)$ は、 Θ から X が生成される確率^{*1} を表すので、この値が大きい Θ ほど尤もらしいと考え、 $p(X|\Theta)$ を Θ の良さを評価する関数 $f(\Theta)$ であるとみなします。この尤度関数はコスト関数の符号を反転したものであるとみなすことで、最適解が解析的に求められない場合は、これまで説明してきた反復最適化技法を利用することができます。最尤推定は幅広く用いられている最も基本的なアプローチですが、観測データ X があまり大きくな場合には、推定結果が不正確になりやすい問題があります。なぜなら、パラメータの値は観測データ X のみで決まり、何らかの制約を加えることができないからです。

最大事後確率推定 (maximum-a-posteriori (MAP) estimation)

MAP 推定では、パラメータ Θ に対する事前分布 $p(\Theta)$ を導入し、事前分布 $p(\Theta)$ と尤度関数 $p(X|\Theta)$ との積を最大化するような Θ^* を点推定することが目標です。

$$\Theta^* = \underset{\Theta}{\operatorname{argmax}} p(X|\Theta)p(\Theta) \quad (2.19)$$

^{*1} 連続側の分布の場合、正確には確率密度ですが、本書では区別せずに「確率」と呼びます。また、パラメータ Θ を確率変数として取り扱わない場合は、 $p(X|\Theta)$ ではなく $p(X; \Theta)$ として表記する場合も多いです。本書では、これらの区別は特にいません。

ただし、MAP 推定では、事前知識を反映して、 $p(\Theta)$ を適切に設定する必要があります。もし、 Θ がさまざまな値を取りうる可能性が高い場合はならかな確率分布を、 Θ がある特定の値の近くを取ることが分かれている場合は急峻な確率分布を設定します。これにより、事前知識^{*2} と実際の観測データとを考慮することで、観測データが小さい場合でも、事前知識に基づく安定したパラメータ推定が可能になります。事前分布が一様分布の場合（事前知識が特ない場合）、MAP 推定は最尤推定と同じ結果を与えます。MAP 推定においても、最尤推定と同様に、標準的な方法を用いた最適化が可能です。

ベイズ推定 (Bayesian estimation)

最尤推定・MAP 推定では、パラメータ Θ を点推定していたのに対し、ベイズ推定では、ベイズの定理を用いることで、事前分布 $p(\Theta)$ と尤度関数 $p(X|\Theta)$ から事後分布 $p(\Theta|X)$ を求めることが目標です。

$$p(\Theta|X) = \frac{p(X|\Theta)p(\Theta)}{p(X)} = \frac{p(X|\Theta)p(\Theta)}{\int p(X|\Theta)p(\Theta)d\Theta} \quad (2.20)$$

本来、 Θ は未知ですから、その推定結果には不確実性が伴います。したがって、観測データが十分にない場合には、100%の確信度をもって Θ の値を一意に決めることは難しく、 Θ のあらゆる可能性を考慮しておくことが望ましいでしょう。ベイズ推定では、 Θ の取りうる値それぞれについて、それがどの程度尤もらしいかという確信度、すなわち確率値を計算します。観測データが増加するにつれて、事後分布は急峻になり、観測データが無限にある場合は、最尤推定で求まる Θ^* に収束します。現実の多くの問題においては、式 (2.20) の分母、すなわち周辺尤度 $p(X)$ を求める際の積分計算を解析的に実行することは困難なため、真の事後分布 $p(\Theta|X)$ を正確に求めることは容易ではありません。主な近似推論方法として、2.2.3 節で説明する変分ベイズ法 (variational Bayesian (VB) methods) と 2.2.4 節で説明するギブスサンプリング (Gibbs sampling) が知られており、いずれも反復計算を行うことで、事後分布 $p(\Theta|X)$ を近似する最適化技法です。

^{*2} 「仮想的な」観測データと解釈することができます。多くの場合、 $p(\Theta)$ には仮想的な観測データの個数と解釈できるパラメータが含まれており、パラメータ推定時に事前分布をどの程度重視するかを自由に制御することができます。

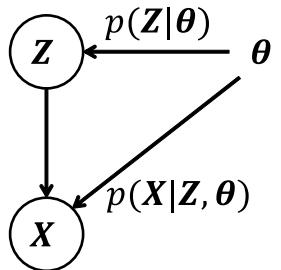


図 2.5 最尤推定のための確率モデル .

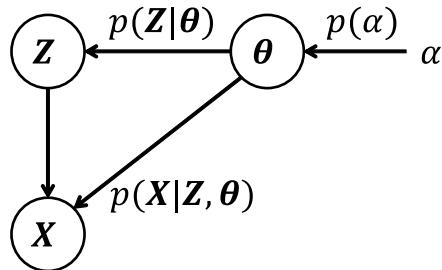


図 2.6 事前分布を導入したベイズモデル .

2.2.1 潜在変数モデル

現実の多くの問題においては、潜在変数モデル (latent variable models) と呼ばれる確率モデルを用いる必要があります。潜在変数モデルでは、パラメータ Θ に加えて、観測変数 (observable variables) X の背後に潜在変数 (latent variables) Z を考えます。

$$p(X, Z | \Theta) = p(X | Z, \Theta)p(Z | \Theta) \quad (2.21)$$

つまり、観測変数はパラメータから直接生成されるのではなく、何らかの潜在変数に影響を受けて生成されると考えます。潜在変数モデルに対して、パラメータの最尤推定を行う場合には、最適化問題

$$\Theta^* = \underset{\Theta}{\operatorname{argmax}} p(X | \Theta) = \underset{\Theta}{\operatorname{argmax}} \int p(X, Z | \Theta) dZ \quad (2.22)$$

を解くことになります。ここで、潜在変数 Z は確率変数ですが、パラメータ Θ は確率変数ではないことに注意してください。

本来、パラメータ Θ は潜在変数 Z 同様に未知であるので、不確実性を取り扱う、すなわち、確率変数として取り扱う方が望ましいでしょう。このとき、パラメータ Θ に対する事前分布 $p(\Theta)$ を導入することで、ベイズモデル

$$p(X, Z, \Theta) = p(X | Z, \Theta)p(Z | \Theta)p(\Theta) \quad (2.23)$$

を定式化することができます。このモデルに対してベイズ推定を行う場合には、ベイズの定理を用いて、パラメータ Θ と潜在変数 Z の事後分布を求めることになります。

$$\begin{aligned}
 p(Z, \Theta | X) &= \frac{p(X|Z, \Theta)p(Z|\Theta)p(\Theta)}{p(X)} \\
 &= \frac{p(X|Z, \Theta)p(Z|\Theta)p(\Theta)}{\int \int p(X|Z, \Theta)p(Z|\Theta)p(\Theta)dZd\Theta} \quad (2.24)
 \end{aligned}$$

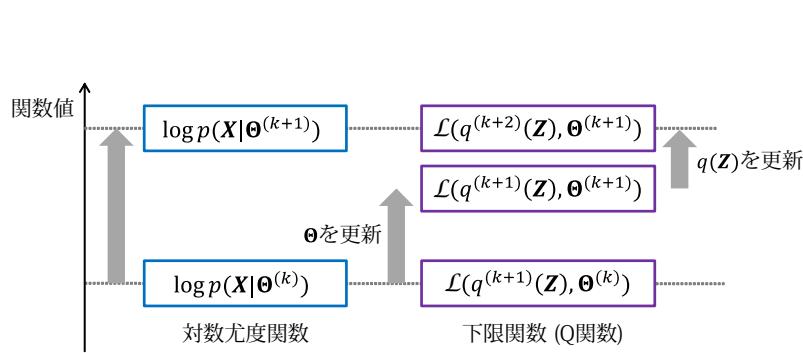
現実には、式 (2.24) の分母の積分を解析的に計算することはできないことがほとんどなので、あとで説明する近似アルゴリズムが必要になります。

2.2.2 最尤推定：EM アルゴリズム

潜在変数モデルに対して最尤推定を行うための決定論的 (deterministic) な手法が Expectation-Maximization (EM) アルゴリズムです。実は、EM アルゴリズムは 2.1.4 章で説明した補助関数法の一種です。図 2.7 に示すように、式 (2.22) において、尤度関数 $p(X|\Theta)$ を直接最大化することは困難なので、その下限関数を最大化することで、間接的に $p(X|\Theta)$ を最大化することを考えます。いま、潜在変数 Z に関する任意の分布 $q(Z)$ を考えて、対数尤度関数 $\log p(X|\Theta)$ の下限関数 $\mathcal{L}(q(Z), \Theta)$ を設計します。

$$\begin{aligned}
 \log p(X|\Theta) &= \log \int p(X, Z|\Theta)dZ \\
 &= \log \int q(Z) \frac{p(X, Z|\Theta)}{q(Z)} dZ \\
 &\geq \int q(Z) \log \frac{p(X, Z|\Theta)}{q(Z)} dZ \\
 &= \mathbb{E}_{q(Z)}[\log p(X, Z|\Theta)] - \mathbb{E}_{q(Z)}[\log q(Z)] \\
 &\stackrel{\text{def}}{=} \mathcal{L}(q(Z), \Theta) \quad (2.25)
 \end{aligned}$$

ここで、対数関数が凸関数であることから、式 (2.12) で与えられるイエンセンの不等式を用いることで、和の対数を対数の和に変換しました。 $\mathcal{L}(q(Z), \Theta)$ は、Q 関数 (Q-function) と呼ばれ、パラメータ Θ の関数であると同時に、関数 $q(Z)$ の汎関数 (functional) となっています。EM アルゴリズムは、Expectation (E) ステップで補助関数 $q(Z)$ に関する最適化を、Maximization (M) ステップでパラメータ Θ に関する最適化を行い、これらを交互に反復します。この手順で、 $\mathcal{L}(q(Z), \Theta)$ は単調非減少 (monotonically non-decreasing) となり、収束性が保障されます。

図 2.7 EM アルゴリズムによるパラメータ \mathbf{X} と補助関数 $q(\Theta)$ の反復最適化 .

まず、E ステップにおいては、パラメータ Θ が既知のもとで、式 (2.25) において等号が成立する、すなわち、 $\mathcal{L}(q(\mathbf{Z}), \Theta)$ を最大化する $q(\mathbf{Z})$ を求めることが目的です。これは、制約条件

$$\int q(\mathbf{Z}) d\mathbf{Z} = 1 \quad (2.26)$$

付きの最大化問題なので、ラグランジュの未定乗数法を用いて解くことができます。まず、未定乗数 λ を導入した関数

$$F(q(\mathbf{Z})) = \int q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} d\mathbf{Z} + \lambda \left(1 - \int q(\mathbf{Z}) d\mathbf{Z} \right) \quad (2.27)$$

を考えます。これを $q(\mathbf{Z})$ で偏微分してゼロとおくと、

$$\frac{\partial F(q(\mathbf{Z}))}{\partial q(\mathbf{Z})} = \log p(\mathbf{X}, \mathbf{Z}|\Theta) - \log q(\mathbf{Z}) - 1 - \lambda = 0 \quad (2.28)$$

を得ます。これを解くと、

$$q(\mathbf{Z}) = e^{-1-\lambda} p(\mathbf{X}, \mathbf{Z}|\Theta) \quad (2.29)$$

となるので、これを式 (2.26) に代入すると、

$$e^{-1-\lambda} \int p(\mathbf{X}, \mathbf{Z}|\Theta) d\mathbf{Z} = 1 \quad (2.30)$$

となり、未定乗数 λ は

$$e^{-1-\lambda} = \frac{1}{\int p(\mathbf{X}, \mathbf{Z}|\Theta) d\mathbf{Z}} \quad (2.31)$$

で与えられます。最終的に、式 (2.31) を式 (2.61) に代入すると、最適な $q(Z)$ を求めることができます。

$$q(Z) = \frac{p(X, Z|\Theta)}{\int p(X, Z|\Theta) dZ} = \frac{p(X, Z|\Theta)}{p(X|\Theta)} = p(Z|X, \Theta) \quad (2.32)$$

次に、M ステップでは、 $q(Z)$ が既知のもとで、 $\mathcal{L}(q(Z), \Theta)$ を最大化する Θ を求めます。式 (2.25) において、 $\mathbb{E}_{q(Z)}[\log p(X, Z|\Theta)]$ （一般に Q 関数と呼ばれています）の最大化を考えればよいことになります。基本的には、 Θ で偏微分してゼロとおくことで、 Θ の更新式が得られます。

アルゴリズム 2.6：EM アルゴリズム

```
Require: 潜在変数モデル  $p(X, Z|\Theta) = p(X|Z, \Theta)p(Z|\Theta)$ 
1: 分布  $q(Z)$  およびパラメータ  $\Theta$  をランダムに初期化
2: while  $\mathcal{L}(q(Z), \Theta)$  が収束していない do
3:   E ステップ:  $q(Z) = p(Z|X, \Theta)$ 
4:   M ステップ:  $\Theta \leftarrow \operatorname{argmax}_{\Theta} \mathbb{E}_{q(Z)}[\log p(X, Z|\Theta)]$ 
5: end while
6: Return 分布  $q(Z)$  およびパラメータ  $\Theta$ 
```

アルゴリズム 2.6 に EM アルゴリズムの手順を示します。EM アルゴリズムでは、E ステップで $q(Z)$ を潜在変数 Z の事後分布と一致させることで、下限関数 $\mathcal{L}(q(Z), \Theta)$ が対数尤度関数 $\log p(X|\Theta)$ に等しくなり、M ステップでパラメータ Θ を最適化することで、対数尤度関数 $\log p(X|\Theta)$ が増加します。このように、直接 $\log p(X|\Theta)$ を増加させることは難しいものの、E ステップをはさむことで最適化を容易にする補助関数法となっています。

2.2.3 ベイズ推定：変分ベイズ法

潜在変数モデルに対してベイズ推定を行うための決定論的な手法が変分ベイズ法 (variational Bayesian method, VB) です。一般に、式 (2.24) において、真の事後分布 $p(Z, \Theta|X)$ を解析的に計算することは困難です。VB で

は、因子分解可能な変分事後分布 $q(Z, \Theta) = q(Z)q(\Theta)$ を考え、真の事後分布 $p(Z, \Theta|X)$ にできる限り近づけるような最適化を行います。したがって、最終的に求まる $q(Z, \Theta)$ は真の事後分布 $p(Z, \Theta|X)$ には一致せず、あくまで事後分布の近似計算手法であることに注意が必要です。

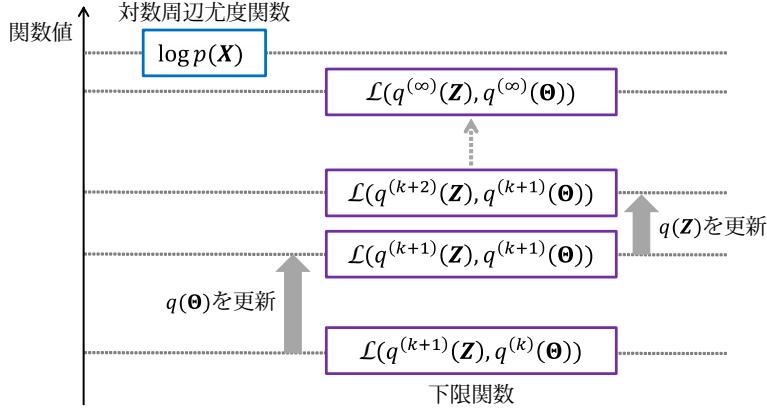
ベイズ推定の難しさは、式 (2.24) の分母である周辺尤度 (marginal likelihood) あるいはエビデンス (evidence) $p(X)$ の解析的な計算が困難な点にあります。したがって、 $p(X)$ を精度良く近似できれば、事後分布 $p(Z, \Theta|X)$ を精度良く近似することができます。EM アルゴリズムと同様、VB も 2.1.4 章で説明した補助関数法の一種となっており、対数周辺尤度 $\log p(X)$ の下限関数を設計し、それを最大化することにより、 $\log p(X)$ のよい近似値を求めます。下限関数は以下の通り導出できます。

$$\begin{aligned} \log p(X) &= \log \int \int p(X, Z, \Theta) dZ d\Theta \\ &= \log \int \int q(Z, \Theta) \frac{p(X, Z, \Theta)}{q(Z, \Theta)} dZ d\Theta \\ &\geq \int \int q(Z, \Theta) \log \frac{p(X, Z, \Theta)}{q(Z, \Theta)} dZ d\Theta \\ &= \int \int q(Z) q(\Theta) \log \frac{p(X, Z, \Theta)}{q(Z) q(\Theta)} dZ d\Theta \\ &= \mathbb{E}_{q(Z)q(\Theta)}[\log p(X, Z, \Theta)] - \mathbb{E}_{q(Z)}[\log q(Z)] - \mathbb{E}_{q(\Theta)}[\log q(\Theta)] \\ &\stackrel{\text{def}}{=} \mathcal{L}(q(Z), q(\Theta)) \end{aligned} \quad (2.33)$$

ここで、対数関数が凸関数なので、式 (2.12) で与えられるエンセンの不等式を用いました。 $\mathcal{L}(q(Z), q(\Theta))$ は、変分下限 (variational lower bound) あるいはエビデンス下限 (evidence lower bound, ELBO) と呼ばれ、関数 $q(Z)$ および $q(\Theta)$ の汎関数です。図 2.8 に示す通り、VB では、VB-E ステップで $q(Z)$ に関する最適化を、VB-M ステップで $q(\Theta)$ に関する最適化を行い、これらを交互に反復します。このように、汎関数を最大化する関数を求める問題を解くことから、VB は変分法の一種となっています。

EM アルゴリズムにおいて、式 (2.25) の等号成立条件は式 (2.32) であるのと同様、VBにおいても、式 (2.33) の等号成立条件は、

$$q(Z, \Theta) = p(Z, \Theta|X) \quad (2.34)$$

図 2.8 変分ベイズ法による補助関数 $q(\mathbf{Z})$ および $q(\boldsymbol{\Theta})$ の反復最適化 .

すなわち， $q(\mathbf{Z}, \boldsymbol{\Theta})$ が真の事後分布 $p(\mathbf{Z}, \boldsymbol{\Theta}|\mathbf{X})$ と等しいときです。しかし， $p(\mathbf{Z}, \boldsymbol{\Theta}|\mathbf{X})$ を解析的に計算することは困難です。そこで，本来， \mathbf{Z} と $\boldsymbol{\Theta}$ は独立ではないので， $p(\mathbf{Z}, \boldsymbol{\Theta}|\mathbf{X}) = p(\mathbf{Z}|\mathbf{X})p(\boldsymbol{\Theta}|\mathbf{X})$ は成立しませんが，VB では， $q(\mathbf{Z}, \boldsymbol{\Theta}) = q(\mathbf{Z})q(\boldsymbol{\Theta})$ という因数分解ができるという強い仮定を置きます。この結果，式 (2.33) の等号は成立しなくなりますが， $\mathcal{L}(q(\mathbf{Z}), q(\boldsymbol{\Theta}))$ を $q(\boldsymbol{\Theta})$ および $q(\mathbf{Z})$ について最大化することで， $\log p(\mathbf{X})$ をできる限り正確に近似できる $q(\boldsymbol{\Theta})$ および $q(\mathbf{Z})$ を探す問題を解くことを考えます。

因数分解の仮定 $q(\mathbf{Z}, \boldsymbol{\Theta}) = q(\mathbf{Z})q(\boldsymbol{\Theta})$ によって引き起こされる近似誤差を評価するため，式 (2.33) の左辺から右辺を引いた差分を計算してみます。

$$\begin{aligned}
 & \log p(\mathbf{X}) - \int \int q(\mathbf{Z}, \boldsymbol{\Theta}) \log \frac{p(\mathbf{X}, \mathbf{Z}, \boldsymbol{\Theta})}{q(\mathbf{Z}, \boldsymbol{\Theta})} d\mathbf{Z} d\boldsymbol{\Theta} \\
 &= \int \int q(\mathbf{Z}, \boldsymbol{\Theta}) \log p(\mathbf{X}) d\mathbf{Z} d\boldsymbol{\Theta} - \int \int q(\mathbf{Z}, \boldsymbol{\Theta}) \log \frac{p(\mathbf{X}, \mathbf{Z}, \boldsymbol{\Theta})}{q(\mathbf{Z}, \boldsymbol{\Theta})} d\mathbf{Z} d\boldsymbol{\Theta} \\
 &= \int \int q(\mathbf{Z}, \boldsymbol{\Theta}) \log \frac{q(\mathbf{Z}, \boldsymbol{\Theta})}{p(\mathbf{Z}, \boldsymbol{\Theta}|\mathbf{X})} d\mathbf{Z} d\boldsymbol{\Theta} \\
 &= \text{KL}(q(\mathbf{Z}, \boldsymbol{\Theta}) \| p(\mathbf{Z}, \boldsymbol{\Theta}|\mathbf{X})) \tag{2.35}
 \end{aligned}$$

ここで， $\text{KL}(q \| p)$ は確率分布 q の確率分布 p に対するカルバック・ライブーダイバージェンス (Kullback-Leibler (KL) divergence) で，必ず非負

値をとり， $q = p$ となるときに限り，最小値の0をとります。したがって， $q(Z, \Theta) = p(Z, \Theta|X)$ であれば，式(2.33)の等号が成立しますが， $q(Z, \Theta) = q(Z)q(\Theta)$ を仮定した場合には，近似誤差が生じます。したがって，VBは，真の事後分布 $p(Z, \Theta|X)$ に対するKL情報量を最小化する変分事後分布 $q(Z)q(\Theta)$ を計算する問題を解いています。

まず，VB-Eステップにおいて， $q(\Theta)$ が既知のもとで，式(2.33)における $\mathcal{L}(q(Z), q(\Theta))$ を最大化する $q(Z)$ を求めます。これは，制約条件

$$\int q(Z)dZ = 1 \quad (2.36)$$

付きの最大化問題なので，EMアルゴリズムと同様に，ラグランジュの未定乗数法を用いて解くことができます。最終的に

$$q(Z) \propto \exp(\mathbb{E}_{q(\Theta)}[\log p(X, Z, \Theta)]) \quad (2.37)$$

を得ます。VB-Mステップにおいては， $q(Z)$ が既知のもとで，式(2.33)における $\mathcal{L}(q(Z), q(\Theta))$ を最大化する $q(\Theta)$ を求めます。VB-Eステップとは Z と Θ の役割が入れ替わっただけなので，同様に導出できます。

$$q(\Theta) \propto \exp(\mathbb{E}_{q(Z)}[\log p(X, Z, \Theta)]) \quad (2.38)$$

を得ます。このように，VBでは，潜在変数とパラメータを区別することなく，いずれも確率変数として対等に取り扱うことができます。

アルゴリズム 2.7： 变分ベイズ法

```

Require: ベイズモデル  $p(X, Z, \Theta) = p(X|Z, \Theta)p(Z|\Theta)p(\Theta)$ 
1: 分布  $q(Z)$  および  $q(\Theta)$  をランダムに初期化
2: while  $\mathcal{L}(q(Z), q(\Theta))$  が収束していない do
3:   VB-Eステップ :  $q(Z) \propto \exp(\mathbb{E}_{q(\Theta)}[\log p(X, Z, \Theta)])$ 
4:   VB-Mステップ :  $q(\Theta) \propto \exp(\mathbb{E}_{q(Z)}[\log p(X, Z, \Theta)])$ 
5: end while
6: Return 分布  $q(Z)$  および  $q(\Theta)$ 

```

アルゴリズム 2.7 に VB を示します。EM アルゴリズム(アルゴリズム 2.6)と VB(アルゴリズム 2.7)を比較すると、E ステップでは潜在変数の事後分布を計算する点で共通していますが、M ステップでは、EM アルゴリズムではパラメータを点推定しているのに対し、VB ではパラメータの事後分布を計算している点で異なります。VB はいずれのステップにおいても、確率変数の期待値を計算していることから、本来は Expectation-Expectation (EE) アルゴリズムと呼ぶべきものであることが分かります。

より一般に、あるベイズモデル $p(\mathbf{X}, \Theta) = p(\mathbf{X}|\Theta)p(\Theta)$ を構成する確率変数 Θ (パラメータや潜在変数の区別を問わない) が、 M 個のグループ $\{\Theta_1, \dots, \Theta_M\}$ に分けられる場合について考えます。このとき、VB では、変分事後分布 $q(\Theta)$ を因子分解できる形 $q(\Theta) = \prod_{m=1}^M q(\Theta_m)$ に限定して、その中で、 $p(\Theta|\mathbf{X})$ に対する KL ダイバージェンスが最小となるものを探します。このときの各グループの変分事後分布の更新則は、

$$q(\Theta_m) \propto \exp \left(\mathbb{E}_{q(\Theta_{-m})} [\log p(\mathbf{X}, \Theta)] \right) \quad (2.39)$$

で与えられます。ここで、 $\neg m$ は m を除いた全てのインデックスを表すものとし、 $q(\Theta_{-m}) = q(\Theta_1) \cdots q(\Theta_{m-1})q(\Theta_{m+1}) \cdots q(\Theta_M)$ です。確率変数 Θ をどのように分割するかが重要で、この分割数 M が少ないほど、独立性の仮定が弱くなるので、 $q(\Theta)$ の $p(\Theta|\mathbf{X})$ に対する近似精度がよくなります。

2.2.4 ベイズ推定：ギブスサンプリング

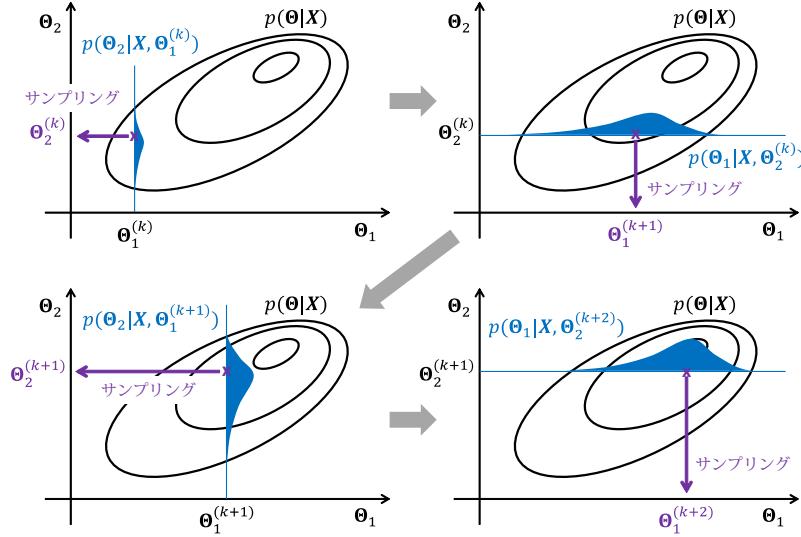
ギブスサンプリング (Gibbs sampling) はマルコフ連鎖モンテカルロ法 (Markov chain Monte Carlo methods, MCMC) の一種で、任意の確率分布に従うサンプルをランダムに生成することができる汎用的な方法です。ベイズ推定の難しさは、式 (2.24) の分母、すなわち事後分布 $p(Z, \Theta|\mathbf{X})$ の正規化項 $p(\mathbf{X})$ が解析的に計算できないことでした。MCMC は、対象となる確率分布の正規化項が計算できない場合にも利用可能なので、ベイズ推定において、事後分布 $p(Z, \Theta|\mathbf{X})$ からのサンプルを得る目的で広く用いられています。特に、ギブスサンプリングは最も効率の良いサンプリング方法で、完全な事後分布 $p(Z, \Theta|\mathbf{X})$ の計算は難しくても、「条件付き」事後分布 $p(Z|\Theta, \mathbf{X})$ や $p(\Theta|Z, \mathbf{X})$ が容易に計算可能であり、既存のアルゴリズムを用いてこれらから容易にサンプリングできる場合に適用できます。

アルゴリズム 2.8： ギブスサンプリング

```
Require: ベイズモデル  $p(X, Z, \Theta) = p(X|Z, \Theta)p(Z|\Theta)p(\Theta)$ 
1: 潜在変数  $Z$  およびパラメータ  $\Theta$  をランダムに初期化
2: while  $p(X, Z, \Theta)$  が収束していない do
3:   GS-E ステップ :  $Z \sim p(Z|\Theta, X)$  に従って  $Z$  をサンプル
4:   GS-M ステップ :  $\Theta \sim p(\Theta|Z, X)$  に従って  $\Theta$  をサンプル
5: end while
6: Return サンプルされた  $Z$  と  $\Theta$  のヒストグラムを  $p(Z, \Theta|X)$ 
   の近似として利用
```

アルゴリズム 2.8 にギブスサンプリングを示します（数学的な証明は教科書^[?]を参照）。VB（アルゴリズム 2.7）とギブスサンプリング（アルゴリズム 2.8）を比較すると、各ステップにおいて、VB では、確率変数の変分事後分布を求める（期待値を計算する）のに対して、ギブスサンプリングでは、確率変数の具体的な値をサンプリングする（一意に決定する）点が異なります。しかし、ベイズ推定では、未知の確率変数の値を一意に決定せずに、その不確実性を考慮する点がポイントのはずです。そこで、決定論的なアルゴリズムである VB とは異なり、確率的なアルゴリズムであるギブスサンプリングは、サンプリングのランダム性を利用することで、不確実性を取り扱います。多くの問題で、VB よりギブスサンプリングの方が実装が簡単で、初期値依存性が低く、局所解に陥りにくい傾向が報告されています。一般に、VB より反復回数は多く必要となるものの、1 回当たりの計算時間が小さいため、全体としての計算時間は短くなることもあります。その一方で、プログラムのデバッグや収束判定が難しいという問題があります。

ギブスサンプリングは本来、事後分布 $p(Z, \Theta|X)$ に従う Z と Θ のサンプルを生成するためのアルゴリズムですが、実際には $p(Z, \Theta|X)$ を最大化する Z と Θ を求めるための最適化技法として利用されます。もし、無限の時間があれば、 Z と Θ の定義域全体に渡るサンプルを得ることができます。そ

図 2.9 ギブスサンプリングによるパラメータの更新 ($\Theta = \{\Theta_1, \Theta_2\}$ の場合).

のヒストグラムは $p(Z, \Theta|X)$ と一致します。しかし、有限の時間では、そのような全空間の探索はできず、 $p(Z, \Theta|X)$ が概ね増加するように Z と Θ が更新されていき、どこかでほぼ収束します。実際には、 $p(Z, \Theta|X)$ が収束するまでのサンプルは捨て (burn-in)，それ以降のサンプルを十分な間隔をあけて取得して、それらの期待値をとることもあります。

より一般に、あるベイズモデル $p(X, \Theta) = p(X|\Theta)p(\Theta)$ を構成する確率変数 Θ (パラメータや潜在変数の区別を問わない) が、 M 個のグループ $\{\Theta_1, \dots, \Theta_M\}$ に分けられる場合について考えます。このとき、事後分布 $p(\Theta|X)$ に従う確率変数 Θ のサンプルを得るには、各 m について

$$\Theta_m \sim p(\Theta_m|X, \Theta_{\neg m}) \quad (2.40)$$

を繰り返すことになります(図 2.9)。VB とは異なり、理論上は、分割数 M がいくらであっても、正しく $p(\Theta|X)$ に従う確率変数 Θ のサンプルが得られます。しかし、 M が小さいほど、 $p(Z, \Theta|X)$ の増加が収束するのが早く、効率的に Θ の最適化を行うことができます。

2.2.5 ベイズ推定：周辺化ギブスサンプリング

周辺化ギブスサンプリング (collapsed Gibbs sampling) は、確率モデルを構成する確率変数のうちの一部を周辺化したうえで、残りの確率変数に対してギブスサンプリングを行う手法です。潜在変数モデルに対して適用する場合は、全ての確率変数の事後分布 $p(Z, \Theta | X)$ からではなく、パラメータ Θ を積分消去 (marginalize out, collapse) した潜在変数 Z の事後分布 $p(Z | X)$ からサンプリングを行います。事後分布の次元が小さくなっているため、通常のギブスサンプリングより収束が早くなる利点があります。

周辺化ギブスサンプリングを用いるには、二つの条件があります。まず、パラメータ Θ が容易に積分消去できることです。すなわち、ベイズモデル $p(X, Z, \Theta) = p(X|Z, \Theta)p(Z|\Theta)p(\Theta)$ に対して、積分計算

$$p(X, Z) = \int p(X|Z, \Theta)p(Z|\Theta)p(\Theta)d\Theta \quad (2.41)$$

を行う必要があります。多くの場合、観測変数 X の生成モデル $p(X|Z, \Theta)$ と潜在変数 Z の生成モデル $p(Z|\Theta)$ は独立したパラメータを持ち、それぞれ独立した事前分布を与えることが多いでしょう。このとき、ベイズモデルは $p(X, Z, \Theta_1, \Theta_2) = p(X|Z, \Theta_1)p(Z|\Theta_2)p(\Theta_1)p(\Theta_2)$ となり、

$$\begin{aligned} p(X, Z) &= \int p(X|Z, \Theta_1)p(\Theta_1)d\Theta_1 \int p(Z|\Theta_2)p(\Theta_2)d\Theta_2 \\ &= p(X|Z)p(Z) \end{aligned} \quad (2.42)$$

を計算する必要があります。 $p(\Theta_1)$ および $p(\Theta_2)$ に共役事前分布を用いた場合は、この積分は容易に解析的に計算できます。

もう一つの条件は、事後分布 $p(Z|X)$ に対して容易にギブスサンプリングが適用できることです。具体的には、潜在変数 Z が N 個のグループ $\{Z_1, \dots, Z_N\}$ に分けられるとすると、各 n について、

$$Z_n \sim p(Z_n | X, Z_{\neg n}) \quad (2.43)$$

を繰り返しますが、このとき、 $p(Z_n | X, Z_{\neg n})$ が解析的に計算でき、簡単にサンプリングできる形の確率分布であることが必要です。多くのモデルでは、 N 個の独立な観測変数 $X = \{x_1, \dots, x_N\}$ が与えられた時に、対応する N 個の潜在変数 $Z = \{z_1, \dots, z_N\}$ が仮定されています。このとき、ある観測

変数 x_n に対応する潜在変数 z_n に着目し、これ以外の潜在変数 $Z_{\neg n}$ の値が全て既知とした場合に、 $z_n \sim p(z_n | X, Z_{\neg n})$ として、 z_n の値を更新することを全ての n について繰り返します。

2.2.6 超パラメータの最適化

ベイズモデル $p(X|\Theta)p(\Theta)$ においては、事前分布 $p(\Theta)$ を適切に設定する必要があります。ここで、事前分布にもパラメータ α が存在することから、 $p(\Theta)$ を $p(\Theta|\alpha)$ と書くことにして、 α の決定方法について考えます。ベイズモデルの観点からは、 α は超パラメータ (hyperparameter) と呼ばれます。パラメータ Θ に関する事前知識があまりない場合は、無情報事前分布 (noninformative prior distribution) に近い事前分布を用いることが一般的です。このとき、パラメータ Θ の事後分布は、ほとんど事前分布の影響を受けず、観測データ X のみで決定されます。一方、事前分布自体を最適化したい場合には、主に三つのアプローチがあります。

経験ベイズ (empirical Bayes)

経験ベイズ法は、第二種の最尤推定 (type-II maximum likelihood estimation) とも呼ばれ、観測データ X に対する超パラメータ α の尤度関数 $p(X|\alpha)$ を最大化するような α^* を求めます。

$$\alpha^* = \operatorname{argmax}_{\alpha} p(X|\alpha) = \operatorname{argmax}_{\alpha} \int p(X|\Theta)p(\Theta|\alpha)d\Theta \quad (2.44)$$

このとき、 $p(X|\alpha)$ は、 Θ から見れば「周辺」尤度関数となっており、多くの現実的なモデルでは、積分計算を解析的に行なうことが困難です。このような場合でも、VB を用いて式 (2.33) に示す通り $p(X|\alpha)$ の下限関数 $\mathcal{L}(q(Z), q(\Theta), \alpha)$ を導出すれば、補助関数法 (2.1.4 章) の原理に従って、 $q(Z)$, $q(\Theta)$ および α を交互に最適化することが可能になります。ただし、最適化すべき関数 $\mathcal{L}(q(Z), q(\Theta), \alpha)$ は解析的に導出できただとしても、一般に、 $\mathcal{L}(q(Z), q(\Theta), \alpha)$ は α に関する複雑な関数となっており、最適な α^* を解析的に求ることは困難です。したがって、最急降下法 (2.1.1 章)、ニュートン法 (2.1.2 章)、準ニュートン法 (2.1.3 章) などの一般的な反復最適化技法を用いる必要があります。乗法更新アルゴリズム (2.1.5 章) が利用できる場合もあります。

階層ベイズ (hierarchical Bayes)

階層ベイズ法では、超パラメータ α に対する事前分布（超事前分布と呼ぶ） $p(\alpha)$ を導入して、事後分布 $p(\alpha|X)$ を計算します。

$$\begin{aligned} p(\alpha|X) &= \frac{p(X|\Theta)p(\Theta|\alpha)p(\alpha)}{p(X)} \\ &= \frac{p(X|\Theta)p(\Theta|\alpha)p(\alpha)}{\int \int p(X|\Theta)p(\Theta|\alpha)p(\alpha)d\Theta d\alpha} \end{aligned} \quad (2.45)$$

経験ベイズ法では、超パラメータ α を点推定しているのに対し、階層ベイズ法では、未知である α の不確実性を適切に取り扱う点で優れています。しかし、一般に、式 (2.45) の分母の積分計算は極めて困難であり、VB もギブスサンプリングも利用できない場合がほとんどです。そのため、一般には、MCMC を用いたサンプリングが行われます。

この方法では、超パラメータ α のベイズ推定が可能になる代わりに、超事前分布 $p(\alpha)$ 自体を適切に設定するという新たな問題が生まれます。しかし、現実には、超パラメータ α に関する事前知識はほとんどないことが多く、 α に対しては無情報事前分布がよく用いられます。このように、ベイズモデルにおいては、階層が上がるにつれて、パラメータの抽象度が高くなるので、曖昧性の高い超事前分布を設定しても問題ないことがほとんどです。むしろ、事前知識がないにもかかわらず、超事前分布を少し変えるだけで、 Θ や α の推定に大きな影響が出る場合、そのベイズモデルは適切ではない可能性があります。

ベイズ最適化 (Bayesian optimization)

経験ベイズ法が、周辺尤度 $p(X|\alpha)$ を最大化する超パラメータ α を求めていたのに対して、ベイズ最適化では、任意の目的関数を最大化する α^* を効率的に探索する方法を提供します。例えば、ベイズモデルに基づく音声認識システムであれば、周辺尤度ではなく、より直接的に、音声認識率を最大化するような α^* を求めたいでしょう。もし、 α が高々 2, 3 個の変数から構成されている場合は、グリッドサーチが利用できるかもしれません。しかし、変数の個数が増えてくると、試すべき組み合わせの数は指数的に爆発します。そもそも、 $f(\alpha)$ を 1 回評価するのに時間がかかる場合は、試行回数をできる限り削減することが望まれます。

ベイズ最適化では、音声認識システムを、 α を引数に取り、認識率を返す未知のブラックボックス関数 $f(\alpha)$ とみなします。これまでの N 回の試行結果から、 N 個の入力 $\{\alpha^{(1)}, \dots, \alpha^{(N)}\}$ に対する関数値 $\{f(\alpha^{(1)}), \dots, f(\alpha^{(N)})\}$ が分かっていたとします。このとき、次に試すべき $\alpha^{(N+1)}$ として、 $f(\alpha)$ が大きな値をとることが判明している α の周辺を調査することが考えられます。一方、まだ探索が進んでいない空間に、より大きな $f(\alpha)$ をとりうる α が存在する可能性もあります。ベイズ最適化では、 $f(\alpha)$ がガウス過程 (Gaussian process) に従うと仮定したうえで、活用 (exploitation) と探索 (exploration) のトレードオフを考慮しながら、次に試すべき $\alpha^{(N+1)}$ を提案します。この提案方法にはいくつかの方法が知られており [?, ?, ?, ?]、どれが良いかは関数の性質によって異なります。汎用的な最適化技法であるため、ソフトウェアも公開されており、簡単に利用可能です [?].

2.3 混合ベイズモデルの学習

本節では、代表的な潜在変数モデルである混合ガウスモデル (Gaussian mixture model, GMM) を例に、EM アルゴリズム、VB、ギブスサンプリング、周辺化ギブスサンプリングを適用する方法について解説します。GMM は最も基本的かつ重要な確率モデルの一つであり、音響信号処理分野における様々な場面で利用されています。

多次元特徴量ベクトルの分布の学習

音響信号から抽出した多次元の特徴量ベクトルの分布を表現するうえで、GMM は最初に検討すべき標準的なモデルです。例えば、音声認識システムにおいては、各音素について、音声スペクトルから抽出した 13 次元程度のメル周波数ケプストラム係数 (mel-frequency cepstrum coefficients) の分布を GMM を用いて表現するのが一般的でした（近年は深層学習を用いる場合が主流）[?]. また、楽曲検索システムにおいて、ある楽曲と類似した楽曲を検索するうえで、各楽曲の音楽音響信号から抽出した音響的特徴量の分布を GMM で表現し、二つの GMM 間の距離を計算することがしばしば行われます [?].

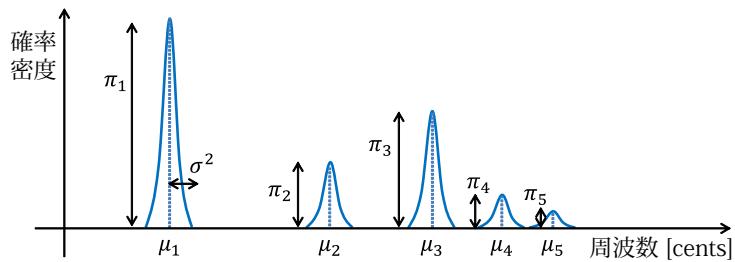


図 2.10 調波構造を表現する GMM . 観測データ \mathbf{X} として振幅スペクトルが与えられた時に、各種パラメータ $\Theta = \{\mu_1, \dots, \mu_5, \pi_1, \dots, \pi_5, \sigma^2\}$ を最尤推定あるいはベイズ推定する問題を解けば、基本周波数 μ_1 が推定できる。

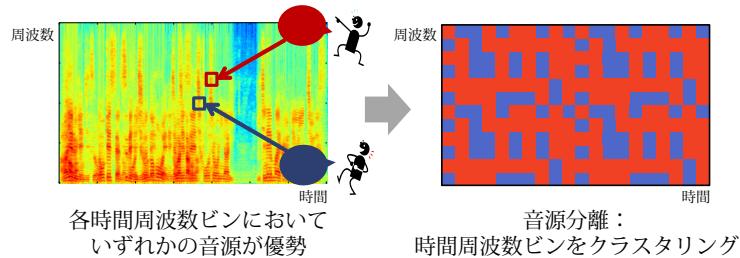


図 2.11 時間周波数クラスタリングに基づくマルチチャネル音源分離。

基本周波数推定

マルチチャネル音源分離

比較的単純にもかかわらず、柔軟な表現力を持つことから、観測データとして N 個の D 次元ベクトル $\mathbf{X} = \{x_1, \dots, x_N\}$ を考える。また、観測データ \mathbf{X} に対応する潜在変数系列を $Z = \{z_1, \dots, z_N\}$ とする。ここでは可算無限個のガウス分布の混合を許容するモデルを考えているので、 z_n は選ばれたガウス分布に対応する次元のみが 1 で他は 0 であるような $K \rightarrow \infty$ 次元のベクトルである。このとき、グラフィカルモデルから変数間の条件つき独立性を考慮すると、完全な同時分布は

$$p(\mathbf{X}, Z, \pi, \boldsymbol{\mu}, \boldsymbol{\Lambda}) = p(\mathbf{X}|Z, \boldsymbol{\mu}, \boldsymbol{\Lambda})p(Z|\pi)p(\pi)p(\boldsymbol{\mu}, \boldsymbol{\Lambda}) \quad (2.46)$$

で与えられる。ここで、 π は無限個のガウス分布に対する混合係数で、無限次元のベクトルである。 μ および Λ は各ガウス分布のパラメータ（平均 μ_k および分散 Λ_k^{-1} ）である。まず、第一項には尤度を設定する。

$$p(\mathbf{X}|\mathbf{Z}, \boldsymbol{\mu}, \boldsymbol{\Lambda}) = \prod_{n=1}^N \prod_{k=1}^{\infty} \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k^{-1})^{z_{n,k}} \quad (2.47)$$

いま、集中度 α ・基底測度 G_0 であるディリクレ過程 $DP(\alpha, G_0)$ を考える。基底測度 G_0 として、 μ, Λ 上の連続分布を与えることにする。このとき、 μ, Λ 上の別の分布 G を $G \sim DP(\alpha, G_0)$ として生成することができる。こうすると、 G は可算無限次元の離散分布となり、ある次元 k の重みが π_k に、実現値が μ_k, Λ_k に対応する。すなわち、無限混合ガウス分布が具体的にひとつ定まる。ここで、 G_0 は G の期待値となっており、 α が大きいほど G_0 に近くなる。ディリクレ過程の一つの実現方法として、ここでは Stick-Breaking Construction (SB 過程) を用いる。SBC は変分ベイズ法を適用するうえで都合が良い DP の表現方法である。このとき、混合係数 π_k は次式で与えられる。

$$\pi_k = v_k \prod_{k'=1}^{k-1} (1 - v_{k'}) \quad (2.48)$$

$$v_k \sim \text{Beta}(1, \alpha) \quad (2.49)$$

このように π_k を v_k に変数変換を行うことで、式 (2.46) の第二項および第三項の積 $p(\mathbf{Z}, \boldsymbol{\pi}) = p(\mathbf{Z}|\boldsymbol{\pi})p(\boldsymbol{\pi})$ は積 $p(\mathbf{Z}, \mathbf{v}) = p(\mathbf{Z}|\mathbf{v})p(\mathbf{v})$ として書き直せる。このとき、各項は以下で与えられる。

$$\begin{aligned} p(\mathbf{Z}|\mathbf{v}) &= \prod_{n=1}^N \prod_{k=1}^{\infty} \pi_k^{z_{n,k}} \\ &= \prod_{n=1}^N \prod_{k=1}^{\infty} \left(v_k \prod_{k'=1}^{k-1} (1 - v_{k'}) \right)^{z_{n,k}} \\ &= \prod_{n=1}^N \left(\prod_{k=1}^{\infty} v_k^{z_{n,k}} \right) \left(\prod_{k=1}^{\infty} \prod_{k'=1}^{k-1} (1 - v_{k'})^{z_{n,k}} \right) \end{aligned}$$

$$\begin{aligned}
&= \prod_{n=1}^N \left(\prod_{k=1}^{\infty} v_k^{z_{n,k}} \right) (1-v_1)^{z_{n,2}} ((1-v_1)(1-v_2))^{z_{n,3}} ((1-v_1)(1-v_2)(1-v_3))^{z_{n,4}} \cdots \\
&= \prod_{n=1}^N \prod_{k=1}^{\infty} v_k^{z_{n,k}} (1-v_k)^{\sum_{k'=k+1}^{\infty} z_{n,k'}} \tag{2.50}
\end{aligned}$$

$$p(\mathbf{v}) = \prod_{k=1}^{\infty} \text{Beta}(v_k | 1, \alpha) = \prod_{k=1}^{\infty} \frac{\Gamma(1+\alpha)}{\Gamma(1)\Gamma(\alpha)} v_k^{1-1} (1-v_k)^{\alpha-1} = \prod_{k=1}^{\infty} \alpha (1-v_k)^{\alpha-1} \tag{2.51}$$

式 (2.46) の第四項には、基底測度 G_0 として、ガウス分布の共役事前分布であるガウス・ウィシャート分布を設定する。

$$p(\boldsymbol{\mu}, \boldsymbol{\Lambda}) = \prod_{k=1}^{\infty} \mathcal{N}(\boldsymbol{\mu}_k | \mathbf{m}_0, (b_0 \boldsymbol{\Lambda}_k)^{-1}) \mathcal{W}(\boldsymbol{\Lambda}_k | \mathbf{W}_0, c_0) \tag{2.52}$$

ここで、 \mathbf{m}_0 、 b_0 、 \mathbf{W}_0 および c_0 はハイパーパラメータである。通常はすべての k について同じ事前分布を与える。

2.3.1 最尤推定：EM アルゴリズム

2.3.2 ベイズ推定：変分ベイズ法

2.4 变分事後分布

ベイズ推定の目的は、観測データ X が与えられたときに、事後分布 $p(Z, v, \boldsymbol{\mu}, \boldsymbol{\Lambda} | X)$ を求めることがある。しかし、これを解析的に計算することは困難なので、变分事後分布 $q(Z, v, \boldsymbol{\mu}, \boldsymbol{\Lambda})$ を導入し、できるかぎり真の事後分布に近づけるよう最適化を行いたい。いま、变分事後分布の潜在変数とパラメータへの因子分解

$$q(Z, v, \boldsymbol{\mu}, \boldsymbol{\Lambda}) = q(Z)q(v, \boldsymbol{\mu}, \boldsymbol{\Lambda}) \tag{2.53}$$

を考える。これがベイズ推定に变分ベイズ法を適用する場合の唯一の仮定である。变分ベイズ法は、事後分布の関数形を解析的に導出可能な形に制限し、その中で最適なものを探す手法である。これはさらに

$$q(Z, v, \boldsymbol{\mu}, \boldsymbol{\Lambda}) = q(Z)q(v)q(\boldsymbol{\mu}|\boldsymbol{\Lambda})q(\boldsymbol{\Lambda}) \tag{2.54}$$

と因子分解できる。あとで見るように、これは仮定や近似ではなく、グラフィカルモデルの構造から必然的に導かれる。

無限混合モデルに変分ベイズ法を適用する場合、さらに、十分に大きいある整数 $k > K$ について、

$$q(z_{n,k>K}) = 0 \quad (2.55)$$

を仮定し、可算無限個あるガウス分布のうち、観測データ中には $K+1$ 番目以降に割り当てられるサンプルが存在しなかったとする。ここで、混合数の打ち切り (truncate) は、事前分布 $p(v)$ 、真の事後分布 $p(v|X)$ あるいは変分事後分布 $q(v)$ に対してではなく、変分事後分布 $q(Z)$ に対して行っていることに注意する。この仮定のもとでは、 $k \leq K$ における $q(v_k)$ を考慮すれば十分であり、 K の値に関して推論結果がネストされることになる（大きな K における結果が小さな K における結果を含む）。したがって、実質的には有限混合モデルに対する事後分布推論を行うが、理論的には完全な無限混合モデルとして取り扱うことができる。一方、 $q(v_K = 1) = 1$ とし、 $K+1$ 回以上の Stick Breaking が発生しないとする仮定では、推論結果はネストされないという違いがある。

このように、SBC に基づくベイズ推論では、式 (2.52) のように、基底測度 G_0 から得られる離散分布 G として、十分に多い K 個のガウス分布を考慮する必要がある。一方、別のディリクレ過程の実現方法である Chinese Restaurant Process (CRP) では、過去の潜在変数系列 $\{z_1, \dots, z_n\}$ から次の潜在変数 z_{n+1} の予測分布を考える。CRP は、有限混合モデルにおけるパラメータ π を積分消去し、混合数を無限としたときの極限と解釈できる。MCMC を利用すれば、混合数が増加した場合に、隨時新たな要素分布 G をサンプルして増やしていくことができる。

2.4.1 VB-E ステップ

まず、VB-E ステップでは因子 $q(Z)$ について考える。一般的な結果を用いると、最適な因子は

$$\log q^*(Z) = \mathbb{E}_{v,\mu,\Lambda} [\log p(\mathbf{X}, Z, v, \mu, \Lambda)] + \text{const.} \quad (2.56)$$

で与えられる。式 (2.46) を代入し、右辺で変数 Z への依存関係だけに興味

あることに注意すると ,

$$\log q^*(\mathbf{Z}) = \mathbb{E}_{\boldsymbol{\mu}, \boldsymbol{\Lambda}} [\log p(\mathbf{X} | \mathbf{Z}, \boldsymbol{\mu}, \boldsymbol{\Lambda})] + \mathbb{E}_{\mathbf{v}} [\log p(\mathbf{Z} | \mathbf{v})] + \text{const.} \quad (2.57)$$

を得る . \mathbf{Z} に依存関係のない項はすべて正規化定数 const. に含まれるので , 必要に応じて計算すればよい .

$$\begin{aligned} \log q^*(\mathbf{Z}) &= \mathbb{E}_{\boldsymbol{\mu}, \boldsymbol{\Lambda}} \left[\sum_{n=1}^N \sum_{k=1}^K z_{n,k} \log \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k^{-1}) \right] \\ &\quad + \mathbb{E}_{\mathbf{v}} \left[\sum_{n=1}^N \sum_{k=1}^K \left(z_{n,k} \log v_k + z_{n,k} \sum_{k'=1}^{k-1} \log(1 - v_{k'}) \right) \right] + \text{const.} \\ &= \sum_{n=1}^N \sum_{k=1}^K z_{n,k} \left(\mathbb{E}_{v_k} [\log v_k] + \sum_{k'=1}^{k-1} \mathbb{E}_{v_{k'}} [\log(1 - v_{k'})] + \mathbb{E}_{\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k} [\log \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k^{-1})] \right) + \text{const.} \\ &= \sum_{n=1}^N \sum_{k=1}^K z_{n,k} \log \rho_{n,k} + \text{const.} \end{aligned} \quad (2.58)$$

ここで ,

$$\log \rho_{n,k} = \mathbb{E}_{v_k} [\log v_k] + \sum_{k'=1}^{k-1} \mathbb{E}_{v_{k'}} [\log(1 - v_{k'})] + \mathbb{E}_{\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k} [\log \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k^{-1})]$$

とした . 上式中の期待値は VB-M ステップにおいて式 (2.72) , 式 (2.73) や
および式 (2.87) として得られる .

式 (2.58) の両辺の対数をとると

$$q^*(\mathbf{Z}) \propto \prod_{n=1}^N \prod_{k=1}^K \rho_{n,k}^{z_{n,k}} \quad (2.60)$$

を得る。この分布は正しく正規化されている必要があること、任意の n, k の値について $z_{n,k}$ は 1 あるいは 0 をとり、すべての k にわたる和が 1 であることに注意すると

$$q^*(\mathbf{Z}) = \prod_{n=1}^N \prod_{k=1}^K \gamma_{n,k}^{z_{n,k}} \quad (2.61)$$

を得る。ここで、量 γ_n はデータ n に対する要素分布 k の負担率であり

$$\gamma_{n,k} = \frac{\rho_{n,k}}{\sum_{k'=1}^K \rho_{n,k'}} \quad (2.62)$$

で与えられる。式 (2.61) で与えられる因子 $q(\mathbf{Z})$ の最適解 $q^*(\mathbf{Z})$ は、式 (2.50) で与えられる事前分布 $p(\mathbf{Z}|\mathbf{v})$ と同じ形をしている。 $q(\mathbf{Z})$ の関数形に関する仮定を導入していないにもかかわらず、式 (2.53) の因子分解とグラフィカルモデルの構造からこの結果は必然的に導かれる。式 (2.61) から、潜在変数 z_n はパラメータ γ_n をもつ多項分布に従うことが分かる。したがって、 $z_{n,k}$ の期待値は次式で与えられる。

$$\mathbb{E}_{\mathbf{z}_n}[z_{n,k}] = \gamma_{n,k} \quad (2.63)$$

2.4.2 VB-M ステップ

次に、VB-M ステップでは因子 $q(\mathbf{v}, \boldsymbol{\mu}, \boldsymbol{\Lambda})$ について考える。一般的な結果を再度用いると、最適な因子は

$$\begin{aligned} \log q^*(\mathbf{v}, \boldsymbol{\mu}, \boldsymbol{\Lambda}) &= \mathbb{E}_{\mathbf{z}} [\log p(\mathbf{v}) + \log p(\mathbf{Z}|\mathbf{v})] + \mathbb{E}_{\mathbf{z}} [\log p(\boldsymbol{\mu}, \boldsymbol{\Lambda}) + \log p(\mathbf{X}|\mathbf{Z}, \boldsymbol{\mu}, \boldsymbol{\Lambda})] + \text{const.} \\ &= \log p(\mathbf{v}) + \mathbb{E}_{\mathbf{z}} [\log p(\mathbf{Z}|\mathbf{v})] + \log p(\boldsymbol{\mu}, \boldsymbol{\Lambda}) + \mathbb{E}_{\mathbf{z}} [\log p(\mathbf{X}|\mathbf{Z}, \boldsymbol{\mu}, \boldsymbol{\Lambda})] + (2.64) \end{aligned}$$

で与えられる。ここで、VB-E ステップの式 (2.63) を用いると、式 (2.64) 中の期待値は次式で計算できる。

$$\mathbb{E}_{\mathbf{z}} [\log p(\mathbf{Z}|\mathbf{v})] = \mathbb{E}_{\mathbf{z}} \left[\sum_{n=1}^N \sum_{k=1}^K \left(z_{n,k} \log v_k + \left(\sum_{k'=k+1}^K z_{n,k'} \right) \log(1 - v_k) \right) \right]$$

$$= \sum_{n=1}^N \sum_{k=1}^K \left(\gamma_{n,k} \log v_k + \left(\sum_{k'=k+1}^K \gamma_{n,k'} \right) \log(1-v_k) \right) \quad (2.65)$$

$$\begin{aligned} \mathbb{E}_z [\log p(\mathbf{X}|\mathbf{Z}, \boldsymbol{\mu}, \boldsymbol{\Lambda})] &= \mathbb{E}_z \left[\sum_{n=1}^N \sum_{k=1}^K z_{n,k} \log \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k^{-1}) \right] \\ &= \sum_{n=1}^N \sum_{k=1}^K \gamma_{n,k} \log \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k^{-1}) \end{aligned} \quad (2.66)$$

式 (2.64) をよく観察すると, v のみを含む項, $\boldsymbol{\mu}, \boldsymbol{\Lambda}$ のみを含む項の和に分解できることがわかる。それぞれはさらに k を含む項ごとの和に分解できる。すなわち, 最適な因子 $q^*(v, \boldsymbol{\mu}, \boldsymbol{\Lambda})$ は

$$q^*(v, \boldsymbol{\mu}, \boldsymbol{\Lambda}) = \prod_{k=1}^K q^*(v_k) \prod_{k=1}^K q^*(\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k) \quad (2.67)$$

と分解できる。このような分解が可能であることもグラフィカルモデルの構造から必然的に導かれる。

式 (2.64) に式 (2.51) および式 (2.52) を代入し, 式 (2.67) と比較することで, 最適な因子 $q^*(v)$ は

$$\log q^*(v_k) = \left(1 + \sum_{n=1}^N \gamma_{n,k} - 1 \right) \log v_k + \left(\alpha + \sum_{n=1}^N \sum_{k'=k+1}^K \gamma_{n,k'} - 1 \right) \log(1-v_k) + \text{(2.68)}$$

で与えられる。両辺の指數をとると, 最適な因子はベータ分布

$$q^*(v_k) = \text{Beta}(v_k | \alpha_k) \quad (2.69)$$

で与えられることが分かる。これらはやはり式 (2.51) で与えられる事前分布と同じ形をしている。このとき, ベータ分布のパラメータ α_k は

$$\alpha_{k,1} = 1 + \sum_{n=1}^N \gamma_{n,k} \quad (2.70)$$

$$\alpha_{k,2} = \alpha + \sum_{n=1}^N \sum_{k'=k+1}^K \gamma_{n,k'} \quad (2.71)$$

で定まる。ここで, 变数 v_k がパラメータ α_k をもつベータ分布に従うことに

着目すると、 $\log v_k$ および $\log(1 - v_k)$ の期待値は標準的な公式から

$$\mathbb{E}_{v_k}[\log v_k] = \psi(\alpha_{k,1}) - \psi(\alpha_{k,1} + \alpha_{k,2}) \quad (2.72)$$

$$\mathbb{E}_{v_k}[\log(1 - v_k)] = \psi(\alpha_{k,2}) - \psi(\alpha_{k,1} + \alpha_{k,2}) \quad (2.73)$$

と計算できる。ここで、 $\psi(\cdot)$ はディガンマ関数（対数ガンマ関数の導関数）である。

最後に、因子 $q^*(\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k)$ について考える。事前分布として共役事前分布を与えたため、事後分布は事前分布と同じガウス・ウィシャート分布になるはずである。この導出を行う前に、十分統計量

$$\mathbb{S}_k[1] \equiv \sum_{n=1}^N \gamma_{n,k} \quad (2.74)$$

$$\mathbb{S}_k[\mathbf{x}] \equiv \sum_{n=1}^N \gamma_{n,k} \mathbf{x}_n \quad (2.75)$$

$$\mathbb{S}_k[\mathbf{x}\mathbf{x}^T] \equiv \sum_{n=1}^N \gamma_{n,k} \mathbf{x}_n \mathbf{x}_n^T \quad (2.76)$$

を定義しておく。いま、式 (2.64) の右辺で興味がある $\boldsymbol{\mu}_k$ および $\boldsymbol{\Lambda}_k$ を含む項を取り出すと

$$\begin{aligned} \log q^*(\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k) &= \log \mathcal{N}(\boldsymbol{\mu}_k | \mathbf{m}_0, (b_0 \boldsymbol{\Lambda}_k)^{-1}) + \log \mathcal{W}(\boldsymbol{\Lambda}_k | \mathbf{W}_0, c_0) + \sum_{n=1}^N \gamma_{n,k} \log \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k^{-1}) + \text{const.} \\ &= \frac{1}{2} \log |\boldsymbol{\Lambda}_k| - \frac{b_0}{2} (\boldsymbol{\mu}_k - \mathbf{m}_0)^T \boldsymbol{\Lambda}_k (\boldsymbol{\mu}_k - \mathbf{m}_0) + \frac{c_0 - D - 1}{2} \log |\boldsymbol{\Lambda}_k| - \frac{1}{2} \text{Tr}(\mathbf{W}_0^{-1} \boldsymbol{\Lambda}_k) \\ &\quad + \frac{1}{2} \sum_{n=1}^N \gamma_{n,k} \log |\boldsymbol{\Lambda}_k| - \frac{1}{2} \sum_{n=1}^N \gamma_{n,k} (\mathbf{x}_n - \boldsymbol{\mu}_k)^T \boldsymbol{\Lambda}_k (\mathbf{x}_n - \boldsymbol{\mu}_k) + \text{const.} \end{aligned} \quad (2.77)$$

を得る。まず、 $p(\boldsymbol{\mu}_k | \boldsymbol{\Lambda}_k)$ に対応する部分、すなわち $\boldsymbol{\mu}_k$ を含む項をとりだすと

$$\log q^*(\boldsymbol{\mu}_k | \boldsymbol{\Lambda}_k) = -\frac{b_0}{2} (\boldsymbol{\mu}_k - \mathbf{m}_0)^T \boldsymbol{\Lambda}_k (\boldsymbol{\mu}_k - \mathbf{m}_0) - \frac{1}{2} \sum_{n=1}^N \gamma_{n,k} (\mathbf{x}_n - \boldsymbol{\mu}_k)^T \boldsymbol{\Lambda}_k (\mathbf{x}_n - \boldsymbol{\mu}_k) + \text{const.}$$

$$\begin{aligned}
&= -\frac{1}{2} \boldsymbol{\mu}_k^T \boldsymbol{\Lambda}_k \boldsymbol{\mu}_k \left(b_0 + \sum_{n=1}^N \gamma_{n,k} \right) + \frac{1}{2} \boldsymbol{\mu}_k^T \boldsymbol{\Lambda}_k \left(b_0 \mathbf{m}_0 + \sum_{n=1}^N \gamma_{n,k} \mathbf{x}_n \right) + \frac{1}{2} \left(b_0 \mathbf{m}_0 + \sum_{n=1}^N \gamma_{n,k} \mathbf{x}_n \right)^T \\
&\quad - \frac{b_0}{2} \mathbf{m}_0^T \boldsymbol{\Lambda}_k \mathbf{m}_0 - \frac{1}{2} \sum_{n=1}^N \gamma_{n,k} \mathbf{x}_n^T \boldsymbol{\Lambda}_k \mathbf{x}_n + \text{const.} \\
&= -\frac{1}{2} \left(b_0 + \mathbb{S}_k[1] \right) \left(\boldsymbol{\mu}_k^T \boldsymbol{\Lambda}_k \boldsymbol{\mu}_k - \boldsymbol{\mu}_k^T \boldsymbol{\Lambda}_k \frac{b_0 \mathbf{m}_0 + \mathbb{S}_k[\mathbf{x}]}{b_0 + \mathbb{S}_k[1]} - \left(\frac{b_0 \mathbf{m}_0 + \mathbb{S}_k[\mathbf{x}]}{b_0 + \mathbb{S}_k[1]} \right)^T \boldsymbol{\Lambda}_k \boldsymbol{\mu}_k \right) \\
&\quad - \frac{b_0}{2} \mathbf{m}_0^T \boldsymbol{\Lambda}_k \mathbf{m}_0 - \frac{1}{2} \sum_{n=1}^N \gamma_{n,k} \mathbf{x}_n^T \boldsymbol{\Lambda}_k \mathbf{x}_n + \text{const.}
\end{aligned}$$

を得る。よって、 $q^*(\boldsymbol{\mu}_k | \boldsymbol{\Lambda}_k)$ はガウス分布

$$q^*(\boldsymbol{\mu}_k | \boldsymbol{\Lambda}_k) = \mathcal{N} \left(\boldsymbol{\mu}_k | \mathbf{m}_k, (b_k \boldsymbol{\Lambda}_k)^{-1} \right) \quad (2.79)$$

となることが分かり、そのパラメータは次式で定まる。

$$b_k = b_0 + \mathbb{S}_k[1] \quad (2.80)$$

$$\mathbf{m}_k = \frac{b_0 \mathbf{m}_0 + \mathbb{S}_k[\mathbf{x}]}{b_0 + \mathbb{S}_k[1]} = \frac{b_0 \mathbf{m}_0 + \mathbb{S}_k[\mathbf{x}]}{b_k} \quad (2.81)$$

ここで、 b_0 は事前に \mathbf{m}_0 を観測した回数、 $\mathbb{S}_k[1]$ は要素分布 k からデータ \mathbf{x} を観測した実効的な回数（負担率の総和）と解釈できる。したがって、 \mathbf{m}_k は事前知識とデータから定まる値との重みつき和となっている。観測データ数が増えるにしたがって b_k は単調増加し、 $\boldsymbol{\mu}_k$ の事後分布の分散は小さくなる（不確かさが減少する）。

次に $q^*(\boldsymbol{\Lambda}_k)$ について考える。 $\log q^*(\boldsymbol{\Lambda}_k) = \log q^*(\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k) - \log q^*(\boldsymbol{\mu}_k | \boldsymbol{\Lambda}_k)$ が成立するので、式 (2.77) から式 (2.79) を引けばよい。このとき、 $\boldsymbol{\Lambda}_k$ に関する項のみを取り出すと

$$\begin{aligned}
\log q^*(\boldsymbol{\Lambda}_k) &= \frac{1}{2} \log |\boldsymbol{\Lambda}_k| - \frac{b_0}{2} (\boldsymbol{\mu}_k - \mathbf{m}_0)^T \boldsymbol{\Lambda}_k (\boldsymbol{\mu}_k - \mathbf{m}_0) + \frac{c_0 - D - 1}{2} \log |\boldsymbol{\Lambda}_k| - \frac{1}{2} \text{Tr} (\mathbf{W}_0^{-1} \boldsymbol{\Lambda}_k) \\
&\quad + \frac{1}{2} \sum_{n=1}^N \gamma_{n,k} \log |\boldsymbol{\Lambda}_k| - \frac{1}{2} \sum_{n=1}^N \gamma_{n,k} (\mathbf{x}_n - \boldsymbol{\mu}_k)^T \boldsymbol{\Lambda}_k (\mathbf{x}_n - \boldsymbol{\mu}_k) \\
&\quad - \frac{1}{2} \log |\boldsymbol{\Lambda}_k| + \frac{b_k}{2} (\boldsymbol{\mu}_k - \mathbf{m}_k)^T \boldsymbol{\Lambda}_k (\boldsymbol{\mu}_k - \mathbf{m}_k) + \text{const.}
\end{aligned}$$

$$\begin{aligned}
&= \frac{c_0 + \mathbb{S}_k[1] - D - 1}{2} \log |\Lambda_k| - \frac{1}{2} \text{Tr} \left(b_0 (\mu_k - m_0) (\mu_k - m_0)^T \Lambda_k \right) \\
&\quad - \frac{1}{2} \text{Tr} (\mathbf{W}_0^{-1} \Lambda_k) - \frac{1}{2} \text{Tr} \left(\sum_{n=1}^N \gamma_{n,k} (\mathbf{x}_n - \mu_k) (\mathbf{x}_n - \mu_k)^T \Lambda_k \right) \\
&\quad + \frac{1}{2} \text{Tr} \left(b_k (\mu_k - m_k) (\mu_k - m_k)^T \Lambda_k \right) + \text{const.}
\end{aligned} \tag{2.82}$$

を得る。ここで、任意の正定値対称行列 Λ とベクトル x について、 $x^T \Lambda x = \text{Tr}(xx^T \Lambda)$ が成立することを用いた。よって、 $q^*(\Lambda_k)$ はウィシャート分布

$$q^*(\Lambda_k) = \mathcal{W}(\Lambda_k | \mathbf{W}_k, c_k) \tag{2.83}$$

となることが分かり、そのパラメータは次式で求まる。

$$\begin{aligned}
c_k &= c_0 + \mathbb{S}_k[1] \\
\mathbf{W}_k^{-1} &= \mathbf{W}_0^{-1} + b_0 (\mu_k - m_0) (\mu_k - m_0)^T \\
&\quad + \sum_{n=1}^N \gamma_{n,k} (\mathbf{x}_n - \mu_k) (\mathbf{x}_n - \mu_k)^T \Lambda_k - b_k (\mu_k - m_k) (\mu_k - m_k)^T \\
&= \mathbf{W}_0^{-1} + b_0 \mathbf{m}_0 \mathbf{m}_0^T - b_0 \mathbf{m}_0 \mu_k^T - b_0 \mu_k \mathbf{m}_0^T + b_0 \mu_k \mu_k^T \\
&\quad + \mathbb{S}_k[\mathbf{x}\mathbf{x}^T] - \mathbb{S}_k[\mathbf{x}] \mu_k^T - \mu_k \mathbb{S}_k[\mathbf{x}]^T + \mathbb{S}_k[1] \mu_k \mu_k^T - b_k \mu_k \mu_k^T + b_k \mathbf{m}_k \mu_k^T + b_k \mu_k \mathbf{m}_k^T - b_k \mathbf{m}_k \mathbf{m}_k^T \\
&= \mathbf{W}_0^{-1} + b_0 \mathbf{m}_0 \mathbf{m}_0^T + \mathbb{S}_k[\mathbf{x}\mathbf{x}^T] - b_k \mathbf{m}_k \mathbf{m}_k^T \\
&\quad + (b_0 + \mathbb{S}_k[1] - b_k) \mu_k \mu_k^T - (b_0 \mathbf{m}_0 + \mathbb{S}_k[\mathbf{x}] - b_k \mathbf{m}_k) \mu_k^T - \mu_k (b_0 \mathbf{m}_0 + \mathbb{S}_k[\mathbf{x}] - b_k \mathbf{m}_k)^T \\
&= \mathbf{W}_0^{-1} + b_0 \mathbf{m}_0 \mathbf{m}_0^T + \mathbb{S}_k[\mathbf{x}\mathbf{x}^T] - b_k \mathbf{m}_k \mathbf{m}_k^T
\end{aligned} \tag{2.85}$$

ここで、式(2.80)および式(2.81)を用いて μ_k を含む項を消去した。自由度 c_k が実効的な観測回数に合わせて自動的に調節されていることが分かる。
実際の計算には上式を用いるのが都合がよいが、これをさらに変形すると

$$\mathbf{W}_k^{-1} = \mathbf{W}_0^{-1} + \frac{b_0 \mathbb{S}_k[1]}{b_0 + \mathbb{S}_k[1]} (\mathbb{E}_k[\mathbf{x}] - \mathbf{m}_0) (\mathbb{E}_k[\mathbf{x}] - \mathbf{m}_0)^T + \mathbb{E}_k [(\mathbf{x} - \mathbb{E}_k[\mathbf{x}])(\mathbf{x} - \mathbb{E}_k[\mathbf{x}])^T] \tag{2.86}$$

が得られ、やはり事前知識とデータから定まる部分との重みつき和となっていることが分かる。

これで最適事後分布 $q^*(\mu_k, \Lambda_k)$ が定まったので、式(2.59)の第三項の期待値について考える。

$$\begin{aligned}
& \mathbb{E}_{\mu_k, \Lambda_k} [\log \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k^{-1})] = \iint q(\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k) \log \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k^{-1}) d\boldsymbol{\mu}_k d\boldsymbol{\Lambda}_k \\
&= \iint q(\boldsymbol{\mu}_k | \boldsymbol{\Lambda}_k) q(\boldsymbol{\Lambda}_k) \left(\frac{1}{2} \log |\boldsymbol{\Lambda}_k| - \frac{D}{2} \log(2\pi) - \frac{1}{2} (\mathbf{x}_n - \boldsymbol{\mu}_k)^T \boldsymbol{\Lambda}_k (\mathbf{x}_n - \boldsymbol{\mu}_k) \right) d\boldsymbol{\mu}_k d\boldsymbol{\Lambda}_k \\
&= -\frac{D}{2} \log(2\pi) + \frac{1}{2} \mathbb{E}_{\boldsymbol{\Lambda}_k} [\log |\boldsymbol{\Lambda}_k|] - \frac{1}{2} \mathbb{E}_{\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k} [(\mathbf{x}_n - \boldsymbol{\mu}_k)^T \boldsymbol{\Lambda}_k (\mathbf{x}_n - \boldsymbol{\mu}_k)] \quad (2.87)
\end{aligned}$$

ここで標準的な公式から

$$\mathbb{E}_{\boldsymbol{\Lambda}_k} [\log |\boldsymbol{\Lambda}_k|] = \sum_{d=1}^D \psi \left(\frac{c_k + 1 - d}{2} \right) + D \log 2 + \log |\mathbf{W}_k| \quad (2.88)$$

となる。ここでもディガンマ関数 ψ を用いた。また、もう一方の期待値を書き直すと次式を得る。

$$\begin{aligned}
& \mathbb{E}_{\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k} [(\mathbf{x}_n - \boldsymbol{\mu}_k)^T \boldsymbol{\Lambda}_k (\mathbf{x}_n - \boldsymbol{\mu}_k)] \\
&= \iint q(\boldsymbol{\mu}_k | \boldsymbol{\Lambda}_k) q(\boldsymbol{\Lambda}_k) (\mathbf{x}_n - \boldsymbol{\mu}_k)^T \boldsymbol{\Lambda}_k (\mathbf{x}_n - \boldsymbol{\mu}_k) d\boldsymbol{\mu}_k d\boldsymbol{\Lambda}_k \\
&= \iint q(\boldsymbol{\mu}_k | \boldsymbol{\Lambda}_k) q(\boldsymbol{\Lambda}_k) (\mathbf{x}_n - \mathbf{m}_k + \mathbf{m}_k - \boldsymbol{\mu}_k)^T \boldsymbol{\Lambda}_k (\mathbf{x}_n - \mathbf{m}_k + \mathbf{m}_k - \boldsymbol{\mu}_k) d\boldsymbol{\mu}_k d\boldsymbol{\Lambda}_k \\
&= \iint q(\boldsymbol{\mu}_k | \boldsymbol{\Lambda}_k) q(\boldsymbol{\Lambda}_k) (\mathbf{x}_n - \mathbf{m}_k)^T \boldsymbol{\Lambda}_k (\mathbf{x}_n - \mathbf{m}_k) d\boldsymbol{\mu}_k d\boldsymbol{\Lambda}_k \\
&\quad + 2 \iint q(\boldsymbol{\mu}_k | \boldsymbol{\Lambda}_k) q(\boldsymbol{\Lambda}_k) (\mathbf{x}_n - \mathbf{m}_k)^T \boldsymbol{\Lambda}_k (\mathbf{m}_k - \boldsymbol{\mu}_k) d\boldsymbol{\mu}_k d\boldsymbol{\Lambda}_k \\
&\quad + \iint q(\boldsymbol{\mu}_k | \boldsymbol{\Lambda}_k) q(\boldsymbol{\Lambda}_k) (\mathbf{m}_k - \boldsymbol{\mu}_k)^T \boldsymbol{\Lambda}_k (\mathbf{m}_k - \boldsymbol{\mu}_k) d\boldsymbol{\mu}_k d\boldsymbol{\Lambda}_k \\
&= c_k (\mathbf{x}_n - \mathbf{m}_k)^T \mathbf{W}_k (\mathbf{x}_n - \mathbf{m}_k) \\
&\quad + b_k^{-1} \int q(\boldsymbol{\Lambda}_k) \int q(\boldsymbol{\mu}_k | \boldsymbol{\Lambda}_k) (\boldsymbol{\mu}_k - \mathbf{m}_k)^T b_k \boldsymbol{\Lambda}_k (\boldsymbol{\mu}_k - \mathbf{m}_k) d\boldsymbol{\mu}_k d\boldsymbol{\Lambda}_k \quad (2.89)
\end{aligned}$$

ここで、標準的な公式

$$\mathbb{E}_{\boldsymbol{\mu}_k} [\boldsymbol{\mu}_k] = \int q(\boldsymbol{\mu}_k | \boldsymbol{\Lambda}_k) \boldsymbol{\mu}_k d\boldsymbol{\mu}_k = \mathbf{m}_k \quad (2.90)$$

$$\mathbb{E}_{\boldsymbol{\Lambda}_k} [\boldsymbol{\Lambda}_k] = \int q(\boldsymbol{\Lambda}_k) \boldsymbol{\Lambda}_k d\boldsymbol{\Lambda}_k = c_k \mathbf{W}_k \quad (2.91)$$

を用いた。また、 $q(\boldsymbol{\mu}_k | \boldsymbol{\Lambda}_k)$ は平均 \mathbf{m}_k 、分散 $b_k \boldsymbol{\Lambda}_k$ のガウス分布であるので次式が成立する。

$$\int q(\boldsymbol{\mu}_k | \boldsymbol{\Lambda}_k) (\boldsymbol{\mu}_k - \mathbf{m}_k) b_k \boldsymbol{\Lambda}_k (\boldsymbol{\mu}_k - \mathbf{m}_k)^T d\boldsymbol{\mu}_k = \mathbf{I} \quad (2.92)$$

すなわち, $q(\boldsymbol{\mu}_k | \boldsymbol{\Lambda}_k)$ のもとでの行列 $(\boldsymbol{\mu}_k - \mathbf{m}_k) b_k \boldsymbol{\Lambda}_k (\boldsymbol{\mu}_k - \mathbf{m}_k)^T$ の対角成分の期待値はすべて 1 である。したがって, $(\boldsymbol{\mu}_k - \mathbf{m}_k)^T b_k \boldsymbol{\Lambda}_k (\boldsymbol{\mu}_k - \mathbf{m}_k)$ の期待値は単位行列 \mathbf{I} の対角成分の総和である D に等しい。最終的に次式を得る。

$$\mathbb{E}_{\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k} \left[(\mathbf{x}_n - \boldsymbol{\mu}_k)^T \boldsymbol{\Lambda}_k (\mathbf{x}_n - \boldsymbol{\mu}_k) \right] = c_k (\mathbf{x}_n - \mathbf{m}_k)^T \mathbf{W}_k (\mathbf{x}_n - \mathbf{m}_k) + D \quad (2.93)$$

最後に, 因子 $q(\alpha)$ について考える。一般的な結果を再度用いると, 最適な因子は

$$\begin{aligned} \log q^*(\alpha) &= \mathbb{E}_{\mathbf{z}, \mathbf{v}, \boldsymbol{\mu}, \boldsymbol{\Lambda}} [p(\mathbf{X}, \mathbf{Z}, \mathbf{v}, \boldsymbol{\mu}, \boldsymbol{\Lambda}, \alpha)] + \text{const.} \\ &= \mathbb{E}_{\mathbf{v}} [\log p(\mathbf{v} | \alpha)] + \log p(\alpha) + \text{const.} \end{aligned} \quad (2.94)$$

で与えられる。ここで, 各項は次式で計算できる。

$$\begin{aligned} \mathbb{E}_{\mathbf{v}} [\log p(\mathbf{v} | \alpha)] &= \sum_{k=1}^K (\log \alpha + (\alpha - 1) \mathbb{E}_{\mathbf{v}} [\log(1 - v_k)]) \\ &= K \log \alpha + \sum_{k=1}^K \mathbb{E}_{\mathbf{v}} [\log(1 - v_k)] \alpha + \text{const.} \\ \log p(\alpha) &= (a_0 - 1) \log \alpha - \frac{\alpha}{\lambda_0} + \text{const.} \end{aligned} \quad (2.95)$$

したがって, 最適な因子 $q^*(\alpha)$ はガンマ分布

$$q^*(\alpha) = \text{Gam}(\alpha | a, \lambda) \quad (2.96)$$

となることが分かり, 各パラメータは次式で与えられる。

$$a = a_0 + K \quad (2.97)$$

$$\lambda^{-1} = \lambda_0^{-1} - \sum_{k=1}^K \mathbb{E}_{\mathbf{v}} [\log(1 - v_k)] \quad (2.98)$$

このとき, 各種の期待値は次式で求まる。

$$\mathbb{E}_{\alpha}[\alpha] = a\lambda \quad (2.99)$$

$$\mathbb{E}_\alpha[\log \alpha] = \psi(a) + \log \lambda \quad (2.100)$$

2.4.3 ベイズ推定：ギブスサンプリング

周辺化ギブスサンプリング(Collapsed Gibbs Sampling)とは、パラメータと潜在変数の空間でそれぞれの値をサンプリングするのではなく、パラメータを積分消去して潜在変数のみの空間につぶしてから(Collapsing)，潜在変数のみの値を直接サンプリングする手法である。まず、すべての変数の完全な同時分布を考えると、

$$\begin{aligned} p(\mathbf{X}, \mathbf{Z}, \mathbf{v}, \boldsymbol{\mu}, \boldsymbol{\Lambda}) &= p(\mathbf{X}|\mathbf{Z}, \boldsymbol{\mu}, \boldsymbol{\Lambda})p(\mathbf{Z}|\mathbf{v})p(\mathbf{v})p(\boldsymbol{\mu}, \boldsymbol{\Lambda}) \\ &= \prod_{n=1}^N \prod_{k=1}^K \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k^{-1})^{z_{n,k}} \prod_{n=1}^N \prod_{k=1}^K v_k^{z_{n,k}} (1 - v_k)^{\sum_{k'=k+1}^K z_{n,k'}} \\ &\quad \prod_{k=1}^K \alpha v_k^{1-1} (1 - v_k)^{\alpha-1} \prod_{k=1}^K \mathcal{N}(\boldsymbol{\mu}_k | \mathbf{m}_0, (b_0 \boldsymbol{\Lambda}_k)^{-1}) \mathcal{W}(\boldsymbol{\Lambda}_k | \mathbf{W}_0, c_0) \end{aligned}$$

で計算できる。いま、パラメータを積分消去した \mathbf{X} および \mathbf{Z} の周辺分布

$$p(\mathbf{X}, \mathbf{Z}) = p(\mathbf{X}|\mathbf{Z})p(\mathbf{Z}) \quad (2.102)$$

を考える。パラメータ v および $\boldsymbol{\mu}, \boldsymbol{\Lambda}$ を積分消去すると、

$$\begin{aligned} p(\mathbf{X}|\mathbf{Z}) &= \prod_{k=1}^K \iint p(\mathbf{X}|\mathbf{Z}, \boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k) p(\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k) d\boldsymbol{\mu}_k d\boldsymbol{\Lambda}_k \\ &= \prod_{k=1}^K \iint \prod_{n=1}^N \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k^{-1})^{z_{n,k}} \mathcal{N}(\boldsymbol{\mu}_k | \mathbf{m}_0, (b_0 \boldsymbol{\Lambda}_k)^{-1}) \mathcal{W}(\boldsymbol{\Lambda}_k | \mathbf{W}_0, c_0) d\boldsymbol{\mu}_k d\boldsymbol{\Lambda}_k \\ &= (2\pi)^{-\frac{DN}{2}} \prod_{k=1}^K \left(\frac{b_0}{b_k^z} \right)^{\frac{D}{2}} \frac{B(\mathbf{W}_0, c_0)}{B(\mathbf{W}_k^z, c_k^z)} \end{aligned} \quad (2.103)$$

$$\begin{aligned} p(\mathbf{Z}) &= \prod_{k=1}^K \int p(\mathbf{Z}|v_k) p(v_k) dv_k \\ &= \alpha^K \prod_{k=1}^K \int v_k^{1+\sum_{n=1}^N z_{n,k}-1} (1 - v_k)^{\alpha+\sum_{n=1}^N \sum_{k'=k+1}^K z_{n,k'}-1} dv_k \end{aligned}$$

$$\begin{aligned}
 &= \alpha^K \prod_{k=1}^K \frac{\Gamma\left(1 + \sum_{n=1}^N z_{n,k}\right) \Gamma\left(\alpha + \sum_{n=1}^N \sum_{k'=k+1}^K z_{n,k'}\right)}{\Gamma\left(1 + \alpha + \sum_{n=1}^N \sum_{k'=k}^K z_{n,k'}\right)} \\
 &= \alpha^K \prod_{k=1}^K \frac{\Gamma(1+n_k) \Gamma(\alpha+n_{>k})}{\Gamma(1+\alpha+n_{\geq k})}
 \end{aligned} \tag{2.104}$$

を得る。ここで、 b_k^z , c_k^z および W_k^z はそれぞれ、式 (2.80), 式 (2.84) および式 (2.85) を用いて b_k , c_k および W_k を計算する際に、負担率 $\gamma_{n,k}$ を潜在変数 $z_{n,k}$ に置き換えて得られる値である。具体的には、式 (2.74) から式 (2.76) で与えられる十分統計量を計算する際に、負担率 $\gamma_{n,k}$ を潜在変数 $z_{n,k}$ に置き換えればよい。 n_k は k 個目のガウス分布に割り当てられたサンプルの個数である。ドット (\cdot) はその変数について足し合わせることを意味する。

いま、あるサンプル x_n に対応する潜在変数 z_n の割り当てを解除して、その予測分布を求めたい。観測データ \mathbf{X} ($\mathbf{X}^{\neg n}$ および x_n) が与えられ、 z_n 以外の潜在変数 $Z^{\neg n}$ の値が判明しているとき、 $z_{n,k} = 1$ となる確率は

$$\begin{aligned} p(z_{n,k} = 1 | \mathbf{x}_n, \mathbf{X}^{\neg n}, \mathbf{Z}^{\neg n}) &\propto p(z_{n,k} = 1, x_n | \mathbf{X}^{\neg n}, \mathbf{Z}^{\neg n}) \\ &= p(z_{n,k} = 1 | \mathbf{Z}^{\neg n}) p(x_n | z_{n,k} = 1, \mathbf{X}^{\neg n} | \mathbf{Z}^{\neg n}) \end{aligned}$$

で与えられる。まず、第一項は次式で計算できる。

$$p(z_{n,k} = 1 | \mathbf{Z}^{\neg n}) = \frac{p(\mathbf{Z})}{p(\mathbf{Z}^{\neg n})} = \frac{1 + n_k^{\neg n}}{1 + \alpha + n_{\geq k}^{\neg n}} \prod_{k'=1}^{k-1} \frac{\alpha + n_{>k'}^{\neg n}}{1 + \alpha + n_{\geq k'}^{\neg n}} \quad (2.106)$$

次に、第二項は $Z^{\neg n}$ が既知かつ $z_{n,k} = 1$ となるときの x_n の予測分布であるので、次式で計算できる。

$$\begin{aligned} &p(x_n | z_{n,k} = 1, \mathbf{X}^{\neg n}, \mathbf{Z}^{\neg n}) \\ &= \iint p(x_n | z_{n,k} = 1, \boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k) p(\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k | \mathbf{X}^{\neg n}, \mathbf{Z}^{\neg n}) d\boldsymbol{\mu}_k d\boldsymbol{\Lambda}_k \\ &= \iint p(x_n | z_{n,k} = 1, \boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k) p(\boldsymbol{\mu}_k | \boldsymbol{\Lambda}_k, \mathbf{X}^{\neg n}, \mathbf{Z}^{\neg n}) p(\boldsymbol{\Lambda}_k | \mathbf{X}^{\neg n}, \mathbf{Z}^{\neg n}) d\boldsymbol{\mu}_k d\boldsymbol{\Lambda}_k \\ &= \mathcal{S}(x_n | \mathbf{m}_{z,k}^{\neg n}, \mathbf{L}_{z,k}^{\neg n}, c_{z,k}^{\neg n} + 1 - D) \end{aligned} \quad (2.107)$$

ここで、ガウス・ウィシャート事後分布 $p(\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k | \mathbf{X}^{\neg n}, \mathbf{Z}^{\neg n})$ のパラメータを $b_{z,k}^{\neg n}, \mathbf{m}_{z,k}^{\neg n}, c_{z,k}^{\neg n}$ および $\mathbf{W}_{z,k}^{\neg n}$ とした。これらの値は、式(2.80), 式(2.81), 式(2.84)および式(2.85)において、期待値 $\gamma_{n,k}$ を $Z^{\neg n}$ で与えられる潜在変数の値 $z_{n,k}$ に置き換え、 n 以外の和をとることで得られる。また、 \mathcal{S} はスチューデント t 分布を表し、そのパラメータ $\mathbf{L}_{z,k}^{\neg n}$ は

$$\mathbf{L}_{z,k}^{\neg n} = \frac{b_{z,k}^{\neg n}}{1 + b_{z,k}^{\neg n}} (c_{z,k}^{\neg n} + 1 - D) \mathbf{W}_{z,k}^{\neg n} \quad (2.108)$$

で与えられる。参考までに、 x_n の予測分布は混合スチューデント t 分布となる。

$$p(x_n | \mathbf{X}^{\neg n}, \mathbf{Z}^{\neg n}) = \sum_{k=1}^K p(z_{n,k} = 1 | \mathbf{Z}^{\neg n}) \mathcal{S}(x_n | \mathbf{m}_{z,k}^{\neg n}, \mathbf{L}_{z,k}^{\neg n}, c_{z,k}^{\neg n} + 1 | 2D) \quad (2.109)$$

これまでの結果をまとめると次式を得る。

$$p(z_{n,k} = 1 | \mathbf{x}_n, \mathbf{X}^{-n}, \mathbf{Z}^{-n}) \propto \frac{1 + n_k^{-n}}{1 + \alpha + n_{\geq k}^{-n}} \prod_{k'=1}^{k-1} \frac{\alpha + n_{>k'}^{-n}}{1 + \alpha + n_{\geq k'}^{-n}} \mathcal{S}(\mathbf{x}_n | \mathbf{m}_{z,k}^{-n}, \mathbf{L}_{z,k}^{-n}, c_{z,k}^{-n} + 1) \quad (2.110)$$

すなわち， z_n は，式 (2.110) で定まる K 次元の多項分布に従う．このよう
な多項分布からサンプリングを行うことは容易である．サンプリングの手順
としては，まず， Z の値をランダムに初期化する．そして，式 (2.110) を用
いて各 n の潜在変数 z_n の値を順番に更新することを繰り返す．更新を繰
り返すことたびに目標分布 $p(Z)$ への収束判定を行い，収束条件が満たされ
ば以降，自己相関がほとんどなくなるように十分長い間隔でサンプルを取得
すればよい．ただし，収束判定は容易ではないため，実際は一定回数の反復
をもって収束したとみなすことが多い．

2.4.4 ベイズ推定：周辺化ギブスサンプリング

一方，CRP に基づく周辺化ギブスサンプリングでは

$$p(z_{n,k} = 1 | \mathbf{Z}^{-n}) = \begin{cases} \frac{n_k^{-n}}{n^{-n} + \alpha} & k \text{ が既存} \\ \frac{\alpha}{n^{-n} + \alpha} & k \text{ が新規} \end{cases} \quad (2.111)$$

$$p(\mathbf{x}_n | z_{n,k} = 1, \mathbf{X}^{-n}, \mathbf{Z}^{-n}) = \begin{cases} \mathcal{S}(\mathbf{x}_n | \mathbf{m}_{z,k}^{-n}, \mathbf{L}_{z,k}^{-n}, c_{z,k}^{-n} + 1 - D) & k \text{ が既存} \\ \mathcal{S}(\mathbf{x}_n | \mathbf{m}_0, \mathbf{L}_0, c_0 + 1 - D) & k \text{ が新規} \end{cases} \quad (2.112)$$

にしたがって z_n をサンプリングするので，考慮すべきガウス分布の個数が
増減する．ここで， L_0 は次式で求められる．

$$\mathbf{L}_0 = \frac{b_0}{1 + b_0} (c_0 + 1 - D) \mathbf{W}_0 \quad (2.113)$$

chapter 3

因子分解

本章では、与えられたデータを複数の要素の積に分解する因子分解技術について説明します。現実の多くの問題において、一見複雑に見えるデータも、実は少数の要素（因子・基底とも呼ぶ）の組み合わせで構成されていることが多いのです。ここでは、非負値行列に対する非負値行列因子分解 (nonnegative matrix factorization, NMF)、ある特殊な形式のテンソル（半正定値行列の集合）に対する半正定値テンソル分解 (positive semidefinite tensor factorization, PSDTF)、非負値整数行列に対する確率的潜在成分分配法 (probabilistic latent component analysis, PLCA) について説明します。また、各手法に対し、確率モデルの最尤推定としての解釈が可能であり、適切なノンパラメトリックベイズ事前分布を導入することで、データに合わせて実効的な基底数（要素数）を自動調節できることを解説します。

3.1 非負値行列因子分解

非負値行列因子分解 (nonnegative matrix factorization: NMF)^[1,2] は、音楽音響信号の各フレームのスペクトルを、少数の基底スペクトルの重み付き線形和で近似するための技術です。ここで、基底スペクトルは、ある楽器音のある音高の平均的なスペクトルや、ある打楽器音の平均的なスペクトルに対応していることが期待されています。すなわち、基底スペクトルは、楽譜上の音符に対応することになり、音符ごとに音源分離を行ったり、自動採

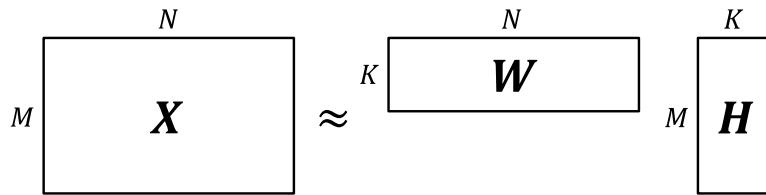


図 3.1 非負値行列因子分解 (NMF) による低ランク近似 .

譜を行ううえで NMF は有用です .

NMF は行列分解の一種ですが , 基底スペクトルと重みをいずれも非負値に限定していることが特徴です . その副次的な効果として , 重みがスパースになるよう誘導されます . なぜなら , 重みが非負値に限定されているため , あるフレームにおいて , いったんある基底スペクトルが利用されると , 他の基底スペクトルを減算することで , その影響を打ち消すことができないからです . すなわち , 利用する基底スペクトルの個数は節約する方がよいということになります . 音響信号全体では K 個の基底スペクトルが必要であるとしても , 各時間フレームでは限られた少数の基底スペクトルのみが実際に発音しているため , NMF によるスパース性の誘発は大変都合がよいのです .

この考え方を進めると , 入力音響信号に合わせて基底数 K を手動で調整する代わりに , 可算無限個の基底の存在を仮定し , 必要な基底だけが自動的に実体化できれば好都合です . 本章では , NMF に基づくモノラル音響信号の音源分離について説明します . まず , コスト関数最小化としての定式化 (最尤推定) と音源分離への適用について説明し , 発展的内容としてノンパラメトリックベイズモデルの学習について解説します .

3.1.1 コスト関数最小化としての定式化

NMF では , 非負値行列 $X = [x_1, \dots, x_N] \in \mathbb{R}_+^{M \times N}$ を , 二つの非負値行列 $W = [w_1, \dots, w_K] \in \mathbb{R}_+^{M \times K}$ および $H = [h_1, \dots, h_K] \in \mathbb{R}_+^{N \times K}$ の積である低ランクな再構成行列 $Y = WH^T$ で近似します (図 5.5).

$$X \approx WH^T \stackrel{\text{def}}{=} Y \quad (3.1)$$

ここで , $w_k \in \mathbb{R}_+^M$ と $h_k \in \mathbb{R}_+^N$ はそれぞれ基底ベクトルとアクティベー

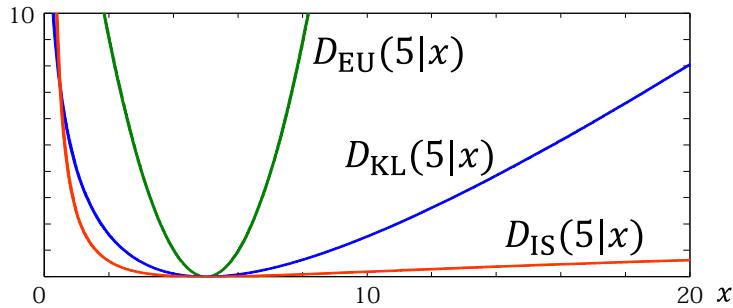


図 3.2 ヨークリッド距離 , KL ダイバージェンス , IS ダイバージェンスに基づくコスト関数 $D(5|x)$. ヨークリッド距離以外は $x = 5$ の左右で非対称であることに注意 .

ションベクトルであり , 基底数は $K \ll \min(M, N)$ とします . 再構成行列を $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_N] \in \mathbb{R}_+^{M \times N}$ とすると , 以下の通り書き直せます .

$$\mathbf{x}_n \approx \sum_{k=1}^K h_{kn} \mathbf{w}_k \stackrel{\text{def}}{=} \mathbf{y}_n \quad (3.2)$$

NMF では , 観測データ \mathbf{X} が与えられたときに , コスト関数

$$\mathcal{D}(\mathbf{X}|\mathbf{Y}) = \sum_{n=1}^N \mathcal{D}(\mathbf{x}_n|\mathbf{y}_n) \quad (3.3)$$

を最小化する \mathbf{Y} (\mathbf{W} および \mathbf{H}) を求めます . 非負値の観測ベクトル \mathbf{x}_n と非負値の再構成ベクトル \mathbf{y}_n との間の誤差 $\mathcal{D}(\mathbf{x}_n|\mathbf{y}_n)$ が小さいほど , 低ランク近似の精度がよいことになります .

誤差 $\mathcal{D}(\mathbf{x}_n|\mathbf{y}_n)$ を評価する尺度として , ヨークリッド距離 , 一般化 Kullback-Leibler (KL) ダイバージェンス [8] , および Itakura-Saito (IS) ダイバージェンス [2] などがよく利用されています .

$$\mathcal{D}_{\text{EU}}(\mathbf{x}_n|\mathbf{y}_n) = \sum_{m=1}^M (x_{nm} - y_{nm})^2 \quad (3.4)$$

$$\mathcal{D}_{\text{KL}}(\mathbf{x}_n|\mathbf{y}_n) = \sum_{m=1}^M \left(x_{nm} \log \frac{x_{nm}}{y_{nm}} - x_{nm} + y_{nm} \right) \quad (3.5)$$

$$\mathcal{D}_{\text{IS}}(\mathbf{x}_n|\mathbf{y}_n) = \sum_{m=1}^M \left(\frac{x_{nm}}{y_{nm}} - \log \frac{x_{nm}}{y_{nm}} - 1 \right) \quad (3.6)$$

例として、図 3.2 に、 x_n および y_n がいずれもスカラ（1 次元のベクトル）である場合のコスト関数の値を示します。これらの関数は常に非負値をとり、 $x_n = y_n$ のときのみ 0 となります。

KL ダイバージェンスと IS ダイバージェンスは、ユークリッド距離などの通常の距離尺度と異なり、非対称性をもっています。

$$\mathcal{D}_{\text{EU}}(\mathbf{x}_n|\mathbf{y}_n) = \mathcal{D}_{\text{EU}}(\mathbf{y}_n|\mathbf{x}_n) \quad (3.7)$$

$$\mathcal{D}_{\text{KL}}(\mathbf{x}_n|\mathbf{y}_n) \neq \mathcal{D}_{\text{KL}}(\mathbf{y}_n|\mathbf{x}_n) \quad (3.8)$$

$$\mathcal{D}_{\text{IS}}(\mathbf{x}_n|\mathbf{y}_n) \neq \mathcal{D}_{\text{IS}}(\mathbf{y}_n|\mathbf{x}_n) \quad (3.9)$$

図 3.2 に示す通り、ユークリッド距離は、 $x = 5$ の左右で対称な二次関数ですが、KL ダイバージェンスや IS ダイバージェンスは、 $x = 5$ から離れるにつれて、 $x < 5$ よりも $x > 5$ の方が値の増加がはるかに緩やかな関数であることが分かります。

また、これらの中では、IS ダイバージェンスのみがスケール不变性を持っています。すなわち、任意の実数 $\alpha > 0$ に対して、以下が成立します。

$$\mathcal{D}_{\text{EU}}(\mathbf{x}_n|\mathbf{y}_n) \neq \mathcal{D}_{\text{EU}}(\alpha \mathbf{x}_n|\alpha \mathbf{y}_n) \quad (3.10)$$

$$\mathcal{D}_{\text{KL}}(\mathbf{x}_n|\mathbf{y}_n) \neq \mathcal{D}_{\text{KL}}(\alpha \mathbf{x}_n|\alpha \mathbf{y}_n) \quad (3.11)$$

$$\mathcal{D}_{\text{IS}}(\mathbf{x}_n|\mathbf{y}_n) = \mathcal{D}_{\text{IS}}(\alpha \mathbf{x}_n|\alpha \mathbf{y}_n) \quad (3.12)$$

音響信号に対して NMF を適用するうえで、スケール不变性は重要な性質です。スケール不变性を持たない場合には、音楽音響信号全体の音量を変化させただけで、NMF で得られる結果が異なってしまいます。そのため、理論的には IS ダイバージェンスが妥当ですが、経験的には KL ダイバージェンスが安定してよい結果を与えることが知られています。

3.1.2 乗法更新アルゴリズムに基づく最適化

コスト関数 $\mathcal{D}(\mathbf{X}|\mathbf{Y})$ を最小化する \mathbf{Y} を直接求めることは困難なため、乗法更新則 (multiplicative updating rules) に基づく反復最適化技法が提案

されています^[9]。本節では、補助関数法（付録??節）に基づく収束性が保証された乗法更新則を紹介します。具体的には、コスト関数 $\mathcal{D}(X|Y)$ の上限関数 $\mathcal{U}(X|Y, \Theta)$ を設計し、 Y と Θ について交互に逐次最小化することで、間接的に $\mathcal{D}(X|Y)$ を逐次最小化します。ここで、 Θ は新たに導入された補助変数であり、 $\mathcal{U}(X|Y, \Theta)$ を Θ について最小化することで、もとの関数 $\mathcal{D}(X|Y)$ と同じ値をとるようにしておく必要があります。

$$\mathcal{D}(X|Y) = \min_{\Theta} \mathcal{U}(X|Y, \Theta) \quad (3.13)$$

効率的な乗法更新則を導出するうえで、 $\mathcal{U}(X|Y, \Theta)$ は、 Y と Θ のいずれに関しても、一方が既知であれば、もう一方に関する最小化を容易に行えることが重要です。特に、 Y と Θ の更新則がいずれも閉形式で記述できることが理想です。そうでないと、 Y と Θ を更新する 1 ステップ内においてすら、最急降下法や（準）ニュートン法などを利用した反復最適化が必要になり、数値的安定性・計算量・収束性などに問題が発生することがあります。

3.1.3 EU-NMF の乗法更新アルゴリズム

ユークリッド距離に基づく NMF (EU-NMF) の乗法更新則を導出します。準備として、二次関数が下に凸であることから、Jensen の不等式（付録??節）を用いて、 $f(z) = \left(\sum_{k=1}^K z_k\right)^2$ の上限関数 $u(z, \lambda)$ を設計します。

$$f(z) = \left(\sum_{k=1}^K \lambda_k \frac{z_k}{\lambda_k}\right)^2 \leq \sum_{k=1}^K \lambda_k \left(\frac{z_k}{\lambda_k}\right)^2 = \sum_{k=1}^K \frac{z_k^2}{\lambda_k} \stackrel{\text{def}}{=} u(z, \lambda) \quad (3.14)$$

ここで、 $z_k \geq 0$ は非負値の変数であり、 $\lambda = \{\lambda_k\}_{k=1}^K$ は

$$\sum_{k=1}^K \lambda_k = 1 \quad (3.15)$$

を満たす非負値の補助変数です。「和の二次関数」の上限関数として、「二次関数の和」が得られていることに注意してください。Jensen の不等式を用いると、和と凸関数（あるいは凹関数）の適用順序を交換できます。

等号成立条件、すなわち、式 (3.15) の制約条件付きで $u(z, \lambda)$ を最小化する λ を求めるには、ラグランジュの未定乗数法を用います。まず、未定乗数

ϕ を用いて、新たな関数

$$F(\lambda, \phi) = \sum_{k=1}^K \frac{z_k^2}{\lambda_k} + \phi \left(1 - \sum_{k=1}^K \lambda_k \right) \quad (3.16)$$

を考えます。ここで、 λ が制約条件を満たす場合には、第二項は 0 であることに注意してください。これを λ_k について偏微分すると

$$\frac{\partial F(\lambda, \phi)}{\partial \lambda_k} = -\frac{z_k^2}{\lambda_k^2} - \phi \quad (3.17)$$

を得ます。 $\frac{\partial F(\lambda, \phi)}{\partial \lambda_k} = 0$ とおくと、 ϕ を用いて λ_k が表せます。

$$\lambda_k = \frac{z_k}{\sqrt{-\phi}} \quad (3.18)$$

これを式 (3.15) に代入すると、

$$1 = \sum_{k=1}^K \lambda_k = \frac{1}{\sqrt{-\phi}} \sum_{k=1}^K z_k \quad (3.19)$$

となるので、未定乗数 ϕ は

$$\sqrt{-\phi} = \sum_{k=1}^K z_k \quad (3.20)$$

で与えられます。これを、式 (3.18) に代入すると、次式を得ます。

$$\lambda_k = \frac{z_k}{\sum_{k=1}^K z_k} \quad (3.21)$$

この結果を用いて、コスト関数 $\mathcal{D}(\mathbf{X}|\mathbf{Y})$ に対して、補助変数 λ を含む上限関数 $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \lambda)$ を導出します。

$$\begin{aligned} \mathcal{D}(\mathbf{X}|\mathbf{Y}) &= \sum_{n=1}^N \sum_{m=1}^M (x_{nm} - y_{nm})^2 \\ &= \sum_{n=1}^N \sum_{m=1}^M \left(x_{nm}^2 - 2x_{nm} \sum_{k=1}^K w_{km} h_{kn} + \left(\sum_{k=1}^K w_{km} h_{kn} \right)^2 \right) \\ &\leq \sum_{n=1}^N \sum_{m=1}^M \left(x_{nm}^2 - 2x_{nm} \sum_{k=1}^K w_{km} h_{kn} + \sum_{k=1}^K \frac{w_{km}^2 h_{kn}^2}{\lambda_{nmk}} \right) \end{aligned}$$

$$\stackrel{\text{def}}{=} \mathcal{U}(\mathbf{X}|\mathbf{Y}, \boldsymbol{\lambda}) \quad (3.22)$$

ここで、式 (3.14) を用いました。補助変数 $\boldsymbol{\lambda} = \{\lambda_{nmk}\}_{n=1, m=1, k=1}^{N, M, K}$ は、 $\sum_{k=1}^K \lambda_{nmk} = 1$ を満たします。また、等号成立条件、すなわち $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \boldsymbol{\lambda})$ を最小化する $\boldsymbol{\lambda}$ は次式で与えられます。

$$\lambda_{nmk} = \frac{w_{km} h_{kn}}{\sum_{k'=1}^K w_{k'm} h_{k'n}} = \frac{w_{km} h_{kn}}{y_{nm}} \quad (3.23)$$

最後に、式 (3.22) を最小化する \mathbf{Y} (\mathbf{W} および \mathbf{H}) を求めます。まず、 $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \boldsymbol{\lambda})$ を w_{km} について偏微分すると、

$$\frac{\partial \mathcal{U}(\mathbf{X}|\mathbf{Y}, \boldsymbol{\lambda})}{\partial w_{km}} = \sum_{n=1}^N \left(-2x_{nm} h_{kn} + \frac{2w_{km} h_{kn}^2}{\lambda_{nmk}} \right) \quad (3.24)$$

を得ます。ここで、 $\frac{\partial \mathcal{U}(\mathbf{X}|\mathbf{Y}, \boldsymbol{\lambda})}{\partial w_{km}} = 0$ とおくと、

$$w_{km} = \frac{\sum_{n=1}^N x_{nm} h_{kn}}{\sum_{n=1}^N \frac{h_{kn}^2}{\lambda_{nmk}}} \quad (3.25)$$

を得ます。同様に、 $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \boldsymbol{\lambda})$ を h_{kn} について偏微分して 0 とおくことで、

$$h_{kn} = \frac{\sum_{m=1}^M x_{nm} w_{km}}{\sum_{m=1}^M \frac{w_{km}^2}{\lambda_{nmk}}} \quad (3.26)$$

を得ます。式 (3.23)、式 (3.25) および式 (3.26) は互いに依存関係にあり、 $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \boldsymbol{\lambda})$ を最小化する $\boldsymbol{\lambda}$ 、 \mathbf{W} および \mathbf{H} を一挙に求めることができません。そのため、これらの式を交互に反復計算することにより、徐々に式 (3.22) を小さくしていくことになります。

NMF の乗法更新則は、補助変数を介さずに表現することもできます。具体的には、式 (3.23) を式 (3.25) および式 (3.26) に代入することで、

$$w_{km} \leftarrow \frac{\sum_{n=1}^N x_{nm} h_{kn}}{\sum_{n=1}^N \frac{h_{kn}^2}{\lambda_{nmk}}} = \frac{\sum_{n=1}^N x_{nm} h_{kn}}{\sum_{n=1}^N \frac{y_{nm} h_{kn}}{w_{km}}} = \frac{\sum_{n=1}^N x_{nm} h_{kn}}{\sum_{n=1}^N y_{nm} h_{kn}} w_{km} \quad (3.27)$$

$$h_{kn} \leftarrow \frac{\sum_{m=1}^M x_{nm} w_{km}}{\sum_{m=1}^M \frac{w_{km}^2}{\lambda_{nmk}}} = \frac{\sum_{m=1}^M x_{nm} w_{km}}{\sum_{m=1}^M y_{nm} w_{km}} h_{kn} \quad (3.28)$$

を得ます。ここで、記号 \leftarrow は、右辺値を左辺値に代入して更新するという意味です。 \leftarrow の右側においては、 λ 、 W および H の更新前の古い値を用います。同じ変数が \leftarrow の左右に登場すると混乱するため、等号ではなく、記号 \leftarrow を用いています。

これらが「乗法」更新則と呼ばれる所以は、自分自身に係数を掛け合わせて更新を行うからです。例えば、式 (3.27) では、 w_{km} に係数 $\frac{\sum_{n=1}^N x_{nm} h_{kn}}{\sum_{n=1}^N y_{nm} h_{kn}} \geq 0$ を掛け合わせています。このとき、係数も非負値であることがから、特別な制約を導入しなくても、 w_{km} の非負値性は自然に保たれます。また、 $w_{km} = 0$ として初期化すると、以降の更新では常に $w_{km} = 0$ が維持されます。

式 (3.27) および式 (3.28) は、行列演算を用いて簡潔に書けます。

$$W \leftarrow \frac{H X^T}{H Y^T} \odot W \quad (3.29)$$

$$H \leftarrow \frac{W X}{W Y} \odot H \quad (3.30)$$

ここで、割り算—や掛け算 \odot は行列の各要素ごとに行うものとします。高速な行列演算が可能なプログラミング言語（例：MATLAB）では、for 文を使わずに上記の更新式を直接コーディングすることができ、CPU や GPU のベクトル演算機能を利用した高速な更新が可能になります。

アルゴリズム 3.1 に、EU-NMF のアルゴリズムを示します（これが実は、ある確率モデルの最尤推定になっていることはあとで示します）。このアルゴリズムは山登り法（NMF ではコスト関数を最小化しているので実際には「山下り法」）の一種であり、大域的な最適解を見つけられる保証はなく、得られる結果は W および H の初期値に大きな影響を受けます。また、スケールの任意性を解消するため、 $\sum_{m=1}^M w_{km} = 1$ を満たすように正規化しておくことにします。具体的には、 w_{km} を更新した結果、 $\sum_{m=1}^M w_{km} = s$ になってしまったとすると、すべての m, n について $w_{km} \leftarrow \frac{1}{s} w_{km}$ および $h_{kn} \leftarrow s h_{kn}$ と更新しておきます。この処理ではコスト関数の値は変化しません。 $\sum_{n=1}^N h_{kn} = 1$ となるようアクティベーションを正規化することもできますが、基底ベクトルを正規化しておくのが一般的です。

アルゴリズム 3.1 : EU-NMF の最尤推定

Require: 非負値行列 $\mathbf{X} \in \mathbb{R}_+^{M \times N}$, 基底数 K

- 1: 非負値行列 $\mathbf{W} \in \mathbb{R}_+^{M \times K}$ をランダムに初期化
- 2: 非負値行列 $\mathbf{H} \in \mathbb{R}_+^{M \times K}$ をランダムに初期化
- 3: **while** 上限関数 $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \lambda)$ が未収束 **do**
- 4: $\mathbf{W} \leftarrow \frac{\mathbf{H}\mathbf{X}^T}{\mathbf{H}\mathbf{Y}^T} \odot \mathbf{W}$
- 5: $\mathbf{H} \leftarrow \frac{\mathbf{W}\mathbf{X}}{\mathbf{W}\mathbf{Y}} \odot \mathbf{H}$
- 6: **end while**
- 7: **Return** 非負値行列 \mathbf{W}, \mathbf{H}

3.1.4 KL-NMF の乗法更新アルゴリズム

EU-NMF とほとんど同様の方法で, KL ダイバージェンスに基づく NMF (KL-NMF) の乗法更新則を導出することができます。準備として, 負の対数関数が下に凸であることから, Jensen の不等式 (付録??節) を用いて, $f(z) = -\log \left(\sum_{k=1}^K z_k \right)$ の上限関数 $u(z, \lambda)$ を設計します。

$$f(z) = -\log \left(\sum_{k=1}^K \lambda_k \frac{z_k}{\lambda_k} \right) \leq -\sum_{k=1}^K \lambda_k \log \frac{z_k}{\lambda_k} \stackrel{\text{def}}{=} u(z, \lambda) \quad (3.31)$$

ここで, $z_k \geq 0$ は非負値の変数であり, $\lambda = \{\lambda_k\}_{k=1}^K$ は

$$\sum_{k=1}^K \lambda_k = 1 \quad (3.32)$$

を満たす非負値の補助変数です。「和の対数関数」の上限関数として、「対数関数の和」が得られていることに注意してください。

等号成立条件, すなわち, 式 (3.32) の制約条件付きで $u(z, \lambda)$ を最小化する λ を求めるには, ラグランジュの未定乗数 ϕ を用いて, 新たな関数

$$F(\boldsymbol{\lambda}, \phi) = \sum_{k=1}^K \lambda_k \log \frac{z_k}{\lambda_k} + \phi \left(1 - \sum_{k=1}^K \lambda_k \right) \quad (3.33)$$

を考えます。これを λ_k について偏微分すると

$$\frac{\partial F(\boldsymbol{\lambda}, \phi)}{\partial \lambda_k} = \log z_k - \log \lambda_k - 1 - \phi \quad (3.34)$$

を得ます。 $\frac{\partial F(\boldsymbol{\lambda}, \phi)}{\partial \lambda_k} = 0$ とおくと、 ϕ を用いて λ_k が表せます。

$$\lambda_k = \frac{z_k}{e^{1+\phi}} \quad (3.35)$$

これを式 (3.32) に代入すると、

$$1 = \sum_{k=1}^K \lambda_k = \frac{1}{e^{1+\phi}} \sum_{k=1}^K z_k \quad (3.36)$$

となるので、未定乗数 ϕ は

$$e^{1+\phi} = \sum_{k=1}^K z_k \quad (3.37)$$

で与えられます。これを、式 (3.35) に代入すると、次式を得ます。

$$\lambda_k = \frac{z_k}{\sum_{k=1}^K z_k} \quad (3.38)$$

この結果を用いて、コスト関数 $\mathcal{D}(\mathbf{X}|\mathbf{Y})$ に対して、補助変数 $\boldsymbol{\lambda}$ を含む上限関数 $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \boldsymbol{\lambda})$ を導出します。

$$\begin{aligned} \mathcal{D}(\mathbf{X}|\mathbf{Y}) &= \sum_{n=1}^N \sum_{m=1}^M (x_{nm} \log x_{nm} - x_{nm} \log y_{nm} - x_{nm} + y_{nm}) \\ &\stackrel{c}{=} \sum_{n=1}^N \sum_{m=1}^M \left(-x_{nm} \log \sum_{k=1}^K w_{km} h_{kn} + \sum_{k=1}^K w_{km} h_{kn} \right) \\ &\leq \sum_{n=1}^N \sum_{m=1}^M \left(-x_{nm} \sum_{k=1}^K \lambda_{nmk} \log \frac{w_{km} h_{kn}}{\lambda_{nmk}} + \sum_{k=1}^K w_{km} h_{kn} \right) \\ &\stackrel{\text{def}}{=} \mathcal{U}(\mathbf{X}|\mathbf{Y}, \boldsymbol{\lambda}) \end{aligned} \quad (3.39)$$

ここで、式 (3.31) を用いました。また、補助変数 λ は、 $\sum_{k=1}^K \lambda_{nmk} = 1$ を満たすものとします。等号成立条件、すなわち $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \boldsymbol{\lambda})$ を最小化する λ は次式で与えられます。

$$\lambda_{nmk} = \frac{w_{km} h_{kn}}{\sum_{k'=1}^K w_{k'm} h_{k'n}} = \frac{w_{km} h_{kn}}{y_{nm}} \quad (3.40)$$

最後に、式 (3.39) を最小化する \mathbf{Y} (\mathbf{W} および \mathbf{H}) を求めます。まず、 $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \boldsymbol{\lambda})$ を w_{km} について偏微分すると、

$$\frac{\partial \mathcal{U}(\mathbf{X}|\mathbf{Y}, \boldsymbol{\lambda})}{\partial w_{km}} = \sum_{n=1}^N \left(-\frac{x_{nm} \lambda_{nmk}}{w_{km}} + h_{kn} \right) \quad (3.41)$$

を得ます。ここで、 $\frac{\partial \mathcal{U}(\mathbf{X}|\mathbf{Y}, \boldsymbol{\lambda})}{\partial w_{km}} = 0$ とおくと、

$$w_{km} = \frac{\sum_{n=1}^N x_{nm} \lambda_{nmk}}{\sum_{n=1}^N h_{kn}} \quad (3.42)$$

を得ます。同様に、 $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \boldsymbol{\lambda})$ を h_{kn} について偏微分して 0 とおくことで、

$$h_{kn} = \frac{\sum_{m=1}^M x_{nm} \lambda_{nmk}}{\sum_{m=1}^M w_{km}} \quad (3.43)$$

を得ます。EU-NMF と同様に、式 (3.40)、式 (3.42) および式 (3.43) を反復することにより、 $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \boldsymbol{\lambda})$ を逐次最小化していくことができます。

補助変数を介さない乗法更新則は、式 (3.40) を式 (3.42) および式 (3.43) に代入することで得られます。

$$w_{km} \leftarrow \frac{\sum_{n=1}^N x_{nm} \frac{w_{km} h_{kn}}{y_{nm}}}{\sum_{n=1}^N h_{kn}} = \frac{\sum_{n=1}^N h_{kn} \frac{x_{nm}}{y_{nm}}}{\sum_{n=1}^N h_{kn}} w_{km} \quad (3.44)$$

$$h_{kn} \leftarrow \frac{\sum_{m=1}^M x_{nm} \frac{w_{km} w_{km}}{y_{nm}}}{\sum_{m=1}^M w_{km}} = \frac{\sum_{m=1}^M w_{km} \frac{x_{nm}}{y_{nm}}}{\sum_{m=1}^M w_{km}} h_{kn} \quad (3.45)$$

全要素が 1 の行列を $\mathbf{1} \in \mathbb{R}^{M \times N}$ とすると、行列演算として記述できます。

$$\mathbf{W} \leftarrow \frac{\mathbf{H} \frac{\mathbf{X}^T}{\mathbf{Y}^T}}{\mathbf{H} \mathbf{1}^T} \odot \mathbf{W} \quad (3.46)$$

$$\mathbf{H} \leftarrow \frac{\mathbf{W} \frac{\mathbf{X}}{\mathbf{Y}}}{\mathbf{W} \mathbf{1}} \odot \mathbf{H} \quad (3.47)$$

アルゴリズム 3.2 : KL-NMF の最尤推定

Require: 非負値行列 $\mathbf{X} \in \mathbb{R}_+^{M \times N}$, 基底数 K

- 1: 非負値行列 $\mathbf{W} \in \mathbb{R}_+^{M \times K}$ をランダムに初期化
- 2: 非負値行列 $\mathbf{H} \in \mathbb{R}_+^{M \times K}$ をランダムに初期化
- 3: while 上限関数 $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \lambda)$ が未収束 do
- 4: $\mathbf{W} \leftarrow \frac{\mathbf{H} \mathbf{X}^T}{\mathbf{H} \mathbf{1}^T} \odot \mathbf{W}$
- 5: $\mathbf{H} \leftarrow \frac{\mathbf{W} \mathbf{X}}{\mathbf{W} \mathbf{1}} \odot \mathbf{H}$
- 6: end while
- 7: Return 非負値行列 \mathbf{W}, \mathbf{H}

アルゴリズム 3.2 に, KL-NMF のアルゴリズムを示します。これも, EU-NMF と同様に, ある確率モデルの最尤推定としての解釈が可能です。

3.1.5 IS-NMF の乗法更新アルゴリズム

IS ダイバージェンスに基づく NMF (IS-NMF) の乗法更新則を導出します。準備として, 逆数関数が下に凸であることから, Jensen の不等式 (付録 ??節) を用いて, $f(z) = \frac{1}{(\sum_{k=1}^K z_k)}$ の上限関数 $u(z, \lambda)$ を設計します。

$$f(z) = \frac{1}{\left(\sum_{k=1}^K \lambda_k \frac{z_k}{\lambda_k}\right)} \leq \sum_{k=1}^K \lambda_k \frac{1}{\frac{z_k}{\lambda_k}} = \sum_{k=1}^K \lambda_k^2 \frac{1}{z_k} \stackrel{\text{def}}{=} u(z, \lambda) \quad (3.48)$$

ここで, $z_k \geq 0$ は非負値の変数であり, $\lambda = \{\lambda_k\}_{k=1}^K$ は

$$\sum_{k=1}^K \lambda_k = 1 \quad (3.49)$$

を満たす非負値の補助変数です。「和の逆数関数」の上限関数として、「逆数関数の和」が得られていることに注意してください。

等号成立条件、すなわち、式 (3.49) の制約条件付きで $u(z, \lambda)$ を最小化する λ を求めるには、未定乗数 ϕ を用いて、新たな関数

$$F(\lambda, \phi) = \sum_{k=1}^K \lambda_k^2 \frac{1}{z_k} + \phi \left(1 - \sum_{k=1}^K \lambda_k \right) \quad (3.50)$$

を考えます。これを λ_k について偏微分すると

$$\frac{\partial F(\lambda, \phi)}{\partial \lambda_k} = \frac{2\lambda_k}{z_k} - \phi \quad (3.51)$$

を得ます。 $\frac{\partial F(\lambda, \phi)}{\partial \lambda_k} = 0$ とおくと、 ϕ を用いて λ_k が表せます。

$$\lambda_k = \frac{\phi}{2} z_k \quad (3.52)$$

これを式 (3.49) に代入すると、

$$1 = \sum_{k=1}^K \lambda_k = \frac{\phi}{2} \sum_{k=1}^K z_k \quad (3.53)$$

となるので、未定乗数 ϕ は

$$\frac{2}{\phi} = \sum_{k=1}^K z_k \quad (3.54)$$

で与えられます。これを、式 (3.52) に代入すると、次式を得ます。

$$\lambda_k = \frac{z_k}{\sum_{k=1}^K z_k} \quad (3.55)$$

さらに、対数関数 $f(z) = \log(z)$ の上限関数 $u(z, \omega)$ を設計します。対数関数は上に凸であることから、任意の点 $\omega > 0$ における接線は常に対数関数の上側にあります。したがって、 ω における一次のテイラー展開を行うと、

$$f(z) \leq \log \omega + \frac{z}{\omega} - 1 \stackrel{\text{def}}{=} u(z, \omega) \quad (3.56)$$

を得ます。等号成立条件、すなわち、 $u(z, \omega)$ を最小化する ω は、 $u(z, \omega)$ を ω で偏微分して 0 とおくことで求まります。

$$\omega = x \quad (3.57)$$

これらの結果を用いて、コスト関数 $\mathcal{D}(\mathbf{X}|\mathbf{Y})$ に対して、補助変数 λ および ω を含む上限関数 $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \lambda, \omega)$ を導出します。

$$\begin{aligned}
 \mathcal{D}(\mathbf{X}|\mathbf{Y}) &= \sum_{n=1}^N \sum_{m=1}^M \left(\frac{x_{nm}}{y_{nm}} - \log \frac{x_{nm}}{y_{nm}} - 1 \right) \\
 &= \sum_{m=1}^M \left(\frac{x_{nm}}{\sum_{k=1}^K w_{km} h_{kn}} - \log \frac{x_{nm}}{\sum_{k=1}^K w_{km} h_{kn}} - 1 \right) \\
 &\stackrel{c}{=} \sum_{m=1}^M \left(\frac{x_{nm}}{\sum_{k=1}^K w_{km} h_{kn}} + \log \sum_{k=1}^K w_{km} h_{kn} \right) \\
 &\leq \sum_{n=1}^N \sum_{m=1}^M \left(\sum_{k=1}^K \frac{x_{nm} \lambda_{nmk}^2}{w_{km} h_{kn}} + \log \omega_{nm} + \frac{1}{\omega_{nm}} \sum_{k=1}^K w_{km} h_{kn} - 1 \right) \\
 &\stackrel{\text{def}}{=} \mathcal{U}(\mathbf{X}|\mathbf{Y}, \lambda, \omega)
 \end{aligned} \tag{3.58}$$

ここで、式 (3.48) および式 (3.56) を用いました。補助変数に関しては、 $\lambda = \{\lambda_{nmk}\}_{n=1, m=1, k=1}^{N, M, K}$ は $\sum_{k=1}^K \lambda_{nmk} = 1$ を満たし、 $\omega = \{\omega_{nm}\}_{n=1, m=1}^{N, M}$ はすべて非負値とします。等号成立条件、すなわち $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \lambda, \omega)$ を最小化する λ および ω は次式で与えられます。

$$\lambda_{nmk} = \frac{w_{km} h_{kn}}{\sum_{k'=1}^K w_{k'm} h_{k'n}} = \frac{w_{km} h_{kn}}{y_{nm}} \tag{3.59}$$

$$\omega_{nm} = \sum_{k=1}^K w_{km} h_{kn} = y_{nm} \tag{3.60}$$

最後に、式 (3.58) を最小化する \mathbf{Y} (\mathbf{W} および \mathbf{H}) を求めます。まず、 $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \lambda, \omega)$ を w_{km} について偏微分すると、

$$\frac{\partial \mathcal{U}(\mathbf{X}|\mathbf{Y}, \lambda, \omega)}{\partial w_{km}} = \sum_{n=1}^N \left(-\frac{x_{nm} \lambda_{nmk}^2}{w_{km}^2 h_{kn}} + \frac{h_{kn}}{\omega_{nm}} \right) \tag{3.61}$$

を得ます。ここで、 $\frac{\partial \mathcal{U}(\mathbf{X}|\mathbf{Y}, \lambda, \omega)}{\partial w_{km}} = 0$ とおくと、

$$w_{km} = \sqrt{\frac{\sum_{n=1}^N \frac{x_{nm} \lambda_{nmk}^2}{h_{kn}}}{\sum_{n=1}^N \frac{h_{kn}}{\omega_{nm}}}} \tag{3.62}$$

を得ます。同様に、 $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \lambda, \omega)$ を h_{kn} について偏微分して 0 とおくと、

$$h_{kn} = \sqrt{\frac{\sum_{m=1}^M \frac{x_{nm}\lambda_{nmk}^2}{w_{km}}}{\sum_{n=1}^N \frac{h_{kn}}{\omega_{nm}}}} \quad (3.63)$$

を得ます。式 (3.59)、式 (3.60)、式 (3.62) および式 (3.63) を反復することにより、 $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \lambda, \omega)$ を逐次最小化していくことができます。

補助変数を介さない乗法更新則は、式 (3.59) および式 (3.60) を、式 (3.62) および式 (3.63) に代入することで得られます。

$$w_{km} \leftarrow \sqrt{\frac{\sum_{n=1}^N \frac{x_{nm} w_{km}^2 h_{kn}^2}{h_{kn} y_{nm}^2}}{\sum_{n=1}^N \frac{h_{kn}}{y_{nm}}}} = \sqrt{\frac{\sum_{n=1}^N h_{kn} \frac{x_{nm}}{y_{nm}^2}}{\sum_{n=1}^N h_{kn} \frac{1}{y_{nm}}}} w_{km} \quad (3.64)$$

$$h_{kn} \leftarrow \sqrt{\frac{\sum_{m=1}^M \frac{x_{nm} w_{km}^2 h_{kn}^2}{w_{km} y_{nm}^2}}{\sum_{m=1}^M \frac{w_{km}}{y_{nm}}}} = \sqrt{\frac{\sum_{m=1}^M w_{km} \frac{x_{nm}}{y_{nm}^2}}{\sum_{m=1}^M w_{km} \frac{1}{y_{nm}}}} h_{kn} \quad (3.65)$$

さらに、これらは行列演算として簡潔に記述することもできます。

$$\mathbf{W} \leftarrow \sqrt{\frac{\mathbf{H} \frac{\mathbf{X}^T}{\mathbf{Y}^T \odot \mathbf{Y}^T}}{\mathbf{H} \frac{\mathbf{1}^T}{\mathbf{Y}^T}}} \odot \mathbf{W} \quad (3.66)$$

$$\mathbf{H} \leftarrow \sqrt{\frac{\mathbf{W} \frac{\mathbf{X}}{\mathbf{Y} \odot \mathbf{Y}}}{\mathbf{W} \frac{\mathbf{1}}{\mathbf{Y}}}} \odot \mathbf{H} \quad (3.67)$$

ここで、 $\mathbf{1} \in \mathbb{R}^{M \times N}$ は、全ての要素が 1 の行列です。 $\sqrt{-}$ は行列の要素ごとに平方根をとるものとします。

アルゴリズム 3.3 に、IS-NMF のアルゴリズムを示します。IS-NMF も、ある確率モデルの最尤推定としての解釈が可能です。アルゴリズム 3.1、アルゴリズム 3.2、アルゴリズム 3.3 を比較すると、EU-NMF では、分子・分母に \mathbf{X} および \mathbf{Y} を、KL-NMF では $\frac{\mathbf{X}}{\mathbf{Y}}$ および $\mathbf{1}$ を、IS-NMF では $\frac{\mathbf{X}}{\mathbf{Y} \odot \mathbf{Y}}$ および $\frac{\mathbf{1}}{\mathbf{Y}}$ をそれぞれ掛けています（比はすべて $\frac{\mathbf{X}}{\mathbf{Y}}$ で等しい）。NMF には \mathbf{W} や \mathbf{H} をスパースに誘導する効果があり、それらの積で計算できる再構成行列 $\mathbf{Y} = \mathbf{WH}$ もスパースになりやすい傾向があります。このとき、 \mathbf{Y} のある要素が非常に小さな値になると、その逆数は非常に大きな値となってしまうため、IS-NMF は数値的に不安定になることがあります。

アルゴリズム 3.3 : IS-NMF の最尤推定

Require: 非負値行列 $\mathbf{X} \in \mathbb{R}_+^{M \times N}$, 基底数 K

- 1: 非負値行列 $\mathbf{W} \in \mathbb{R}_+^{M \times K}$ をランダムに初期化
- 2: 非負値行列 $\mathbf{H} \in \mathbb{R}_+^{M \times K}$ をランダムに初期化
- 3: while 上限関数 $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \lambda, \omega)$ が未収束 do
- 4: $\mathbf{W} \leftarrow \sqrt{\frac{\mathbf{H} \frac{\mathbf{X}^T}{\mathbf{Y}^T \odot \mathbf{Y}^T}}{\mathbf{H} \frac{1^T}{\mathbf{Y}^T}}} \odot \mathbf{W}$
- 5: $\mathbf{H} \leftarrow \sqrt{\frac{\mathbf{W} \frac{\mathbf{X}}{\mathbf{Y} \odot \mathbf{Y}}}{\mathbf{W} \frac{1}{\mathbf{Y}}}} \odot \mathbf{H}$
- 6: end while
- 7: Return 非負値行列 \mathbf{W}, \mathbf{H}

3.1.6 β -NMF の乗法更新アルゴリズム

これまで議論してきた EU-NMF, KL-NMF, IS-NMF を統一的に取り扱うことができる NMF として, β ダイバージェンスに基づく β -NMF が提案されています [?]. 観測ベクトル \mathbf{x}_n と再構成ベクトル \mathbf{y}_n との近似誤差を表す β ダイバージェンスは次式で定義されます.

$$\mathcal{D}_\beta(\mathbf{x}_n|\mathbf{y}_n) = \sum_{m=1}^M \left(\frac{x_{nm}^\beta}{\beta(\beta-1)} + \frac{y_{nm}^\beta}{\beta} - \frac{x_{nm}y_{nm}^{\beta-1}}{\beta-1} \right) \quad (3.68)$$

ここで, $\beta \in \mathbb{R}$ は $0, 1$ を除く任意の実数であり, $\beta = 2$ のときユークリッド距離に一致します. また, $\beta \rightarrow 1$ および $\beta \rightarrow 0$ の極限を考えると,

$$\begin{aligned} \lim_{\beta \rightarrow 1} \mathcal{D}_\beta(\mathbf{x}_n|\mathbf{y}_n) &= \lim_{\beta \rightarrow 1} \sum_{m=1}^N \left(x_{nm} \frac{x_{nm}^{\beta-1} - y_{nm}^{\beta-1}}{1-\beta} - \frac{x_{nm}^\beta - y_{nm}^\beta}{\beta} \right) \\ &= \sum_{m=1}^N (x_{nm}(\log x_{nm} - \log y_{nm}) - (x_{nm} - y_{nm})) \end{aligned} \quad (3.69)$$

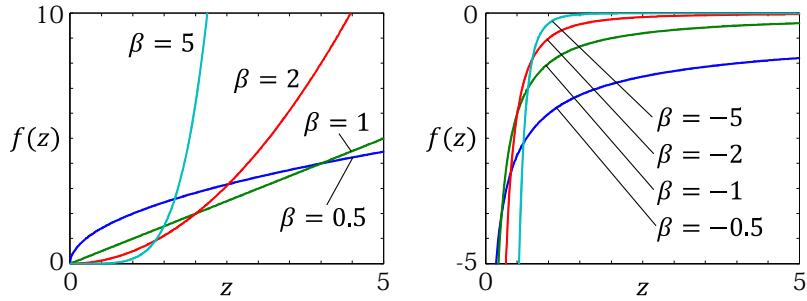


図 3.3 いくつかの β に対する関数 $f_\beta(z) = \frac{1}{\beta}z^\beta$ のプロット . $\beta = 1$ を境界として , $\beta \geq 1$ のときは下に凸 (convex) , $\beta \leq 1$ のときは上に凸 (concave) となることに注意 .

$$\begin{aligned} \lim_{\beta \rightarrow 0} \mathcal{D}_\beta(\mathbf{x}_n | \mathbf{y}_n) &= \lim_{\beta \rightarrow 0} \sum_{m=1}^N \left(x_{nm} \frac{y_{nm}^{\beta-1}}{1-\beta} - \frac{x_{nm}^\beta - y_{nm}^\beta}{\beta} + \frac{x_{nm}^\beta}{\beta-1} \right) \\ &= \sum_{m=1}^N \left(\frac{x_{nm}}{y_{nm}} - \log \frac{x_{nm}}{y_{nm}} - 1 \right) \end{aligned} \quad (3.70)$$

となるので , $\beta \rightarrow 1$ のとき KL ダイバージェンス , $\beta \rightarrow 0$ のとき IS ダイバージェンスと等価であることが分かります . このように , β の値を調節することで , EU-NMF , KL-NMF , IS-NMF をはじめ , それらの中間的な性質をもつ NMF を統一的に記述できます .

β ダイバージェンスに基づく NMF (β -NMF) の乗法更新則を導出しましょう . 式 (3.68) で与えられるコスト関数をみると , 第一項は定数ですが , 第二項および第三項は , β の値によって下に凸であるか上に凸であるかが決まります . そのため , β の値で場合分けを行う必要があります .

まず , 準備として , 関数 $f_\beta(z) = \frac{1}{\beta}z^\beta$ ($z > 0$) について考えます (図 3.3) . $\beta \leq 1$ の場合は , $f_\beta(z)$ は上に凸であることから , 任意の点 $\omega > 0$ における一次の泰ラーラー展開 (ω における接線方程式) を考えることで , $f_\beta(z)$ の上限関数 $u_\beta(z, \omega)$ を設計できます .

$$f_\beta(z) \leq \frac{1}{\beta} \omega^\beta + \omega^{\beta-1} (z - \omega) = \frac{\beta+1}{\beta} \omega^\beta + \omega^{\beta-1} z \stackrel{\text{def}}{=} u_\beta(z, \omega) \quad (3.71)$$

ここで , 等号成立条件 , すなわち , $u_\beta(z, \omega)$ を最小化する ω は , $u_\beta(z, \omega)$ を

ω で偏微分して 0 とおくことで求まります（導出は EU-NMF, KL-NMF, IS-NMF などと同様なので省略）。

$$\omega = x \quad (3.72)$$

一方、 $\beta \geq 1$ の場合は、 $f_\beta(z)$ は下に凸であることから、Jensen の不等式を用いて、 $f'_\beta(z) = \frac{1}{\beta} \left(\sum_{k=1}^K z_k \right)^\beta$ の上限関数 $u'_\beta(z, \lambda)$ を設計できます。

$$f'_\beta(z) = \frac{1}{\beta} \left(\sum_{k=1}^K \lambda_k \frac{z_k}{\lambda_k} \right)^\beta \leq \frac{1}{\beta} \sum_{k=1}^K \lambda_k \left(\frac{z_k}{\lambda_k} \right)^\beta \stackrel{\text{def}}{=} u'_\beta(z, \lambda) \quad (3.73)$$

ここで、補助変数 $\lambda = \{\lambda_k\}_{k=1}^K$ は、 $\sum_{k=1}^K \lambda_k = 1$ を満たします。等号成立条件、すなわち、 $u'_\beta(z, \lambda)$ を最小化する λ は、 $u'_\beta(z, \lambda)$ を λ で偏微分して 0 とおくことで求まります（導出は省略）。

$$\lambda_k = \frac{z_k}{\sum_{k=1}^K z_k} \quad (3.74)$$

また、関数 $g'_\beta(z) = -\frac{1}{\beta-1} \left(\sum_{k=1}^K z_k \right)^{\beta-1}$ および関数 $g_\beta(z) = -\frac{1}{\beta-1} z^{\beta-1}$ についても、 $g'_\beta(z) = -f'_{\beta-1}(z)$ および $g_\beta(z) = -f_{\beta-1}(z)$ となることを用いて、上限関数をそれぞれ設計できます。

$$g'_\beta(z) \leq -u'_{\beta-1}(z, \lambda) \quad (\beta \leq 2) \quad (3.75)$$

$$g_\beta(z) \leq -u_{\beta-1}(z, \omega) \quad (\beta \geq 2) \quad (3.76)$$

各場合における等号成立条件、すなわち、上限関数を最小化する補助変数 λ あるいは ω は、次式で与えられます。

$$\lambda_k = \frac{z_k}{\sum_{k=1}^K z_k} \quad (\beta \leq 2) \quad (3.77)$$

$$\omega = x \quad (\beta \geq 2) \quad (3.78)$$

これらの結果を用いて、コスト関数 $\mathcal{D}(\mathbf{X}|\mathbf{Y})$ に対して、補助変数 λ および ω を含む上限関数 $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \lambda, \omega)$ を導出します。

$$\mathcal{D}(\mathbf{X}|\mathbf{Y}) \stackrel{c}{=} \sum_{n=1}^N \sum_{m=1}^M \left(\frac{1}{\beta} y_{nm}^\beta - x_{nm} \frac{1}{\beta-1} y_{nm}^{\beta-1} \right)$$

$$= \begin{cases} \sum_{n=1}^N \sum_{m=1}^M \left(u_\beta(y_{nm}, \omega_{nm}) - x_{nm} u'_{\beta-1}(y_{nm}, \lambda_{nm}) \right) & (\beta < 1) \\ \sum_{n=1}^N \sum_{m=1}^M \left(u'_\beta(y_{nm}, \lambda_{nm}) - x_{nm} u'_{\beta-1}(y_{nm}, \lambda_{nm}) \right) & (1 \leq \beta \leq 2) \\ \sum_{n=1}^N \sum_{m=1}^M \left(u'_\beta(y_{nm}, \lambda_{nm}) - x_{nm} u_{\beta-1}(y_{nm}, \omega_{nm}) \right) & (\beta > 2) \end{cases}$$

$\stackrel{\text{def}}{=} \mathcal{U}(X|Y, \lambda, \omega) \quad (3.79)$

ここで、 $y_{nm} = \{y_{nmk}\}_{k=1}^K = \{w_{km} h_{kn}\}_{k=1}^K$ 、 $\lambda_{nm} = \{\lambda_{nmk}\}_{k=1}^K$ と定義しました。いずれの場合においても、等号成立条件、すなわち $\mathcal{U}(X|Y, \lambda, \omega)$ を最小化する λ および ω は次式で与えられます。

$$\lambda_{nmk} = \frac{w_{km} h_{kn}}{y_{nm}} \quad (3.80)$$

$$\omega_{nm} = y_{nm} \quad (3.81)$$

最後に、 $\mathcal{U}(X|Y, \lambda, \omega)$ を最小化する Y (W および H) を求めます。
 $\mathcal{U}(X|Y, \lambda, \omega)$ は、 w_{km} あるいは h_{kt} に関する一次導関数および二次導関数を計算することにより、いずれの変数についても凸関数であることが分かります^[?]。したがって、 $\mathcal{U}(X|Y, \lambda, \omega)$ を w_{km} について偏微分してゼロとおくことにより、 H が与えられたもとの w_{km} の最適解が得られます。

$$w_{km} = \begin{cases} \left(\frac{\sum_{n=1}^N \lambda_{nm}^{2-\beta} x_{nm} h_{kn}^{\beta-1}}{\sum_{n=1}^N \omega_{nm}^{\beta-1} h_{kn}^{\beta-1}} \right)^{\frac{1}{2-\beta}} & (\beta < 1) \\ \frac{\sum_{n=1}^N \lambda_{nm}^{2-\beta} x_{nm} h_{kn}^{\beta-1}}{\sum_{n=1}^N \omega_{nm}^{1-\beta} h_{kn}^{\beta}} & (1 \leq \beta \leq 2) \\ \left(\frac{\sum_{n=1}^N \lambda_{nm}^{\beta-2} x_{nm} h_{kn}}{\sum_{n=1}^N \omega_{nm}^{1-\beta} h_{kn}^{\beta}} \right)^{\frac{1}{\beta-1}} & (\beta > 2) \end{cases} \quad (3.82)$$

補助関数を介さない乗法更新則は、式 (3.80) および式 (3.81) を式 (3.82) に代入することで得られます。

$$w_{km} \leftarrow \left(\frac{\sum_{n=1}^N h_{kn} y_{nm}^{\beta-2} x_{nm}}{\sum_{n=1}^N h_{kn} y_{nm}^{\beta-1}} \right)^{\psi(\beta)} w_{km} \quad (3.83)$$

ここで、 $\psi(\beta)$ は β の関数であり、次式で与えられます。

$$\psi(\beta) = \begin{cases} \frac{1}{2-\beta} & (\beta < 1) \\ 1 & (1 \leq \beta \leq 2) \\ \frac{1}{\beta-1} & (\beta > 2) \end{cases} \quad (3.84)$$

h_{kn} の乗法更新則も同様にして導出できます。

$$h_{kn} \leftarrow \left(\frac{\sum_{m=1}^M w_{km} y_{nm}^{\beta-2} x_{nm}}{\sum_{m=1}^M w_{km} y_{nm}^{\beta-1}} \right)^{\psi(\beta)} h_{kn} \quad (3.85)$$

さらに、これらは行列演算として簡潔に記述することもできます。

$$\mathbf{W} \leftarrow \left(\frac{\mathbf{H} (\mathbf{X} \odot \mathbf{Y}^{\beta-2})^T}{\mathbf{H} (\mathbf{Y}^{\beta-1})^T} \right)^{\psi(\beta)} \odot \mathbf{W} \quad (3.86)$$

$$\mathbf{H} \leftarrow \left(\frac{\mathbf{W} (\mathbf{X} \odot \mathbf{Y}^{\beta-2})^T}{\mathbf{W} (\mathbf{Y}^{\beta-1})^T} \right)^{\psi(\beta)} \odot \mathbf{H} \quad (3.87)$$

ここで、 Z^α は、行列 Z の各要素を α 乗することを意味するものとします。

アルゴリズム 3.4 : β -NMF の最尤推定

Require: 非負値行列 $\mathbf{X} \in \mathbb{R}_+^{M \times N}$, 基底数 K , 任意の実数 β

1: 非負値行列 $\mathbf{W} \in \mathbb{R}_+^{M \times K}$ をランダムに初期化

2: 非負値行列 $\mathbf{H} \in \mathbb{R}_+^{M \times K}$ をランダムに初期化

$$3: \psi(\beta) = \begin{cases} \frac{1}{2-\beta} & (\beta < 1) \\ 1 & (1 \leq \beta \leq 2) \\ \frac{1}{\beta-1} & (\beta > 2) \end{cases}$$

4: while 上限関数 $\mathcal{U}(\mathbf{X}|\mathbf{Y}, \lambda, \omega)$ が未収束 do

$$5: \quad \mathbf{W} \leftarrow \left(\frac{\mathbf{H} (\mathbf{X} \odot \mathbf{Y}^{\beta-2})^T}{\mathbf{H} (\mathbf{Y}^{\beta-1})^T} \right)^{\psi(\beta)} \odot \mathbf{W}$$

$$6: \quad \mathbf{H} \leftarrow \left(\frac{\mathbf{W} (\mathbf{X} \odot \mathbf{Y}^{\beta-2})^T}{\mathbf{W} (\mathbf{Y}^{\beta-1})^T} \right)^{\psi(\beta)} \odot \mathbf{H}$$

7: end while

8: Return 非負値行列 \mathbf{W}, \mathbf{H}

アルゴリズム 3.4 に、 β -NMF のアルゴリズムを示します。これは、 $\beta = 2$ とすると EU-NMF におけるアルゴリズム 3.1 に、 $\beta = 1$ とすると KL-

NMF におけるアルゴリズム 3.2 に , $\beta = 0$ とすると IS-NMF におけるアルゴリズム 3.3 に帰着することから , 統一的な乗法更新アルゴリズムとなっていることが分かります .

3.1.7 乗法更新アルゴリズム

NMF に対する乗法更新アルゴリズムは一意に定まるものではなく , さまざまなバリエーションが存在します . これまで紹介してきた補助関数法に基づく乗法更新アルゴリズム (例 : アルゴリズム 3.4) は , コスト関数の上限関数を導入し , 上限関数を逐次最小化することにより , もとのコスト関数を間接的に逐次最小化することができます . さらに , 収束性が保証されているという好ましい性質をもちます . ただし , このようにして得られた反復更新則が , 乗法更新型の形式をとっていたことは必然的ではありません . つまり , 私たちは最初から乗法更新則を導出しようと考えていたわけではなく , あくまで補助関数法に基づくコスト関数最小化の枠組みに従った結果 , たまたま乗法更新則が得られたということです .

本節では , より直接的に NMF の乗法更新則を導出するアプローチについて説明します . 研究コミュニティにおいて , 乗法更新則といえば , このアプローチで導出されたものを指すことが一般的です . しかし , 必ずしも最も好ましい乗法更新則になっているとは限らないことに注意が必要です . まず , 多くの場合で , このアプローチで導出される乗法更新則には収束性が保証されません . また , 経験的にはほとんどの場合で収束するとしても , 収束速度が最も高速であるとも限りません .

具体的に , β -NMF の乗法更新則を導出してみましょう . まず , 最小化すべきコスト関数は次式で与えられます .

$$\mathcal{D}(X|Y) \stackrel{c}{=} \sum_{n=1}^N \sum_{m=1}^M \left(\frac{1}{\beta} y_{nm}^\beta - x_{nm} \frac{1}{\beta-1} y_{nm}^{\beta-1} \right) \quad (3.88)$$

私たちの目的は , $\mathcal{D}(X|Y)$ を逐次最小化するような乗法更新則

$$w_{km} \leftarrow \eta_{km} w_{km} \quad (3.89)$$

$$h_{kn} \leftarrow \zeta_{kn} h_{kn} \quad (3.90)$$

を見つけることです .

適切な係数 η_{km} および ζ_{kn} を求めるには、まず、 $\mathcal{D}(\mathbf{X}|\mathbf{Y})$ を w_{km} について偏微分します。

$$\begin{aligned}\frac{\partial \mathcal{D}(\mathbf{X}|\mathbf{Y})}{\partial w_{km}} &= \frac{\partial \mathcal{D}(\mathbf{X}|\mathbf{Y})}{\partial y_{nm}} \frac{\partial y_{nm}}{\partial w_{km}} \\ &= \sum_{n=1}^N y_{nm}^{\beta-1} h_{kn} - \sum_{n=1}^N x_{nm} y_{nm}^{\beta-2} h_{kn} \\ &\stackrel{\text{def}}{=} \kappa_{km}^+ - \kappa_{km}^-\end{aligned}\quad (3.91)$$

ここで、 $y_{nm} = \sum_{k=1}^K w_{km} h_{kn}$ であることを用いました。式 (3.91) では、第一項 κ_{km}^+ および第二項 κ_{km}^- はいずれも非負値であることから、 $\mathcal{D}(\mathbf{X}|\mathbf{Y})$ の偏微分が、二つの非負値の差によって表現されていることが分かります。ここで、 κ_{km}^+ を分母に、 κ_{km}^- を分子として、 η_{km} を

$$\eta_{km} = \frac{\kappa_{km}^-}{\kappa_{km}^+} = \frac{\sum_{n=1}^N x_{nm} y_{nm}^{\beta-2} h_{kn}}{\sum_{n=1}^N y_{nm}^{\beta-1} h_{kn}} \quad (3.92)$$

で計算することにします。

h_{kn} についても同様であり、最終的に得られる乗法更新則を行列演算形式で記述すると以下の通りです。

$$\mathbf{W} \leftarrow \frac{\mathbf{H} (\mathbf{X} \odot \mathbf{Y}^{\beta-2})^T}{\mathbf{H} (\mathbf{Y}^{\beta-1})^T} \odot \mathbf{W} \quad (3.93)$$

$$\mathbf{H} \leftarrow \frac{\mathbf{W} (\mathbf{X} \odot \mathbf{Y}^{\beta-2})^T}{\mathbf{W} (\mathbf{Y}^{\beta-1})^T} \odot \mathbf{H} \quad (3.94)$$

これらの乗法更新則については、 $1 \leq \beta \leq 2$ のときに限り、収束性が証明されています。式 (3.93) および式 (3.94) と、式 (3.86) および式 (3.87) を比較すると、係数部分に $\psi(\beta)$ 乗がかかっているかどうかの違いがあります。

3.2 非負値行列因子分解の確率的な解釈

NMF は最尤推定であってもスパースな解が得られやすいが、適切な事前分布を導入してベイズ推定を行うことで、よりスパースな解を得ることができます。更に、ノンパラメトリックベイズモデルを定式化すれば、基底数を

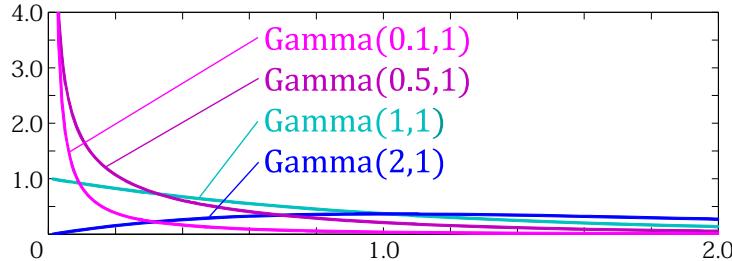


図 3.4 形状母数が異なる幾つかのガンマ分布 .

$K \rightarrow \infty$ とした場合でもスパースな学習が可能になる。すなわち、観測行列 X に合わせて高々有限個の基底がアクティベートされるような機構が実現できる。具体的には、ガンマ過程あるいはベータ過程を事前分布に用いることになる。

3.2.1 確率モデルの最尤推定としての定式化

3.2.2 ノンパラメトリックベイズモデル

ガンマ過程に基づく NMF のノンパラメトリックベイズモデル (GaP-NMF) について説明する。まず、式 (3.2) に対し、 K 次元の非負値ベクトル $\theta = [\theta_1, \theta_2, \dots, \theta_K]$ を導入する。

$$\mathbf{x}_n \approx \sum_{k=1}^K \theta_k h_{kn} \mathbf{w}_k \stackrel{\text{def}}{=} \mathbf{y}_n \quad (3.95)$$

ここで、 $\theta_k \geq 0$ は基底 k の大域的な重みである。この θ に対し、観測データ X を表現するのに必要な基底 k 以外の要素 θ_k がゼロとなるようなスパースな学習を行いたい。

ノンパラメトリックベイズモデルを定式化するため、 θ, W, H に対して事前分布を導入する。まず、 W 及び H の各要素は非負値であるので、ガンマ事前分布を用いると都合が良い。

$$w_{km} \sim \text{Gamma}(a_0^w, b_0^w) \quad (3.96)$$

$$h_{kn} \sim \text{Gamma}(a_0^h, b_0^h) \quad (3.97)$$

ここで, $a_0^* > 0$ 及び $b_0^* > 0$ はそれぞれ, ガンマ分布の形状母数と逆尺度母数である. 更に, θ に対しても同様にガンマ事前分布を仮定する.

$$\theta_k \sim \text{Gamma}\left(\frac{\alpha c}{K}, \alpha\right) \quad (3.98)$$

ここで, $\alpha > 0$ 及び $c > 0$ は超パラメータである. ガンマ分布の形状母数が小さくなるほど 0 が出る確率が大きくなる(図 3.4). ただし, $\mathbb{E}_{\text{prior}}[\theta_k] = \frac{c}{K}$, $\mathbb{E}_{\text{prior}}[\sum_k \theta_k] = c$ である.

ここで, 式 (3.96), 式 (3.97) 及び式 (3.98) で構成される有限モデルに対して, $K \rightarrow \infty$ となる極限を考えると, 以下のガンマ過程が得られる.

$$G \sim \text{GaP}(\alpha, G_0) \quad (3.99)$$

ここで, G_0 は空間 U ($w \in \mathbb{R}_+^M$ と $h \in \mathbb{R}_+^N$ の直積空間) 上に定義された基底測度であり, $G_0(U) = c$ を満たす(図??). このとき, G は U 上の離散測度となり, 空間 U の任意の分割 $\{U_i\}_{i=1}^I$ に対して

$$G(U_i) \sim \text{Gamma}(\alpha G_0(U_i), \alpha) \quad (3.100)$$

が成立している. ただし, $\mathbb{E}[G] = G_0$ である. 微小区間への分割を $\{U_k\}_{k=1}^\infty$ とすると, $G(U_k) = \theta_k$ である. α は集中度と呼ばれ, α が小さくなるほど θ はよりスパースになる. 計算機上では $K \rightarrow \infty$ は扱えないが, K を α に比べて十分大きな値に設定すれば, 式 (3.98) はガンマ過程の良い近似となる(weak-limit approximation).

3.2.3 GaP-KL-NMF のベイズ推定

まず, GaP-KL-NMF に対する VB を導出する. 式 (??) で与えられる変分下限 $\mathcal{L}(q)$ の第一項は式 (5.10) で計算できる対数ボアソン尤度の期待値であるが, 依然として解析的に計算できない. そのため, 凹関数 $f(x) = \log(x)$ に対して Jensen の不等式を用いると更なる変分下限

$$\begin{aligned} & \mathbb{E}_q[\log p(\mathbf{X}|\boldsymbol{\theta}, \mathbf{W}, \mathbf{H})] \\ & \stackrel{c}{=} \mathbb{E}_q\left[\sum_{nm} \left(x_{nm} \log \sum_k y_{knm} - \sum_k y_{knm} \right)\right] \end{aligned}$$

Algorithm 1 GaP-KL-NMF のペイズ推定

Require: 非負値行列 $\mathbf{X} \in \mathbb{R}_+^{M \times N}$, 最大基底数 K , ガンマ過程の集中度 α , ガンマ分布のパラメータ $a_0^w, b_0^w, a_0^h, b_0^h$

- 1: 変分事後分布 $q(\boldsymbol{\theta}), q(\mathbf{W}), q(\mathbf{H})$ をランダムに初期化
- 2: **while** not converged **do**
- 3: $\lambda_{knm} \propto \mathbb{E}_q[\theta_k w_{km} h_{kn}]$
- 4: $q(\theta_k) = \text{Gamma}(\frac{\alpha c}{K} + \sum_{nm} \lambda_{knm} x_{nm},$
- 5: $\alpha + \sum_{nm} \mathbb{E}_q[w_{km} h_{kn}])$
- 6: $q(w_{km}) = \text{Gamma}(a_0^w + \sum_n \lambda_{knm} x_{nm},$
- 7: $b_0^w + \sum_n \mathbb{E}_q[\theta_k h_{kn}])$
- 8: $q(h_{kn}) = \text{Gamma}(a_0^h + \sum_m \lambda_{knm} x_{nm},$
- 9: $b_0^h + \sum_m \mathbb{E}_q[\theta_k w_{km}])$
- 10: **end while**
- 11: **Return** 変分事後分布 $q(\boldsymbol{\theta}), q(\mathbf{W}), q(\mathbf{H})$

$$\begin{aligned}
&= \sum_{nm} x_{nm} \mathbb{E}_q \left[\log \sum_k \lambda_{knm} \frac{y_{knm}}{\lambda_{knm}} \right] \\
&\quad - \sum_{knm} \mathbb{E}_q [y_{knm}] \\
&\geq \sum_{nm} x_{nm} \sum_k \lambda_{knm} \mathbb{E}_q \left[\log \frac{y_{knm}}{\lambda_{knm}} \right] \\
&\quad - \sum_{knm} \mathbb{E}_q [y_{knm}] \\
&\stackrel{\text{def}}{=} \mathbb{E}_q [\log q(\mathbf{X} | \boldsymbol{\theta}, \mathbf{W}, \mathbf{H})]
\end{aligned} \tag{3.101}$$

を得る。ここで、 λ_{knm} は $\sum_k \lambda_{knm} = 1$ を満たす補助変数である。等号成立条件（変分下限が最大となる条件）はラグランジュの未定乗数法を用いて求めることができ、 $\lambda_{knm} \propto \mathbb{E}_q[y_{knm}]$ となる。

最後に、各パラメータに対する変分事後分布を導出する。実際には式(??)で与えられる元の変分下限 $\mathcal{L}(q)$ ではなく、式(3.101)を用いて得られた異なる変分下限を最大化することになる。その結果、式(??), (??), (??)において、 $\log p(\mathbf{X}, \boldsymbol{\theta}, \mathbf{W}, \mathbf{H})$ の代わりに次式を用いることになる。

$$\log q(\mathbf{X}, \boldsymbol{\theta}, \mathbf{W}, \mathbf{H}) = \log q(\mathbf{X} | \boldsymbol{\theta}, \mathbf{W}, \mathbf{H})$$

Algorithm 2 GaP-IS-NMF のベイズ推定

Require: 非負値行列 $\mathbf{X} \in \mathbb{R}_+^{M \times N}$, 最大基底数 K , ガンマ過程の集中度 α , ガンマ分布のパラメータ $a_0^w, b_0^w, a_0^h, b_0^h$

- 1: 変分事後分布 $q(\theta), q(\mathbf{W}), q(\mathbf{H})$ をランダムに初期化
- 2: **while** not converged **do**
- 3: $\lambda_{knm} \propto \mathbb{E}_q[\theta_k^{-1} w_{km}^{-1} h_{kn}^{-1}]^{-1}$
- 4: $\omega_{nm} \propto \sum_k \mathbb{E}_q[\theta_k w_{km} h_{kn}]$
- 5: $q(\theta_k) = \text{GIG}(\frac{\alpha c}{K}, \alpha + \sum_{nm} \omega_{nm}^{-1} \mathbb{E}_q[w_{km}] \mathbb{E}_q[h_{kn}],$
 $\sum_{nm} x_{nm} \lambda_{knm}^2 \mathbb{E}_q[w_{mk}^{-1}] \mathbb{E}_q[h_{kn}^{-1}])$
- 6: $q(w_{km}) = \text{GIG}(a_0^w, b_0^w + \sum_n \omega_{nm}^{-1} \mathbb{E}_q[\theta_k] \mathbb{E}_q[h_{kn}],$
 $\sum_n x_{nm} \lambda_{knm}^2 \mathbb{E}_q[\theta_k^{-1}] \mathbb{E}_q[h_{kn}^{-1}])$
- 7: $q(h_{kn}) = \text{GIG}(a_0^h, \sum_m \omega_{nm}^{-1} \mathbb{E}_q[\theta_k] \mathbb{E}_q[w_{km}],$
 $\sum_m x_{nm} \lambda_{knm}^2 \mathbb{E}_q[\theta_k^{-1}] \mathbb{E}_q[w_{km}^{-1}])$
- 11: **end while**
- 12: **Return** 変分事後分布 $q(\theta), q(\mathbf{W}), q(\mathbf{H})$

$$+ \log p(\theta) + \log p(\mathbf{W}) + \log p(\mathbf{H}) \quad (3.102)$$

具体的には, 最適な変分事後分布 $q(\theta)$ は, θ に関連する項のみを取り出すと以下の通り計算できる .

$$\begin{aligned} \log q(\theta) &\stackrel{c}{=} \sum_{knm} x_{nm} \lambda_{knm} \log \theta_k \\ &\quad - \sum_{knm} \theta_k \mathbb{E}_q[w_{km}] \mathbb{E}_q[h_{kn}] \\ &\quad + \sum_k \left(\left(\frac{\alpha c}{K} - 1 \right) \log \theta_k - \alpha \theta_k \right) \end{aligned} \quad (3.103)$$

従って, θ_k の事後分布はガンマ分布となる . 同様に, 最適な $q(\mathbf{W})$ や $q(\mathbf{H})$ もガンマ分布として求まる . Algorithm 1 に更新則を示す . 反復ごとに, $\mathbb{E}[\theta_k]$ が十分に小さい基底 k を削除していくけば, 実効的な基底数 K_+ が自動的に定まる .

3.2.4 GaP-IS-NMF のベイズ推定

次に, GaP-IS-NMF に対する VB^[3] を導出する . 式 (??) で与えられる変分下限 $\mathcal{L}(q)$ の第一項は式 (5.6) で与えられる対数指數尤度の期待値であり ,

やはり解析的に計算できない。そこで、凹関数 $f(x) = -\frac{1}{x}$ に対して Jensen の不等式を考える。

$$-\frac{1}{\sum_{k=1}^K x_k} = -\frac{1}{\sum_{k=1}^K \lambda_k \frac{x_k}{\lambda_k}} \geq \sum_{k=1}^K \frac{\lambda_k^2}{x_k} \quad (3.104)$$

ここで、 λ_k は $\sum_k \lambda_k = 1$ を満たす補助変数であり、等号成立条件は $\lambda_k \propto x_k$ である。更に、凸関数 $g(x) = -\log(x)$ に対する 1 次のテイラー展開 (ω における接線) を考える。

$$-\log(x) \geq -\log(\omega) - \frac{x}{\omega} + 1 \quad (3.105)$$

ここで、 ω は補助変数であり、等号成立条件は $\omega = x$ である。これら二つの不等式を用いると変分下限 $\mathbb{E}_q[\log q(\mathbf{X}|\boldsymbol{\theta}, \mathbf{W}, \mathbf{H})]$ を得る。

$$\begin{aligned} & \mathbb{E}_q[\log p(\mathbf{X}|\boldsymbol{\theta}, \mathbf{W}, \mathbf{H})] \\ & \stackrel{c}{=} \mathbb{E}_q \left[\sum_{nm} \left(-x_{nm} (y_{nm})^{-1} - \log(y_{nm}) \right) \right] \\ & \geq - \sum_{nm} x_{nm} \mathbb{E}_q \left[\sum_k \frac{\lambda_{knm}^2}{y_{knm}} \right] \\ & \quad - \sum_{nm} \left(\log(\omega_{nm}) + \mathbb{E}_q \left[\frac{y_{nm}}{\omega_{nm}} \right] - 1 \right) \\ & \stackrel{\text{def}}{=} \mathbb{E}_q[\log q(\mathbf{X}|\boldsymbol{\theta}, \mathbf{W}, \mathbf{H})] \end{aligned} \quad (3.106)$$

各パラメータに対する変分事後分布は、3.2.3 節と同様に求められる。具体的には、最適な変分事後分布 $q(\boldsymbol{\theta})$ は、 $\boldsymbol{\theta}$ に関連する項のみを取り出すと

$$\begin{aligned} \log q(\boldsymbol{\theta}) & \stackrel{c}{=} - \sum_{knm} x_{nm} \lambda_{knm}^2 \theta_k^{-1} \mathbb{E}_q[w_{km}^{-1}] \mathbb{E}_q[h_{kn}^{-1}] \\ & \quad - \sum_{knm} \omega_{nm}^{-1} \theta_k \mathbb{E}_q[w_{km}] \mathbb{E}_q[h_{kn}] \\ & \quad + \sum_k \left(\left(\frac{\alpha c}{K} - 1 \right) \log \theta_k - \alpha \theta_k \right) \end{aligned} \quad (3.107)$$

従って、 θ_k の事後分布は Generalized Inverse Gaussian (GIG) 分布となることが分かる（詳細は^[3] 参照）。Algorithm 2 に更新則を示す。

3.2.5 BP-KL-NMF のベイズ推定

3.3 半正定値テンソル分解

このような無限次元の空間 ($K \rightarrow \infty$) におけるスパースな学習は、ノンパラメトリックベイズモデルを用いて実現することができる^[3]。近年、NMF の数学的に自然な拡張である半正定値テンソル分解 (Positive Semidefinite Tensor Factorization: PSDTF)^[4,5] が提案され、優れた音源分離結果を達成している。

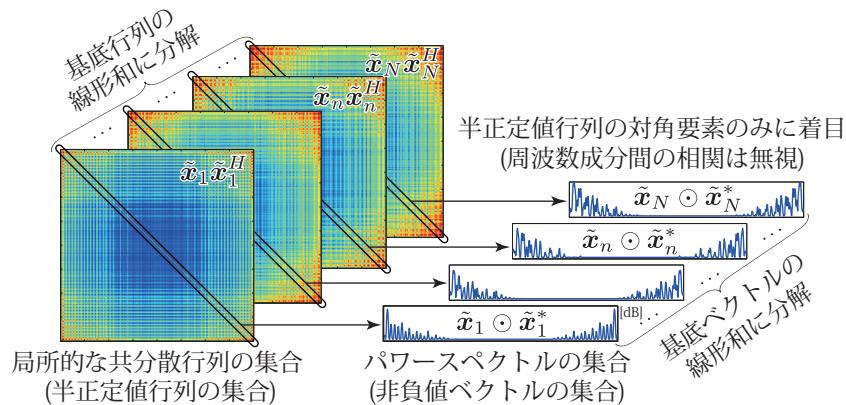
本章では、NMF の自然な拡張である半正定値テンソル分解^[4,5](PSD TF)について解説する。PSD TF では、各フレーム n の複素スペクトル \tilde{x}_n の自己共分散 $X_n = \tilde{x}_n \tilde{x}_n^H$ 、すなわち半正定値行列を少数の半正定値行列の和に分解する(図 3.5)。一方、NMF では、上記行列の対角成分(パワースペクトル) $x_n = \tilde{x}_n \odot \tilde{x}_n^*$ 、すなわち非負値ベクトルを少数の非負値ベクトルの和に分解する。行列の半正定値性はベクトルの非負値性の拡張概念であり、非負値テンソル分解 (Nonnegative Tensor Factorization: NTF) と PSD TF とは異なる。

音源分離においては、観測スペクトログラム \tilde{X} から、式(5.2)を満たす音源スペクトログラム \tilde{X}_k の位相を推定できる。音源信号の周期と短時間フーリエ変換の窓長 M が異なると、音源信号の巡回定常性の仮定が成り立たないため、周波数ビン間の相関を取り扱える利点は大きい。マルチチャネル音源分離において、マイク間の相關行列を分解する際にも同様のモデルが提案されている^[10]。

3.3.1 コスト関数最小化としての定式化

PSD TF では、観測データとして 3 階のテンソル $\mathbf{X} = [X_1, \dots, X_n] \in \mathbb{C}^{M \times M \times N}$ に対する分解を行う。各要素 $X_n \succeq \mathbf{0} \in \mathbb{C}^{M \times M}$ は半正定値行列とする。今、各 X_n を K 個の半正定値行列 $\{\mathbf{W}_k\}_{k=1}^K$ (基底行列) の凸結合で近似したい。

$$\mathbf{X}_n \approx \sum_{k=1}^K h_{kn} \mathbf{W}_k \stackrel{\text{def}}{=} \mathbf{Y}_n \quad (3.108)$$



ここで, $h_{kn} \geq 0$ は \mathbf{X}_n における基底行列 \mathbf{W}_k の重みである. 観測行列 \mathbf{X}_n と再構成行列 \mathbf{Y}_n との間の誤差 $\mathcal{D}(\mathbf{X}_n|\mathbf{Y}_n)$ を評価する尺度として, 非負値ベクトル間の KL ダイバージェンスや IS ダイバージェンスの拡張である, 半正定値行列間の von-Neumann (vN) ダイバージェンスや Log-Determinant (LD) ダイバージェンスがある^[11].

$$\begin{aligned} \mathcal{D}_{\text{vN}}(\mathbf{X}_n|\mathbf{Y}_n) &= \text{tr} (\mathbf{X}_n \log \mathbf{X}_n - \mathbf{X}_n \log \mathbf{Y}_n \\ &\quad - \mathbf{X}_n + \mathbf{Y}_n) \end{aligned} \quad (3.109)$$

$$\begin{aligned} \mathcal{D}_{\text{LD}}(\mathbf{X}_n|\mathbf{Y}_n) &= \text{tr} (\mathbf{X}_n \mathbf{Y}_n^{-1}) \\ &\quad - \log |\mathbf{X}_n \mathbf{Y}_n^{-1}| - M \end{aligned} \quad (3.110)$$

3.3.2 乗法更新アルゴリズムに基づく最適化

コスト関数 $\mathcal{D}(\mathbf{X}|\mathbf{Y}) = \sum_n \mathcal{D}(\mathbf{X}_n|\mathbf{Y}_n)$ を最小化する $\mathbf{H} = [h_1, \dots, h_K] \in \mathbb{R}^{N \times K}$ 及び $\mathbf{W} = [\mathbf{W}_1, \dots, \mathbf{W}_K] \in \mathbb{C}^{M \times M \times K}$ を求めるため, LD-PSDTF に対しても乗法更新アルゴリズム^[4,5]が提案されている. 更新則は Algorithm?3 で与えられる(導出は文献^[4,5]参照). h_{kn} の非負性と \mathbf{W}_k の半正定値性は自然に保たれているが, $\text{tr}(\mathbf{W}_k) = 1$ を満たすよう, 反復ごとに \mathbf{W}_k 及び h_k をスケーリングしておく.

3.4 確率的潜在成分解析

Probabilistic Latent Component Analysis

3.4.1 確率モデルの最尤推定としての定式化

3.4.2 ノンパラメトリックベイズモデル

3.4.3 音源分離への応用

前節の議論を踏まえて、式 (??), (??), (??), (??) で定義されるノンパラメトリックベイズ KL-NMF (GaP-KL-NMF) あるいは式 (??), (??), (??), (??) で定義されるノンパラメトリックベイズ IS-NMF (GaP-IS-NMF) に対する変分ベイズ法 (Variational Bayes: VB) について述べる。今、観測データ \mathbf{X} が与えられたときに、ベイズの定理を用いて未知パラメータ $\theta, \mathbf{W}, \mathbf{H}$ の事後分布

$$p(\theta, \mathbf{W}, \mathbf{H} | \mathbf{X}) = \frac{p(\mathbf{X}, \theta, \mathbf{W}, \mathbf{H})}{p(\mathbf{X})} \quad (3.111)$$

を計算したい。しかし、周辺尤度 $p(\mathbf{X})$ は解析的に計算できないため、対数周辺尤度の変分下限 $\mathcal{L}(q)$ を構成し、逐次最大化を行うことで $p(\mathbf{X})$ を近似することを考える。すなわち、変分分布 $q(\theta, \mathbf{W}, \mathbf{H})$ に対し、凹関数 $f(x) = \log(x)$ に対して Jensen の不等式を用いると以下を得る。

$$\begin{aligned} & \log p(\mathbf{X}) \\ &= \log \int q(\theta, \mathbf{W}, \mathbf{H}) \frac{p(\mathbf{X}, \theta, \mathbf{W}, \mathbf{H})}{q(\theta, \mathbf{W}, \mathbf{H})} d\theta d\mathbf{W} d\mathbf{H} \\ &\geq \int q(\theta, \mathbf{W}, \mathbf{H}) \log \frac{p(\mathbf{X}, \theta, \mathbf{W}, \mathbf{H})}{q(\theta, \mathbf{W}, \mathbf{H})} d\theta d\mathbf{W} d\mathbf{H} \\ &= \mathbb{E}_{q(\theta, \mathbf{W}, \mathbf{H})} [\log p(\mathbf{X}, \theta, \mathbf{W}, \mathbf{H})] \\ &\quad - \mathbb{E}_{q(\theta, \mathbf{W}, \mathbf{H})} [\log q(\theta, \mathbf{W}, \mathbf{H})] \\ &\stackrel{\text{def}}{=} \mathcal{L}(q) \end{aligned} \quad (3.112)$$

等号成立条件は $q(\theta, \mathbf{W}, \mathbf{H}) = p(\theta, \mathbf{W}, \mathbf{H} | \mathbf{X})$ であり、このとき $\mathcal{L}(q)$ が最

大値をとる。しかし、真の事後分布 $p(\theta, \mathbf{W}, \mathbf{H} | \mathbf{X})$ は計算困難であるため、变分事後分布を因子分解可能な形 $q(\theta, \mathbf{W}, \mathbf{H}) = q(\theta)q(\mathbf{W})q(\mathbf{H})$ に限定し、その中でも以下で計算できる变分下限

$$\begin{aligned}\mathcal{L}(q) &= \mathbb{E}_q[\log p(\mathbf{X} | \theta, \mathbf{W}, \mathbf{H})] + \mathbb{E}_q[\log p(\theta)] \\ &\quad + \mathbb{E}_q[\log p(\mathbf{W})] + \mathbb{E}_q[\log p(\mathbf{H})] \\ &\quad + H(q(\theta)) + H(q(\mathbf{W})) + H(q(\mathbf{H}))\end{aligned}\quad (3.113)$$

を最大化するものを求めたい。ここで、 $H(\cdot)$ はエントロピーを表す。これは、变分事後分布 $q(\theta)q(\mathbf{W})q(\mathbf{H})$ の真の事後分布 $p(\theta, \mathbf{W}, \mathbf{H} | \mathbf{X})$ に対する KL ダイバージェンスを最小化することと等価である。式 (3.113) を逐次最大化するには、以下の更新式を収束するまで繰り返せば良い。

$$q(\theta) \propto \exp(\mathbb{E}_{q(\mathbf{H}, \mathbf{W})}[\log p(\mathbf{X}, \theta, \mathbf{W}, \mathbf{H})]) \quad (3.114)$$

$$q(\mathbf{H}) \propto \exp(\mathbb{E}_{q(\theta, \mathbf{W})}[\log p(\mathbf{X}, \theta, \mathbf{W}, \mathbf{H})]) \quad (3.115)$$

$$q(\mathbf{W}) \propto \exp(\mathbb{E}_{q(\theta, \mathbf{H})}[\log p(\mathbf{X}, \theta, \mathbf{W}, \mathbf{H})]) \quad (3.116)$$

Chapt er 4

音声信号処理

[音声 .]

4.1 音声分析合成

我々人間は、喉頭と声帯を用いた発声と声道による調音をおおよそ独立に制御しながら声を発します。音声分析合成とは、音声生成過程を模擬したモデルを用いて音声信号の各短時間区間ににおける声帯の音源特性と声道の共振特性を推定し、音声認識、音声合成、音声変換、音声符号化などに役立てるための技術です。音声信号のみから声帯音源特性と声道共振特性を推定する問題は、 $XY = 10$ という式から X と Y を推定する問題と似ていて、何らかの仮定を置かない限り解が一意に定まりません。従って、声帯音源や声道に関してどのような仮定を置くかが問題解決のポイントになります。これらの仮定の置き方に応じてこれまで多くの手法が提案されていますが、中でも特に有名かつ基礎的なのが線形予測分析と呼ぶ手法で、音声音響信号処理の分野に多大な影響を与え、統計的手法による音声情報処理の枠組を生むきっかけの一つになった重要技術です。また、現在も携帯電話や VoIP の音声符号化圧縮方式の基礎技術として用いられています。そこで、4.1.1 ではまず線形予測分析の理論を解説します。

4.1.1 線形予測分析

以下、線形予測分析の目的とそれを実現するための最適化問題、確率モ

ルの最尤パラメータ推定問題としての解釈、パラメータ推定のためのアルゴリズム、音声の生成過程との関係性、周波数領域での解釈、などについて述べていきます。読み進めていくうちにいかに奥深くエレガントな理論であるかを分かっていただけるのではないかと思います。

4.1.1.1 問題の定式化

線形予測分析の目的自体は大変シンプルで、信号の現時刻の標本値を過去の標本値の線形結合でできるだけ良く近似できるように結合係数を決めることです。これが「線形予測」分析と呼ばれる所以です。音声信号のように新しい時刻間で強い相関があるような信号の場合、その相関を活かすことで信号を少ないパラメータで表現できるだろうという考え方がベースになっています。線形予測分析がなぜ音声の声帯の音源特性と声道の共振特性を推定する手法になっているのかについてはおいおい説明していくことにして、ここではまず以上の問題を具体的に定式化していくことにします。

音声信号は大域的に見れば非定常ですが、短い区間に区切ればそれぞれの区間では近似的に定常と見なせます。そこで、ある短区間の音声信号の標本値を x_1, x_2, \dots, x_N とし、この系列に対し定常性を仮定します。線形予測分析は、時刻 t_n の信号の標本値 x_n を時刻 t_n より過去の標本値 $x_{n-1}, x_{n-2}, \dots, x_{n-P}$ の線形結合で予測できるようにすることが目的で、

$$\mathcal{J}(\mathbf{a}) = \sum_n \left(x_n - \sum_{p=1}^P a_p x_{n-p} \right)^2 \quad (4.1)$$

を最小化する結合係数 $\mathbf{a} = (a_1, \dots, a_P)^\top$ を求める最適化問題として定式化されます。この結合係数を「予測係数」と呼びます。この最適化問題は、 $\mathbf{x} = (x_1, \dots, x_N)^\top$ を自己回帰過程

$$x_n = \sum_{p=1}^P a_p x_{n-p} + y_n \quad (4.2)$$

$$y_n \stackrel{iid}{\sim} \mathcal{N}(y_n; 0, \sigma^2) \quad (4.3)$$

から生成された観測値系列と仮定した場合の $\mathbf{a} = (a_1, \dots, a_P)^\top$ の最尤推定問題と等価になります^[?]。このことを以下で確認しましょう。ただし、 \mathcal{N}

は正規分布の確率密度関数

$$\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \Sigma) = \frac{1}{(2\pi)^{N/2} |\det(\Sigma)|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \Sigma^{-1} (\mathbf{x}-\boldsymbol{\mu})} \quad (4.4)$$

を表すものとします。式(??)は y_n が独立に同一（平均 0, 分散 σ^2 ）の正規分布に従うことを意味します。 y_n は $\sum_{p=1}^P a_p x_{n-p}$ による x_n の予測の誤差を表す確率変数なので、「予測誤差」と呼びます。 $\mathbf{y} = (y_1, \dots, y_T)^\top$ とし、

$$\Psi = \begin{bmatrix} 1 & & & & & 0 \\ -a_1 & \ddots & & & & \\ \vdots & \ddots & \ddots & & & \\ -a_P & & \ddots & \ddots & & \\ 0 & \ddots & & \ddots & \ddots & \\ & & -a_P & \cdots & -a_1 & 1 \end{bmatrix} \quad (4.5)$$

と置くと、式(??)は

$$\Psi \mathbf{x} = \mathbf{y} \quad (4.6)$$

のように書けます。 Ψ は対角成分がすべて 1 の下三角行列なので $\det(\Psi) = 1$ が言え、逆行列をもちます。従って、 $\mathbf{y} \sim \mathcal{N}(\mathbf{y}; \mathbf{0}, \sigma^2 \mathbf{I})$ および式(??)より、 \mathbf{x} は平均が 0, 分散共分散行列が $\sigma^2 \Psi^{-1} \Psi^{-\top}$ の正規分布に従います。

$$\mathbf{x} \sim \mathcal{N}(\mathbf{x}; \mathbf{0}, \sigma^2 \Psi^{-1} \Psi^{-\top}) \quad (4.7)$$

$\det(\Psi) = 1$ より、 $\mathbf{x} = (x_1, \dots, x_N)^\top$ が観測された下での a の対数尤度は

$$\log p(\mathbf{x}|a) = -\frac{N}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_n \left(x_n - \sum_{p=1}^P a_p x_{n-p} \right)^2 \quad (4.8)$$

となるので、 a によらない項を除けば式(??)の正負を逆転したものと等しくなります。以上よりたしかに式(??)を a に関して最小化することと式(??)を a に関して最大化することは等価であることが分かります。

$\mathcal{J}(a)$ を a_q に関して偏微分して 0 と置き、

$$\frac{\partial \mathcal{J}(a)}{\partial a_q} = 2 \left(\sum_p a_p \sum_n x_{n-p} x_{n-q} - \sum_n x_n x_{n-q} \right) = 0 \quad (4.9)$$

を $q = 1, \dots, P$ について連立させると、

$$\begin{bmatrix} r_{1,1} & r_{1,2} & \cdots & r_{1,P} \\ r_{2,1} & r_{2,2} & \cdots & r_{2,P} \\ \vdots & \vdots & & \vdots \\ r_{P,1} & r_{P,2} & \cdots & r_{P,P} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_P \end{bmatrix} = \begin{bmatrix} r_{0,1} \\ r_{0,2} \\ \vdots \\ r_{0,P} \end{bmatrix} \quad (4.10)$$

$$r_{q,p} = \sum_n x_{n-p} x_{n-q} \quad (4.11)$$

という形を得ます。よって $\mathcal{J}(a)$ を最小化する a は上式を解くことで得られます。ここで、 $\{x_n\}$ がエルゴード的である（集合平均と時間平均が等しい）ならば $r_{q,p}$ は x_n の自己相関関数 $\mathbb{E}[x_{n-p} x_{n-q}]$ となり、さらに x_n が弱定常である（自己相関が時間差のみに依存する）ならば、 $r_{q,p} = v_{|p-q|}$ と置くことができます。このとき、式 (??) は

$$\begin{bmatrix} v_0 & v_1 & \cdots & v_{P-1} \\ v_1 & v_0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & v_1 \\ v_{P-1} & \cdots & v_1 & v_0 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_P \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_P \end{bmatrix} \quad (4.12)$$

と書けます。この特殊な形の連立一次方程式を Yule-Walker 方程式といい、次章の方法により解を効率的に計算することができます。

4.1.2 Levinson-Durbin アルゴリズム

ここでは、式 (??) の Yule-Walker 方程式の解を計算するための効率的なアルゴリズムについて述べます。一般的な連立一次方程式の解法としては Gauss の消去法が知られていますが、式 (??) を Gauss の消去法を用いて解く場合、計算オーダーは $\mathcal{O}(P^3)$ になります。係数行列が実対称行列であることを活かせば Cholesky 分解を用いた方法、係数行列が Toeplitz 行列であることを活かせば Levinson-Durbin アルゴリズムと呼ぶ方法を使うことができます。この場合の計算オーダーはいずれも $\mathcal{O}(P^2)$ となります。一方、今解きたい方程式は、係数行列が Toeplitz 型であるとともに実対称でもある特殊なクラスに属しています。このようなクラスの連立一次方程式の解は、Levinson-Durbin アルゴリズムで $\mathcal{O}(P \log P)$ の計算オーダーで計算することができます。

式 (??) を満たす a を $\hat{a}^{(P)} = (\hat{a}_1^{(P)}, \dots, \hat{a}_P^{(P)})^\top$ とし、これを P 次の最適な予測係数と呼ぶことにします。Levinson-Durbin アルゴリズムでは、 m 次の最適な予測係数 $\hat{a}^{(m)}$ と $m+1$ 次の最適な予測係数 $\hat{a}^{(m+1)}$ の間の関係式を用いて $\hat{a}^{(1)}$ から $\hat{a}^{(P)}$ を再帰的に解いていくことが基本方針となります。

まず、 m 次の最適な予測係数 $\hat{a}^{(m)}$ が満たすべき Yule-Walker 方程式（式 (??) において $P = m$ としたもの）を考えます。式 (??) において右辺を左辺に移項し、

$$\sigma_m^2 = v_0 - \sum_{p=1}^m \hat{a}_p^{(m)} v_p \quad (4.13)$$

と置けば、式 (??) を

$$\begin{bmatrix} v_0 & v_1 & \cdots & v_m \\ v_1 & v_0 & \cdots & v_{m-1} \\ \vdots & \vdots & \ddots & \vdots \\ v_m & v_{m-1} & \cdots & v_0 \end{bmatrix} \begin{bmatrix} 1 \\ -\hat{a}_1^{(m)} \\ \vdots \\ -\hat{a}_q^{(m)} \end{bmatrix} = \begin{bmatrix} \sigma_m^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (4.14)$$

と書き換えることができます。同様に、 $m+1$ 次の最適な予測係数は

$$\begin{bmatrix} v_0 & v_1 & \cdots & v_{m+1} \\ v_1 & v_0 & \cdots & v_m \\ \vdots & \vdots & \ddots & \vdots \\ v_{m+1} & v_m & \cdots & v_0 \end{bmatrix} \begin{bmatrix} 1 \\ -\hat{a}_1^{(m+1)} \\ \vdots \\ -\hat{a}_{m+1}^{(m+1)} \end{bmatrix} = \begin{bmatrix} \sigma_{m+1}^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (4.15)$$

を満たします。ここで、係数行列の構造を活かしながら、式 (??) を式 (??) と同じ形とサイズになるように等価変形し、 $\hat{a}^{(m)} = (\hat{a}_1^{(m)}, \dots, \hat{a}_m^{(m)})^\top$ と $\hat{a}^{(m+1)} = (\hat{a}_1^{(m+1)}, \dots, \hat{a}_{m+1}^{(m+1)})^\top$ の関係を導きます。まず、式 (??) の左辺の行列の $m+2$ 列目に $(v_{m+1}, v_m, \dots, v_1)^\top$ という列を追加し、この列による影響をなくすためベクトル $(1, -\hat{a}_1^{(m)}, \dots, -\hat{a}_m^{(m)})^\top$ の第 $m+2$ 要素に 0 を追加することで、

$$\left[\begin{array}{ccccc|c} v_0 & v_1 & \cdots & v_q & v_{m+1} \\ v_1 & v_0 & \cdots & v_{m-1} & v_m \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ v_m & v_{m-1} & \cdots & v_0 & v_1 \end{array} \right] \left[\begin{array}{c} 1 \\ -\hat{a}_1^{(m)} \\ \vdots \\ -\hat{a}_m^{(m)} \\ 0 \end{array} \right] = \left[\begin{array}{c} \sigma_m^2 \\ 0 \\ \vdots \\ 0 \end{array} \right] \quad (4.16)$$

と書き換えることができます。次に、式(??)の左辺の行列の $m+2$ 行目に $(v_{m+1}, v_m, \dots, v_1, v_0)$ という行を追加し、右辺のベクトルの第 $m+2$ 要素に

$$w_m = v_{m+1} - \sum_{p=1}^m \hat{a}_p^{(m)} v_{m-p+1} \quad (4.17)$$

を追加することで、

$$\left[\begin{array}{ccccc|c} v_0 & v_1 & \cdots & v_m & v_{m+1} \\ v_1 & v_0 & \cdots & v_{m-1} & v_m \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ v_m & v_{m-1} & \cdots & v_0 & v_1 \\ v_{m+1} & v_m & \cdots & v_1 & v_0 \end{array} \right] \left[\begin{array}{c} 1 \\ -\hat{a}_1^{(m)} \\ \vdots \\ -\hat{a}_m^{(m)} \\ 0 \end{array} \right] = \left[\begin{array}{c} \sigma_m^2 \\ 0 \\ \vdots \\ 0 \\ w_m \end{array} \right] \quad (4.18)$$

と書くことができます。これで式(??)の左辺の行列を式(??)の左辺の行列と同じ形とサイズにすることができました。ここで、左辺の行列が対称かつ Toeplitz 型になっているので、式(??)において左辺と右辺のベクトルの要素の順序を反転させたもの

$$\left[\begin{array}{ccccc|c} v_0 & v_1 & \cdots & v_m & v_{m+1} \\ v_1 & v_0 & \cdots & v_{m-1} & v_m \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ v_m & v_{m-1} & \cdots & v_0 & v_1 \\ v_{m+1} & v_m & \cdots & v_1 & v_0 \end{array} \right] \left[\begin{array}{c} 0 \\ -\hat{a}_m^{(m)} \\ \vdots \\ -\hat{a}_1^{(m)} \\ 1 \end{array} \right] = \left[\begin{array}{c} w_m \\ 0 \\ \vdots \\ 0 \\ \sigma_m^2 \end{array} \right] \quad (4.19)$$

も同様に成立します。式(??)と式(??)の左辺の行列は等しいので、式(??)から式(??)に任意の値 K_m を乗じたものを引くことで、

$$\begin{bmatrix} v_0 & v_1 & \cdots & v_m & v_{m+1} \\ v_1 & v_0 & \cdots & v_{m-1} & v_m \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ v_m & v_{m-1} & \cdots & v_0 & v_1 \\ v_{m+1} & v_m & \cdots & v_1 & v_0 \end{bmatrix} \begin{bmatrix} 1 - 0 \\ -\hat{a}_1^{(m)} + K_m \hat{a}_m^{(m)} \\ \vdots \\ -\hat{a}_m^{(m)} + K_m \hat{a}_1^{(m)} \\ 0 - K_m \end{bmatrix} = \begin{bmatrix} \sigma_m^2 - K_m w_m \\ 0 \\ \vdots \\ 0 \\ w_m - K_m \sigma_m^2 \end{bmatrix} \quad (4.20)$$

という形を得ます。さてここで、式(??)と式(??)を比較してみましょう。
 K_m は任意に選んで良いので、式(??)の右辺の第 $m+2$ 要素が式(??)と同様に0になるように

$$K_m = \frac{w_m}{\sigma_m^2} \quad (4.21)$$

と選べば、式(??)を式(??)と同じ形にすることができます。よって、そのときの二式を比較することで、

$$\begin{aligned} \sigma_{m+1}^2 &= \sigma_m^2 - K_m w_m \\ \hat{a}_p^{(m+1)} &= \hat{a}_p^{(m)} - K_m \hat{a}_{m-p+1}^{(m)} \quad (p = 1, \dots, m) \\ \hat{a}_{m+1}^{(m+1)} &= K_m \end{aligned} \quad (4.22)$$

という関係式を得ることができます。式(??)は m 次の最適な予測係数から $m+1$ 次の最適な予測係数を導く再帰式となっているため、Levinson-Durbin再帰式と言います。1次の最適な予測係数 $\hat{a}_1^{(1)}$ は式(??)より

$$\hat{a}_1^{(1)} = \frac{v_1}{v_0} \quad (4.23)$$

と容易に求まり、式(??)と式(??)より σ_1^2 と w_1 はそれぞれ

$$\sigma_1^2 = v_0 - \hat{a}_1^{(1)} v_1 \quad (4.24)$$

$$w_1 = v_2 - \hat{a}_1^{(1)} v_1 \quad (4.25)$$

となるので、これらを初期値として上述の再帰式を用いて $\hat{a}^{(2)} = (\hat{a}_1^{(2)}, \hat{a}_2^{(2)})^\top$, $\hat{a}^{(3)} = (\hat{a}_1^{(3)}, \hat{a}_2^{(3)}, \hat{a}_3^{(3)})^\top$, ..., $\hat{a}^{(P)} = (\hat{a}_1^{(P)}, \dots, \hat{a}_P^{(P)})^\top$ を再帰的に求めていくことができます。以上が Levinson-Durbin アルゴリズムです。

ところで, Levinson-Durbin 再帰式の導出の過程で導かれた式(??)の K_m は偏自己相関係数 (Partial Correlation Coefficients; PARCOR) と呼ばれていて, 実は物理的に重要な意味をもちます. このことは??節で詳しく述べることとします.

以上で求めた P 次の最適な予測係数 \hat{a} を Ψ に代入すれば, 式(??)より x から予測誤差系列 y を得ることができます. 逆に, 予測係数 \hat{a} と予測誤差系列 y のペアから,

$$x_1 = y_1, \quad x_2 = y_2 + \hat{a}_1 x_1, \quad x_3 = y_3 + \hat{a}_1 x_2 + \hat{a}_2 x_1, \quad \dots \quad (4.26)$$

のように x の要素を逐次的に復元することができます.

4.1.3 声道スペクトル推定としての解釈

音声信号は, 声帯によって発せられ声道による共振を受けた音波が, 口唇から放射されたものです. 声道を線形フィルタと見なせば音声信号は声帯音源信号を入力した線形系の応答と見なせます(図??). このような音声信号の生成過程のモデル化はソースフィルタ理論と呼ばれています[?, ?]. 予測誤差 $\{y_n\}$ を声帯音源信号, 予測係数 $a = (a_1, \dots, a_P)^\top$ を声道のフィルタ特性(声道スペクトル)を特徴付けるパラメータと見なせば, 線形予測分析はソースフィルタ理論の解釈の下で周波数領域において音声の声道スペクトル推定を行っていることに相当します. 以下ではこのことを示します.

音声信号 $\{x_n\}$ と予測誤差系列 $\{y_n\}$ の関係は

$$y_n = x_n - \sum_{p=1}^P a_p x_{n-p} \quad (4.27)$$

で与えられます. $X(z)$ と $Y(z)$ をそれぞれ $\{x_n\}$ と $\{y_n\}$ の z 変換(?章参照)とすると,

$$Y(z) = A(z)X(z) \quad (4.28)$$

$$A(z) = 1 - a_1 z^{-1} - a_2 z^{-2} \cdots - a_P z^{-P} \quad (4.29)$$

となるため, 音声信号を入力として予測誤差系列を出力とする系は $A(z)$ を伝達関数とする線形時不变系で表されます. 逆に, 予測誤差系列 $\{y_n\}$ を入力として音声信号 $\{x_n\}$ を出力とする系の伝達関数は $A(z)$ の逆数

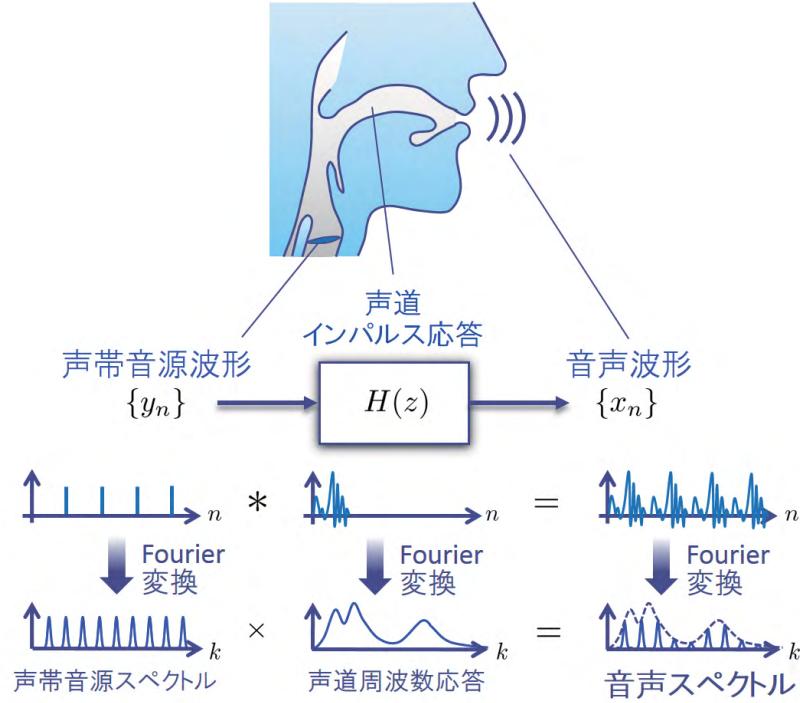


図 4.1 ソースフィルタ理論による音声生成過程モデル

$$H(z) = \frac{1}{A(z)} \quad (4.30)$$

となります。この伝達関数は零をもたず極のみをもちますが、? 章で述べたとおりこのように極のみからなるシステムを全極システムといいます。すなわち、もし音声信号の生成過程が線形時不变系で表せて、予測誤差系列 $\{y_n\}$ を声帯音源信号と見なすことができれば、声道の共振特性を全極型の伝達関数で表現することになります。 $k = 1, \dots, N$ を周波数インデックスとすると、 z 変換において $z = e^{2\pi j(k-1)/N}$ としたものは離散 Fourier 変換と一致します。従って、 $H(e^{2\pi j(k-1)/N})$ は全極システムの周波数応答となります。また、 $|H(e^{2\pi j(k-1)/N})|^2$ を全極スペクトルと呼びます。実は声道フィルタと

して全極システムを仮定することは声道を無損失系の等長音響管でモデル化していることに相当します。このことは??節で詳しく述べることにして、ここでは線形予測分析が周波数領域で何を行っていることに相当するのかを明らかにします。

??節では線形予測分析は時間領域では式(??)を尤度関数とした予測係数 a の最尤推定問題として定式化されることを示しました。ここでは、 x の離散 Fourier 変換の尤度関数を導きます。 F を各行に異なる周波数の複素正弦波が格納された離散 Fourier 変換行列

$$F = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & e^{2\pi j 1/N} & e^{2\pi j 2/N} & \cdots & e^{2\pi j (N-1)/N} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & e^{2\pi j (N-1)/N} & e^{2\pi j (N-1)2/N} & \cdots & e^{2\pi j (N-1)(N-1)/N} \end{bmatrix} \quad (4.31)$$

とすると、 x の離散 Fourier 変換は $\chi = Fx$ で与えられ、 $\chi = (\chi_1, \dots, \chi_N)^T$ の第 k 要素は周波数 k の成分を表します。 F はユニタリ行列で $|\det(F)| = 1$ なので、確率密度関数の変数変換により、 χ は平均が 0、分散共分散行列が $\sigma^2 F \Psi^{-1} \Psi^{-T} F^H$ の複素正規分布に従います。

$$\chi \sim \mathcal{N}_{\mathbb{C}}(\chi; 0, \sigma^2 F \Psi^{-1} \Psi^{-T} F^H) \quad (4.32)$$

ただし、 $\mathcal{N}_{\mathbb{C}}(x; \mu, \Sigma) = \frac{1}{\pi^N |\det(\Sigma)|} e^{-(x-\mu)^H \Sigma^{-1} (x-\mu)}$ です。ここで、分析窓の両端点において信号が巡回していると仮定し、式(?)の代わりに Ψ を

$$\Psi = \begin{bmatrix} 1 & -a_P & \cdots & -a_1 \\ -a_1 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & -a_P \\ -a_P & \ddots & \ddots & \ddots \\ 0 & -a_P & \cdots & -a_1 & 1 \end{bmatrix} \quad (4.33)$$

のような巡回行列とします。巡回行列同士の積は巡回行列になり、また、巡回行列は離散 Fourier 変換行列により対角化されるため、

$$\begin{aligned} \sigma^2 F \Psi^{-1} \Psi^{-T} F^H &= \sigma^2 (F \Psi^T \Psi F^H)^{-1} \\ &= \text{diag}(\lambda_1, \dots, \lambda_T) \end{aligned} \quad (4.34)$$

となります。ただし、 λ_k は $\sigma^2 \Psi^{-1} \Psi^{-T}$ の固有値

$$\lambda_k = \frac{\sigma^2}{|A(e^{2\pi j(k-1)/N})|^2} \quad (4.35)$$

$$A(z) = 1 - a_1 z^{-1} - \cdots - a_P z^{-P} \quad (4.36)$$

で与えられ、周波数 k における全極スペクトルを表します。式 (??) および式 (??) より、所与の χ の下での a の対数尤度は

$$\log p(\chi | a) = - \sum_k \left(\log \pi \lambda_k + \frac{|\chi_k|^2}{\lambda_k} \right) \quad (4.37)$$

となり、もし λ_k に何も制約がなく k ごとに自由度をもつならば

$$\frac{\partial \log p(\chi | a)}{\partial \lambda_k} = -\frac{1}{\lambda_k} + \frac{|\chi_k|^2}{\lambda_k^2} = 0 \Rightarrow \lambda_k = |\chi_k|^2 \quad (4.38)$$

より、 $\lambda_k = |\chi_k|^2$ のときに最大になります。よって、式 (??) に $\lambda_k = |\chi_k|^2$ を代入したものから式 (??) を引いたもの

$$\begin{aligned} D_{IS}(\lambda_k || |\chi_k|^2) &= - \sum_k (\log \pi |\chi_k|^2 + 1) + \sum_k \left(\log \pi \lambda_k + \frac{|\chi_k|^2}{\lambda_k} \right) \\ &= \sum_k \left(\frac{|\chi_k|^2}{\lambda_k} - \log \frac{|\chi_k|^2}{\lambda_k} - 1 \right) \\ &\geq 0 \end{aligned} \quad (4.39)$$

は信号のパワースペクトル $|\chi_k|^2$ と全極スペクトル λ_k の離れ具合を表す非負の尺度となります。これを板倉齋藤距離と呼びます^[?]。図??を見ても分かるように、板倉齋藤距離は^{*1} 非対称で、 λ_k が $|\chi_k|^2$ を下回る場合により過大なペナルティを課す誤差関数です。このため式 (??) は、 λ_k が $|\chi_k|^2$ をできるだけ下回らず $|\chi_k|^2$ のピークの近くを通るような関数となっているときほど小さい値になります。通常、線形予測分析により音声分析を行う際、次数 P は 10 から 20 程度に設定することが多いですが、この場合、線形予測分析では音声のスペクトルの包絡線（スペクトル包絡）を推定していると見なすことができます。実際、図??を見ると、たしかに全極スペクトルが音声

^{*1} よって、板倉齋藤距離は距離の公理を満たさないので厳密には距離ではありませんが、伝統的にこのように呼ばれています。板倉齋藤歪みや板倉齋藤ダイバージェンスという呼称も見られます。

スペクトルのピークの近くを通るように推定されていることが分かります。

さて、式(??)を最大化する a が時間領域で求めた最尤解と同じになることを確認しておきましょう。証明は例えば^[?]に譲りますが、 $N \gg P$ においては Ψ が式(??)のような巡回行列で与えられる場合でも $\det(\Psi) \simeq 1$ となることが示せるので、式(??)の左辺の行列式は $\det(\sigma^2 F \Psi^{-1} \Psi^{-T} F^H) = \sigma^{2N} \det(F)^2 \det(\Psi)^{-2} \simeq \sigma^{2N}$ と近似できます。また、右辺の行列式は $\prod_k \lambda_k$ となるので $\sum_k \log \lambda_k = N \log \sigma^2$ が言えます。従って、式(??)は

$$\log p(\chi|a) = -N \log(\pi\sigma^2) - \sum_k \frac{|\chi_k|^2}{\sigma^2} \left| 1 - \sum_p a_p e^{-2\pi j p(k-1)/N} \right|^2 \quad (4.40)$$

と書けます。対数尤度 $\log p(\chi|a)$ の a_q に関する偏微分

$$\frac{\partial \log p(\chi|a)}{\partial a_q} = \sum_k \frac{|\chi_k|^2}{\sigma^2} 2\operatorname{Re} \left[e^{2\pi j q(k-1)/N} - \sum_p a_p e^{2\pi j (q-p)(k-1)/N} \right]$$

を 0 と置いた式

$$\sum_p a_p \operatorname{Re} \left[\sum_k |\chi_k|^2 e^{2\pi j (q-p)(k-1)/N} \right] = \operatorname{Re} \left[\sum_k |\chi_k|^2 e^{2\pi j q(k-1)/N} \right]$$

を $q = 1, \dots, P$ について連立させることで、最尤の a が満たすべき方程式

$$\begin{bmatrix} v_0 & v_{-1} & \cdots & v_{1-P} \\ v_1 & v_0 & \cdots & v_{2-P} \\ \vdots & \vdots & \ddots & \vdots \\ v_{P-1} & v_{P-2} & \cdots & v_0 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_P \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_P \end{bmatrix} \quad (4.41)$$

$$v_q = \operatorname{Re} \left[\sum_k |\chi_k|^2 e^{2\pi j q(k-1)/N} \right] \quad (4.42)$$

が得られます。ここで、式(??)は音声信号 $\{x_n\}$ のパワースペクトル $\{|\chi_k|^2\}$ の逆 Fourier 変換となっているため、 $\{v_q\}$ は式(??)における $\{v_q\}$ と同様、 $\{x_n\}$ の自己相関関数を表す変数となっています。また、 $\{|\chi_k|^2\}$ は実関数なので、 $\{v_q\}$ は偶関数 ($v_q = v_{-q}$) となります。よって、式(??)は式(??)と同じ形の Yule-Walker 方程式に帰着します。

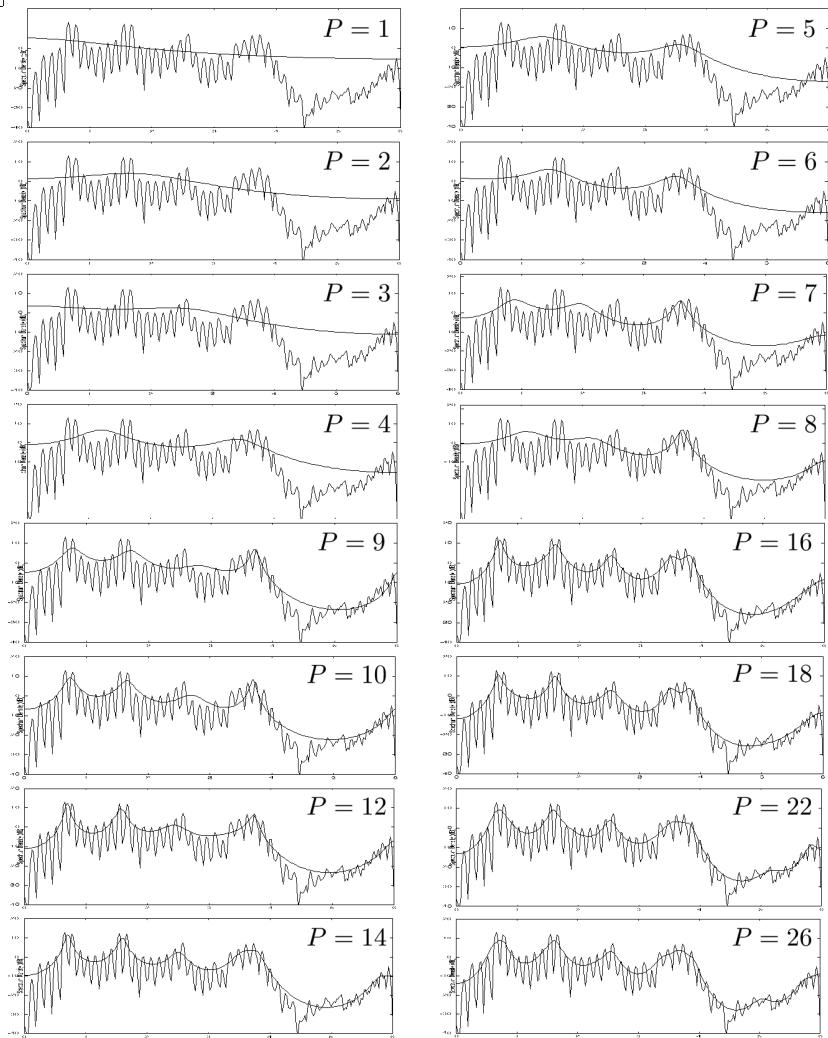


図 4.2 音声スペクトルと次数 P の線形予測分析により推定された全極スペクトル ($P = 1, \dots, 8, 9, 10, 12, 14, 16, 18, 22, 26$)

4.1.4 無損失系等長音響管による声道モデルとしての解釈

本節では全極システムを仮定することが無損失系等長音響管により声道をモデル化していることに相当することを示します。そこでまず、一定の断面積の管の中を進行する音波の波動方程式を導き、これをもとに、断面積が不連続に変化する管内を音波がどう伝播していくかを考えます。

音波は空気などの媒質の中を伝播する粗密波で、平均圧力を基準として音波によって引き起こされる圧力の変化分を音圧といいます。空間中のある断面に圧力変動が伝わると、その断面上で、媒質を構成する微小粒子が平衡位置から変位させられます。このときの粒子の速度を粒子速度といいます。音波は音圧と粒子速度の二つの量によって特徴付けられ、これらの量を支配する方程式を波動方程式と呼びます。ここでは、断面積が S で d 軸方向に伸びている管を考え、この管と d と $d + \Delta d$ の二つの面で囲まれた微小部分（図 ??）における音圧と粒子速度の関係を導くため、媒質の運動量の保存を表す運動方程式と質量の保存を表す連続の方程式を導入します。

まず運動方程式を導きます。時刻 t における d での音圧を $P(t, d)$ とすれば $d + \Delta d$ での音圧は $P(t, d) + \frac{\partial P(t, d)}{\partial d} \Delta d$ となります。ゆえに、この微小部分には左右逆向きの力 $P(t, d)S$ と $(P(t, d) + \frac{\partial P(t, d)}{\partial d} \Delta d)S$ が加わるので、その差の力 $-\frac{\partial P(t, d)}{\partial d} \Delta d S$ により微小部分の粒子が動かされます。管内の媒質の密度を ρ とすると微小部分の質量は $\rho \Delta d S$ になり、さらに粒子の変位を $\zeta(t, d)$ とすると粒子の加速度は $\frac{\partial^2 \zeta(t, d)}{\partial t^2}$ になるので、

$$-\frac{\partial P(t, d)}{\partial d} \Delta d S = \rho \Delta d S \frac{\partial^2 \zeta(t, d)}{\partial t^2} \quad (4.43)$$

という運動方程式を得ます。この式に粒子速度 $v(t, d) = \frac{\partial \zeta(t, d)}{\partial t}$ を代入した

$$-\frac{\partial P(t, d)}{\partial d} = \rho \frac{\partial v(t, d)}{\partial t} \quad (4.44)$$

が音圧と粒子速度の関係を表す一つ目の式となります。

次に連続の方程式を導きます。上記運動方程式に従って d の位置にあった粒子が $\zeta(t, d)$ だけ変位したとすると、 $d + \Delta d$ にあった粒子の変位は $\zeta(t, d) + \frac{\partial \zeta(t, d)}{\partial d} \Delta d$ となります。よって、微小部分の体積 $\Delta d S$ は変位の差と断面積の積 $\frac{\partial \zeta(t, d)}{\partial d} \Delta d S$ だけ膨張します。媒質の体積減少の割合 $(\frac{\partial \zeta(t, d)}{\partial d} \Delta d S) / (\Delta d S)$ は加えられた圧力 $P(t, d)$ に比例するので、比例定数

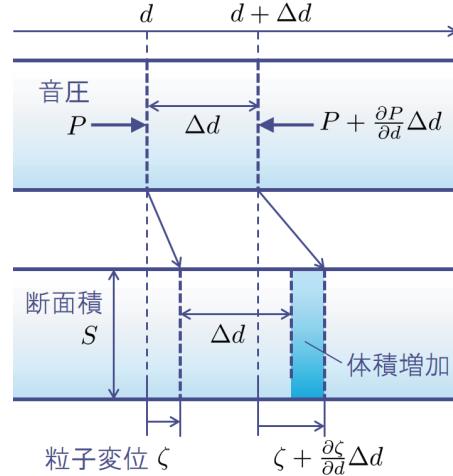


図 4.3 ***

を K とすると

$$P(t, d) = -K \frac{\partial \zeta(t, d)}{\partial d} \quad (4.45)$$

という関係式が得られます。 K は体積弾性率と呼ばれ、媒質によって決まる値です。この式の両辺を時間 t で微分し、粒子速度 $v(t, d) = \frac{\partial \zeta(t, d)}{\partial t}$ を代入すると

$$-\frac{\partial P(t, d)}{\partial t} = K \frac{\partial v(t, d)}{\partial d} \quad (4.46)$$

という式を得ます。これが連続の方程式で、音圧と粒子速度の関係を表す二つの式となります。

これらの二つの関係式を同時に記述したものが波動方程式です。そこで、両式を統合化することを考えます。式 (??) の両辺を d で微分した式と式 (??) の両辺を t で微分した式を用いれば、 v を消去し P だけの式に書き換えることができます。逆に、式 (??) の両辺を t で微分した式と式 (??) の両辺を d で微分した式を用いることで v だけの式に書き換えることができます。この操作により P と v に関する波動方程式

$$\frac{\partial^2 P(t, d)}{\partial d^2} = \frac{1}{c^2} \frac{\partial^2 P(t, d)}{\partial t^2} \quad (4.47)$$

$$\frac{\partial^2 v(t, d)}{\partial d^2} = \frac{1}{c^2} \frac{\partial^2 v(t, d)}{\partial t^2} \quad (4.48)$$

$$c = \sqrt{\frac{K}{\rho}} \quad (4.49)$$

を得ます。これらの式は P と v の振る舞いを別々に表現したもので、両者の関係を記述した形にはなっていません。そこで、 P と v の関係を記述しやすくする目的で

$$v(t, d) = -\frac{\partial \phi(t, d)}{\partial d} \quad (4.50)$$

を満たす速度ポテンシャルと呼ぶ関数 $\phi(t, d)$ を導入します。式 (??) を運動方程式 (式 (??)) に代入し、 d に関して積分すると

$$P(t, d) = \rho \frac{\partial \phi(t, d)}{\partial t} + C \quad (4.51)$$

を得ます。ここで、 C は積分定数ですが、音波が伝播していない $\phi = 0$ のとき音圧が $P = 0$ となるように $C = 0$ と設定することで

$$P(t, d) = \rho \frac{\partial \phi(t, d)}{\partial t} \quad (4.52)$$

を得ます。式 (??), (??) より、速度ポテンシャル $\phi(t, d)$ は P と v を結びつける関数となっていることが分かります。あとは式 (??), (??) を連続の方程式 (式 (??)) に代入することで、 P と v に関する波動方程式を速度ポテンシャル $\phi(t, d)$ を介して

$$\frac{\partial^2 \phi(t, d)}{\partial d^2} = \frac{1}{c^2} \frac{\partial^2 \phi(t, d)}{\partial t^2} \quad (4.53)$$

のよう一つの方程式で記述することができます。***の場合の式 (??) の解は

$$\phi(t, d) = A e^{j\omega(t-d/c)} + B e^{j\omega(t+d/c)} \quad (4.54)$$

以上より、一定の断面積の管内を一方向に伝播する音波の波動方程式は断面積 S に依存しないことが分かりましたが、以下で断面積が不連続変化する音響管を伝播する音波の振る舞いを考えるために、粒子速度 $v(t, d)$ の代わりに

図 4.4 ***

断面積 S に依存する体積速度 $u(t, d) = Sv(t, d)$ という量を扱います。

今、図??のように長さが等しく断面積が異なる M 個の管を連結した非一様音響管を考え、この管内を長さ方向にのみ伝播する音波を考えます。 m 番目の区間の管の断面積を S_m とすると、この管内を伝播する音波の速度ポテンシャル $\phi_m(t, d)$ は

$$\frac{\partial^2 \phi_m(t, d)}{\partial d^2} = \frac{1}{c^2} \frac{\partial^2 \phi_m(t, d)}{\partial t^2} \quad (4.55)$$

を満たし、一般解は

$$\phi_m(t, d) = Ae^{j\omega(t-d/c)} + Be^{j\omega(t+d/c)} \quad (4.56)$$

で与えられます。体積速度 $u_m(t, d)$ と速度ポテンシャル $\phi_m(t, d)$ の関係は

$$u_m(t, d) = -S_m \frac{\partial \phi_m(t, d)}{\partial d} \quad (4.57)$$

で与えられ、音圧 $P_m(t, d)$ と速度ポテンシャル $\phi_m(t, d)$ の関係は式 (??) で与えられるので、

$$u_m(t, d) = u_m^+(t, d) - u_m^-(t, d) \quad (4.58)$$

$$P_m(t, d) = \frac{\rho c}{S_m} (u_m^+(t, d) + u_m^-(t, d)) \quad (4.59)$$

となります。ただし、 $u_m^+(t, d), u_m^-(t, d)$ は

$$u_m^+(t, d) = \frac{j\omega S_m A}{c} e^{j\omega(t-d/c)} \quad (4.60)$$

$$u_m^-(t, d) = \frac{j\omega S_m B}{c} e^{j\omega(t+d/c)} \quad (4.61)$$

で与えられます。

4.2 韵律分析合成

4.2.1 背景と目的

普段我々私たちは会話をするとき言葉を使って相手にメッセージを伝えま

すが、言葉とともに声の高低を効果的に使いながら声に表情をつけ、声の調子や意図や、その人っぽさなどのさまざまな情報を相手に伝えています。

声の高低の時間変化を表す基本周波数パターンは大きく分けてフレーズ成分とアクセント成分と呼ぶ二つの成分からなります。フレーズ成分とは文や句の全体に及ぶ範囲で緩やかに変化する基本周波数パターンのこと、話者の調子や意図、文の区切りや係り受けを表現するのに重要な役割を担います。一方、アクセント成分とは各単語の中で急峻に変化する基本周波数パターンのこと、単語の意味や方言の違いに関係します。例えば「はし」という単語は「は」が高い場合と低い場合とでは意味が違いますし、「おいおい」という単語も先頭の「お」が高い場合と低い場合とで意味が違います。特に日本語の場合、単語アクセントと単語の意味の関係は方言によって異なるので、単語内の基本周波数パターンを変えて話せば異なった方言になります。また、フレーズ成分とアクセント成分の大きさは、文や句や単語の強弱を表すいわゆるメリハリに相当します。これらが大きい場合と小さい場合とでは話し方の印象は大きく変わり、メリハリをつけることで発話の中で注目すべき文や単語を相手に示すことができるようになります。以上のように、基本周波数パターンは様々さまざまな情報を持っており、言葉に勝るとも劣らないくらい音声コミュニケーションにおいて大きな役割を果たしています。基本周波数パターン分析とは、***。

基本周波数パターンは声帯を伸縮させる甲状腺骨という部位により制御されています。藤崎らによってこの制御メカニズムを模擬した物理モデルが提案されており(図??), 藤崎モデルと呼ばれています。甲状腺骨の運動がどのような基本周波数パターンをもたらすかを簡潔に説明した方程式で、日本語を含む多言語の音声の基本周波数パターンを極めて良く表現できることが知られています。このモデルでは、甲状腺骨の二つの独立な運動(並進と回転)に伴う声帯の長さの変化の合計と対数基本周波数の変化が比例関係にあり、それぞれの運動がイントネーションとアクセントに関与しているという仮定がベースとなっています。このモデルに基づき、音声から甲状腺骨がどう動いたかを推定することができれば、その声にそっくりの基本周波数パターンを再現することができ、さらにその数値を変えてやれば甲状腺骨が異なる動きをした場合の基本周波数パターンの音声を再合成することができるようになります。ところが、この逆問題は一筋縄ではいかず長らく未解決問

題とされていました。例えば $7+3$ を解くのは簡単でも $X+Y=10$ となる X と Y を一意に決められないのと同じで、フレーズ成分とアクセント成分から基本周波数パターンを得る方程式は与えられていたとしても基本周波数パターンのみからフレーズ成分とアクセント成分を一意に決めるることはできないからです。しかしまったく全く手がかりがないわけではありません。自然音声におけるフレーズやアクセントのタイミングや強度には統計的な偏りがあります。藤崎モデルの確率モデル化により統計的アプローチにより基本周波数パターンから藤崎モデルのパラメータを推定する（フレーズ成分とアクセント成分に分解する）手法が亀岡らによって提案されています。??節以降で、藤崎モデルについて概説した上で亀岡らの手法を紹介します。

4.2.2 骨格筋の弾性特性

藤崎モデルでは、

1. 声帯の伸びの合計と対数基本周波数の変化が比例関係にある
2. 対数基本周波数がフレーズ成分とアクセント成分とベースライン成分の三成分の和で表される

という二つの重要な仮定がベースとなっていました。これらの仮定の妥当性に対する藤崎による説明は以下のとおりです。

l を筋肉の長さとすると、声帯筋を含む骨格筋において、長さ方向に加わる張力 T と剛性 dT/dl （筋肉を単位長さ変化させるのに必要な力）の間には

$$\frac{dT}{dl} = a + bT \quad (4.62)$$

のような線形の関係が成り立つことが多くの実測結果により示されています [?, ?]。ただし、 a は $T = 0$ における筋肉の剛性です。声帯筋の静止張力を T_0 とし、そのときの声帯筋の長さを l_0 とすると、この微分方程式より張力 T と声帯筋の長さ l の関係を表す式

$$T = \left(T_0 + \frac{a}{b} \right) e^{b(l-l_0)} - \frac{a}{b} \quad (4.63)$$

が導かれます。ここで、 $T_0 \gg a/b$ のとき、式 (??) は

$$T = T_0 e^{b\delta} \quad (4.64)$$

と近似できます。ただし、 $\delta = l - l_0$ は張力が T_0 から T へ変化した際の声帯筋の長さの変化を表します。一方、一定の張力 T で張られた密度 ρ の弹性膜の固有振動周波数 f_0 は、膜の大きさによって決まる定数 c_0 を用いて

$$f_0 = c_0 \sqrt{T/\rho} \quad (4.65)$$

と表されます。従って、式(??)と式(??)より、声帯の伸びが δ のときの基本周波数の対数 $y = \log f_0$ は

$$y = \log c_0 \sqrt{T_0/\rho} + \frac{b}{2} \delta \quad (4.66)$$

となり、 δ と y が線形の関係にあることが示されます。声帯の伸び δ が時間変化する場合、 t を時刻とすると基本周波数パターン $y(t)$ は

$$y(t) = \log c_0 \sqrt{T_0/\rho} + \frac{b}{2} \delta(t) \quad (4.67)$$

のように固定項と時間変化成分の和として表されます。

声帯の長さは甲状軟骨の平行移動運動と回転運動によって変化します。平行移動運動によって生じる変化分を $\delta_p(t)$ 、回転運動によって生じる変化分を $\delta_a(t)$ とすると、これらの二つの運動が微小で互いに独立と見なせる範囲では声帯の長さの変化は $\delta_p(t)$ と $\delta_a(t)$ の和となります。なお、甲状軟骨の平行移動の時定数は、回転の時定数よりもはるかに大きいため、多くの言語に共通する特徴として $\delta_p(t)$ が句単位の比較的緩やかな音調の表現に、 $\delta_a(t)$ が語または音節単位の比較的急激で局所的な音調の表現に用いられています。

4.2.3 基本周波数パターン生成過程モデル [?]

藤崎モデルでは、甲状軟骨の二つの独立な運動（平行移動と回転）に伴う声帯の長さの変化の合計が F_0 の時間的变化をもたらすと解釈され、声帯の伸びと対数 F_0 の変化が比例関係にあるという仮定に基づき F_0 パターンがモデル化されます。

甲状軟骨の平行移動運動に関係する F_0 パターンをフレーズ成分、回転運動に関係する F_0 パターンをアクセント成分と呼び、それぞれ $y_p(t)$, $y_a(t)$ とします。ただし、 t は時刻です。 $y_p(t)$ の生成過程（フレーズ制御機構）はフレーズ指令と呼ぶパルス波を入力とした臨界制動の二次線形系、 $y_a(t)$ の生成過程（アクセント制御機構）はアクセント指令と呼ぶ矩形波を入力とした

臨界制動の二次線形系により表現されます。以上の二つの成分と、声帯の物理的性質によって決まるベースライン成分 y_b の和 $y_p(t) + y_a(t) + y_b$ で F_0 パターン $y(t)$ を表したものが藤崎モデルです。フレーズ成分は短区間の上昇のあとに緩やかな下降をなす成分で、アクセント成分は急激な上昇下降をなす成分であるため、多くの言語に共通して前者が句単位の比較的緩やかな音調を、後者が語または音節単位の比較的急激で局所的な音調を表現する役割を担っていると考えられています。フレーズ制御機構およびアクセント制御機構は

$$G_p(t) = \begin{cases} \alpha^2 t e^{-\alpha t} & (t \geq 0) \\ 0 & (t < 0) \end{cases} \quad (4.68)$$

$$G_a(t) = \begin{cases} \beta^2 t e^{-\beta t} & (t \geq 0) \\ 0 & (t < 0) \end{cases} \quad (4.69)$$

で与えられるインパルス応答 $G_p(t), G_a(t)$ によって特徴づけられます。 α と β はそれぞれの制御機構の固有角周波数であるが、これらは話者ごとにほぼ一定値をとること、話者の個人差も言語による差も比較的小さいことが確かめられており、およそ $\alpha = 3\text{rad/s}$, $\beta = 20\text{rad/s}$ 程度であることが経験的に知られています。これらを用いて F_0 パターン $y(t)$ は具体的に

$$\begin{aligned} y(t) = & y_b + \sum_i A_{p,i} G_p(t - T_{0,i}) \\ & + \sum_j A_{a,j} \{ S_a(t - T_{1,j}) - S_a(t - T_{2,j}) \} \end{aligned} \quad (4.70)$$

と表されます。ただし、 $A_{p,i}$ と $T_{0,i}$ はそれぞれ i 番目のフレーズ指令の大きさと生起時刻であり、 $A_{a,j}$ と $T_{1,j}$ 、 $T_{2,j}$ はそれぞれ j 番目のアクセント指令の振幅と始端時刻と終端時刻を表します。ところで、アクセント制御機構のインパルス応答 $G_a(t)$ は $S_a(t)$ の時間微分ゆえ

$$G_a(t) = \begin{cases} \beta^2 t e^{-\beta t} & (t \geq 0) \\ 0 & (t < 0) \end{cases} \quad (4.71)$$

となり、 $G_p(t)$ と同形であることが分かります。従って、アクセント成分

$y_a(t)$ は複数の矩形波からなるアクセント指令関数 $u_a(t)$ と $G_a(t)$ の畠み込みによって表されます。

本章では、藤崎モデルの離散時間表現を得るために、連続時間システムのフレーズ制御機構およびアクセント制御機構に対して後退差分変換により離散化を行います。まず、フレーズ制御機構の伝達関数 (Laplace 変換) は $\mathcal{G}_p(s) = \mathcal{L}[G_p(t)] = \alpha^2/(s + \alpha)^2$ で与えられます。後退差分変換は、時間微分演算子 s を z 領域における後退差分演算子 $s \simeq (1 - z^{-1})/t_0$ に置き換える変換であり (t_0 は変換先の離散時間表現におけるサンプリング周期)、この変換により、 $\mathcal{G}_p^{-1}(s)$ の逆システムの伝達関数を z 領域で

$$\mathcal{H}_p^{-1}(z) = a_2 z^{-2} + a_1 z^{-1} + a_0 \quad (4.72)$$

と書くことができます。ただし、

$$a_2 = (\psi - 1)^2, \quad a_1 = -2\psi(\psi - 1), \quad a_0 = \psi^2 \quad (4.73)$$

および、 $\psi = 1 + 1/(\alpha t_0)$ です。ここで、 k を離散時刻インデックスとし、フレーズ指令関数およびフレーズ成分の離散時間表現をそれぞれ $u_p[k]$ および $y_p[k]$ とすると、 $y_p[k]$ は、単一のパラメータ ψ (あるいは α) によって特性が決まる拘束つき全極モデルからの出力

$$u_p[k] = a_0 y_p[k] + a_1 y_p[k - 1] + a_2 y_p[k - 2] \quad (4.74)$$

と見なすことができます。同様に、アクセント指令関数 $u_a[k]$ とアクセント成分 $y_a[k]$ の関係も

$$u_a[k] = b_0 y_a[k] + b_1 y_a[k - 1] + b_2 y_a[k - 2] \quad (4.75)$$

と書くことができます。ただし、 $b_2 = (\varphi - 1)^2$, $b_1 = -2\varphi(\varphi - 1)$, $b_0 = \varphi^2$, $\varphi = 1 + 1/(\beta t_0)$ です。ベースライン成分 $y_b(t)$ の離散時間表現を $y_b[k]$ とすると、藤崎モデルの離散時間表現はこれらの三成分の和 $y[k] = y_p[k] + y_a[k] + y_b[k]$ で与えられます。

次に、 $u_p[k]$ と $u_a[k]$ をモデル化します。藤崎モデルにおいて、フレーズ指令とアクセント指令に関して以下のようないくつかの規則が定められています。

(A1) フレーズ指令はパルス波、アクセント指令は矩形波である。(A2) 発話の開始または先行フレーズ内のアクセント指令終了後にフレーズ指令が発

生する。また、フレーズ指令の後にアクセント開始指令が発生する。つまり、アクセント指令発生中はフレーズ指令は発生しない。(A3) アクセント指令の開始した後には必ずアクセント終了指令が発生する。つまり、隣接するアクセント指令同士は重なり合うことはない。

藤崎モデルのパラメータ推定における難しさの一つは、これらの制約の下で最適推定をいかにして行えるかという点にあろう。特に、上記の(A2), (A3) より、 $u_p[k]$ と $u_a[k]$ は相互に依存し合う関係がある点に注意が必要である。提案法では、これを解決するため隠れマルコフモデル(HMM)を用いて指令入力信号を確率モデル化する。まず、 $\mathbf{o}[k] := (u_p[k], u_a[k])^T$ を、

$$\mathbf{o}[k] \sim \mathcal{N}(\boldsymbol{\nu}[k], \boldsymbol{\Upsilon}) \quad (4.76)$$

$$\boldsymbol{\nu}[k] := \begin{bmatrix} \mu_p[k] \\ \mu_a[k] \end{bmatrix}, \quad \boldsymbol{\Upsilon} := \begin{bmatrix} \sigma_p^2 & 0 \\ 0 & \sigma_a^2 \end{bmatrix} \quad (4.77)$$

のように正規分布する確率変数と見なし、平均 $\boldsymbol{\nu}[k]$ が図??のような状態遷移に伴って変化するモデルを考える。これは HMM に他ならず、このように $\mathbf{o}[k]$ を HMM でモデル化したことにより、状態遷移の経路制限(状態遷移確率の設定)を通して $\boldsymbol{\nu}[k]$ に対して上記の(A1)~(A3)を満たすような制約を与えることが可能となる。提案する HMM の構成は以下のとおりである。

出力値系列: $\{\mathbf{o}[k]\}_{k=1}^K$
 状態集合: $S := \{p_0, p_1, a_0, \dots, a_N\}$
 状態系列: $\{s_k\}_{k=1}^K$
 状態出力分布: $P(\mathbf{o}[k]|s_k = i) = \mathcal{N}(\mathbf{c}_i[k], \boldsymbol{\Upsilon})$

$$\mathbf{c}_i[k] = \begin{cases} (0, 0)^T & (i = p_0, a_0) \\ (A_p[k], 0)^T & (i = p_1) \\ (0, A_a^{(n)})^T & (i = a_n) \end{cases}$$

状態遷移確率: $\phi_{i',i} := \log P(s_k = i | s_{k-1} = i')$

簡単のため状態遷移確率を定数とすると、以上より指令入力モデルにおいて推定すべきパラメータは、フレーズ指令の大きさ $A_p[k]$ 、状態遷移系列 s_k 、アクセント指令の大きさ $\{A_a^{(n)}\}_{n=1}^N$ 、指令入力信号の分散 σ_p^2, σ_a^2 であり、これらをまとめて θ_u と記す。また、平均値系列 $\{\mu_p[k]\}_{k=1}^K$ および $\{\mu_a[k]\}_{k=1}^K$

図 4.5 Command function modeling with HMM.

は、状態遷移系列 $\{s_k\}_{k=1}^K$ が与えられたもとで $(\mu_p[k], \mu_a[k])^T \leftarrow c_{s_k}[k]$ で与えられる。

前述の指令入力モデルに基づき $y = (y[1], \dots, y[K])^T$ の確率密度関数を導く。式 (??), (??) より, $u_p := (u_p[1], \dots, u_p[K])^T$, $u_a := (u_a[1], \dots, u_a[K])^T$, $\mu_p := (\mu_p[1], \dots, \mu_p[K])^T$, $\mu_a := (\mu_a[1], \dots, \mu_a[K])^T$ とすると,

$$u_p | \theta_u \sim \mathcal{N}(\mu_p, \Sigma_p), \quad \Sigma_p = \sigma_p^2 I \quad (4.78)$$

$$u_a | \theta_u \sim \mathcal{N}(\mu_a, \Sigma_a), \quad \Sigma_a = \sigma_a^2 I \quad (4.79)$$

が言える。??章で得た関係式より、フレーズ成分 $y_p := (y_p[1], \dots, y_p[K])^T$ と u_p の関係、および、アクセント成分 $y_a := (y_a[1], \dots, y_a[K])^T$ と u_a の関係は、

$$A := \begin{bmatrix} a_0 & O \\ a_1 a_0 & O \\ a_2 a_1 a_0 & O \\ \ddots \ddots \ddots & O \\ O & a_2 a_1 a_0 \end{bmatrix}, \quad B := \begin{bmatrix} b_0 & O \\ b_1 b_0 & O \\ b_2 b_1 a_0 & O \\ \ddots \ddots \ddots & O \\ O & b_2 b_1 b_0 \end{bmatrix} \quad (4.80)$$

と置くと、それぞれ $u_p = Ay_p$, $u_a = By_a$ のように表現できることから、式 (??), (??) より

$$y_p | \theta_u, \alpha \sim \mathcal{N}(A^{-1}\mu_p, A^{-1}\Sigma_p(A^{-1})^T) \quad (4.81)$$

$$y_a | \theta_u, \beta \sim \mathcal{N}(B^{-1}\mu_a, B^{-1}\Sigma_a(B^{-1})^T) \quad (4.82)$$

が導かれる。ベース成分 $y_b[k]$ についても、同様に白色 Gauss 性雑音 $\epsilon_b[k]$ に起因する確率変数 $y_b[k] = \mu_b + \epsilon_b[k]$ と仮定し、 $\epsilon_b[k] \sim \mathcal{N}(0, \sigma_b^2)$ とし、同様に、 $\epsilon_\xi[j]$ と $\epsilon_{\xi'}[j']$ は $(\xi, j) \neq (\xi', j')$ のとき独立とすると、

$$y_b | \mu_b \sim \mathcal{N}(\mu_b \mathbf{1}, \Sigma_b) \quad (4.83)$$

が言える。ただし、 $\Sigma_b = \sigma_b^2 I$ であり、 $\theta_b := \{\mu_b, \sigma_b^2\}$ と置く。仮定より、 y_p , y_a , y_b は独立なので、 $\Theta := \{\theta_u, \alpha, \beta, \theta_b\}$ が与えられたもとでの F_0 パターン $y = y_p + y_a + y_b$ の確率密度関数は、式 (??), (??) と式 (??) より

$$\mathbf{y}|\Theta \sim \mathcal{N}(\mathbf{A}^{-1}\boldsymbol{\mu}_p + \mathbf{B}^{-1}\boldsymbol{\mu}_a + \mu_b \mathbf{1}, \mathbf{A}^{-1}\boldsymbol{\Sigma}_p(\mathbf{A}^{-1})^T + \mathbf{B}^{-1}\boldsymbol{\Sigma}_a(\mathbf{B}^{-1})^T + \boldsymbol{\Sigma}_b) \quad (4.84)$$

で与えられる。以上より、

$$\begin{aligned} P(\mathbf{y}|\Theta) &= \frac{|\boldsymbol{\Sigma}|^{1/2}}{(2\pi)^{T/2}} \exp \left\{ -\frac{1}{2}(\mathbf{y} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{y} - \boldsymbol{\mu}) \right\} \\ \boldsymbol{\mu} &= \mathbf{A}^{-1}\boldsymbol{\mu}_p + \mathbf{B}^{-1}\boldsymbol{\mu}_a + \mu_b \mathbf{1} \\ \boldsymbol{\Sigma} &= \mathbf{A}^{-1}\boldsymbol{\Sigma}_p(\mathbf{A}^T)^{-1} + \mathbf{B}^{-1}\boldsymbol{\Sigma}_a(\mathbf{B}^T)^{-1} + \boldsymbol{\Sigma}_b \end{aligned} \quad (4.85)$$

が、 F_0 パターン \mathbf{y} が与えられたときの藤崎モデルパラメータ Θ の尤度関数である。

Θ の事前確率については、各要素は独立で、状態遷移系列 $\{s[k]\}_{k=1}^K$ と制御パラメータの ψ と φ 以外のパラメータは一様に分布すると仮定し、 $P(\Theta) \propto P(\psi)P(\varphi)P(s_1)\prod_{k=2}^K P(s_k|s_{k-1})$ とする。

$P(\Theta|\mathbf{y})$ を最大化する問題（式（??）に相当）は解析的に解くことはできないが、 $\mathbf{x} := (\mathbf{y}_p^T, \mathbf{y}_a^T, \mathbf{y}_b^T)^T$ を完全データと見なすことで EM アルゴリズムによる不完全データ問題に帰着できる。この場合、完全データの対数尤度は、先に見たとおり、

$$\log P(\mathbf{x}|\Theta) \stackrel{c}{=} \frac{1}{2} \log |\boldsymbol{\Lambda}^{-1}| - \frac{1}{2}(\mathbf{x} - \mathbf{m})^T \boldsymbol{\Lambda}^{-1}(\mathbf{x} - \mathbf{m})$$

$$\mathbf{x} := \begin{bmatrix} \mathbf{y}_p \\ \mathbf{y}_a \\ \mathbf{y}_b \end{bmatrix}, \quad \mathbf{m} := \begin{bmatrix} \mathbf{A}^{-1}\boldsymbol{\mu}_p \\ \mathbf{B}^{-1}\boldsymbol{\mu}_a \\ \mu_b \mathbf{1} \end{bmatrix} \quad (4.86)$$

$$\boldsymbol{\Lambda}^{-1} := \begin{bmatrix} \mathbf{A}^T \boldsymbol{\Sigma}_p^{-1} \mathbf{A} & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{B}^T \boldsymbol{\Sigma}_a^{-1} \mathbf{B} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \boldsymbol{\Sigma}_b^{-1} \end{bmatrix} \quad (4.87)$$

で与えられる。このとき、Q 関数 $Q(\Theta, \Theta')$ は、

$$\begin{aligned} Q(\Theta, \Theta') &\stackrel{c}{=} \frac{1}{2} \left[\log |\boldsymbol{\Lambda}^{-1}| - \text{tr}(\boldsymbol{\Lambda}^{-1} \mathbb{E}[\mathbf{x}\mathbf{x}^T|\mathbf{y}; \Theta']) \right. \\ &\quad \left. + 2\mathbf{m}^T \boldsymbol{\Lambda}^{-1} \mathbb{E}[\mathbf{x}|\mathbf{y}; \Theta'] - \mathbf{m}^T \boldsymbol{\Lambda}^{-1} \mathbf{m} \right] + \log P(\Theta) \end{aligned} \quad (4.88)$$

となる。ここで、 $\mathbf{y} = \mathbf{Hx}$ （ただし、 $\mathbf{H} = [\mathbf{I} \ \mathbf{I} \ \mathbf{I}]$ ）であるから、 $\mathbb{E}[\mathbf{x}|\mathbf{y}; \Theta]$ と

$\mathbb{E}[xx^T|y; \Theta]$ は、具体的に

$$\mathbb{E}[x|y; \Theta] = m + \Lambda H^T (H \Lambda H^T)^{-1} (y - Hm) \quad (4.89)$$

$$\begin{aligned} \mathbb{E}[xx^T|y; \Theta] &= \Lambda - \Lambda H^T (H \Lambda H^T)^{-1} H \Lambda \\ &\quad + \mathbb{E}[x|y; \Theta] \mathbb{E}[x|y; \Theta]^T \end{aligned} \quad (4.90)$$

と書ける。E ステップでは、直前のステップで更新されたモデルパラメータを Θ' に代入し、上記に基づいて $\mathbb{E}[x|y; \Theta']$ と $\mathbb{E}[xx^T|y; \Theta']$ が算出される。 y_p, y_a, y_b に対応するように $\mathbb{E}[x|y; \Theta']$ および $\mathbb{E}[xx^T|y; \Theta']$ を

$$\mathbb{E}[x|y; \Theta'] = \begin{bmatrix} \bar{x}_p \\ \bar{x}_a \\ \bar{x}_b \end{bmatrix}, \quad \mathbb{E}[xx^T|y; \Theta'] = \begin{bmatrix} R_p & * & * \\ * & R_a & * \\ * & * & R_b \end{bmatrix} \quad (4.91)$$

のように区分表現すると、Q 関数は

$$\begin{aligned} Q(\Theta, \Theta') &\stackrel{c}{=} \frac{1}{2} \left[\log |A^T \Sigma_p^{-1} A| + \log |B^T \Sigma_a^{-1} B| + \log |\Sigma_b^{-1}| \right. \\ &\quad - \text{tr}(A^T \Sigma_p^{-1} A R_p) + 2\mu_p^T \Sigma_p^{-1} A \bar{x}_p - \mu_p^T \Sigma_p^{-1} \mu_p \\ &\quad - \text{tr}(B^T \Sigma_a^{-1} B R_a) + 2\mu_a^T \Sigma_a^{-1} B \bar{x}_a - \mu_a^T \Sigma_a^{-1} \mu_a \\ &\quad - \text{tr}(\Sigma_b^{-1} R_b) + 2\mu_b^T \Sigma_b^{-1} \bar{x}_b - \mu_b^T \Sigma_b^{-1} \mu_b \left. \right] \\ &\quad + \log P(\Theta) \end{aligned} \quad (4.92)$$

と書き直せて、これを用いて各パラメータについて M ステップの更新式を求めることができる。

1) 状態系列: Q 関数の中で $s := \{s_k\}_{k=1}^K$ に関係する項は

$$\begin{aligned} \mathcal{I}_1(s) &:= -\frac{1}{2} \sum_{k=1}^K (\mathbf{o}[k] - \mathbf{c}_{s_k}[k])^T \Upsilon^{-1} (\mathbf{o}[k] - \mathbf{c}_{s_k}[k]) \\ &\quad + \log P(s_1) + \sum_{k=2}^K \log P(s_k|s_{k-1}) \end{aligned} \quad (4.93)$$

となる。ただし、 $\mathbf{o}[k] := ([A\bar{x}_p]_k, [B\bar{x}_a]_k)^T$ であり、 $[\cdot]_k$ はベクトルの k 番目の要素を表す。これを最大化する状態遷移系列 $\{s_k\}_{k=1}^K$ は動的計画法により効率的に解くことができる。まず、すべての状態 i について $\delta_1(i)$ を

$\delta_1(i) = -\frac{1}{2}(\mathbf{o}[1] - \mathbf{c}_i[1])^T \boldsymbol{\Upsilon}^{-1}(\mathbf{o}[1] - \mathbf{c}_i[1])$ と置くと, $k = 2, \dots, K$ について逐次的に $\delta_k(i)$ を $\delta_k(i) = \max_{i'} [\delta_{k-1}(i') - \frac{1}{2}(\mathbf{o}[k] - \mathbf{c}_i[k])^T \boldsymbol{\Upsilon}^{-1}(\mathbf{o}[k] - \mathbf{c}_i[k]) + \phi_{i',i}]$ により計算していくことができる。各ステップで選択される状態番号 $\Psi_k(i) = \operatorname{argmax}_{i'} [\delta_{k-1}(i') + \phi_{i',i}]$ を記憶しておくことで, $k = K$ まで到達後に $s_{k-1} = \Psi_k(s_k)$ ($k = K, \dots, 2$) により選択された状態番号を辿っていくことができ, 最適経路 s_1, \dots, s_K を得ることができる。

2) フレーズ制御パラメータ: ψ の事前分布を $\psi \sim \mathcal{N}(\mu_\psi, 1/\nu_\psi^2)$ とする。式(??)より, \mathbf{A} は定数行列 $\mathbf{U}_2, \mathbf{U}_1, \mathbf{U}_0$ を用いて $\mathbf{A} = \mathbf{U}_2\psi^2 + \mathbf{U}_1\psi + \mathbf{U}_0$ と表せるので, Q 関数の ψ に関する偏導関数は 4 次式となり,

$$\begin{aligned} & 2\operatorname{tr}(\mathbf{U}_2^T \mathbf{U}_2 \mathbf{R}_p)\psi^4 + 3\operatorname{tr}(\mathbf{U}_2^T \mathbf{U}_1 \mathbf{R}_p)\psi^3 \\ & + \{\operatorname{tr}((2\mathbf{U}_2^T \mathbf{U}_0 + \mathbf{U}_1^T \mathbf{U}_1) \mathbf{R}_p) - 2\boldsymbol{\mu}_p^T \mathbf{U}_2 \bar{\mathbf{x}}_p + \sigma_p^2 \nu_\psi^2\}\psi^2 \\ & + \{\operatorname{tr}(\mathbf{U}_1^T \mathbf{U}_0 \mathbf{R}_p) - \boldsymbol{\mu}_p^T \mathbf{U}_1 \bar{\mathbf{x}}_p - 2\sigma_p^2 \nu_\psi^2 \mu_\psi\}\psi - 2K\sigma_p^2 \end{aligned}$$

の根を解くことで極値を求める能够である。求まつた 4 つの極値の中で $\mathcal{I}_2(\psi)$ を最大にする ψ が更新値となる。

3) アクセント制御パラメータ: 同様に φ の事前分布を $\varphi \sim \mathcal{N}(\mu_\varphi, 1/\nu_\varphi^2)$ とする。更新値の導出過程は 4) と同様なので省略する。

4) その他:

$$A_p[k] = \hat{u}_p[k], \quad (k \in \mathcal{T}_{p1}) \quad (4.94)$$

$$A_a^{(n)} = \frac{1}{|\mathcal{T}_{a_n}|} \sum_{k \in \mathcal{T}_{a_n}} [\mathbf{B} \bar{\mathbf{x}}_a]_k, \quad \mathcal{T}_{a_n} = \{k | s_k = a_n\} \quad (4.95)$$

$$\mu_b = \mathbf{1}^T \bar{\mathbf{x}}_b / T \quad (4.96)$$

$$\sigma_p^2 = (\operatorname{tr}(\mathbf{A}^T \mathbf{A} \mathbf{R}_p) - 2\boldsymbol{\mu}_p^T \mathbf{A} \bar{\mathbf{x}}_p + \boldsymbol{\mu}_p^T \boldsymbol{\mu}_p) / K \quad (4.97)$$

$$\sigma_a^2 = (\operatorname{tr}(\mathbf{B}^T \mathbf{B} \mathbf{R}_a) - 2\boldsymbol{\mu}_a^T \mathbf{B} \bar{\mathbf{x}}_a + \boldsymbol{\mu}_a^T \boldsymbol{\mu}_a) / K \quad (4.98)$$

$$\sigma_b^2 = (\operatorname{tr}(\mathbf{R}_b) - 2\mu_b \mathbf{1}^T \bar{\mathbf{x}}_b) / K + \mu_b^2 \quad (4.99)$$

c h a p t e r

5

音楽信号解析

本章では、モノラル音楽音響信号の解析技術について解説します。まず、音楽音響信号を音符単位に分解（音源分離・自動採譜）するうえで有用な非負値行列因子分解 (nonnegative matrix factorization: NMF) および確率的潜在成分解析 (probabilistic latent component analysis: PLCA) を説明します。両手法とともに、ある確率モデルの最尤推定として解釈することが可能ですが、NMF は階乗モデル (factorial model)、PLCA は混合モデル (mixture model) に対応している点で異なります。また、両手法とともに、これらの性質の違いを考慮した適切な事前分布を導入することにより、ノンパラメトリックベイズモデルの定式化とベイズ推定が可能になります。一方、音楽音響信号を異なる音響的性質をもつ音源信号に分解する技術についても解説します。例えば、音楽音響信号を調波音と打楽器音に分離する技術、歌声と伴奏音に分離する技術などが近年盛んに研究されています。

5.1 音楽音響信号の構成要素への分解

音楽情報処理においては、解析対象となる音楽音響信号はモノラル(1チャネル)であることを想定するのが一般的です。市販 CD に録音されている音楽音響信号は、ステレオ(左右2チャネル)形式であることがほとんどですが、通常のマルチチャネル信号処理技術(??章)は多くの場合使えません。ポピュラー音楽の制作過程においては、各楽器パートを別々のマイクで個別

に録音しておき、定位感を演出するために左右の音量バランスのみを調節してから、全パートをミキシングすることが広く行われています。この場合、各楽器パートのステレオ信号には、実際に2つのマイクを楽器の前において録音した時とは異なり、左右チャネルで位相差がありません。本書では、左右チャネルの音量差に着目する音響信号解析手法は取り扱わず、ステレオ信号はあらかじめモノラル信号に変換してから処理を行うものとします。

解析対象の情報を一切与えずに、すなわちブラインド環境下において、モノラルの音楽音響信号を構成要素に分解することは非常に難しい課題です。ここで、何を構成要素とみなすべきかは、タスクに合わせて十分な検討が必要です。例えば、自動採譜では、音楽音響信号を楽譜に変換することを目的としているため、構成要素は音符に対応していることが望まれます。一方、音楽音響信号を歌声・ギター・ベース・ドラムといった楽器パートごとに分離したいのであれば、構成要素は楽器音の音色に対応していなければなりません。このように、構成要素(parts)を定義すれば、それらの組み合わせによって複雑な音楽音響信号が生成される過程を考えることにより、独自の確率的生成モデルを定式化することができます(順問題)。いったん確率モデルが定式化できれば、観測変数として音楽音響信号が与えられた時に、潜在変数である構成要素や生成過程における各種のパラメータを推定する問題を解くことが目標になります(逆問題)。

モノラル音響信号の分離は数学的に不良設定問題(劣決定)であり、なんらかの制約や基準なしでは最適解を一意に定めることができません。簡単に言えば、 $X + Y = 10$ を満たす X や Y を一意に定められないのと同じことで、 X や Y が満たすべき性質を適切に表現し、その制約がどの程度満たされているかを数値化することにより、はじめて最適な X や Y を求めることができます。したがって、音楽音響信号を分解するには、音響信号に内在する「スパース性」や「低ランク性」といった何らかの性質を音の聞き分けの手がかりに用いる必要があります。具体的に言えば、このような構成要素の持つ性質を音響信号の確率的生成モデルに取り込むことにより、尤度最大化という統一的な基準のもとで、最適な構成要素を推定することができるようになります。さらに、構成要素を直接観測できないがゆえの不確実性をベイズ的に適切に取り扱うことも可能になります。

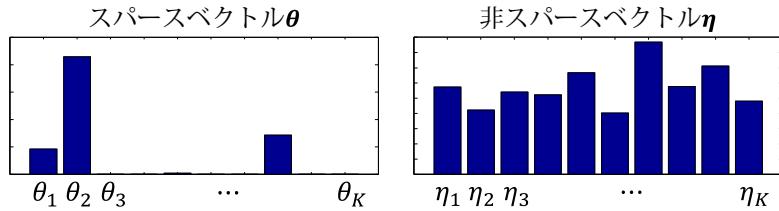


図 5.1 K 次元のスパースベクトル θ と非スパースベクトル η . 各ベクトルは各次元 k ($1 \leq k \leq K$) に対して $\theta_k \sim \text{Gamma}(0.1, 0.1)$, $\eta_k \sim \text{Gamma}(10, 10)$ としてランダムに生成. ただし, $\mathbb{E}[\theta_k] = \mathbb{E}[\eta_k] = 1$ であることに注意.

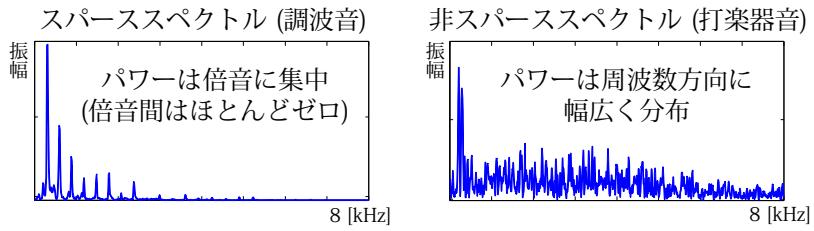


図 5.2 調波構造を持つ楽器音（ピアノ）のパワースペクトルと調波構造を持たない打楽器音（スネアドラム）のパワースペクトル.

5.1.1 スパース性

ベクトルや行列がスパースであるとは, ほとんどの要素がゼロをとる状態を指します（図 5.1）. 一見複雑に思える音楽音響信号も, 高々有限個の楽器音が重なりあってできています. 例えば, あるピアノ曲を考えてみると, 楽曲中に出現する音高は音域や調に依存することから, ピアノの持つ 88 種類の音高（鍵盤の個数）の使われやすさに大きな偏りがあります. また, 音楽音響信号の構成要素そのものにもスパース性が存在します. 例えば, 調波構造を持つ楽器音は, 倍音周波数付近にパワーが集中しており, 倍音間にはほとんどパワーが存在しません（図 5.2）. 一方, バスドラムやスネアドラムのように, 明確な音高を持たない打楽器音は, 周波数方向に幅広くパワーが分布しており, スパースではありません. 図 5.1 に示すように, 適切な確率分布を用いれば, スパース性を表現することができます.

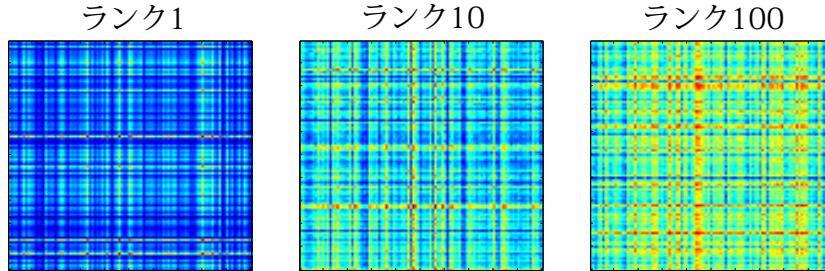


図 5.3 ランク $K = 1, 10, 100$ の行列の例 . 各行列 \mathbf{X} は , $A_{nk} \sim \text{Gamma}(10, 10)$ および $B_{km} \sim \text{Gamma}(10, 10)$ として行列 $\mathbf{A} \in \mathbb{R}^{N \times K}$ および $\mathbf{B} \in \mathbb{R}^{K \times M}$ をランダムに生成したあと , $\mathbf{X} = \mathbf{AB}$ として計算 .

5.1.2 低ランク性

ある行列 $\mathbf{X} \in \mathbb{R}^{N \times M}$ が低ランクであるとは , $K \ll M, N$ として ,

$$\mathbf{X} = \mathbf{AB} \quad (5.1)$$

となるような行列 $\mathbf{A} \in \mathbb{R}^{N \times K}$ および $\mathbf{B} \in \mathbb{R}^{K \times M}$ が存在することをいいます . ここで , K はランク (階数) と呼ばれ , もとの行列 \mathbf{X} の行数 M や列数 N よりはるかに小さい値をとります . 究極的には , $K = 1$ である場合に \mathbf{X} はランク 1 の行列となり , $\mathbf{X} = ab^T$ となるようなベクトル $a \in \mathbb{R}^N$ および $B \in \mathbb{R}^M$ の直積で表現されます .

低ランクの行列を可視化してみると , 布地の織り目のような縦横の模様があることに気づきます (図 5.3) . 特にランク 1 の場合 , 行列を任意の場所で縦方向にスライスしてみると , 抜き出されるベクトルの全体的な形状は一定であり , そのスケールのみが変化することになります (場所ごとに直径が変化する金太郎あめを想像してください) . 行列を任意の場所で横方向にスライスする場合も同じことです .

実際の楽器音のスペクトログラムも , 同様の縦横の模様を持っています (図 5.3) . 特に , バスドラムやスネアドラムといった打楽器音は , 同じ形状のパワースペクトルが何度も繰り返し出現し , 各発音時刻の後では , パワースペクトルの形状が保たれたまま音量が徐々に減衰していく現象がみられることから , 低ランクな行列 (理想的にはランク 1 の行列) で近似することは

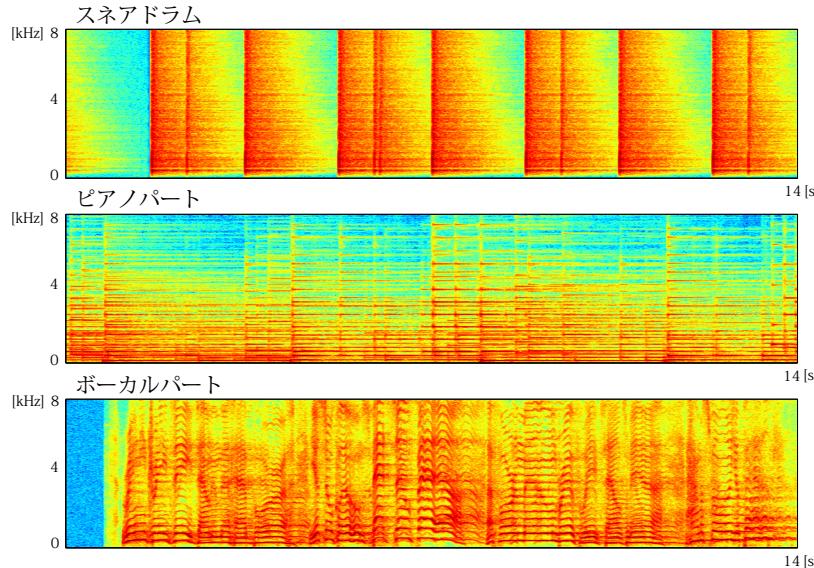


図 5.4 スネアドラム、ピアノパート、ボーカルパートのパワースペクトログラム。これらをそれぞれ行列とみなしたときに、この順番にランクが増加すると考えられる。研究コミュニティベースの音源分離コンテスト SiSEC 2008^[?]において配布されている開発用データに含まれる楽曲 “bearlin roads” を利用。

ある程度妥当であると考えられます。ピアノパートは複数の音高の重畠で成り立っているため、ランク 1 の行列で近似することは困難なもの、やはり低ランク構造を持ちそうなことが分かります。一方、ボーカルパートは、時間方向・周波数方向にパワースペクトルが複雑に変化しているため、低ランクの行列とはみなすことは困難です。

5.1.3 音源分離への応用

NMF では、基底ベクトル w_k 及びアクティベーションベクトル h_k がスパースになりやすい。一方、同様の行列分解形式 $X = WH$ を持つ主成分分析 (Principal Component Analysis: PCA) では、行列要素は負値をとることが許されている。従って、式 (3.2) のように入力を基底の線形和で表現

する際に、基底の加減算による細かな調節が可能となる。一方、NMFでは、基底の加算しか許されず、いったんアクティベートされた基底の影響を打ち消すことはできない。そのため、各基底 w_k が x_n 中の局所的な「パーツ」に対応し、少數の基底で x_n を表現する方が都合が良い。NMFは当初、顔画像（ピクセル値ベクトル）の集合に対して適用され、目・鼻・口といった顔のパーツ画像に対応する基底ベクトルが得られることが分かった。このようなパーツに基づく分解表現は、音楽音響信号の分解と相性が良い。なぜなら、調波音のスペクトルは周波数軸上でスパースであり、混合音スペクトルは局所的な周波数帯域上のパーツの組み合わせとみなせるからである。

観測信号の複素スペクトログラムを $\tilde{\mathbf{X}} = [\tilde{x}_1, \dots, \tilde{x}_N] \in \mathbb{C}^{M \times N}$, k 番目の音源信号の複素スペクトログラムを $\tilde{\mathbf{X}}_k = [\tilde{x}_{k1}, \dots, \tilde{x}_{kN}] \in \mathbb{C}^{M \times N}$ とする。 M は周波数ビン数、 N はフレーム数である。観測した混合音が K 個の音源信号の瞬時混合であると仮定すると、以下が成立する。

$$\tilde{\mathbf{X}} = \sum_{k=1}^K \tilde{\mathbf{X}}_k \quad \left(\tilde{x}_n = \sum_{k=1}^K \tilde{x}_{kn} \right) \quad (5.2)$$

観測変数 $\tilde{\mathbf{X}}$ を潜在変数 $\tilde{\mathbf{X}}_k$ に分解する問題は不良設定であるので、 $\tilde{\mathbf{X}}_k$ に対応するパワースペクトログラム $\mathbf{X}_k = [x_{k1}, \dots, x_{kN}] \in \mathbb{R}_+^{M \times N}$ ($x_{knm} = |\tilde{x}_{knm}|^2$) は、ランク 1 行列 $\mathbf{Y}_k = [y_{k1}, \dots, y_{kN}] \in \mathbb{R}_+^{M \times N}$ で近似する（図 5.5）。

$$\mathbf{X}_k \approx \mathbf{w}_k \mathbf{h}_k^T \stackrel{\text{def}}{=} \mathbf{Y}_k \quad (5.3)$$

すなわち、 \mathbf{Y}_k の任意のフレーム n におけるパワースペクトル y_{kn} は基底スペクトル $\mathbf{w}_k \in \mathbb{R}^M$ を重み h_{kn} でスケーリングするだけで得られるという仮定をおいた ($y_{kn} = h_{kn} w_k$)。

まず、潜在変数 \tilde{x}_{kn} が y_{kn} で定まる対角共分散行列を持つ複素ガウス分布に従うことを仮定する。

$$\tilde{x}_{kn} \sim \mathcal{N}_c(\mathbf{0}, \text{diag}(\mathbf{y}_{kn})) \quad (5.4)$$

ただし、 $\text{diag}(\eta)$ はベクトル η を対角成分に持つ対角行列を表す。式 (5.2) に着目すると、複素ガウス分布の再生性から

$$\tilde{x}_n \sim \mathcal{N}_c(\mathbf{0}, \text{diag}(\mathbf{y}_n)) \quad (5.5)$$

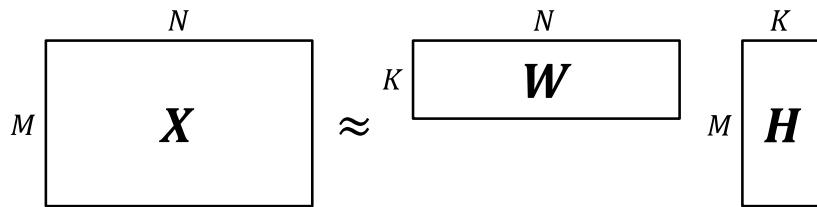


図 5.5 パワースペクトログラムに対する IS ダイバージェンスに基づく非負値行列分解 (IS-NMF) の適用結果 .

を得る . ただし , $y_n = \sum_k y_{kn}$ である . 従って , $x_{nm} = |\tilde{x}_{nm}|^2$ は指數分布に従うことが分かる .

$$x_{nm} \sim \text{Exponential}(y_{nm}) \quad (5.6)$$

ここで , 式 (5.5) の対数をとって符号反転させると , 式 (3.6) と定数項を除いて等しい . 従って , 式 (5.5) の最大化 (最尤推定) は式 (3.6) の最小化と等価であり , IS-NMF の適用が適切であると分かる .

最終的に , 式 (5.4) 及び式 (5.5) に着目すると , \tilde{x}_n が与えられたときの \tilde{x}_{kn} の事後分布は複素ガウス分布になることが分かり , その平均と分散は

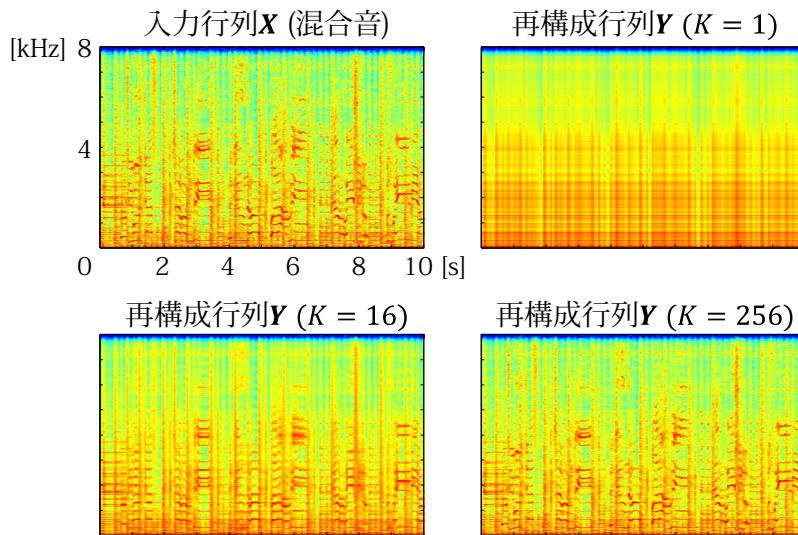
$$\mathbb{E}[\tilde{x}_{kn}|\tilde{x}_n] = \text{diag}(\mathbf{y}_{kn})\text{diag}(\mathbf{y}_n)^{-1}\tilde{x}_n \quad (5.7)$$

$$\begin{aligned} \mathbb{V}[\tilde{x}_{kn}|\tilde{x}_n] &= \text{diag}(\mathbf{y}_{kn}) \\ &\quad - \text{diag}(\mathbf{y}_{kn})\text{diag}(\mathbf{y}_n)^{-1}\text{diag}(\mathbf{y}_{kn}) \end{aligned} \quad (5.8)$$

で与えられる . この処理はウィナーフィルタリングと呼ばれ , \tilde{X}_k の位相は \tilde{X} の位相と同一であるという仮定が置かれている . 最後に , 逆フーリエ変換を用いて , $\mathbb{E}[\tilde{X}_k|\tilde{X}]$ から k 番目の音源信号を復元することができる .

基底数 K を変化させながら IS-NMF を適用した結果を図 5.6 に示す . K が小さすぎると近似が荒く , K を大きくしすぎると物理的な意味を持たない極めて局所的な基底ばかりになる . このことから , 基底数 K を適切に定める重要な性質が分かる .

実際には , 性質の良くない局所解に陥りやすい IS-NMF の代わりに KL-NMF が利用される場合が多い . このとき , X や X_k は振幅スペクトログラ

図 5.6 異なる基底数 K に対する IS-NMF の結果 .

ムとするのが一般的である ($x_{knm} = |\tilde{x}_{knm}|$)。

まず、潜在変数 x_{knm} が y_{knm} で定まるポアソン分布に従うことを仮定する。

$$x_{knm} \sim \text{Poisson}(y_{knm}) \quad (5.9)$$

ここで、複数の音源信号の重畠における振幅スペクトルの加法性を仮定すると（実際には成立しないことに注意）、ポアソン分布の再生性から

$$x_{nm} \sim \text{Poisson}(y_{nm}) \quad (5.10)$$

を得る。ここで、式 (5.10) の対数をとって符号反転させると、式 (3.5) と定数項を除いて等しい。従って、式 (5.10) の最大化（最尤推定）は式 (3.5) の最小化と等価である。

5.1.4 音源分離への応用

式 (5.2) を満たすように、 \tilde{x}_n を $\{\tilde{x}_{kn}\}_{k=1}^K$ の和に分解したい。まず、潜在

変数 \tilde{x}_{kn} が共分散行列 \mathbf{Y}_{kn} を持つ複素ガウス分布に従うことを仮定する。

$$\tilde{\mathbf{x}}_{kn} \sim \mathcal{N}_c(\mathbf{0}, \mathbf{Y}_{kn}) \quad (5.11)$$

ここで、式(5.4)のように共分散行列を対角行列に限定しないことで、周波数ピクセル間の相関を考慮している。式(5.2)と複素ガウス分布の再生性から

$$\tilde{\mathbf{x}}_n \sim \mathcal{N}_c(\mathbf{0}, \mathbf{Y}_n) \quad (5.12)$$

を得る。ただし、 $\mathbf{Y}_n = \sum_k \mathbf{Y}_{kn}$ である。ここで、式(5.12)の対数をとって符号反転させると、式(3.110)と定数項を除いて等しい。従って、式(5.12)の最大化は式(3.110)の最小化と等価であり、LD-PSDTF を用いて \mathbf{Y}_n や \mathbf{Y}_{kn} を求めることができる。

最終的に、式(5.11)、(5.12)から、 $\tilde{\mathbf{x}}_n$ が与えられたときの $\tilde{\mathbf{x}}_{kn}$ の事後分布は複素ガウス分布になることが分かり、その平均と分散は次式となる。

$$\mathbb{E}[\tilde{\mathbf{x}}_{kn} | \tilde{\mathbf{x}}_n] = \mathbf{Y}_{kn} \mathbf{Y}_n^{-1} \tilde{\mathbf{x}}_n \quad (5.13)$$

$$\mathbb{V}[\tilde{\mathbf{x}}_{kn} | \tilde{\mathbf{x}}_n] = \mathbf{Y}_{kn} - \mathbf{Y}_{kn} \mathbf{Y}_n^{-1} \mathbf{Y}_{kn} \quad (5.14)$$

ここで、式(5.7)とは異なり、 $\tilde{\mathbf{X}}_k$ の位相は $\tilde{\mathbf{X}}$ の位相とは異なる点に注意する。IS-NMF のように各周波数ピクセル n, m ごとではなく、各フレーム n ごとに一挙に分離を行うことで、周波数ピクセル間の相関を考慮しながら高品質な分離が可能となる。

5.1.5 確率モデルの最尤推定としての定式化

文献 [4,5]において、NMF と同様にガンマ過程を用いて基底数を $K \rightarrow \infty$ としたノンパラメトリックベイズモデルが提案されている。今後の課題として、vN-PSDTF に対する乗法更新則の導出や計算量の削減が挙げられる。

5.2 楽器パートの分離

5.2.1 歌声・伴奏音の分離

最後に、スパース性に基づいて音楽音響信号を楽器種別に分離する技術を紹介する。ロバスト主成分分析 (RPCA) は、入力行列 \mathbf{X} を低ランク行列 \mathbf{L} とスパース行列 \mathbf{S} の和に分解する。具体的には、 $\mathbf{X} = \mathbf{L} + \mathbf{S}$ を満たすとい

Algorithm 3 LD-PSDTF の最尤推定

Require: テンソル $\mathbf{X} \in \mathbb{C}^{M \times M \times N}$, 基底数 K

- 1: 基底テンソル $\mathbf{W} \in \mathbb{C}^{M \times M \times K}$ をランダムに初期化
- 2: 非負値行列 $\mathbf{H} \in \mathbb{R}_+^{M \times K}$ をランダムに初期化
- 3: **while** not converged **do**
- 4: $\mathbf{P}_k = \sum_{n=1}^N h_{kn} \mathbf{Y}_n^{-1}$
- 5: $\mathbf{Q}_k = \sum_{n=1}^N h_{kn} \mathbf{Y}_n^{-1} \mathbf{X}_n \mathbf{Y}_n^{-1}$
- 6: コレスキー分解 $\mathbf{Q}_k = \mathbf{L}_k \mathbf{L}_k^T$
- 7: $\mathbf{W}_k \leftarrow \mathbf{W}_k \mathbf{L}_k (\mathbf{L}_k^T \mathbf{W}_k \mathbf{P}_k \mathbf{W}_k \mathbf{L}_k)^{-\frac{1}{2}} \mathbf{L}_k^T \mathbf{W}_k$
- 8: $h_{kn} \leftarrow h_{kn} \left(\frac{\text{tr}(\mathbf{Y}_n^{-1} \mathbf{W}_k \mathbf{Y}_n^{-1} \mathbf{x}_n)}{\text{tr}(\mathbf{Y}_n^{-1} \mathbf{W}_k)} \right)^{\frac{1}{2}}$
- 9: **end while**
- 10: **Return** 基底テンソル \mathbf{W} , 非負値行列 \mathbf{H}

う制約のもとで、以下で定義される最適化問題を解く。

$$\text{minimize } \|\mathbf{L}\|_* + \lambda \|\mathbf{S}\|_1 \quad (5.15)$$

ここで、 $\|\cdot\|_*$, $\|\cdot\|_1$ はそれぞれ核ノルム, L1 ノルムである。厳密ではないが拡張ラグランジュ法に基づく効率的な解法を Algorithm 4 に示す。5.1 章で議論したとおり、混合音スペクトログラムを \mathbf{X} として与えると、伴奏音スペクトログラム \mathbf{L} と歌声スペクトログラム \mathbf{S} が得られる。また、メディアンフィルタを時間方向・周波数方向に適用することで調波音・打楽器音を分離すること (Harmonic/Percussive Source Separation: HPSS) も可能である^[7]。これらの技術による分離結果を図 5.7 に示す。

5.2.2 調波音・非調和音の分離

HPSS のいくつかの手法を紹介。

5.2.3 音色に基づく分離

ソースフィルタモデル

5.3 おわりに

本稿では、スパース性に基づく音楽音響信号分解技術として、非負値行列分

Algorithm 4 Inexact ALM に基づく RPCA

Require: 入力行列 $\mathbf{X} \in \mathbb{R}^{M \times N}$, 重み係数 λ

- 1: 初期化 : $\mathbf{Y} = \mathbf{X} / \max(\|\mathbf{X}\|_2, \lambda^{-1} \|\mathbf{X}\|_\infty) \in \mathbb{R}^{M \times N}$
- 2: 初期化 : $\mathbf{S} = \mathbf{0} \in \mathbb{R}^{M \times N}$
- 3: 初期化 : $\mu > 0$ (e.g., $\mu = 1.25/\|\mathbf{X}\|_2$)
- 4: 初期化 : $\rho > 1$ (e.g., $\rho = 1.5$)
- 5: **while** not converged **do**
- 6: $[\mathbf{U}, \Sigma, \mathbf{V}] = \text{SVD}(\mathbf{X} - \mathbf{S} + \mu^{-1} \mathbf{Y})$
- 7: $\mathbf{L} \leftarrow \mathbf{U} \mathcal{F}_{\mu^{-1}}(\Sigma) \mathbf{V}^T$
- 8: $\mathbf{S} \leftarrow \mathcal{F}_{\lambda\mu^{-1}}(\mathbf{X} - \mathbf{L} + \mu^{-1} \mathbf{Y})$
- 9: $\mathbf{Y} \leftarrow \mathbf{Y} + \mu(\mathbf{X} - \mathbf{L} - \mathbf{S})$
- 10: $\mu \leftarrow \rho\mu$
- 11: **end while**
- 12: **Return** 低ランク行列 \mathbf{L} , スパース行列 $\mathbf{S} \in \mathbb{R}^{M \times N}$

$$\ast \mathcal{F}_\epsilon[x] = x - \epsilon \text{ (if } x > \epsilon\text{), } x + \epsilon \text{ (if } x < -\epsilon\text{), } 0 \text{ (otherwise)}$$

解 (NMF), 半正定値テンソル分解 (PSDTF), ロバスト主成分分析 (RPCA), 調波・打楽器分離音 (HPSS) を紹介した。これらは音声情報処理分野の影響を受けながら、音楽情報処理分野で独自の発展を遂げている。例えば、NMF (音響モデル) はソース・フィルタ型の楽器音モデルや、楽譜の生成モデル (言語モデル) と統合する試みが進んでいる。本稿が読者の理解の一助になれば幸いである。

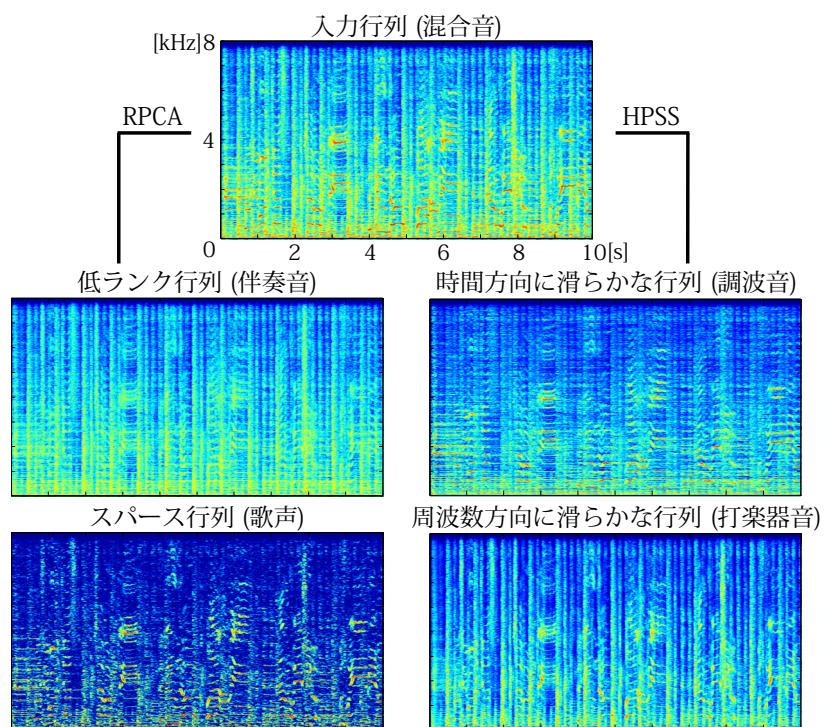


図 5.7 音楽音響信号に対する RPCA・HPSS の適用結果 .

B i b l i o g r a p h y h y

参考文献

- [1] D. D. Lee and H. S. Seung, “Learning the parts of objects by non-negative matrix factorization,” *Nature*, **401**, 788–791 (1999).
- [2] C. Févotte, N. Bertin, and J.-L. Durrieu, “Nonnegative matrix factorization with the Itakura-Saito divergence: With application to music analysis,” *Neural Computation*, **21**(3), 793–830 (2009).
- [3] M. Hoffman, D. Blei, and P. Cook, “Bayesian nonparametric matrix factorization for recorded music,” *ICML*, 439–446 (2010).
- [4] K. Yoshii, R. Tomioka, D. Mochihashi, and M. Goto, “Infinite positive semidefinite tensor factorization for source separation of mixture signals,” *ICML*, 576–584 (2013).
- [5] K. Yoshii, R. Tomioka, D. Mochihashi, and M. Goto, “Beyond NMF: Time-domain audio source separation without phase reconstruction,” *ISMIR*, 369–374 (2013).
- [6] Z. Lin, M. Chen, and Y. Ma, “The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices,” *Math. Prog.* (2010).
- [7] D. FitzGerald, “Harmonic/percussive separation using median filtering,” *DAFx* (2010).
- [8] P. Smaragdis and J. C. Brown, “Non-negative matrix factorization for polyphonic music transcription,” *WASPAA*, 177–180 (2003).
- [9] 亀岡弘和, “非負値行列因子分解の音響信号処理への応用,” 日本音響学会誌, 68(11), 559–565 (2012).
- [10] H. Sawada, H. Kameoka, S. Araki, and N. Ueda, “Multichannel extensions of non-negative matrix factorization with

- complex-valued data,” *IEEE Trans. TASLP*, **21**(5), 971–982 (2013).
- [11] B. Kulis, M. Sustik, and I. Dhillon, “Low-rank kernel learning with Bregman matrix divergences,” *JMLR*, **10**, 341–376 (2009).

c h a p t e r

6

マイクロホンアレイ音響信号解析

[本章では、マイクロホンアレイ音響信号解析技術について説明します。]

6.1 音源定位

6.2 音源分離

6.3 残響除去

6.4 音源定位・音源分離・残響除去の統合モデル