

脳を学ぶ上で重要な数学シリーズ  
発展数学編

後藤 優仁

2021 年 12 月 15 日

# 目次

第 1 章	はじめに	3
第 2 章	数学的テクニック	5
2.1	ラグランジュの未定乗数法	5
2.1.1	定義と用法	5
2.1.2	証明	6
2.1.3	例題	7
第 3 章	情報理論	9
3.1	エントロピー	9
3.2	KL 距離	16
3.2.1	他の分布間距離	19
3.3	条件付きエントロピー	20
3.4	相互情報量	20
3.5	Transfer Entropy	21
第 4 章	信号処理	24
4.1	独立成分分析	24
4.2	正準相関分析	26
4.2.1	定義と用法	26
4.2.2	導出	27
4.2.3	固有値問題へ	29
4.2.4	例題	30
第 5 章	機械学習	34
5.1	EM アルゴリズム	34
5.1.1	不完全データ集合	35
5.1.2	EM アルゴリズムの概要	37
5.1.3	変分下界	37
5.1.4	E ステップ	40
5.1.5	M ステップ	42
5.2	変分ベイズ推定	44

第 6 章	知覚心理実験	47
6.1	信号検出理論 . . . . .	47
6.1.1	感度 . . . . .	51
6.1.2	反応バイアス . . . . .	51
6.1.3	まとめ . . . . .	52

# 第1章 はじめに

さて、この数学シリーズの中でもこの advanced はかなり異質で、筆者が脳神経科学を研究する上で関わっていった様々な (あまり一般的ではない) 議論を展開するために必要になった数学的知識をまとめる場です。筆者は脳活動の非線形ダイナミクスの機能的な役割を研究しています。脳は多数の非線形素子が結合した大自由度の力学系とみなすことができ、多様なダイナミクスを示します。計算論的神経科学の観点で、神経系が示す同期、自発活動、誘発活動ダイナミクス、ノイズ誘発同期、... といった様々な現象を捉え、その情報処理メカニズムの解明を試みています。

こうした作業には、信号処理は勿論、非線形ダイナミクス (つまり力学系)、情報理論、複雑系、機械学習... 多種多様な数学的、理論的知識が求められます。この勉強については終わりがなく、本当に役に立つのかも分かりません。ただ確実に言えるのは、

「○○をやっている時の脳は△△領域で～～ ms 後に  $\times$  Hz の波が同期している！！」

なんてことだけ見ても、脳を理解する事は出来ないということです。その活動が何故大事なのか、その活動を通してどんな処理をしているのか... こうしたところまで理解してこそその科学だと、筆者含め計算論的神経科学者たちは考えています。

実験的にデータを集めるだけでは理解に至らず、その背景で何が行われているのか、どんな表現がなされているのか。そんなところまで考えないとだよね、というスタンスです。その必要性が分からん、そんなに大事だろうか。そう思う人はユニークな思考実験的論文があるので是非読んでみてください [1][2]。何をもって脳の理解とするか？

この問に答えるのが、有名なマーの3レベルという概念？お話？ [3] です。

- 計算理論
- 表現とアルゴリズム
- ハードウェアによる実装

この3段階を踏み、相互に対応付ける事が脳の理解に大切だ、とする話です。計算理論は、我々が脳を使ってどんな「計算」を行っているのか、行っているべきなのか、といった議論。自由エネルギー原理だとかの話、すなわち脳が採用している戦略を考える所です。ハードウェアによる実装は、多くの神経科学者がやっているように脳のどの部分でどんな活動が起きていてといった解剖学・生理学的知見。最後に表現とアルゴリズムは、ハードウェアの実装を使っていかに計算理論で提案された処理を実行する

のか、になります。

彼に言わせれば、これまで主流の神経科学はハードウェアの実装ばかりだったわけですね。計算理論と表現に関する議論は、無論ありましたがあまり活発ではなかった。

ここから先、どう考えるかは個人の自由だと思います。計算理論や表現についても考えていこうとするか、そんなに色々手を出しても回収しきれないと見切りをつけるか、あるいは他の人が結び付けてくれる事を期待して実験データを提供するに集中するのか...

筆者は、計算理論と実装を結ぶ、表現の研究者になりたいと考えた次第です。一番勉強する事が多いような気もしますが、楽しんでやっていきます。

長くなりましたが、本書はそんなモチベーションのもと、計算理論やアルゴリズムについて学習したことをまとめていくものにします。なので神経に本当に役立つのか、理解が正しいのか、様々な問題があると思いますが、まあ教科書ではなく筆者のノートだと思って見てください。結構やってみると楽しいです。また本稿はその性質上、随所で本や論文を引用しながら議論を展開していきます。筆者の拙い理解での説明では不十分だったり不適切だったりすることも少なくないはずなので、気になるところは適宜参照してください。

ps. なんて最初は考えていましたが、結局その他にも勉強する事が増えてきたのでもはや「その他集」といったキメラになりました。

## 第2章 数学的テクニック

本稿をまとめていくにあたって、basic で解説したような基礎の微分積分や線形代数、統計ではたちうち出来なくなってしまったので、どうしても学習が必要になってしまったもの達をここにまとめておきます。いずれは basic とかに飛ばされるのかも知れません。

### 2.1 ラグランジュの未定乗数法

こいつ、本当に色々なところで出てきます。機械学習でも、ICA や CCA などの手法もそうだし、とにかく「またお前か!」という感じでした。初めて見た時は意味わからんすぎたので飛ばしていましたが、ちゃんと勉強したら案外そんな事もなかったのでもとめます。いやほんと、こいつが出来るようになるだけで大分学習が楽になる気がします。

#### 2.1.1 定義と用法

まず、式を載せてみます。本来は多次元に拡張可能なものなのですが、とりあえずここでは簡単のため2次元で載せます。

##### ラグランジュの未定乗数法

制約条件  $g(x, y) = 0$  の元で、 $f(x, y)$  を最大化する  $(x, y)$  を求める問題を考えるとき、ラグランジュ乗数  $\lambda$  を用いてラグランジュ関数  $L$  を

$$L(x, y, \lambda) = f(x, y) - \lambda g(x, y) \quad (2.1)$$

と置くと、ある  $(x_0, y_0)$  が題意の条件を満たすなら、ある  $\lambda_0$  が存在し、 $(x_0, y_0, \lambda_0)$  において

$$\frac{\partial L}{\partial x} = \frac{\partial L}{\partial y} = \frac{\partial L}{\partial \lambda} = 0 \quad (2.2)$$

が成り立つ。

といったようなものです。

何言ってるか訳分からんですよね。2変数関数の極大値問題を考えていて、それが分からないから悩んでいる人に対して「じゃあ3変数にしようか!!」とか意味が分からんし、ここで深くは触れないですけど、実際ラグランジュ使って解いたところで、これが出てきた解=最大値(最小値)となるとは限ら

なくて、あくまで停留点、つまり極値の候補しか分かんないという。うーん。意味あるのかそれ... 意味わからんし... なんて考えてしまい、勉強をやめていました。しかしこれから確認するように、考え方はシンプルだし、とてもありがたいものです。

まず、用語ですがラグランジュの”未定”乗数法というからには、未定なんらかの値を掛け算して解く方法？みたいに捉えられますね。この未定乗数が $\lambda$ の事です。それから、何故未定かって正味 $\lambda$ の値そのものには興味がないというか、どうでもいいのです。 $\lambda$ を使って表せるという事実が大事。

## 2.1.2 証明

さて、まずは式 (2.2) の各項について考えていきます。あえて3つ目、 $\partial L / \partial \lambda = 0$  から考えていきます。と言っても、計算すればすぐ分かりますので式をどうぞ。

$$\begin{aligned}\frac{\partial L}{\partial \lambda} &= \frac{\partial(f(x, y) - \lambda g(x, y))}{\partial \lambda} \\ &= \frac{\partial f(x, y)}{\partial \lambda} - \lambda \frac{\partial g(x, y)}{\partial \lambda} \\ &= -g(x, y) = 0\end{aligned}\tag{2.3}$$

普通に、関数 $L$ を $\lambda$ で偏微分するとこうなりますね。これは結局、制約条件である $g(x, y) = 0$ の言い換えです。特に意味なし。以上！

では残りの二つについて、すなわち $\partial L / \partial x = \partial L / \partial y = 0$ について考えます。同じく $L$ を展開していくと

$$\begin{aligned}\frac{\partial L}{\partial x} &= \frac{\partial f(x, y)}{\partial x} - \lambda \frac{\partial g(x, y)}{\partial x} = 0 \\ \therefore \frac{\partial f(x, y)}{\partial x} &= \lambda \frac{\partial g(x, y)}{\partial x}\end{aligned}\tag{2.4}$$

$$\begin{aligned}\frac{\partial L}{\partial y} &= \frac{\partial f(x, y)}{\partial y} - \lambda \frac{\partial g(x, y)}{\partial y} = 0 \\ \therefore \frac{\partial f(x, y)}{\partial y} &= \lambda \frac{\partial g(x, y)}{\partial y}\end{aligned}\tag{2.5}$$

と、それぞれ表せます。ただの移項なのでここまでは大丈夫でしょう。そして、でてきたこの二つの式をベクトルの形に整理すると次のようになります。

$$\begin{pmatrix} \frac{\partial f(x, y)}{\partial x} \\ \frac{\partial f(x, y)}{\partial y} \end{pmatrix} = \lambda \begin{pmatrix} \frac{\partial g(x, y)}{\partial x} \\ \frac{\partial g(x, y)}{\partial y} \end{pmatrix}\tag{2.6}$$

綺麗になりました。さて、さらにここで、一般にこれらのように関数をそれぞれの変数で偏微分して並べたやつって勾配 ( $\nabla$ ) と言うんです。なので更にこの式 (2.6) を以下のように表します。

$$\nabla f = \lambda \nabla g \quad (2.7)$$

さて、この意味するところは  $f$  の勾配は  $g$  の勾配が  $\lambda$  倍されたものであるという事になります。勾配とは、その点で取った接線に垂直な線、法線ベクトルを指すんです。よって、この式の意味するところは  $f$  と  $g$  の法線ベクトルが定数倍、並行であるという事になります。

逆説的になりますが、法線ベクトルが並行という事は、この二つの関数が接している、接線を共有しているという事になります。

具体例に戻ります。ある制約条件  $g(x, y) = 0$  の元、ある一定数  $k$  に対して  $f(x, y) = k$  を満たす  $f(x, y)$  を考えると、こいつは  $(x, y)$  平面上の曲線となり、関数  $f$  の等高線ばいものになります。考えたい問題は  $f$  の  $g(x, y)$  の元での最大化なので、この等高線と制約条件の曲線とがちょうど接する時が極値になります。

何故なら、接しておらず交差している場合、制約条件を満たす範囲に  $f(x, y) > k, f(x, y) < k$  の領域が存在してしまうためです。それならそっちにずれた方が値が大きく (小さく) なりますよね。制約条件にぎりぎり引っかかる、かすめる時が極値になっているわけです。ここら辺は普通に微分の考え方ですね。制約条件を満たしつつ、与えられた関数に従う点を求めるのだから、接点を求めるわけですね。

### 2.1.3 例題

$g(x, y) = x + y - 1 = 0$  の制約条件のもと、関数  $f(x, y) = 1 - x^2 - y^2$  の最大値を考えてみます。制約条件の元の極値問題なので、ラグランジュの未定乗数法を使います。まずラグランジュ関数  $L$  は以下ようになります。

$$L(x, y, \lambda) = 1 - x^2 - y^2 - \lambda(x + y - 1) \quad (2.8)$$

続いて、 $x, y, \lambda$  のそれぞれについて偏微分すると、

$$\frac{\partial L}{\partial x} = -2x - \lambda = 0 \quad (2.9)$$

$$\frac{\partial L}{\partial y} = -2y - \lambda = 0 \quad (2.10)$$

$$\frac{\partial L}{\partial \lambda} = -x - y + 1 = 0 \quad (2.11)$$



が導けます．この連立方程式を解くと，関数  $f$  の制約条件  $g$  の元での最大値は  $(\frac{1}{2}, \frac{1}{2})$  であり，対応するラグランジュ乗数は 1 である事が分かりました．

簡単ですね！！

## 第3章 情報理論

近年は神経科学に情報理論の議論を輸入するのが流行りになっている気がします。脳波の解析もだし、情報処理の理論もそうだし、いろんなどこで見るのでとりあえず勉強。関係する研究は以下とか

- 自由エネルギー原理 [4]
- Phase Amplitude Coupling の評価. [5]
- 相互情報量
- トランスファーエントロピー

\*まだちゃんと引っ張ってきてない

### 3.1 エントロピー

はじめにエントロピーの考え方を導入しましょう。まずは離散確率変数  $x$  を考えます。観測者がこの変数に対するある値を観測したとき、どれだけの情報量、surprise を得られるのか。これを考える概念がエントロピーです。

直観的に、起きそうもない事象が得られたら情報量は大きいし、その逆も然りですよ。宝くじで1等があたるのはめちゃくちゃびっくりする、つまり情報量大きいけど、参加賞的なのもらっても何も思いません。つまり情報量は確率分布  $p(x)$  に依存していて、その値によって定まる単調な関数  $h(x)$  といえます。

また、2つの事象  $x, y$  を考えたとき、これらが独立なら両方を観測したときの情報量は別個に観測したときの情報量の和と等しい(式 3.1) はず。宝くじが当たる事による surprise によって、帰りに頭に雷が落ちてくる事によって生じる surprise が小さくみたいになることはないですよ？あるかも。ないって事にしてください。

よって、二つの事象が独立なのであれば情報量について以下の式 (3.1) が成り立ちます。

$$h(x, y) = h(x) + h(y) \quad (3.1)$$

次に、これらの事象の同時確率についても単純に積で求められます(式 3.2)。宝くじが当たり、かつ頭に雷が降ってくる確率です。

$$p(x, y) = p(x)p(y) \quad (3.2)$$

もし式 (3.1, 3.2) が分からないようなら基本的な確率が出来てないので、statistics.pdf で勉強してみてください。

さて、ここはちょっとテクいです。

この二つの関係から、「 $x$  と  $y$  の確率をかける操作をしたものに対して何らかの処理をしたものが、何らかの処理をした  $x$  と  $y$  の和になっている」ので、関数  $h(\cdot)$  は対数をとっている事が分かり（ほら、掛け算って対数だと足し算じゃん？）,

$$h(x) = -\log p(x) \quad (3.3)$$

がいえます。  $\log(x)$  は単調増加で、確率  $p(x)$  は常に 0 から 1 の範囲をとるため、  $h(x)$  の値を常に正にするため符号を反転させている事に注意です。ここで対数の底に 2 を採用するのが情報理論での一般の使い方で、その場合は  $h(x)$  の単位は bit になるようです。

これを使って、信号のサイズ、情報量 (bit) を算出してるわけですね。あとデータの圧縮なんかにも関係すると思いますがそこまでは知らないし触れませんし触れられません。

次に、この値を分布全体に適用する事を考えます。つまり、確率分布そのものが与える情報量です。その指標として、確率変数  $x$  の分布  $p(x)$  に関して  $h(x) = -\log p(x)$  の期待値をとることで、情報量の平均を定義します。確率変数の期待値の一般的な計算です。これも分からなければやはり statistics に行ってください。

$$H[x] = -\sum_x p(x) \log p(x) \quad (3.4)$$

式 (3.4) に定義する量をシャノンエントロピーといいます [6]。意外と簡単ですね。もっと難しいと思ってました。

ついでにこれを連続変数にすると微分エントロピー (式 3.5) が求まります。

$$H[x] = -\int p(x) \log p(x) dx \quad (3.5)$$

言うまでもないと思いますが、  $\sum_x p(x)$  も  $\int p(x)$  も 1 です。

さて、次に当然浮かぶ疑問は、どんな確率分布だとどんなエントロピーが算出されるのか、です。

ちゃんと数学的に証明することも出来るっぽいけど面倒だしそこにあんまり興味ないので、simulation してみます。

とりあえず一様分布で確認してみます。0.1 から 1 の値をとる、サンプル数 10 の一様分布 (図 3.1) のエントロピーを算出します (コード 3.1)。

Listing 3.1: エントロピーの計算

```
1 x = [0.1:0.1:1];  
2 px = zeros(1,10)+0.1;  
3 H = - sum(px .* \log(px));
```

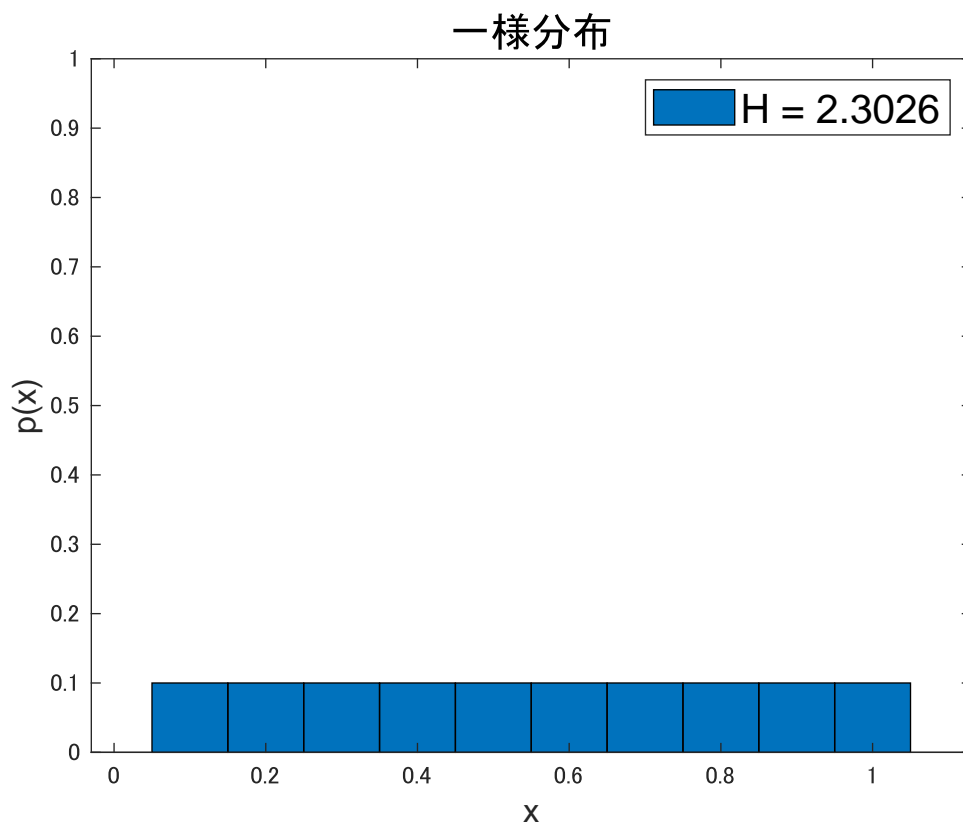


図 3.1: 0.1 が 10 この一様分布

エントロピーは 2.3026 でした。次に、同じ一様分布でもサンプル数が多いとどうなるのか試します。先ほどと同じ条件の、100 個のデータ (図 3.1) です。

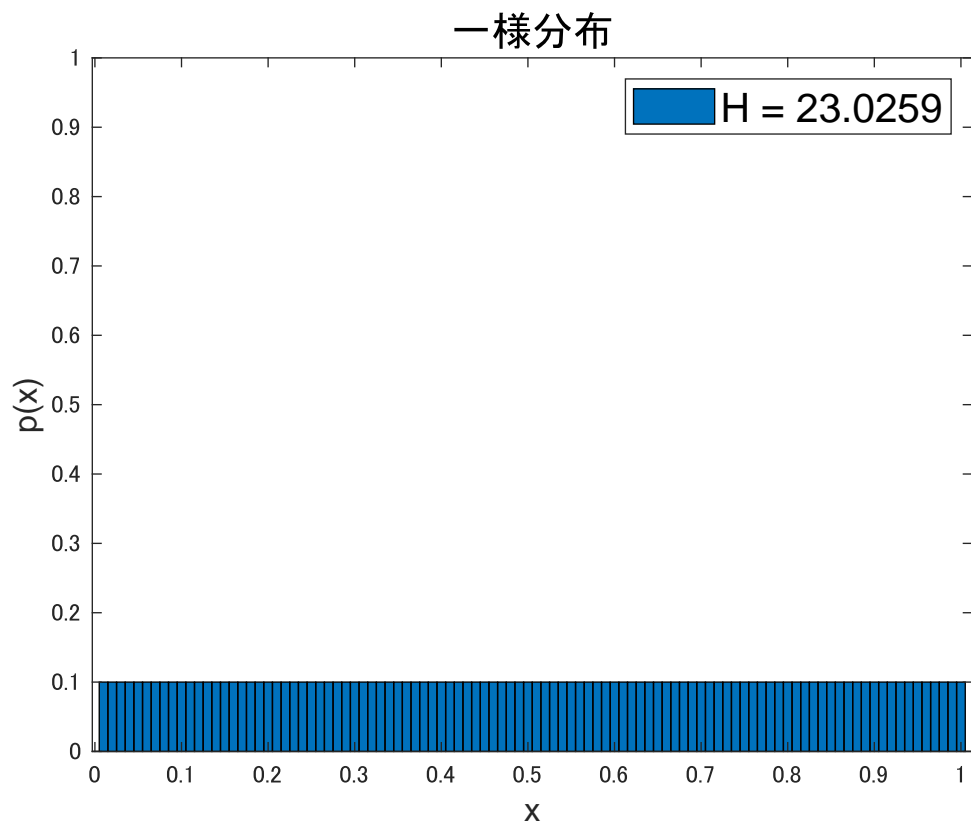


図 3.2: 0.01 が 100 この一様分布

エントロピーは 23.0259. 大きくなりましたね. データ数に応じてエントロピー自体は大きくなるばい  
です.

次に山を持たせた分布で見えます.

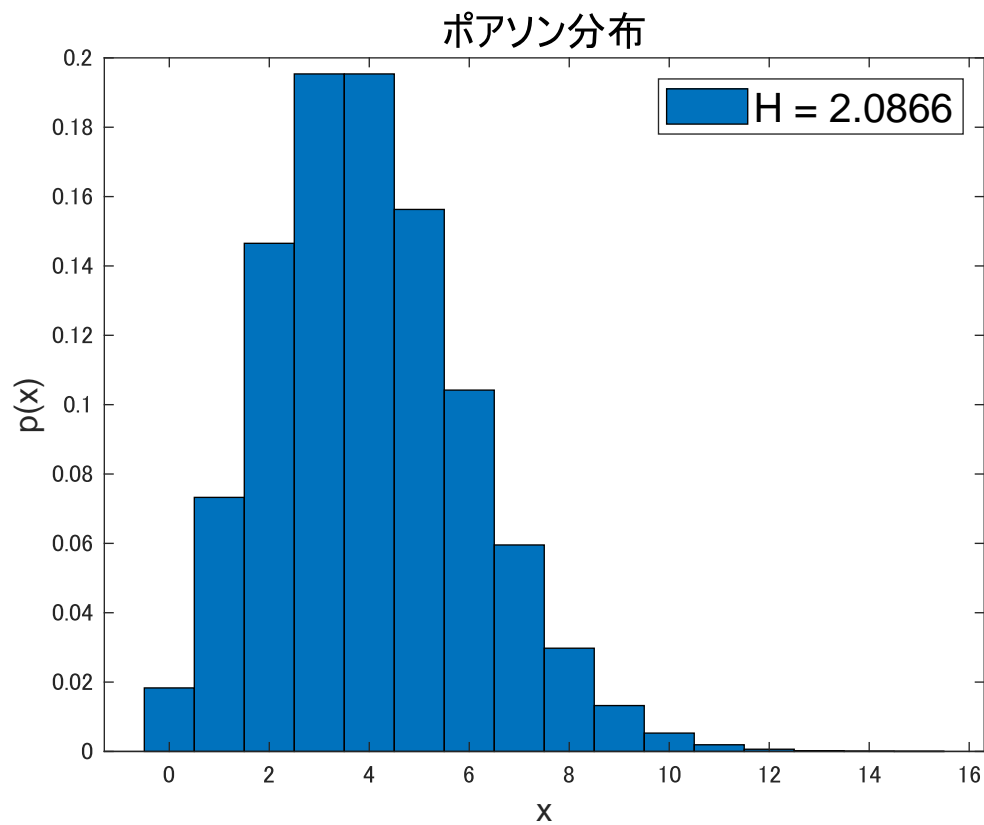


図 3.3: ポアソン分布のエントロピーの例

まずはポアソン分布 (図 3.1) でやってみました. 一様分布より小さな値を取っていますね. 次は正規分布に近い二項分布 (図 3.1) だとどうなるか見てみましょう.

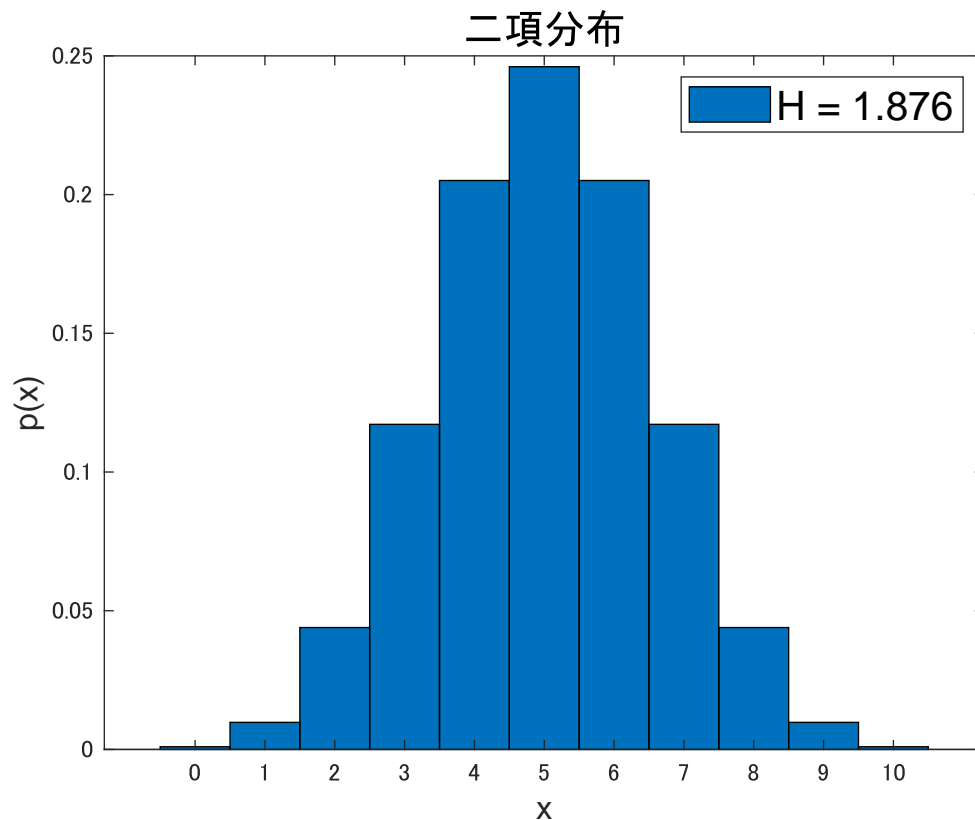


図 3.4: 二項分布のエントロピーの例

エントロピーは1.876. 一様分布に比べるとかなり小さく, ポアソン分布より小さいです. 本当はもっと山の位置動かしたり急峻にさせてみたりと比較したいですが, 飽きたので結論. (離散) エントロピーは確率分布  $p(x)$  が一様分布に近づく程大きくなり, 一様分布の時に最大になります.  $p(x) * \log(p(x))$  なのでまあ, 考えればそうかなって感じ. 証明は結構めんどくさいぽいです.

この性質から, 状態の予測が困難であるほどエントロピーが高い事になるので, 不確かさ (Uncertainty) の指標として用いられる事が多い理由がわかるかと思います.

ちなみに微分エントロピー, つまり連続の場合はガウス分布が最もエントロピー高いようです.

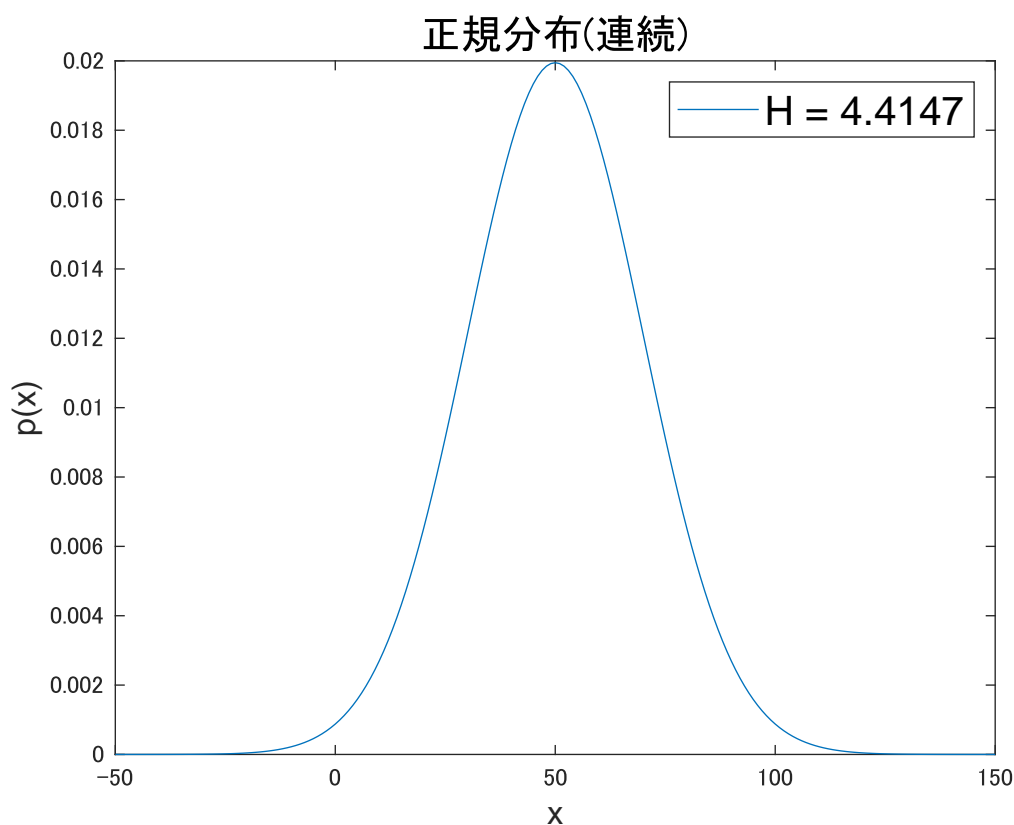


図 3.5: 正規分布のエントロピーの例

一応証明していくか！！

くくく、この資料を最初にまとめた時の僕には無理だったが、奴は四天王の中でも最弱... 今の僕にかかれれば造作ありません。

まず平均  $\mu$ , 分散が  $\sigma^2$  となる分布を考え、そいつらの中でエントロピーを最大化すれば良いので制約条件として、

- $\int p(x)dx = 1$
- $\int xp(x)dx = \mu$
- $\int (x - \mu)^2 p(x)dx = \sigma^2$

を与え、制約条件下での最大化問題なのでラグランジュ未定乗数法 (2.1) をもってくればワンパンで解けて

$$p(x) = \frac{1}{(2\pi\sigma^2)^{\frac{1}{2}}} \exp\left\{-\frac{(x - \mu)^2}{2\sigma^2}\right\} \quad (3.6)$$



と、たしかに正規分布が出てくるわけですね。正規分布の式の導出、統計編だと「なんとなく指数分布をいじると～」みたいに visual を近付けるみたいな方法でやりましたが正しくはこれで導くのが多いっぽいですね。

てなわけで連続の確率分布においてエントロピー最強は正規分布になるのでした。

他にもエントロピーについて見れる性質があって、まず  $p(x)$  が 0 は困ります。log にかけた時に計算がこわれるので。無限に吹っ飛びます。

あと、一様分布の比較から分かるようにデータ数が多いほどエントロピーも増大するっぽいですね。これもまあ普通に総和とってるんだから当たり前か？

まあこんな感じなので、エントロピーの値だけを見てなにかしらの議論をすることは難しそうですね。エントロピーを増大/減少させるといった変化の議論だとか、そういった使い方の方がナチュラルに見えます。

## 3.2 KL 距離

さて、このエントロピーがどんな事に使えるのか考えていきます。エントロピーは分布の特徴を表す量になっていたわけなので、これを使うと二つの分布の比較、なんてことも出来ることになります。直観的には、全く同じ特徴の分布同士ならそのエントロピーに差はないし、違う分布なら差がある、という感じです。

式にしてみましょう。まず、微分エントロピー  $(-\int p(x) \log p(x))$  は確率変数  $x$  の分布  $p(x)$  の元での期待値でした。なら、ここで新しい分布  $q(x)$  を考えたとき、仮にこの分布が同一 ( $p(x) = q(x)$ ) であれば、

$$\int p(x) \log q(x) dx - \int p(x) \log p(x) dx = 0 \quad (3.7)$$

が成り立つ事になります。同じ分布の元で考えた同じ確率変数の期待値だから、当たり前です。分布  $p(x)$  の元で見た  $q(x)$  の期待値が、分布  $p(x)$  の元で見た  $p(x)$  の期待値と等しい、ということです。逆にこの分布が異なるものであるほど、この計算の結果は大きな値を取る事になります。

てなわけで、この量をちゃんと正負の調整した上で、「分布  $p(x), q(x)$  の相対エントロピー、あるいはカルバック-ライブラー距離、またはカルバックライブラーダイバージェンス」として以下の式で定義します [6]。

$$\mathbb{D}_{KL}(p||q) = - \int p(x) \log q(x) dx - (- \int p(x) \log p(x) dx) \quad (3.8)$$

$$= - \int p(x) \log \frac{q(x)}{p(x)} dx \quad (3.9)$$

一寸ややこしく見えますが，基本的には式 (3.7) を  $\log$  について整理しただけです．  
簡単ですね．

KL 距離の性質ですが，まず  $\mathbb{D}_{KL}(p||q) \geq 0$  です．距離だし．  
等号が成り立つのは分布  $p(x), q(x)$  が等しいときのみ (図 3.2) です．

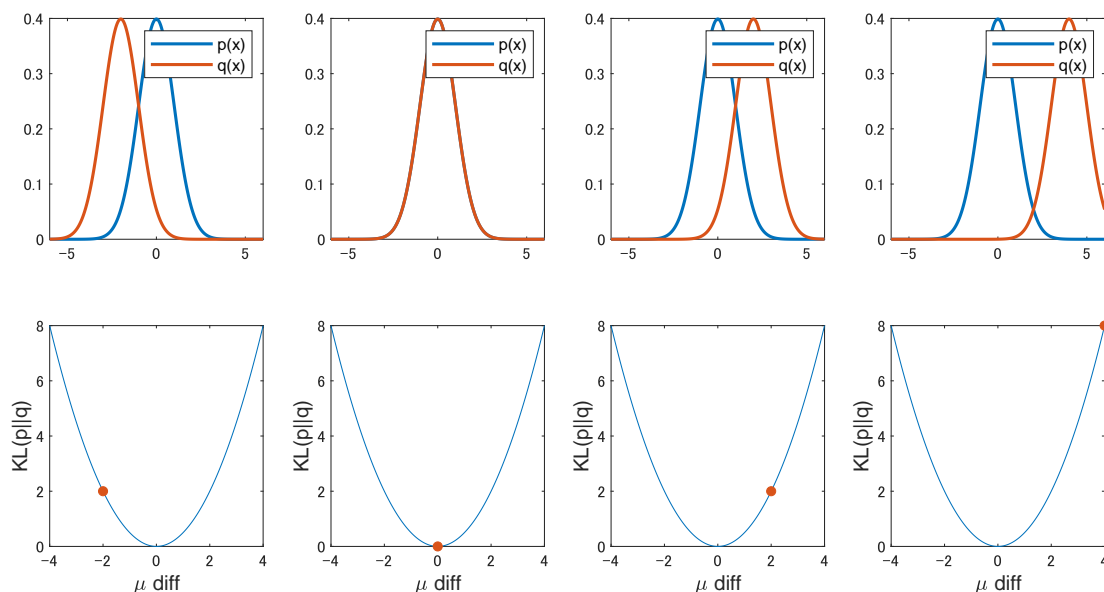


図 3.6: 二つの，標準偏差 1 で平均値の異なる正規分布間の  $\text{KLD}(p \rightarrow q)$

それから  $\mathbb{D}_{KL}(p||q) \neq \mathbb{D}_{KL}(q||p)$  なことにも気を付けてください．分布  $p(x)$  の元で見た  $q(x)$  の期待値と，分布  $q(x)$  の元で見た  $p(x)$  の期待値とは別物ですからね．

それから，対数なので KLD は以下のような表記のこともあります．一緒です． $\log$  の計算の性質を思い出してください．割り算は引き算です (?)

$$\mathbb{D}_{KL}(p||q) = \int p(x) \log \frac{p(x)}{q(x)} dx \quad (3.10)$$

留意してください．

あと、KL “距離” と日本語で呼んでいますが厳密には距離じゃないので注意が必要です. というのも、KLD は以下に示す距離の公理 [7] を満たしていないからです.

#### 距離の公理

2 点  $A, B$  が与えられたとき, 実数  $d(A, B)$  を与える規則で, 次の性質を満たすものを距離という.

- $d(A, B) \geq 0$
- $d(A, B) = 0 \Leftrightarrow A = B$
- $d(A, B) = d(B, A)$
- $d(A, B) + d(B, C) \geq d(A, C)$

このうち、KLD が満たしていないのは为什么呢？

そう、3 つめの対称性ですね！ $\mathbb{D}_{KL}(p||q) \neq \mathbb{D}_{KL}(q||p)$  でした.

あと 4 つ目、三角不等式も怪しいと思うんですね. 呼んでた資料とかでは特に対称性のとこだけネチネチと言われてましたが、三角不等式はどうなんでしょう？

直観的には微妙だと思ってて、だから KLD の値を単純に比較したりだとかの議論は出来ない気がしている.

じゃあ KLD はどう使うんだよって話ですが、最小化したい量として導入してるのが多い気がします.

つまり、予測分布を真の分布に近づけたい、だとかですね. この時に最小化する、分布と分布との距離として使われる量です.

あとは、一様分布と得られたデータ分布との距離を測る、なんて使い方もありました [5]. この場合は正規化的なのして、 $\mathbb{D}_{KL}(P||U)$  (where  $U$  is the uniform dist) を 0-1 の値にして使っていましたね.

いまのとこ個人的に分からないのは、 $\mathbb{D}_{KL}(A||B)$  と  $\mathbb{D}_{KL}(C||D)$  の値を比較した議論 (たとえば、 $A-B$  は  $C-D$  の 3 倍離れている!) なんてのは出来るのかなってところです.

ユークリッドなら自明に出来ると思うんですけど、これだとなんか出来ない気がする. 三角不等式も怪しいし.

どうなんでしょう？ 今後の課題になってます.

余談ですが、式 (3.8) の右辺第一項、 $-\int p(x) \log q(x) dx$  は交差エントロピー  $H(p, q)$  とも言います。分布  $p(x)$  の元で見た  $q(x)$  の期待値なので、 $p(x)$  の分布を想定したとき、 $q(x)$  がどれだけ予測しにくいのかも捉えられます。

これだけでも、交差エントロピー  $H(p, q)$  は正解値と推定値の比較なんかの用途で使えるようです。

じゃあ KLD と何が違うのか、というと、ここからは個人的な予想ですが...

問題なのは  $p(x)$  自体の分布が既にもってる情報量、つまり  $H(p)$  なんだと思います。

交差エントロピーは計算式をみれば分かるように、 $p(x)$  自体のエントロピーの影響を受けた数値になってしまうため、なんというか「どれくらい外れているか」の指標に使うにはフェアじゃない気がします。

なので交差エントロピーの値から、 $p(x)$  自体が持っているエントロピーの値を差し引いた量が知りたいわけですね。そうすると式 (3.8) は

$$\mathbb{D}_{KL}(p||q) = H(p, q) - H(p) \quad (3.11)$$

とも捉えられますね。あってるのかな？

### 3.2.1 他の分布間距離

確率分布同士の距離を測る指標は  $\mathbb{D}_{KL}$  だけでなく、他にも以下のようなものがあるっぽいです [8].

$$\chi^2(Q||P) := \sum_{i=1}^k \frac{(p_i - q_i)^2}{p_i} \quad \chi^2 \text{ 統計量} \quad (3.12)$$

$$L_1(Q||P) := \int |Q(x) - P(x)| dx \quad L_1 \text{ ノルム} \quad (3.13)$$

$$L_2(Q||P) := \int \{Q(x) - P(x)\}^2 dx \quad L_2 \text{ ノルム} \quad (3.14)$$

$$I_K(Q||P) := \int \{\sqrt{Q(x)} - \sqrt{P(x)}\}^2 dx \quad \text{ヘリンジャー距離} \quad (3.15)$$

$$\mathbb{D}(Q||P) := \int f\left(\frac{Q(x)}{P(x)}\right) Q(x) dx \quad \text{f-ダイバージェンス} \quad (3.16)$$

$$I_\lambda(Q||P) := \int \left\{ \left(\frac{Q(x)}{P(x)}\right)^\lambda - 1 \right\} Q(x) dx \quad \text{一般化情報量} \quad (3.17)$$

$$\mathbb{D}_{KL}(Q||P) := \int \log\left(\frac{Q(x)}{P(x)}\right) Q(x) dx \quad \text{KL 情報量} \quad (3.18)$$

どれがどんな時にどう使われるとかは調べてないです，でもこれでいくと上3つはよく見る気がする．でもまあ全体的に似てるポイですね．なんとなく哲学というか考え方はどれも似たりよったりな気がします．

### 3.3 条件付きエントロピー

エントロピーは式 (3.5) に示す量でしたが，これを条件付き確率  $p(x, y)$  に拡張して考えます．今  $x$  が既知である場合，同時分布  $p(x, y)$  について  $y$  を特定するための情報は  $-\log p(y|x)$  なので（これはいいよね？条件付き確率です），その合計は

$$H(y|x) = - \iint p(y, x) \log p(y|x) dy dx \quad (3.19)$$

で表され，これは条件付きエントロピーといいます [6][8]．さらにこれを使えば

$$H(x, y) = H(y|x) + H(x) \quad (3.20)$$

と書けますね！エントロピーは対数なので，確率の乗法を意味しています．つまり  $x$  と  $y$  の同時分布を記述する情報量は， $x$  単体の情報量と  $x$  が与えられた元での  $y$  の情報量との和になるわけですね．

### 3.4 相互情報量

KL 距離は分布と分布の距離を測れる便利な指標でした．

これを使った、これまた便利そうな指標の一つが相互情報量です。二つの変数  $x, y$  を考えて、こいつらの同時分布  $p(x, y)$  が得られたとします。この時、この変数 2 人の間にどんな関係があるのか確認したくなりますよね。他人なのか、それとも親密な関係なのか... まあつまり独立かどうかです。

さて、KL 距離はこの独立性の検証的な使い方が可能で、それがまさに相互情報量の計算です。式 (3.21) を見た方が早いでしょう [8]。

$$MI(x, y) = H(x) + H(y) - H(x, y) \quad (3.21)$$

$x$  と  $y$  が独立であった場合の同時確率の情報量と独立でないときの情報量の離れ具合を見るわけですね。例のごとく対数なので、要は  $p(x)p(y)$  と  $p(x, y)$  です。これは KLD を使えば式 (3.22) のように表せます。

$$MI(x, y) := \mathbb{D}_{KL}(p(x, y) || p(x)p(y)) = - \iint p(x, y) \log \frac{p(x)p(y)}{p(x, y)} dx dy \quad (3.22)$$

あら簡単。変数  $x, y$  の同時分布と周辺分布積との KLD を見るだけです。KLD なので、両者が同じ、つまり  $x$  と  $y$  が独立である時に限って 0 になる量ってわけですね。

てことは、 $x$  と  $y$  がずぶずぶの関係であるほど値が大きくなるわけだから、 $y$  の値を知る事によって  $x$  の不確実性が減った度合を表すと言えます [6]。

KLD 同様、符号反転で以下の表記 (式 3.23) もあります。

$$MI(x, y) := \iint p(x, y) \log \frac{p(x, y)}{p(x)p(y)} dx dy \quad (3.23)$$

式 (3.22) を見れば分かりますが、相互情報量は対称性を持っており、 $MI(x, y) = MI(y, x)$  です。なのでどっちがどっちにどの程度依存しているみたいな議論までは出来ません。

### 3.5 Transfer Entropy

相互情報量は 2 つの確率変数間の相互依存度のような指標でしたが、どっちがどっちに依存している関係なのか、みたいな因果性まで見れたらカッコいいですよね。そう、それが Transfer Entropy です。

え？ いや例えば領域 A の活動が元に領域 B の活動が起きてるとか言いたいじゃないですか。

少し話はそれますが、やはり我々神経科学者にとって 21 世紀入ってからの大きな問題の一つが因果性の検証だと思うんですよね。いろんなデータが実験的に得られているけど、基本的には相関でしかなくて。「So what?」なんですよ。なので電気刺激や磁気刺激、古いけど破壊法だったりオプトジェネティクス、最近ではニューロフィードバックなんて手法を使って、追加実験的に因果性の検証をしていくのが流行りになっています。

でも面倒なので、どうせなら計測した脳活動だけで因果関係まで言えたら嬉しいよねってモチベーションで考えられるのが effective connectivity とかで、Transfer entropy はまさに Effective connectivity の一種です。

閑話休題。

本題ですが、Transfer Entropy の基本的な考え方はこんな感じっぽいんです。まず、因果といってもあくまで情報理論的な観点で見た因果「y の結果、x が起きた」です。ちょっと緩いわけですね。実際、まじな因果の検証とか無理では？とも思いますが。

この「y の結果～」という表現からも分かるように、概念の背景に時間軸がひそんでいます。相互情報量はある同時刻の活動のみを比較するような処理をしていましたが、ここに経時的な変化の考慮も踏まえ、経時的な変化における確率変数同士の依存度を見ていく必要があります。

大丈夫ですかね？ここまでは前提です。

ではコアになる考え方ですが、もし仮に y の結果として x が起きているのであれば、x 単体の時系列を使った  $x_{t+1}$  の予測よりは x と y の値を使った  $x_{t+1}$  の方が精度が高いですよ。式にするとこう。

$$\begin{aligned} p(x_{t+1}|\mathbf{x}_t) \\ p(x_{t+1}|\mathbf{x}_t, \mathbf{y}_t) \end{aligned}$$

こいつらの値を比較したとき、 $x_{t+1}$  の値が  $\mathbf{y}_t$  に全く依存していないのであれば、両者に差はなく等しくなるはずですね。

ふう。

ここまで来たらあとは相互情報量の時と一緒に。KLD を使って

$$T_{y \rightarrow x} := \mathbb{D}_{KL}(p(x_{t+1}|\mathbf{x}_t, \mathbf{y}_t) || p(x_{t+1}|\mathbf{x}_t)) = - \sum p(x_{t+1}|\mathbf{x}_t, \mathbf{y}_t) \log \frac{p(x_{t+1}|\mathbf{x}_t)}{p(x_{t+1}|\mathbf{x}_t, \mathbf{y}_t)} \quad (3.24)$$

で定義される量を,  $y$  から  $x$  への Transfer entropy とします. 例によって符号反転で

$$T_{y \rightarrow x} = \sum p(x_{t+1}|\mathbf{x}_t, \mathbf{y}_t) \log \frac{p(x_{t+1}|\mathbf{x}_t, \mathbf{y}_t)}{p(x_{t+1}|\mathbf{x}_t)} \quad (3.25)$$

とも表します. この量は  $\mathbf{y}$  を知る事によって減少した  $x_{t+1}$  の不確かさです. 相互情報量とは似ているようで異なります.

また例のごとく式から分かるように  $T_{y \rightarrow x} \neq T_{x \rightarrow y}$  です. このことから, Transfer Entropy には向きが含まれており, したがって因果性の議論に使えるわけですね [9].

因果性といえば他に有名なのは Granger Causality ですが, こいつらの比較はまた今度気が向いたらやってみます.



## 第4章 信号処理

信号処理の基本は Analysis.pdf の方に載せていますが，理解が必須ではない，というか少し難しいものはこちらにあげていきます．

### 4.1 独立成分分析

Independent Component Analysis (ICA) と呼ばれる多変量解析の手法です．だいたい 90 年代頃に確立されました．まずどういったものかと言うと，観測された信号を独立な複数の信号の線形な重ね合わせとして再表現するものです．

といってもよくわからないと思うので，我々も実は日頃から ICA をやっていますよという話からしましょう．皆さん毎日夜には駅前の居酒屋やバーで楽しくお酒を嗜んでいる事かと思いますが，実はこの時我々の脳は独立成分分析をしているのです．

飲み屋では多くの人間が同時に声を発し，食器の音やどこかのコールの音，厨房の音と無数の音が同時に我々の耳＝脳に送られてきます．しかしどういうわけか，その音がどの誰の声か，厨房の音なのか食器の音なのか判断できますよね．つまり，与えられた音の時系列データ（これは一つのデータに重ねられて聴こえてる）を複数の信号源に分解している，そう，独立成分分析をしているわけです．

我々は特に意識せずとも当たり前のように出来るこの作業ですが，実は PC にやらせるととんでもなく難しかったのです．それをどうにか実現しましょうという事で出来たのが現在の ICA です．

脳波の研究に何の役に立つのかですが，端的に言えばノイズの除去や脳活動の特徴抽出です．脳波はだいたい 64ch の電極を使って計測するわけですが，そこには残念ながら眼球運動由来の電位や筋肉の電位など，脳波ではない成分も乗ってしまっています．これらは脳波に比べて電位も大きいので，脳波の解析の際に邪魔以外のなにものでもなく，親の仇のように憎むべき存在です．

もう分かるかと思いますが，“脳波” 信号を分解し，真の脳波信号と眼電，筋電，などに分解してあげるのに使えるのが独立成分分析です．実用では，こうして分解したデータのうち眼電や筋電由来と思われる成分だけ取り除いてあげれば綺麗に脳波だけ見れるわけですね．すくなくとも目的はそんなとこです．

長くなりましたが日本語はこれくらいにして，早速数理に入りましょう．まず元の時系列信号を  $\mathbf{x}(t)$  とします．これが，初めに仮定より線形な ( $N \geq 2$ ) 個の信号源 ( $\mathbf{s}(t) = [s_1(t), s_2(t), \dots, s_N(t)]$ ) からの信号の重ね合わせによって  $N$  個のセンサに観測されているとします．ここで信号源とセンサーの個数が一緒

なのに違和感を覚える人もいると思いますが、とりあえず置いてください。実際脳波の ICA も基本的にチャンネル数と同じ数に分解します。するとこの関係は

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t)^T \quad (4.1)$$

と表せます。ここで  $\mathbf{A}$  は  $N \times N$  の係数行列で、 $a_{ij}$  は  $i$  番目のセンサで観測される  $j$  番目の信号源からの信号の係数です。この値が大きいのは信号源の影響を強く受け、小さいのはあまり受けていない事を意味します。

ではここで、今回の問題で求めたいのは  $\mathbf{s}(t)$  ですよね。なので  $\hat{\mathbf{s}}(t)$  的な行列として  $\mathbf{y}(t)$  を定義し、

$$\mathbf{y}(t) = \mathbf{W}\mathbf{x}(t) \quad (4.2)$$

という式をたてます。実数の分離行列  $\mathbf{W}$  を元信号にかけて  $\mathbf{y}(t) = [y_1(t), y_2(t), \dots, y_N(t)]$  で表される分離信号に分解する事を考えます。分離行列も  $N \times N$  の成分をもっていて、それぞれのセンサの値をどれだけ反映させて分離信号を生成するのか、を分離信号の数だけ行います。この  $\mathbf{y}(t)$  が綺麗に独立するように  $\mathbf{W}$  を更新していくのが独立成分分析の手順です。

ではその独立性をどう表すか、というところですが、 $\mathbb{D}_{KL}$  を使います。正確には相互情報量です。相互情報量は、複数の変数があったときにその同時分布と周辺分布の積との間の KL 距離を測るものでしたね。両者が全く同じ分布である時、0 になるものだったので、相互情報量が 0 になる時は変数が全て独立である、という事を表せるのでした。これを使います。

$$p(\mathbf{y}) = p(y_1, y_2, \dots, y_N) \quad \text{同時分布} \quad (4.3)$$

$$p(\mathbf{y}) = \prod_{i=1}^N p(y_i) \quad \text{周辺分布} \quad (4.4)$$

こいつらの KL 距離が相互情報量だから

$$\mathbb{D}_{KL}(\mathbf{y}) = \int p(\mathbf{y}) \log \frac{p(\mathbf{y})}{\prod_{i=1}^N p(y_i)} d\mathbf{y} \quad (4.5)$$

と表せます。この時、式 (4.5) が 0 になるのは  $p(y_1, y_2, \dots, y_N) = \prod_{i=1}^N p(y_i)$  が成り立つ、つまり独立な信号源として仮定した  $\mathbf{y}$  がちゃんと独立である時になります。よって最小化しましょう。

$$\begin{aligned} \hat{\mathbf{W}} &= \arg \min_{\mathbf{W}} \mathbb{D}_{KL}(\mathbf{y}|\mathbf{W}) \\ &= \arg \min_{\mathbf{W}} \int p(\mathbf{y}|\mathbf{W}) \log \frac{p(\mathbf{y}|\mathbf{W})}{\prod_{i=1}^N p(y_i|\mathbf{W})} d\mathbf{y} \end{aligned} \quad (4.6)$$

式 (4.5) が 0 になるように分離行列  $\mathbf{W}$  を逐次更新していけばいいわけですね。最小化する逐次更新の仕方はいくつかあって、最有力?なのが勾配法を用いるものです。勾配法についてまだちゃんと勉強できていないので今は式だけ。

$$\mathbf{W} \leftarrow \mathbf{W} - \eta \mathbf{E}[\phi(\mathbf{y})\mathbf{y}^T - I]\mathbf{W}^{-T} \quad (4.7)$$

この更新式 (4.6) に従って評価式 (4.5) を更新していけば相互情報量を最小化できて、無事に元の  $\mathbf{x}(t)$  を独立成分  $\mathbf{y}(t)$  を取り出すことが出来るわけですね。正直最後だけまだ分かん。でも最小化したってのは分かるのでとりあえずヨシ！

## 4.2 正準相関分析

PCA とかの仲間的なやつです。与えられたデータ群に含まれる共通成分を抜き出す目的で用いられます。では他のそうした手法と何が違うのか、どうやって計算されるのかをまとめていきます。

### 4.2.1 定義と用法

正準相関分析 (Canonical Correlation Analysis: CCA) は、二つのデータ集合の間に含まれる共通成分を抜き出します。三つ以上の場合にはちょっと難しくなりますが例のごとく一般 CCA みたいなものがあるのですがここでは置いておきます。

とにかく、データ集合  $\mathbf{x}, \mathbf{y}$  が与えられた時、この二つの集合に共通して含まれる成分をどうにかして取り出すのが CCA の仕事です。

ここまで聞いただけだと特にすごさが分からないと思いますが、こいつのすごいのはそれぞれのベクトルの成分は対応している必要がないし、そもそも次元も異なって良いです。つまり、 $\mathbf{x}$  が 64 次元のベクトルなのに対し  $\mathbf{y}$  が 16 次元とかでも良いわけですね。成分同士まったく対応を考えてないからこそできる技です。

脳波なら、64 チャンネル脳波計で取ったデータと 16 チャンネル脳波計で取ったデータから共通の成分を抜き出すといったような事に使えるわけですね。便利です。

ではこれを、数学的にどうやっていくのか考えます。結論から言うと、CCA は与えられたそれぞれの観測データを線形変換した値間の相関係数が最も高くなるようにそれぞれの線形変換のパラメータを求める手法です。以下に数式を使って考えていきます。

### 正準相関分析

$m$  次元の観測ベクトル  $\mathbf{x}$  と  $n$  次元の観測ベクトル  $\mathbf{y}$  がある時、それぞれ  $T$  個のサンプルがあった場合に得られる  $(x_1, y_1) \dots (x_T, y_T)$  について、以下に示す条件を満たす  $\mathbf{a}, \mathbf{b}$  によってそれぞれを線形変換する。

$$u(\mathbf{x}) = \mathbf{a}^\top \mathbf{x}, v(\mathbf{y}) = \mathbf{b}^\top \mathbf{y} \quad (4.8)$$

$$\mathbf{a}, \mathbf{b} = \arg \max_{\mathbf{a}, \mathbf{b}} \mathbf{a}^\top \mathbf{V}_{xy} \mathbf{b} \quad (4.9)$$

$$s.t. \quad \mathbf{a}^\top \mathbf{V}_{xx} \mathbf{a} = \mathbf{a}^\top \mathbf{V}_{yy} \mathbf{a} = 1$$

#### 4.2.2 導出

さて、見ていきます。ここで観測ベクトルはたとえば試行 1 と試行 2 の脳波データとか、被験者 1 と被験者 2 の脳波データとか、そんな感じです。それらの行列に対し、何らかの線形変換をして新しく  $\mathbf{u}, \mathbf{v}$  を求めています。こいつらが何かはとりあえず置いておいて、その線形変換に用いるパラメータ  $\mathbf{a}, \mathbf{b}$  を最適化するのが CCA の仕事、というわけですね。

では、その最適化とは具体的に何かというと、線形変換の結果得られる  $\mathbf{u}, \mathbf{v}$  の相関係数が最大になるように  $\mathbf{a}, \mathbf{b}$  を選ぶという事になっています。これが  $\arg \max$  で行われている処理ですね。ただし、それだけではなく制約条件として、何やら変な奴らが 1 になるという制約を課しています。

まず、相関係数を求めるという話ですがここで相関係数をどうやって求めるのかというと、以下のようになります。

$$\rho(\mathbf{a}, \mathbf{b}) = \frac{E[u(\mathbf{x})v(\mathbf{y})]}{\sqrt{E[u(\mathbf{x})^2]} \sqrt{E[v(\mathbf{y})^2]}} \quad (4.10)$$

$$= \frac{E[(\mathbf{a}^\top \mathbf{x})(\mathbf{b}^\top \mathbf{y})]}{\sqrt{E[(\mathbf{a}^\top \mathbf{x})(\mathbf{a}^\top \mathbf{x})]} \sqrt{E[(\mathbf{b}^\top \mathbf{y})(\mathbf{b}^\top \mathbf{y})]}} \quad (4.11)$$

$$= \frac{\mathbf{a}^\top E[\mathbf{x}\mathbf{y}^\top] \mathbf{b}}{\sqrt{\mathbf{a}^\top E[\mathbf{x}\mathbf{x}^\top] \mathbf{a}} \sqrt{\mathbf{b}^\top E[\mathbf{y}\mathbf{y}^\top] \mathbf{b}}} \quad (4.12)$$

$$= \frac{\mathbf{a}^\top \mathbf{V}_{xy} \mathbf{b}}{\sqrt{\mathbf{a}^\top \mathbf{V}_{xx} \mathbf{a}} \sqrt{\mathbf{b}^\top \mathbf{V}_{yy} \mathbf{b}}} \quad (4.13)$$

$$(4.14)$$

ここで、式 (4.12) から式 (4.13) への変形では  $E[\mathbf{x}, \mathbf{y}^\top] = \mathbf{V}_{xy}$ ,  $E[\mathbf{x}\mathbf{x}^\top] = \mathbf{V}_{xx}$ ,  $E[\mathbf{y}\mathbf{y}^\top] = \mathbf{V}_{yy}$  としています。これらは、分散共分散行列で得られるそれぞれの成分です。

おっと、説明を忘れていました。ここで用いる観測データですが、どちらも先にサンプル平均が0になるように調整しておきます。つまり全体から平均を引けば良いです。

さて、そんな分散の概念を導入する事で式 (4.13) の形に相関係数を整形できました。

ここで、少し頭のいい操作をする事で更にこいつを簡略化させる事で、CCA の式が導出できます。分母と分子、それぞれ  $\mathbf{a}$  と  $\mathbf{b}$  がいますよね。これ、結局  $\mathbf{a}$  と  $\mathbf{b}$  で打ち消しあうので、こいつらにそれぞれどんな正の数をかけても  $\rho(\mathbf{a}, \mathbf{b})$  の値は変わらない事が分かります。

これを利用しちゃいます。何をかけても良いんだから、 $\mathbf{a}^\top \mathbf{V}_{xx} \mathbf{a} = \mathbf{a}^\top \mathbf{V}_{yy} \mathbf{a} = 1$  となるように標準偏差の値を整形しちゃいましょう。すると、分母が全部消えるので結局、CCA が解くべき問題 (相関係数  $\rho(\mathbf{a}, \mathbf{b})$  の最大化) は、以下の問題に帰着します。

$$\mathbf{a}, \mathbf{b} = \arg \max_{\mathbf{a}, \mathbf{b}} \mathbf{a}^\top \mathbf{V}_{xy} \mathbf{b} \quad \text{s.t.} \quad \mathbf{a}^\top \mathbf{V}_{xx} \mathbf{a} = \mathbf{a}^\top \mathbf{V}_{yy} \mathbf{a} = 1 \quad (4.15)$$

先程いれた仮定、標準偏差が1になるという制約条件を満たしつつ、相関係数を最大にする線形変換をする  $\mathbf{a}, \mathbf{b}$  を求めるという事ですね。制約付き2次関数の最大化問題です。

では、こいつをどうやって解いていくかというところで、ラグランジュの未定乗数法が出てきます。ここで使うために最初の章にまとめる羽目になりました。分からない人は参照してください。とにかく、ラグランジュの未定乗数法を使うと式 (4.15) は以下の関数の極値問題になります。

$$L(\mathbf{a}, \mathbf{b}, \lambda_a, \lambda_b) = \mathbf{a}^\top \mathbf{V}_{xy} \mathbf{b} + \lambda_a (1 - \mathbf{a}^\top \mathbf{V}_{xx} \mathbf{a}) + \lambda_b (1 - \mathbf{b}^\top \mathbf{V}_{yy} \mathbf{b}) \quad (4.16)$$

ということで、ラグランジュ関数 (4.16) を  $\mathbf{a}, \mathbf{b}$  で微分して0とおくと、

$$\mathbf{V}_{xy} \mathbf{b} = 2\lambda_a \mathbf{V}_{xx} \mathbf{a}, \quad \mathbf{V}_{xy}^\top \mathbf{a} = 2\lambda_b \mathbf{V}_{yy} \mathbf{b} \quad (4.17)$$

となる。これを行列の形でまとめると以下のようになります。

$$\begin{bmatrix} 0 & \mathbf{V}_{xy} \\ \mathbf{V}_{xy}^\top & 0 \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} = \lambda \begin{bmatrix} \mathbf{V}_{xx} & 0 \\ 0 & \mathbf{V}_{yy} \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} \quad (4.18)$$

一般化固有値問題に帰着できました。ここで、 $\lambda_a = \lambda_b = \lambda$  としたのは、以下の根拠によります。式 (4.17) のそれぞれに左から  $\mathbf{a}^\top, \mathbf{b}^\top$  をかけてやると

$$\mathbf{a}^\top \mathbf{V}_{xy} \mathbf{b} = 2\lambda_a \mathbf{V}_{xx} \mathbf{a} \quad (4.19)$$

$$= 2\lambda_a \quad (4.20)$$

$$\mathbf{b}^\top \mathbf{V}_{xy}^\top \mathbf{a} = \mathbf{a}^\top \mathbf{V}_{xy} \mathbf{b} \quad (4.21)$$

$$= 2\lambda_b \mathbf{V}_{yy} \mathbf{b} \quad (4.22)$$

$$= 2\lambda_b \quad (4.23)$$

$$\therefore \lambda_a = \lambda_b \quad (4.24)$$

$\mathbf{V}_{xx} \mathbf{a}, \mathbf{V}_{yy} \mathbf{b}$ が消えるのは、制約条件より1としていたからです。ということで固有値は等しいわけですが、更に言うとこれらの固有値 $\lambda$ は得られる $u(x), v(y)$ の相関係数 $(\rho(\mathbf{a}, \mathbf{b}) = \mathbf{a}^\top \mathbf{V}_{xy} \mathbf{b})$ であるという事も分かります。

つまり、相関係数の最大化は最も大きな固有値を選ぶ事に相当し、CCAのタスクは最大の固有値に対応する固有ベクトルによって定まる $\mathbf{a}, \mathbf{b}$ を求めるという事にまとまりました。

### 4.2.3 固有値問題へ

さて、もうほぼ終わりですがあとは、自分で実装する時に簡単なように一般化固有値問題からただの固有値問題に落とし込みます。

まず、式(4.17)の $\mathbf{V}_{xy} \mathbf{b}$ について、左から逆行列 $\mathbf{V}_{xx}^{-1}$ をかける事で以下のような操作をします。

$$\mathbf{V}_{xx}^{-1} \mathbf{V}_{xy} \mathbf{b} = \mathbf{V}_{xx}^{-1} 2\lambda_a \mathbf{V}_{xx} \mathbf{a} \quad (4.25)$$

$$= \lambda \mathbf{a} \mathbf{a} = \frac{\mathbf{V}_{xx}^{-1} \mathbf{V}_{xy} \mathbf{b}}{\lambda} \quad (4.26)$$

これで、 $\mathbf{a}$ について $\mathbf{b}$ と $\lambda$ を使って表せたので、式(4.17)のもう一方の式の方に代入します。すると

$$\begin{aligned} \mathbf{V}_{xy}^\top \mathbf{a} &= 2\lambda_b \mathbf{V}_{yy} \mathbf{b} \\ \mathbf{V}_{xy}^\top \frac{\mathbf{V}_{xx}^{-1} \mathbf{V}_{xy} \mathbf{b}}{\lambda} &= \lambda \mathbf{V}_{yy} \mathbf{b} \\ \mathbf{V}_{xy}^\top \mathbf{V}_{xx}^{-1} \mathbf{V}_{xy} \mathbf{b} &= \lambda^2 \mathbf{V}_{yy} \mathbf{b} \\ \mathbf{V}_{yy}^{-1} \mathbf{V}_{xy}^\top \mathbf{V}_{xx}^{-1} \mathbf{V}_{xy} \mathbf{b} &= \lambda^2 \mathbf{b} \\ \mathbf{V}_{yy}^{-1} \mathbf{V}_{yx} \mathbf{V}_{xx}^{-1} \mathbf{V}_{xy} \mathbf{b} &= \lambda^2 \mathbf{b} \end{aligned} \quad (4.27)$$

というように変形する事が出来ます。同様に $\mathbf{a}$ についても同じ作業をする事で、

$$V_{xx}^{-1}V_{xy}V_{yy}^{-1}V_{yx}\mathbf{a} = \lambda^2\mathbf{a} \quad (4.28)$$

$$V_{yy}^{-1}V_{yx}V_{xx}^{-1}V_{xy}\mathbf{b} = \lambda^2\mathbf{b} \quad (4.29)$$

とすることが出来ます。まとめると、線形変換  $\mathbf{a}, \mathbf{b}$  はそれぞれ、式 (4.28) の固有値問題を解くことで求める事が出来ます。

#### 4.2.4 例題

たとえば、以下のような階段状に変化する二つの信号を、ノイズの中から検出する問題を考えます。

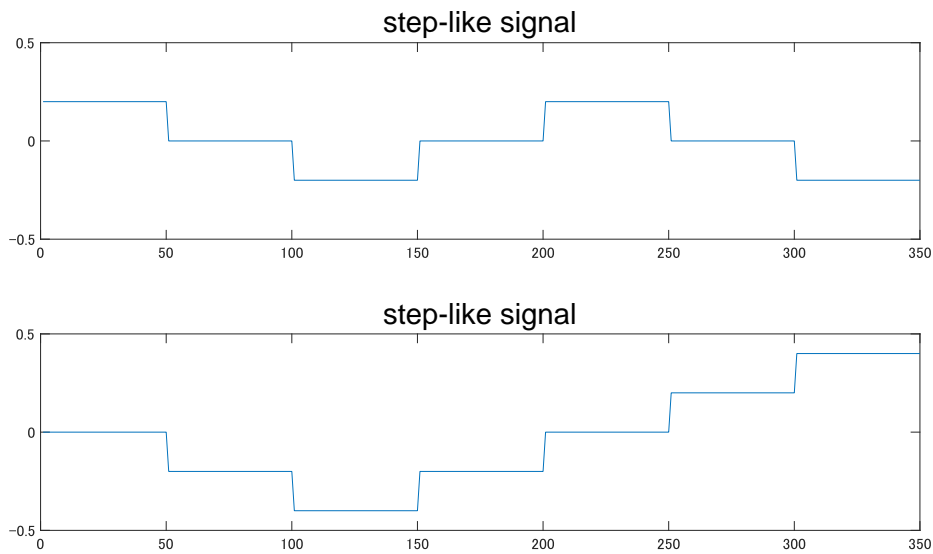


図 4.1: てきとうに作った信号。脳波の背景には実はこんなものが隠れていた！という仮定のもと以降を進めます。

次に、こいつらをノイズに埋めます。CCA は二つのデータ群の間で行う処理だったので、こいつらをそれぞれ電極 1,2 の真の成分として、これが 2 回の試行 (X,Y) で得られたと仮定して次のようなデータ群を生成します。

## Original signals

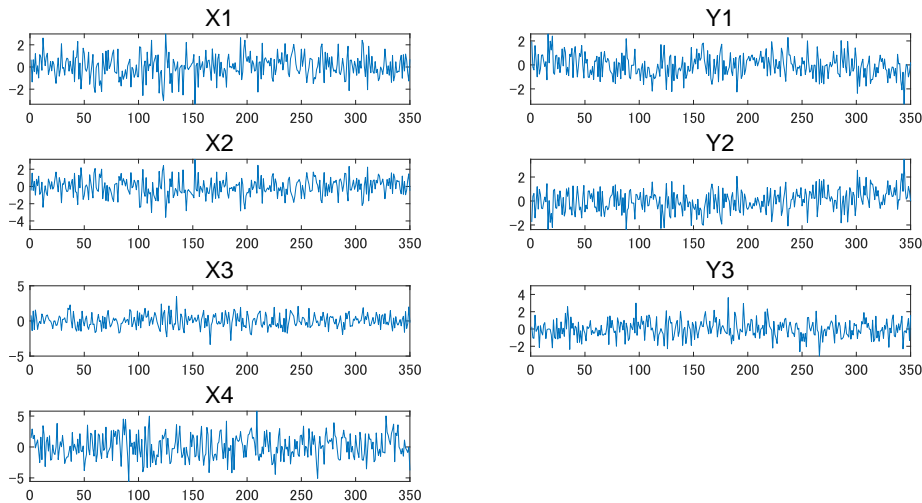


図 4.2: それっぽく作ってみた信号群. X1 と Y1, X2 と Y2 の組み合わせにはそれぞれ共通した階段状の信号が背景に含まれている. X3,X4,Y3 はただのノイズ.

ここで、CCA にやってほしいのは X1,X2,Y1,Y2 に含まれている、先程生成したような共通成分を抜き出す作業です. X1 と Y1, X2 と Y2 がそれぞれ背景に共通の信号が含まれている組み合わせです. ちなみにそれ以外の X3 とかはてきとうに作ったノイズなので、全く共通成分を含んでいません.

また、確認ですがデータ群の次元数は共通してなくても良かったんですね. 今回は X は 4 次元、つまり脳波なら 4 チャンネル、に対して Y は 3 次元でやってみます.

さて、CCA にかけた結果がこちらです.



## Generated signals

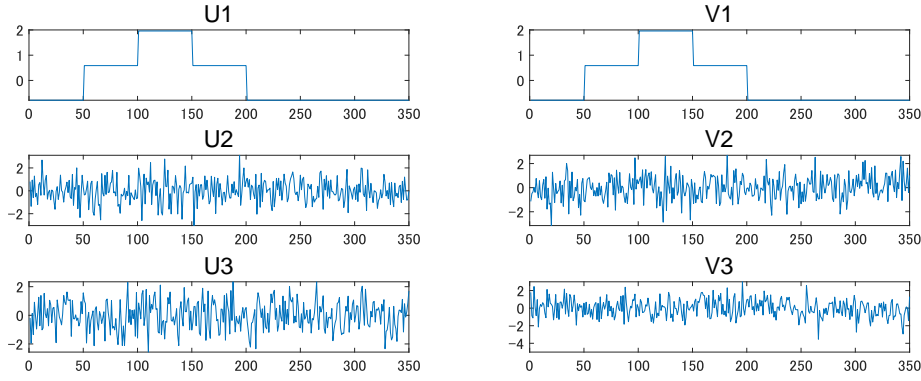


図 4.3: CCA の結果生成された正準相関変数.  $U$  は  $X$  の,  $V$  は  $Y$  の線形変換によって為されている. このうち, 同じ数字同士の相関係数が高い程,  $X$  と  $Y$  に共通に含まれていた成分であるという事になる. 今回だと 1 に定義した階段 2 つの足し合わせのようなものが現れ, 2 以降はノイズが並んでいる.

たしかに, 第一正準相関変数として階段状の成分を抜き出せていますね. 要求を満たしています. さて, 他にも, 知りたいのは第一正準相関変数に線形変換する際に,  $X$  と  $Y$  のうちそれぞれの成分を重視しているのかを調べたくなります. 脳波であれば, 64 チャンネルのうちどの電極群がこのコンポーネントに対応しているのかなんて事を調べるわけですね.

これは線形変換する際の固有ベクトルを確認すれば分かります. 見てみると,  $A$  と  $B$  はそれぞれ

$$A = \begin{bmatrix} -3.4271 & 0.5217 & 0.2541 \\ -3.4271 & -0.3639 & 0.1660 \\ 0.0000 & 0.2522 & 0.4342 \\ 0.0000 & 0.1141 & -0.4466 \end{bmatrix} \quad (4.30)$$

$$B = \begin{bmatrix} -3.4271 & -0.2025 & 0.5748 \\ -3.4271 & 0.5143 & -0.2501 \\ -0.0000 & 0.6879 & 0.7074 \end{bmatrix} \quad (4.31)$$

となっています. それぞれ, 行は  $X$  と  $Y$  の何行目がどれだけ寄与しているか, 列は第何正準相関変数についてかを表しています. こう見ると, 第一正準相関変数である  $U1$ ,  $V1$  はそれぞれ,  $X$  と  $Y$  の第一, 第二成分 (チャンネル) を使って算出されていて, 他のデータは見えてない事が分かります. 勿論, 共通成分をいれたのは第一第二の成分のみだったので, これは正しい挙動です.

これ，脳波で考えるならたとえば前頭 (Fp1, Fp2, Fz, F4) と頭頂 (Cz, C3, C4...) あたりの寄与が大きいので，こいつらの活動が共通している！！なんて主張に使うことが出来そうですね。

あるいはより発展的には，別に次元数は一致しなくて良いし同じデータでなくても良いので，脳波とMRの結果でだとか，いろいろと応用する事が出来るかもしれません。

いずれ，ICA や PCA など，他の線形変換を使った手法との違いとかも考えられると良いのかも。

## 第5章 機械学習

個人的に PRML を読んで頑張っていたり，モデリングとか自由エネルギー原理とか勉強する上でやっぱちょくちょく出てくるのが機械学習系の知見というか学習なので，それもまとめていこうと思います．ありえんむずいから合ってるのか自信ないけど頑張るぞい．

### 5.1 EM アルゴリズム

EM アルゴリズム，いろんなところでよく聞きますよね．うちの助教さん曰く，自由エネルギー原理の前に変分ベイズやるべきで，その前に EM アルゴリズムはやっとくべき，だそうで．まあそらそうって感もあるので頑張って学習してみましたよって事でまとめます．

#### EM アルゴリズム

確率モデルのパラメータ  $\theta$  を最尤推定する手法の一つ．観測可能な変数だけでなく，観測が不可能な潜在変数も観測データの振る舞いに影響している場合に使う

ということで，最尤推定の問題を解く手法の一つということでした．

最尤推定は以下の式でいうところの，尤度  $p(x|\theta)$  を最大にするようなパラメータを推定する手法でした．

$$p(\theta|x) \propto p(x|\theta)p(\theta) \quad (5.1)$$

与えられたデータ集合を  $X$ ，尤度関数を  $L$

$$L(\theta) = p(x_1, x_2, \dots, x_n|\theta) \quad (5.2)$$

として定義すると，最尤推定で解く問題は

$$\hat{\theta} = \arg \max_{\theta} L(\theta) \quad (5.3)$$

であり，これはつまり最尤推定解は

$$\frac{\partial}{\partial \theta} \log L(\theta) = 0 \quad (5.4)$$

の解でした。統計編でやってましたね。しかしここで、実は統計編の時には確認していなかった穴があります。その穴を解決するための手法の一つが EM アルゴリズムです。

### 5.1.1 不完全データ集合

さて、その穴とは何かを考えていきます。たとえば「ある人の知覚実験の結果」を観測変数として試行を繰り返すと、その結果はきつとなんらかの（普通は正規分布が妥当そう）分布に従いますよね。

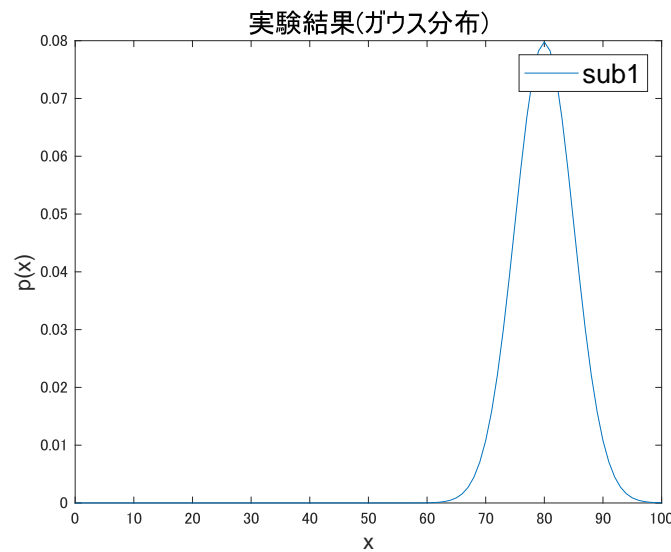


図 5.1: 被験者 1 人の実験結果

では次に、「2 人の被験者の知覚実験の結果」を観測変数とするとどうなるかというと、

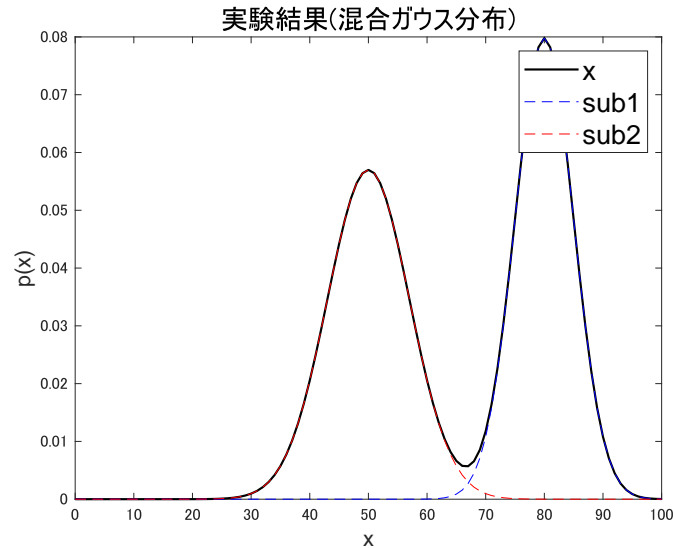


図 5.2: x:被験者 2 人の実験結果

図 (5.1.1) のようになるはずです。

どっちの被験者の成績なのかが分からない場合、二つのガウス分布の足し合わせで出来る分布によってデータ  $x$  が生成されることになります。ここでは、平均点が 80 点と 50 点の 2 人の被験者を想定します。こうしてガウス分布を混ぜて作られる分布を混合ガウス分布と言います。

被験者が 1 人の場合、観測変数  $x$  が  $\theta(\mu, \sigma)$  のパラメータに従って生成されているので普通に尤度関数を定義できました。が 2 人の場合どうでしょう。観測変数  $x$  は平均と分散だけでなく、どちらの被験者の結果なのかというもう一つのパラメータが必要になります。そこで、混合ガウス分布の式として

$$p(x|\mu, \sigma, \pi) = \sum_j \pi_j N(x|\mu, \sigma) \quad (5.5)$$

を考えます。  $\pi$  はどっちの被験者が採用されるか...? 多いか...? のパラメータです。ガウス分布に対する重みですね。というわけで、このパラメータ  $\pi$  による被験者の決定結果として隠れた変数  $z$  を定義します。これはどっちの被験者が、つまりどちらのガウス分布からデータが生成されるかです。

するとここで考える尤度は  $p(x|\theta)$  ではなく  $p(x, z|\theta)$  になります。ここで  $\theta$  も  $(\mu, \sigma)$  から  $(\mu, \sigma, \pi)$  になっています。

では尤度関数を定義して、偏微分して最尤解を求めよう... と思ったところで問題が生じます。  $z$  が観測できないことです。観測できているのはあくまで  $x$  であって、その背景にあった  $z$  の値は不明です。こうなると、普通に最尤法を解くことはできなくなってしまうわけですね。

これが最尤法の穴です。このように、データが生成される確率  $p(x, z)$  があつた時にこのうち  $z$  の情報が観測できないというようなデータ集合のことを不完全データ集合と言います。EM アルゴリズムとは、

不完全データ集合に対する最尤推定問題を解くための手法である，ということになりそうです。ではようやく，中身に入っていきます。

### 5.1.2 EM アルゴリズムの概要

数式を追っていくのはちょっとだけ大変なので，まずはモチベだけ確認します。EM アルゴリズムは， $p(x, z|\theta)$  をどうにかして解き，最尤な  $\theta$  を求めるための手法でした。ただしこの時，問題になるのは変数  $z$  が観測できないということです。

そのため，まず変数  $z$  も確率変数であると考えて，尤度  $p(x, z|\theta)$  を  $z$  周りで周辺化することで普通の尤度  $p(x|\theta)$  と等しいよねって考えます。つまり

$$p(x|\theta) = \sum_z p(x, z|\theta) \quad (5.6)$$

ですね。こうして，普通の最尤推定の時の尤度の最大化ではなく，尤度の期待値の最大化問題に置き換え，近似的に  $\theta$  を求めています。んで，そのために二つのステップ，E ステップと M ステップを繰り返し計算していくから EM アルゴリズムと呼ばれます。そのステップはなんぞやってのは式を見てからにします。

### 5.1.3 変分下界

てことで，例のごとく式 (5.6) を対数化し，対数尤度の形に式を書き直します。

$$\log p(x|\theta) = \log \sum_z p(x, z|\theta) \quad (5.7)$$

ふむ。ここで新たな問題が生じます。なんかこれ，log-sum の形が出てきちゃっています。これは解析的には扱いにくくて困る形です。てことで次の課題は，どうにかしてこの log-sum くんを扱える形に料理することです。

いきなりですがここからなんかすごいアクロバットな事します。僕は思いつかないような技なんですけど...

$$\log p(x|\theta) = \log \sum_z p(x, z|\theta) \quad (5.8)$$

$$= \log \sum_z q(z) \frac{p(x, z|\theta)}{q(z)} \quad (5.9)$$

と，確率変数である  $z$  についての分布  $q(z)$  を導入し，右辺に  $\frac{q(z)}{q(z)}$  をかけます。当然この分数は 1 なのでかけても問題はありません。さて，こうすると嬉しい事があって，対数と期待値に関する不等式として

### Jansen の不等式

$$\log E[X] \geq E \log[X]$$

というものがあります。これを使うことでさらに

$$\log p(x|\theta) = \log \sum_z p(x, z|\theta) \quad (5.10)$$

$$= \log \sum_z q(z) \frac{p(x, z|\theta)}{q(z)} \geq \sum_z q(z) \log \frac{p(x, z|\theta)}{q(z)} \quad (5.11)$$

という関係式をつくることができます。  
この右側の項を

$$L(q, \theta) = \sum_z q(z) \log \frac{p(x, z|\theta)}{q(z)} \quad (5.12)$$

として定義すると、式 (5.7) は

$$\log p(x|\theta) \geq L(q, \theta) \quad (5.13)$$

という形に変わります。多分何やっているのか分からなくなってる頃だと思うので式を落ち着いて考えてみます。まず式 (5.13) の左辺は対数尤度ですね。最尤推定において最大化したいやつです。普通だったら  $\theta$  で偏微分して最大化するやつ。そいつが、右辺の  $L$  と等号付きの不等式で表されています。

つまり求めたい対数尤度は常に関数  $L$  以上であるという関係が見えるわけですね。これを使うと、 $L$  の最大化を進めていけばそれに従って尤度の方も最大化が進んでいくことが分かります。

そう、ここで何をしたかったのかと言うと、今回求めるのが無理 ( $z$  のせい) な尤度関数の最大化問題を、別の関数である  $L$  の最大化問題に置き換えたわけですね！

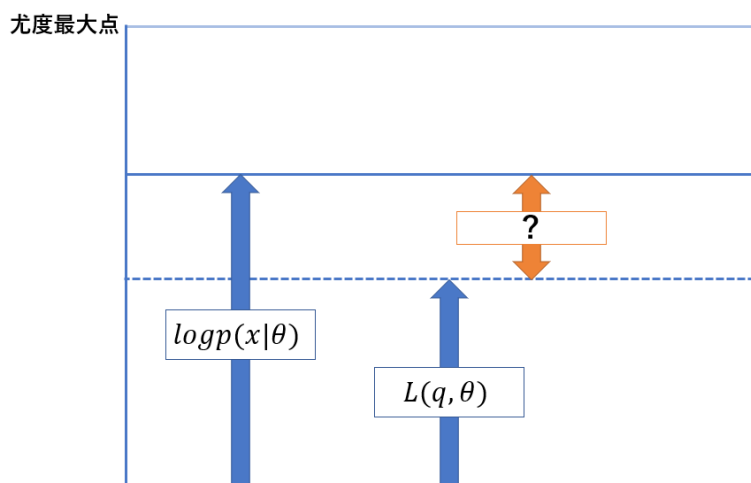


図 5.3: 尤度と L の関係

以上の関係は図 (5.1.3) のようになります。ここで気になるのは、まず尤度と L の間には 0 以上の「差」があることです。この差を 0 に近づけていくことで、 $\log p(x|\theta) = L(q, \theta)$  の関係までは L を大きくすることが出来ます。

まずはこの問題を考えます。とりあえず計算してみましょうか。

$$\begin{aligned}
 & \log p(x|\theta) - L(q, \theta) \\
 &= \log p(x|\theta) - \sum_z q(z) \log \frac{p(x, z|\theta)}{q(z)} \\
 &= \log p(x|\theta) - \sum_z q(z) \log \frac{p(z|x, \theta)p(x|\theta)}{q(z)} \\
 &= \sum_z q(z) \log p(x|\theta) - \sum_z q(z) (\log p(z|x, \theta) + \log p(x|\theta) - \log q(z)) \\
 &= - \sum_z q(z) (\log p(z|x, \theta) - \log q(z)) \\
 &= - \sum_z q(z) \log \frac{p(z|x, \theta)}{q(z)} \tag{5.14}
 \end{aligned}$$

色々やっていますが、まず 1 行目は普通に図 (5.1.3) でいうオレンジ矢印部分です。こいつを L の中身を代入して書き直し、乗法定理を使って 3 行目の形に書き直し、さらに log の計算を足し算の形にばらしたのが 4 行目です。ここで左側の項にも  $\sum q(z)$  がかかっていますが、これは確率分布だから 1 になるので特に意味をなしません。んで  $\sum q(z)$  で全体をくくって打ち消しあわせて、分数の形になおしてみたのが 5.14 になります。

この形、よく見ると KLD になっているの分かるでしょうか。



$$\begin{aligned}\log p(x|\theta) - L(q, \theta) &= - \sum_z q(z) \log \frac{p(z|x, \theta)}{q(z)} \\ &= KL[q(z) || p(z|x, \theta)]\end{aligned}$$

になっていますよね。てことで、オレンジの部分は  $q(z)$  と  $p(z)$  の KLD でした。

さて、ここからついに EM アルゴリズムの中身に入っていきます。

まずもう一度与えられている問題を考えると、最尤推定をするために最大化したいのは尤度  $\log p(x|\theta)$  なんだけど、でもこいつに未知の観測不可な要素も孕まれていると考えると普通に尤度の計算できねえじゃん。ということで  $\log p(x, z|\theta)$  を考えて、こいつをごちゃごちゃした結果として関数  $L$  の最大化をしていこうという問題にすり替えられたのでした。

関数  $L$  は  $L(q, \theta)$  とあるので、いじれる変数は 2 つです。これらそれぞれの変数を使って関数の最大化をしていけばいいわけですね。

それぞれ、片方を固定した状態でもう片方をその条件下で最適化する、という作業を繰り返していくのが EM アルゴリズムです。E ステップと M ステップというのは、この 2 変数の最適化ステップに対応しています。つまり

$$\begin{aligned}\text{E-step:} \quad q &= \arg \max_q L(q, \theta) \\ \text{M-step:} \quad \hat{\theta} &= \arg \max_{\theta} L(q, \theta)\end{aligned}$$

を交互に解く問題です。順番に見ましょう。

#### 5.1.4 E ステップ

まず、

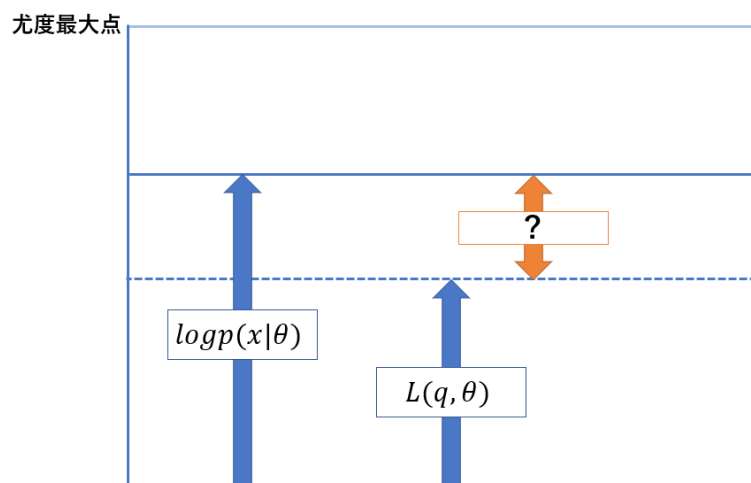


図 5.4: 尤度と L の関係

を見直して明らかなのは、 $p(x|\theta)$  と  $L(q, \theta)$  の間にある、 $KL(q||p)$  を最小化する必要があるということです。

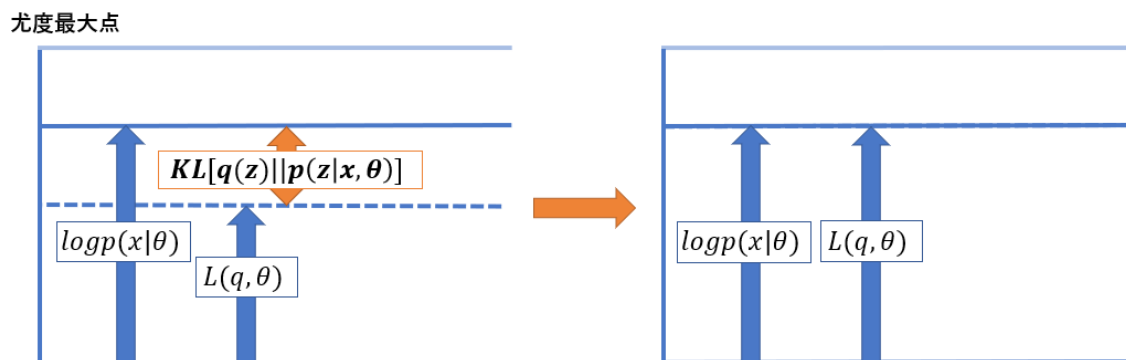


図 5.5: E ステップ

$L(q, \theta)$  の  $\theta$  を固定して、 $q$  をいじって  $KL(q||p)$  を最小化するステップが E ステップです。

$$q = \arg \max_q L(q, \theta) \quad (5.15)$$

E は expectation, つまり期待値です。もともと

$$\log p(x|\theta) = L(q, \theta) + KL(q||p)$$

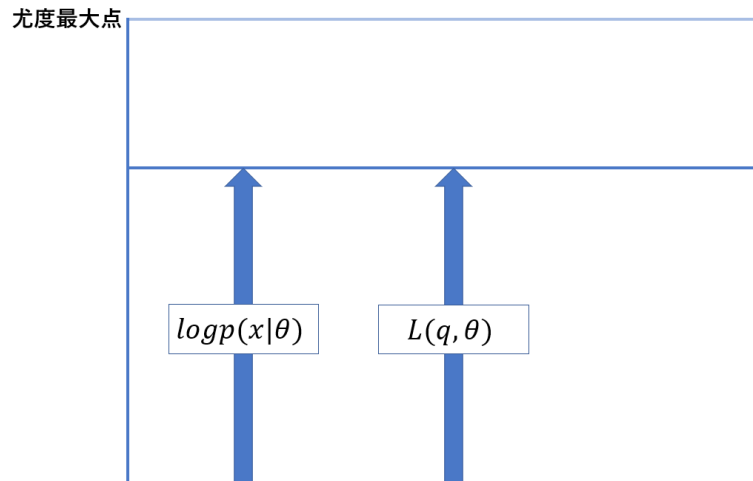
なので、ここで KLD が最小化されれば L の最大化になるわけですね。さらに、KLD が 0 ってことは  $q = p$  になる、つまり

$$q(z) = p(z|x, \theta)$$

となることを意味します。q の分布が p の事後分布と等しくなったことが分かります。

### 5.1.5 M ステップ

次に M ステップです。今、E ステップによって



の状態にまで更新されています。ここから何をやれば良いのかですが、真の最大値はまだまだ上にいます。なので  $p(x|\theta)$  と一緒に下界を押し上げる必要があります。

E ステップで KLD が最小化され、固定された  $\theta$  の元では  $\log p(x|\theta)$  が等しくなりましたが、ここで今度は  $\theta$  の方を最適化します。q を固定し、 $\theta$  を更新するわけですね。

$$\hat{\theta} = \arg \max_{\theta} L(q, \theta) \quad (5.16)$$

$\theta$  は  $\log p(x|\theta)$  も KLD も条件に入っている変数なので、L の  $\theta$  を更新すると他も更新されます。

まず、L は  $\theta$  について最大化なので当然上に伸びます。

また KLD は 0 以上の値をとるもので、E ステップで 0 になっている状態なので KLD は変わらないか増加するかです。で、元々最大化したい量である  $\log p(x|\theta)$  は

$$\log p(x|\theta) = L(q, \theta) + KL[q||p]$$

があるので右辺が増加することによって一緒に増加します。こうして全体が上に押し上げられる感じですね。M は Maximization の M で、最大化を意味します。

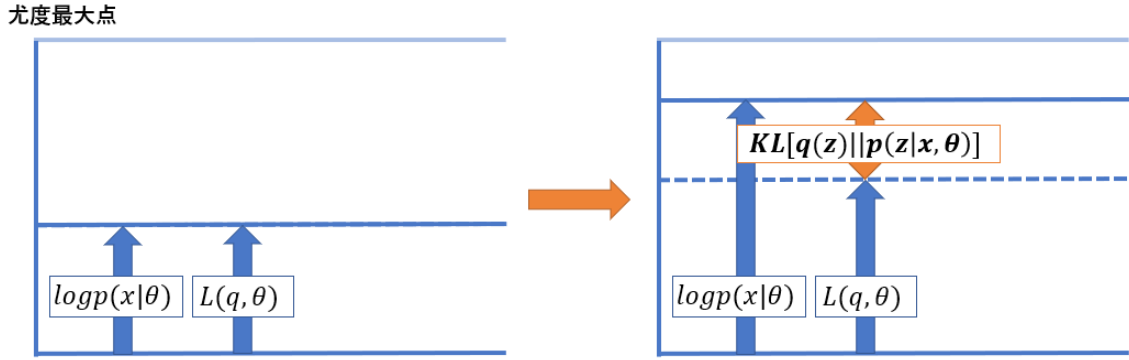


図 5.6: M ステップ

計算はまず  $q(z)$  を固定して、 $\theta$  の最大化をしていくので、E ステップの時に使っていた  $\theta$  を  $\theta_{old}$  としておきます。

L 関数の中身を式 (5.16) にある E ステップの結果を使って書き換えると

$$L(q, \theta) = \sum q(z) \log \frac{p(x, z|\theta)}{q(z)} \quad (5.17)$$

$$= \sum p(z|x, \theta_{old}) \log p(x, z|\theta) - \sum p(z|x, \theta_{old}) \log p(z|x, \theta_{old}) \quad (5.18)$$

となります。q の分布が事後分布と等しくなるってやつですね。このうち、第二項は  $\theta$  に依存していないので最大化の際は定数と考えていいので、すると

$$L(q, \theta) = \sum p(z|x, \theta_{old}) \log p(x, z|\theta) + const. \quad (5.19)$$

$$= Q(\theta, \theta_{old}) + const. \quad (5.20)$$

と、Q 関数の最大化問題に置き換えて考えられます。Q 関数は完全データの対数尤度  $\log p(x, z|\theta)$  の、E で求めた事後分布  $p(z|x, \theta_{old})$  による期待値であると考えられます。こいつの最大化によって全体を押し上げるのが M ステップでした。ちなみに、ここで定数としたやつはエントロピーとよばれる量です。が、EM アルゴリズムでは関係ないのでスルー。

$$\theta_{new} = \arg \max_{\theta} Q(\theta, \theta_{old}) \quad (5.21)$$

これが実質の M ステップです。更新の結果 KLD がまた生じているので、またこいつを最小化する E ステップをやり直します。で、また KLD が 0 になったら  $\theta$  を変えて押し上げる M ステップ。この繰り返しのすることで L を限界まで最大化すると、一緒に値が更新されている対数尤度関数も最大化されるってわけですね。

これをやっているうちに、 $\theta$  を更新しても KLD が増加しないところに来ます。こうなったときが最適化終了のタイミングです。

Expectation と Maximization を繰り返す、期待値最大化法が EM アルゴリズムの正体でした。

**EM アルゴリズム** 1. E ステップ: 現在推定されている潜在変数の分布に基づいて、モデルの尤度の期待値を計算。  
2. M ステップ: E ステップの尤度期待値を最大化するパラメータ  $\theta$  を求める。  
3. M ステップで求めたパラメータは、次の E ステップで使われる潜在変数の分布を決定するために用いられる。  
4. この 2 ステップの繰り返しで値が更新されなくなったところで最適化終了

## 5.2 変分ベイズ推定

EM アルゴリズムは隠れ変数として  $z$  がある時、変分下界を使ってパラメータ  $\theta$  を点推定する最尤法の解法でした。が、これにはいくつか課題点があります。

- $\theta$  の分布がなだらかだったり多峰の時うまくいかない
- 隠れ変数が 1 層  $z$  の時しか使えない
- 過学習してしまう

だそうです。結構つらいですね。とりあえず解決法として考えられるのは点での推定じゃなくて確率分布にしちゃうことと、なんらかの形で正則化して過学習を防ぐことでしょうか。後者はともかく、前者は簡単です。ベイズ推定を使えばいいですね。

最尤推定と最大事後確率推定、ベイズ推定については統計編にまとめていますが、最尤法は尤度が最大になる  $\theta$  を、最大事後確率推定は事後確率が最大になる  $\theta$  を点推定する手法なのに対し、ベイズ推定は事後分布自体を使う手法でした。

そのベイズの考え方をいれた EM アルゴリズムが変分ベイズ EM アルゴリズム?VB-EM アルゴリズムです。最尤法の時の EM アルゴリズムでは  $z$  は確率変数だったけど  $\theta$  は最大化できる特定の値でしたね。これも確率変数にしちゃうのがこれからやることです。

まず、やりたいのはこれまでに得られたデータ集合  $X$  の確率、 $p(X)$  を最大化するパラメータの事後分布を考えることです。これはベイズモデルが持つて、推定の根拠となるものなのでモデルエビデンスなんて言います。

$$p(X) = \iint p(X, z, \theta) dz d\theta \quad (5.22)$$

を最大化するんですね。

最尤法の時同様、確率分布  $q$  を導入して、式 (5.22) の最大化に必要な変分下界を求めます。

$$\log p(X) = \log \iint p(X, z, \theta) dz d\theta \quad (5.23)$$

$$= \log \iint q(z, \theta|X) \frac{p(X, z, \theta)}{q(z, \theta|X)} dz d\theta \geq \iint q(z, \theta|X) \log \frac{p(X, z, \theta)}{q(z, \theta|X)} dz d\theta = L(q(z, \theta)) \quad (5.24)$$

ということで、変分下界  $L(q(z, \theta))$  が求まりました。が、問題なのは推定したい  $q(z, \theta)$  が同時分布の形をとっていることです。それぞれの全ての組み合わせを考慮しているわけですね。これだと面倒なので、変分ベイズ推定では  $q$  の変数間に独立性を仮定します。つまり

$$q(z, \theta) = q(z)q(\theta) \quad (5.25)$$

ですね。このように各確率変数に独立性の仮定を置いて推定したい  $q(z, \theta)$  を複数の確率分布の積で表現することを平均場近似と言います。なので平均場近似をやると

$$\begin{aligned} \log p(X) &\geq \iint q(z, \theta|X) \log \frac{p(X, z, \theta)}{q(z, \theta|X)} dz d\theta \\ &= \iint q(z)q(\theta) \log \frac{p(X, z, \theta)}{q(z)q(\theta)} dz d\theta \\ &= L(q(z, \theta)) \end{aligned} \quad (5.26)$$

と置き換えることができ、解く事のできる形に  $L$  を変形できます。この  $L$  が変分自由エネルギーと呼ばれる量です。自由エネルギー原理にでできますよね、変分自由エネルギー。

ともかく、こうして変数  $z, \theta$  を分離することが出来たのでそれぞれを逐次更新して  $L$  を最大化していきます。

面倒なので  $z, \theta$  をまとめ (隠れ変数が1層以上であることも考慮して)、 $Z$  を定義します。

$$q(Z) = \prod_{i=1}^M q_i(Z_i), \quad Z_i (i = 1, \dots, M) \quad (5.27)$$

で、この  $q(Z)$  の中で  $L$  を最大にするものを探します。式 (5.26) を書き直すと

$$L(q) = \int q(Z) \log \frac{p(X, Z)}{q(Z)} dZ \quad (5.28)$$

$$= \int q(Z) (\log p(X, Z) - \log q(Z)) dZ \quad (5.29)$$

$$= \int \prod_i q_i(Z_i) (\log p(X, Z) - \sum_i \log q_i(Z_i)) dZ \quad (5.30)$$

$$= \int \prod_i q_i \log p(X, Z) dZ - \int \prod_i q_i \sum_i \log q_i dZ \quad (5.31)$$

となります。2行目から3行目の変形に式 (5.14) を使っています。log の中身はかけざんが足し算になるときに注意ですね。

ここでちょっと面白いことをします。今  $L$  は  $q_i(Z_i)$  で書かれていますが、こいつをさらに  $q_j(Z_j)$  を使って整理してみます。すると

$$L(q) = \int \prod_i q_i \log p(X, Z) dZ - \int \prod_i q_i \sum_i \log q_i dZ \quad (5.32)$$

$$= \iint q_j \prod_i q_i (\log p(X, Z) - \sum_i \log q_i) dZ_{i \neq j} dZ_j \quad (5.33)$$

となります。これは元々  $Z$  にいていた変数が  $z, \theta$  で、二重積分だったことを思い出せば大丈夫だと思います。こいつをさらに変形していくと

$$L(q) = \int \prod_i q_i \log p(X, Z) dZ - \int \prod_i q_i \sum_i \log q_i dZ \quad (5.34)$$

$$= \iint q_j \prod_{i \neq j} q_i (\log p(X, Z) - \sum_i \log q_i) dZ_{i \neq j} dZ_j \quad (5.35)$$

$$= \int q_j \int \prod_{i \neq j} \quad (5.36)$$

## 第6章 知覚心理実験

やはり勉強して思うのは、心理屋さんって頭良いよなあという事です。よく知らない数理や工学系の人達は心理屋さんを馬鹿にしがちで、というのも「心理の人達は定性的な議論しかしてない。科学じゃない」なんて偏見に基づいていると思います。

これについては尤もな指摘でもあるので一概に否定は出来ません（実際まだまだアンケート取って終わりとかそういう人達もいますし、彼らについてはたしかに定性的な議論になっている事は否めないと思う）。しかし、中にはどうかして定性的な結果を定量的に評価しようなんて事を考え続けている心理屋さんたちもいて、彼らは我々が小馬鹿にしているようなデータを使って「まじか!」となるような指標をだしていたりします。本章は自分が勉強した、主に知覚心理実験で用いられているそういった理論、指標について触れます。

### 6.1 信号検出理論

信号検出理論 (Signal Detection Theory) はその名の通り、ある特定の信号があったかどうかを検出する我々のプロセスをモデル化したもので、たとえば A という刺激を学習して他の刺激と区別するようなタスクを考えた際に適用できます。この場合、被験者は A 刺激を検知したら Yes と、そうじゃないものであったら No と回答します。この時、どれだけ正確に Yes/No を答えられるか (正答率) が被験者の成績として一般に評価されているわけですが、もう少し考えてみようぜ、という話です。

というのも、実際自分でやってみれば分かるのですが被験者の性格によっても正答率とかは変わってきます。Yes と No が 25:75 でランダムイズされた実験で考えてみましょう。被験者が保守的 (筆者はこのタイプ) な場合、微妙だなと思う刺激に対しては No と答えるため、全体的に Yes と答える回数が減ります。その結果、Yes/No 比が 15:85 になったけど Yes と答えた刺激の中での正答率は 100% という人がいる一方、逆にとりあえず Yes と答える人は 40:60 になっていて、刺激 A を Yes と答える確率は 100% だけでもそうじゃないのについても Yes 判定してしまっている人がいたり... 単純に正答率だけでは比べるのが難しい、というかフェアじゃないよねという事態が生じます。そこで使われるのが SDT です。SDT は、2 値の正解データ (Yes or No) と、2 値の回答データ (yes or no) の組み合わせそれぞれの確率について対応表を作成するところから始まります。統計でおなじみの偽陽性とか偽陰性のあれと一緒にです。

表 6.1: 信号検出理論での反応の分類表

	ありと判断	なしと判断
あり	ヒット率 (Hit Rate)	ミス率 (Miss Rate)
なし	誤警報率 (False Alarm Rate)	正棄却率 (Correct Rejection Rate)



これら4つの指標 (HR, MR, FAR, CRR) を用いて議論するのが信号検出理論の基本です。それぞれ、ヒット率は実際に信号があった際にちゃんと検知できた割合、ミス率は本当は信号があったのに検知できなかった割合、誤警報率は実際には信号はなかったのにあったと勘違いした割合、正棄却率は実際に信号がなくて、検知もしなかった場合を意味します。ヒット率と正棄却率が100%になるのが最強の被験者ですね。が実際にはそんなことはなくて、被験者ごとに4つの値はめちゃくちゃです。保守的な人はどうしてもミス率が上がるし、積極的な人は誤警報率が上がります。どっちが優秀とか比べようがないですよ。なのでこの4指標をそのまま使うのでは不十分です。勿論ただの正答率で議論する研究よりはマシですが。

これらの指標を元に、信号のありなしがどういった判断によってなされたのかをモデル化していくのが信号検出理論です。そんなに難しくないです。

まず、このモデルが置いている仮定から確認します。それは計測・観測ノイズの存在です。我々の感覚器や脳は常に正確に反応できるわけではないし、たとえば触覚刺激を検出する課題だとしても偶然吹いた風だとか服がこすれるだとか、実験で統制している刺激以外にも様々な要因によって信号を受け取っています。これによって、全く同じ刺激を与えられたとしても反応は若干異なるわけで、これらをノイズとして考えると、基本的には与える信号とは無関係に常にノイズが与えられているため、信号があったとき (Yes)、なかった時 (No) どちらも反応は確率分布の形をとることになります。自然界のノイズは基本的にガウスノイズとして仮定できるため、正規分布を仮定します。感覚は常に正規分布しているというのは感覚処理の話や計算論的神経科学の観点からも違和感なく喉を通りますね。

では、被験者が信号を検出するときとしない時の差について考えていきます。まず信号がないとき、これはただノイズのみに依存する正規分布です。信号があるときには、同じように正規分布だが信号の強度に依存してその平均がずれた正規分布が生じます。ノイズ分布とノイズ+信号分布の分散についてはどうするかですが、面倒なので等分散性を仮定しておきます。

信号検出の課題とはつまり、この枠組みで語るなら得られた刺激がどちらの分布によって生じているのかを判断する仕事になります。しかし問題は、被験者が保守的だったり積極的だったりすると、判断の基準が変わります。保守的な人はノイズ分布はもちろんノイズ+信号分布の一部から大部分までも含めてノイズ分布から生じたと推定し、積極的な人は逆にノイズ分布の一部も含めて信号分布であると推定する事が考えられます。つまり、二つの分布の平均のほかにもう一つ、被験者個人の反応バイアス、Yes/No のどちらに分類しやすいかという変数も発生します。

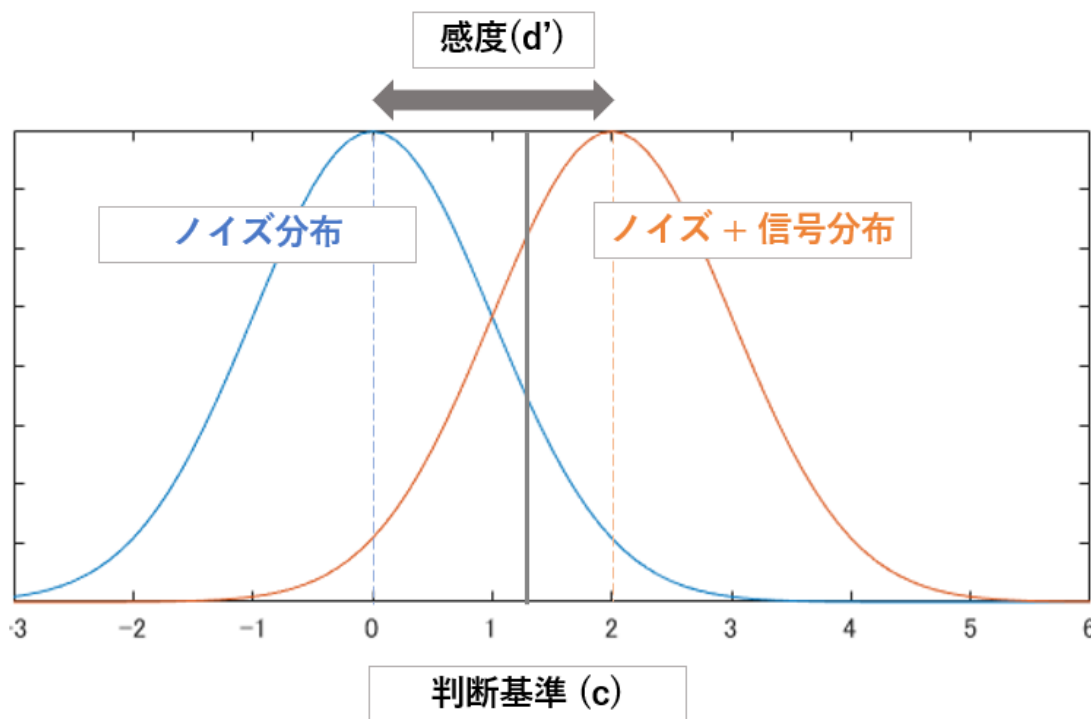


図 6.1: 信号検出理論で仮定する二つの正規分布

二つの分布が分かり、被験者ごとの判断基準 (例... 合計値が 10 以上だったら Yes) が分かっただけでは、先程の表にあったヒット率やミス率も計算できます。単に分布を二つ並べて、線を引いて、線の右左のどちら側かと、本来はどちらの分布に含まれているかで 4 パターンに分けられますからね。これがそれぞれ先程の 4 指標に対応しているわけです。

図にすると以下 (図 6.1, 6.1) のようになります。Hit, Miss, False Alarm, Correct Rejection のそれぞれがどちらの分布に属していて、どちらの分布として判定されたのかを確認してください。

では逆に考えると、4 指標の値が分かれば二つの分布 (の距離) も判断基準も推定できる事になります。この、分布間の距離を感度 ( $d'$ )、判断基準を反応バイアス ( $c$ ) として推定するのが信号検出理論です。

それぞれ算出法を考えます。感度は定義上は分布の距離なので

$$d' = \frac{\mu_{s+n} - \mu_n}{\sigma_n} \quad (6.1)$$

となります。これはノイズ+信号分布とノイズ分布の平均の差をノイズ分布の標準偏差で割るということで、つまり分布間の距離がノイズ分布の標準偏差何個分に相当するのかの指標です。多くの場合、ノイズ分布は標準偏差が 1 の正規分布を仮定するので、その場合は単純に平均の差を意味します。こうして算出される感度 ( $d'$ ) は、二つの分布に対する被験者の心的距離です。区別しやすいもの程距離は離れていて、つまり感度も高くなります。いわゆる”優秀”な成績は感度が高い状態ですね。これは分布間の距離なので、被験者が積極的か保守的かなどの判断基準 ( $c$ ) の影響を受けない弁別能力を反映している事が分かるかと思います。

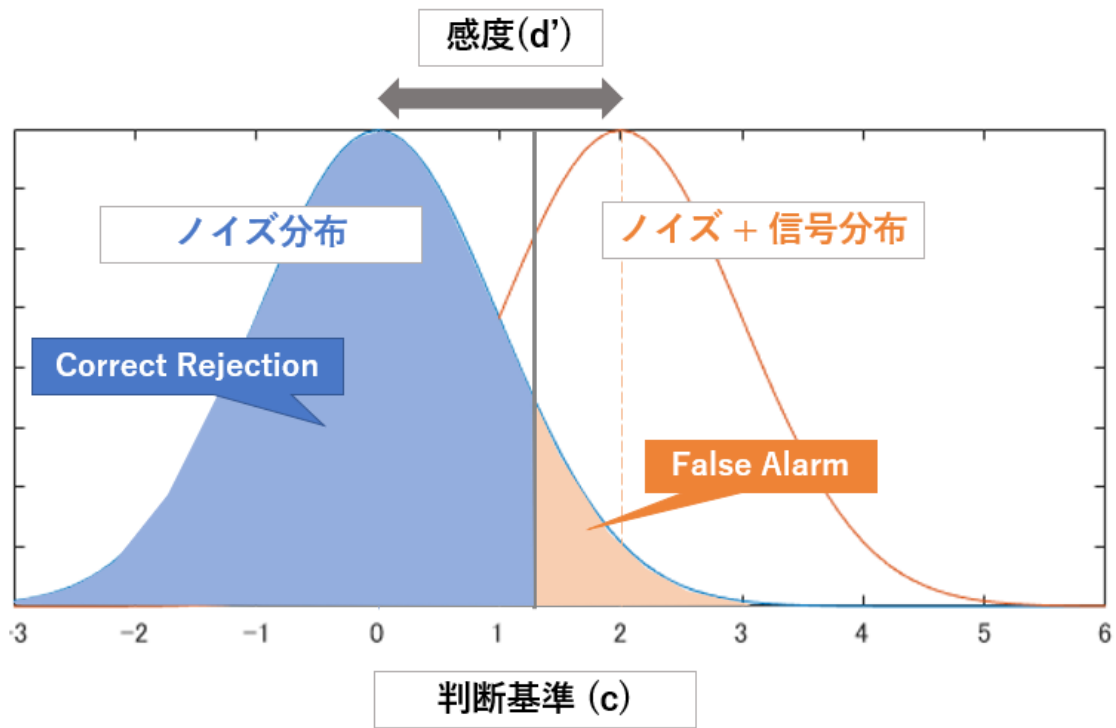


図 6.2: ノイズ分布の二つの領域

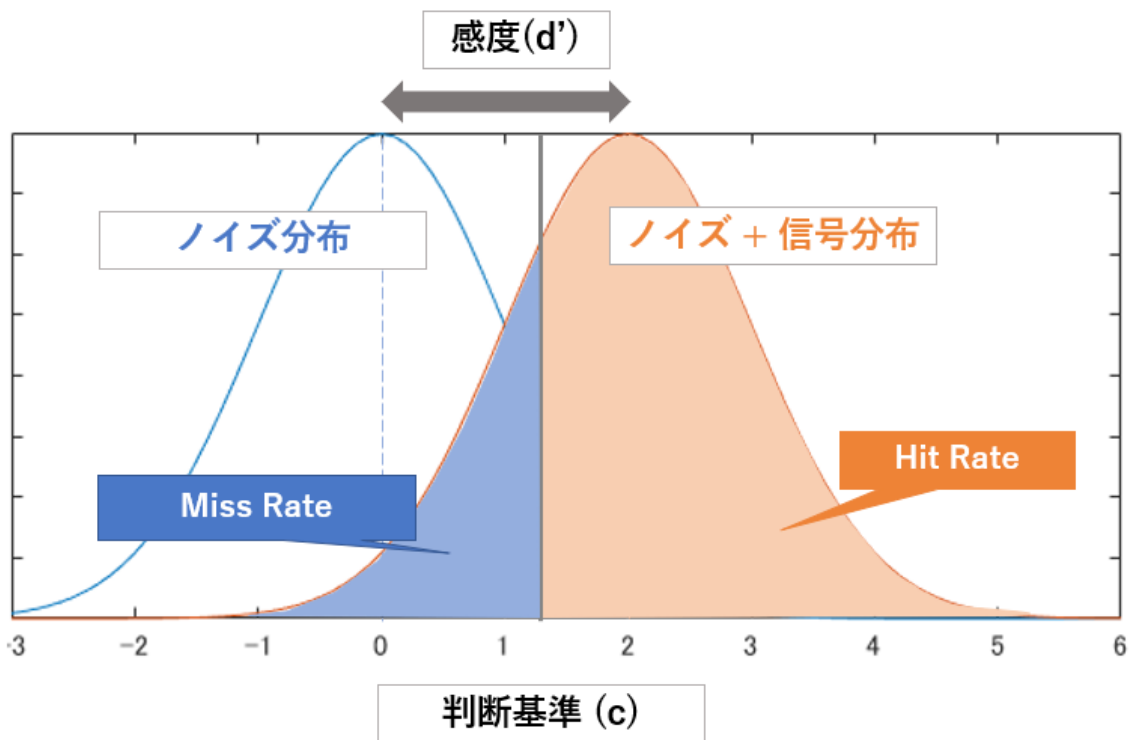


図 6.3: 信号分布の二つの領域

### 6.1.1 感度

感度は刺激が強く（極端に）なればなるほど高くなるし、学習によっても高くなる傾向があります。実際の場合では、より正確には分布の平均位置を出すのは難しいので次のようなステップで感度を求めます。

まず、信号分布に対するヒット率（ヒットとミスの割合）と、ノイズ分布に対する誤警報率（CR と FA の割合）を算出します。これは以下の式で得られます。

$$\text{ヒット率} = p(\text{yes}|SN) = \frac{\text{ヒット数}}{\text{ヒット数} + \text{ミス数}} \quad (6.2)$$

$$\text{誤警報率} = p(\text{yes}|N) = \frac{\text{誤警報数}}{\text{誤警報数} + \text{正棄却数}} \quad (6.3)$$

これらは、それぞれ判断基準より上の信号分布、ノイズ分布の面積に相当します。ノイズ分布と信号分布は平均が違っただけで同形の分布だったので、標準正規分布において右側の面積を積分すると計算したヒット率や誤警報率になる点をそれぞれ求めてやれば、その数値の差はすなわち分布の平均の差と等しくなる事が分かるかと思います。図 (6.1, 6.1) のオレンジの部分と比較してみてください。同じ灰色の線（判断基準）に対し、オレンジ部分はかなり異なりますね。この差がすなわち分布間の距離ということです。よって以下の計算をします。

$$d' = Z(\text{ヒット率}) - Z(\text{誤警報率}) \quad (6.4)$$

ここで、 $Z()$  は Z 得点です。z 得点とは、標準正規分布の平均を基準にした標準偏差数です。その分布の平均からどれだけ離れているかですね。これを二つの分布それぞれについて算出し、その差を求める事で分布間の距離を測るわけです。Z を取らないといけない理由は、そもそも独立な異なる分布を比較する際には標準化する必要があったのでしたね。

Listing 6.1: 感度の計算

```
1  dat_ph = hit / (hit + miss);
2  dat_pf = fa / (fa + cr);
3  d' = norminv(dat_ph) - norminv(dat_pf);
```

こうして得られた感度は、大きい値であるほどしっかりとノイズ分布と信号分布を区別できるという事になります。これを使って、A の被験者と B の被験者のどちらが性格に Yes と答えられているかといったような事が分かりますね。

### 6.1.2 反応バイアス

反応バイアスは、被験者がそもそも Yes/No のどちらかに偏って答えやすいかどうかの指標です。「Yes をうまく検知できたら報酬をあげます」とか、「なるべく間違って Yes と答えなさい」といった指示が実験者から与えられていたり、本人の性格だったりによってどちらに答えやすいかは被験者ごとに異なります。感度はこのバイアスの影響を受けない指標だったわけですが、バイアス自体は以下のようにして求めれます。

$$C = \frac{1}{2}(Z(\text{ヒット率} + \text{誤警報率})) \quad (6.5)$$

これは、それぞれの分布の平均が判断基準からどれだけ離れているかの値 ( $Z()$ ) の平均です。簡単ですね。この値が負だと被験者は Yes と答えやすく、正だと No と答えやすい、そして 0 だとバイアスがないという事になります。

### 6.1.3 まとめ

信号検出理論は様々な知覚実験に利用できます。たとえば、ある特定の刺激パターン（音でも光でも顔画像でも）を学習し、それを検知したら Yes と答えるといった課題はよく見られると思います。これを単純に正答率で評価してしまうと、被験者によって様々な反応バイアスがあるためにあまり正しくない評価になりかねません。そういう時に感度を計算すると便利ですね。そうする事で、たとえば疾患患者は感度が著しく低く、反応バイアスは健常者と変わらないとかそういった議論ができるようになるわけですね。

## 関連図書

- [1] Lazepnik, Y. (2002). “Can a biologist fix a radio?-Or, what I learned while studying apoptosis.” *Cancer Cell*, 2(3), 179-182.
- [2] Jonas, Eric, and Konrad Paul Kording. (2017). ”Could a neuroscientist understand a microprocessor?.” *PLoS computational biology* 13.1.
- [3] Marr. (1982). “Vision.”
- [4] Friston, K. (2010). “The free-energy principle: A unified brain theory?” *Nature Reviews Neuroscience*. Volume 11, Issue 2, February 2010, 127-138
- [5] Adriano B. L. Tort, et al. (2010). “Measuring Phase-Amplitude Coupling Between Neuronal Oscillations of Different Frequencies” *Journal of Neurophysiology* 104:1195-1210.
- [6] Christopher M. Bishop. (2006). “Pattern Recognition and Machine Learning”
- [7] Wikipedia
- [8] yumaloo. “Kullback-Leibler Divergence についてまとめる”  
<https://yul.hatenablog.com/entry/2019/01/07/152738>
- [9] Katunori Kitano. “Transfer entropy を用いた神経回路の解析” *Annual Review 神経* 2017 I. Basic Neuroscience.