

# 深層学習を用いた脳波信号からの運動意図推定

佐野 光  
Sano Hikaru

2018 年 1 月 4 日

# 数学的表記について

本文中で用いられる数学的な表記の中で、慣習的に用いているものを記載する。ここに記載した数式は再度本文中で明示する場合もあるが、特に誤解を招かない場合には予告なしに用いる。

$x$	ベクトル（スカラーを表す場合には本文中で明記して用いる）	(1)
$ x $	ユークリッドノルム	(2)
$ x _p$	$l_p$ ノルム	(3)
$X$	行列（他の用途の場合には明記する）	(4)
$\cdot^T$	転置記号	(5)
$\cdot^{-1}$	逆行列記号	(6)
$I$	単位行列	(7)
$\text{diag}(x)$	ベクトル $x$ の要素を対角成分に持つ対角行列	(8)
$\mathbb{R}$	実数集合	(9)
$\mathbb{R}^m$	$m$ 次元実数集合	(10)
$\mathbb{R}^{m \times n}$	$m$ 次元実数集合と $n$ 次元実数集合の直積集合	(11)
$\mathbb{I}(S)$	指示関数 ( $S$ が真の時に 1 を返し、その他の場合に 0 を返す)	(12)
$\{a, b\}$	$a$ と $b$ を元を持つ集合	(13)
$(a, b)$	$a < b$ で $a$ と $b$ を端点とする閉区間	(14)
$[a, b]$	$a < b$ で $a$ と $b$ を端点とする開区間	(15)
$a \in A$	$a$ は集合 $A$ の元	(16)
$B \subset A$	$B$ は集合 $A$ の真部分集合	(17)
$B \subseteq A$	$B$ は集合 $A$ の部分集合	(18)
$\pi$	円周率	(19)
$e$	ネイピア数	(20)
$\log(\cdot)$	ネイピア数を底とした対数関数	(21)
$\max_x f(w, x)$	$x$ に関する $f$ の最大値	(22)
$\min_x f(w, x)$	$x$ に関する $f$ の最小値	(23)
$\arg \max_x f(w, x)$	$x$ に関して $f$ が最大となるときの $x$	(24)
$\arg \min_x f(w, x)$	$x$ に関して $f$ が最小となるときの $x$	(25)
$p(\cdot)$	確率密度関数あるいは確率質量関数（慣習的に確率分布と表記する）	(26)
$q(\cdot)$	確率密度関数あるいは確率質量関数（慣習的に確率分布と表記する）	(27)
$p(x_1, x_2)$	$x_1, x_2$ の同時確率	(28)
$p(x_1   x_2)$	$x_2$ に条件付けられた $x_1$ の条件付き確率	(29)
$\mathcal{N}(\mu, \sigma^2)$	スカラーの平均 $\mu$ 、分散 $\sigma^2$ の 1 次元ガウス分布	(30)
$\mathcal{N}(\mu, \Sigma)$	平均ベクトル $\mu$ 、分散共分散行列 $\Sigma$ の多次元ガウス分布	(31)
$\mathbb{E}[\cdot]$	期待値演算	(32)
$\mathbb{E}[x_1   x_2]$	$x_2$ に条件付けられた $x_1$ の条件付き期待値	(33)
$\mathbb{E}_{X \in C}[f(X)]$	集合 $C$ に属する $X$ による $f(X)$ の期待値	(34)
$\mathbb{E}_{p(x)}[f(x)]$	確率密度関数 $p(x)$ による $f(x)$ の期待値	(35)
$\mathcal{KL}(p   q)$	$p$ から $q$ へのカルバック・ライブラーダイバージェンス	(36)
i.i.d.	独立同分布から生起することを表す	(37)
$\mathcal{D}$	データ集合	(38)
$x_{a:b}$	$x_a, x_{a+1}, \dots, x_b$	(39)

# 目次

1	研究背景	1
1.1	研究背景	1
1.2	BCI の概要	1
1.2.1	外部刺激型 BCI	1
1.2.2	運動想起型 BCI	1
1.3	研究目的	1
2	BCI のための要素技術	2
2.1	周波数解析	2
2.1.1	4 種類のフーリエ変換	2
2.1.2	パワースペクトル密度	4
2.2	多変量解析	4
2.2.1	Principal Component Analysis(PCA)	4
2.2.2	Indipendent Component Analysis(ICA)	7
2.2.3	Blind Source Separation(BSS)	7
2.3	分類手法	10
2.3.1	Linear Discriminant Analysis(LDA)	10
2.3.2	Support Vector Machine(SVM)	12
2.3.3	Logistic Regression(LR)	15
3	従来の運動想起型 BCI	17
3.1	特徴量工学	17
3.1.1	頭皮領域と空間フィルタ	17
3.1.2	脳波と時間周波数解析	19
3.2	Common Spatial Pattern とその派生手法	20
3.2.1	脳波信号の定式化	20
3.2.2	Common Spatial Pattern(CSP)	20
3.2.3	Common Sparse Spatio Spectral Pattern(CSSSP)	22
3.2.4	Filter Bank Common Spatial Pattern(FBCSP)	23
3.3	従来手法のまとめ	23
3.3.1	特徴量の観点からの従来手法	24
3.3.2	構成の観点からの従来手法	24
4	提案手法	26
4.1	ニューラルネットワーク	26
4.1.1	ニューラルネットワークに用いられる層	26
4.1.2	活性化関数	28
4.1.3	過学習抑制手法	29

5	実験方法	30
6	結果と考察	31
Appendix		32
A.1	セクション . . . . .	32
A.1.1	サブセクション . . . . .	32
参考文献		34

# 研究背景

## 1.1 研究背景

Brain Computer Interface(BCI) とは、脳とコンピュータを何らかの方法で接続し脳活動とコンピュータの動作を関連付ける技術の総称である。BCI は脳とコンピュータの接続方法に関して「侵襲式」と「非侵襲式」の2つに分類され、実応用を考える上では非常に重要な項目となる。侵襲式と非侵襲式とは、BCI を構築する際に外科手術を要するか否かを指す言葉であり、非侵襲式では外科手術を伴わない接続方法を用いて BCI を構築する。

非侵襲式は侵襲式に比べ人間への負担が明らかに少なく、BCI の実用化において大きなアドバンテージを有する。一方で、脳活動の計測精度という観点では劣り、BCI の構築のためには多くの工学的問題を解決する必要がある。具体的には、計測時に混入するノイズの除去の問題や、脳活動の信号源の推定の問題を解決しなければならないと考えられている。本稿では特に脳活動を頭皮上の電位変化である Electroencephalogram ( EEG ) として計測し、計測データに応じてコンピュータが特定の処理を行うタイプの BCI に着目する。

## 1.2 BCI の概要

### 1.2.1 外部刺激型 BCI

### 1.2.2 運動想起型 BCI

本研究で着目する運動想起型 BCI とは、人間が肢体運動を能動的に想像したことを検知する BCI である。主な応用としては麻痺患者のリハビリテーションが想定されているが、他にも肢体運動を外部機器の操作に対応させることで依り汎用的な応用が期待できる。

## 1.3 研究目的

運動想起型 BCI は人間が自発的な活動を行った際に生じる脳波を検知する必要があるが、外部刺激に対する脳の活動に比べて被験者による個人差が激しく、検知アルゴリズムを手で設計するには限界がある。従って、各個人毎に BCI の設計をしなければならないケースにおいて検知アルゴリズムを自動で獲得する手法を開発することは重要な課題である。そこで本研究では、個々の脳波を事前に計測・収集しておくことで、人手での難解な脳波解析を回避しつつ高い精度で運動想起を検知できる機会学習モデルの構築を行う。特に近年、画像認識や音声認識の分野で高い性能を示している深層学習に着目し、脳波からの運動意図推定に適したニューラルネットワーク開発を行う。

## BCI のための要素技術

この章では BCI に用いられる要素技術について述べる。ここで紹介する技術は BCI に限らずデータを扱う場面で広く持ちいられている手法であるため、一般的な定式化と用途を述べた後に、脳波解析における応用方法や問題点について述べる。

### 2.1 周波数解析

スカラーの時間波形  $x(t)$  に対して、周波数スペクトルを算出する際に用いられるフーリエ変換について述べる。 $t$  が連続量である時  $x(t)$  をアナログ信号と呼び、離散的であればデジタル信号と呼ぶ。このセクションではアナログ信号とデジタル信号が混在するため、アナログ信号に関しては  $x(t)$  と表記し、デジタル信号を  $x_n$  と表記することで明確に区別する。

#### 2.1.1 4 種類のフーリエ変換

時間を連続的に扱うか離散的に扱うかの違いだけでなく、周波数を連続的に扱うか否かによってもフーリエ変換の式は異なっており、 $2 \times 2 = 4$  つのフーリエ変換がある。通常はフーリエ変換について議論する場合は時間も周波数も連続であるとみなした式を使う場合が多いが、コンピュータ上で数値の処理を行う場合は原則離散的であるため、4 つのフーリエ変換について全て簡単に説明する。

##### フーリエ変換

スカラーの時間波形  $x(t)$  に対して、フーリエ変換は以下の (2.1) で表され、逆フーリエ変換の式は (2.2) で表される。

$$X(\Omega) = \int_{-\infty}^{\infty} x(t) \exp(-i\Omega t) dt \quad (2.1)$$

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\Omega) \exp(i\Omega t) d\Omega \quad (2.2)$$

ここで、 $i$  は虚数単位であり  $X(\Omega)$  は  $\Omega$  を変数とするスカラー関数である。逆フーリエ変換はスカラーの時間波形  $x(t)$  をある周波数  $\Omega$  の波形  $\exp(i\Omega t)$  の線型結合によって表現した式である。このときの結合係数が  $X(\Omega)$  である。工学的な立場では時間波形  $x(t)$  が与えられた時に、波形  $\exp(i\Omega t)$  による線型結合をすることで、波形にどのような周波数の波形がどれくらいの割合で含まれているかを知りたいケースがある。それを可能にするのがフーリエ変換であり、 $x(t)$  から結合係数  $X(\Omega)$  を算出することが可能である。通常はこの時の結合係数  $X(\Omega)$  を周波数スペクトルと呼ぶ。逆フーリエ変換 (2.2) に登場する定数倍  $1/2\pi$  は、フーリエ変換  $\mathcal{F}(\cdot)$  と逆フーリエ変換  $\mathcal{F}^{-1}(\cdot)$  が、フーリエ変換可能な時間波形  $x(t)$  に対して  $\mathcal{F}^{-1}(\mathcal{F}(x(t))) = x(t)$  となるように調整するための係数である。この係数は逆フーリエ変換の式ではなくフーリエ変換の式に付いていてもよく、あるいは両方の式に平方根の形で分配されていても構わない。またフーリエ変換可能な時間波

形とは以下を満たす  $x(t)$  である

$$\int_{-\infty}^{\infty} |x(t)| dt < \infty \quad (2.3)$$

#### 離散周波数フーリエ変換

取りうる周波数を離散的にした場合のフーリエ変換について説明する。離散周波数フーリエ変換という命名は便宜的にこの論文内で行っているものであり一般的ではない。通常、ここで紹介するフーリエ変換は「フーリエ級数展開」として知られている。歴史的には微分方程式を解くために開発され、フーリエ変換よりも先に発見されている。離散周波数フーリエ変換と離散周波数逆フーリエ変換はそれぞれ (2.4) と (2.5) で表される。

$$X_k = \frac{1}{T_0} \int_{-\frac{T_0}{2}}^{\frac{T_0}{2}} x(t) \exp(-i\Omega_0 kt) dt \quad (2.4)$$

$$x(t) = \sum_{k=-\infty}^{\infty} X_k \exp(i\Omega_0 kt) \quad (2.5)$$

離散周波数逆フーリエ変換は、スカラーの時間波形  $x(t)$  を離散的な周波数  $\Omega_0 k$  の波形  $\exp(i\Omega_0 kt)$  の線型結合によって表現した式である。このときの結合係数が  $X_k$  である。ここに  $T_0$  は時間波形の周期であり、 $\Omega_0 = 2\pi/T_0$  を基本周波数と呼ぶ。離散周波数フーリエ変換が、時間波形が与えられた時の周波数スペクトルを算出する役割を担うことは、フーリエ変換と同様である。フーリエ変換との最たる違いは、時間波形に対して周期性を仮定している点であり、その周期は既知でなければならない。時間波形が周期性を持つ場合には周波数は離散的な値を取る。

#### 離散時間フーリエ変換

取りうる時間を離散的にした場合のフーリエ変換について説明する。離散時間フーリエ変換と離散時間逆フーリエ変換はそれぞれ (2.6) と (2.7) で表される。

$$X(\omega) = \sum_{n=-\infty}^{\infty} x_n \exp(-i\omega n) \quad (2.6)$$

$$x_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega) \exp(i\omega n) d\omega \quad (2.7)$$

離散時間逆フーリエ変換が、時間波形  $x_n$  を周波数スペクトル  $X(\omega)$  を結合係数とした  $\exp(i\omega n)$  の線型結合を表しているのはこれまでと同様である。しかし、離散時間フーリエ変換では周波数スペクトル  $X(\omega)$  が周期的な関数となることは強調しておかねばならない。

#### 離散フーリエ変換

離散フーリエ変換は、取りうる時間も周波数も離散的であるとした場合のフーリエ変換である。離散フーリエ変換と離散逆フーリエ変換はそれぞれ (2.8) と (2.9) で表される。

$$X_k = \sum_{n=0}^{N-1} x_n \exp\left(-i\frac{2\pi}{N} kn\right) \quad (2.8)$$

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k \exp\left(i\frac{2\pi}{N} kn\right) \quad (2.9)$$

ここで  $N$  はサンプル時間点数の意味で周期である。時間波形に対する周期は  $N$  であるが、周波数スペクトルに対する周期も  $N$  となる。仮に 1 周期が 1000 点のサンプル点によって構成される時間波形に対し離散フーリエ変換を用いた場合は、得られる周波数スペクトルは 1000 点で周期を有する形式となる。離散フーリエ変換は有限の数列から有限の数列への変換であり、他のフーリエ変

換と異なり無限大を扱う必要はないためコンピュータ上で計算を実行することが可能である。高速フーリエ変換と呼ばれる実応用で頻繁に用いられるアルゴリズムは、離散フーリエ変換 (2.8) を高速に実行する手続きのことである。

## 2.1.2 パワースペクトル密度

パワースペクトル密度  $PSD(\Omega)$  とは時間信号  $x(t)$  の周波数スペクトルを  $X(\Omega)$  とした時、

$$PSD(\Omega) = |X(\Omega)|^2 \quad (2.10)$$

に相当する関数である。信号  $x(t)$  のエネルギーが周波数  $\Omega$  に関してどのように分布するかを示している。コンピュータでパワースペクトル密度を計算する場合は (2.8) で計算される  $X_k$  の二乗  $|X_k|^2$  を算出する。しかし、数学的な定式化を行う上では以下の定義が用いられ、 $|X_k|^2$  はピリオドグラムと呼び区別する。

$$PSD(\Omega) = \int_{-\infty}^{\infty} R(\tau) \exp(-i\Omega\tau) d\tau \quad (2.11)$$

ここに、 $R(\tau)$  は

$$R(\tau) = \mathbb{E}[x(t)x(t+\tau)] \quad (2.12)$$

であり、信号  $x(t)$  の自己相関関数である。(2.11) は、ウィーナー・ヒンチンの定理としても知られており、自己相関関数のフーリエ変換がパワースペクトル密度になることを示している。この定義から明らかなようにパワースペクトル密度とは統計量であって、解析的に算出されるのではなく、推定されるものである。

ピリオドグラムはパワースペクトル密度の推定を行う手段として用いられる。一般的にピリオドグラムによる推定値は平均と標準偏差が同じ大きさを持ち、時系列の長さを長くしても推定誤差は改善されない。時系列を長くすることで周波数分解能を高くすることはできるが、個々の周波数でのスペクトルの相対誤差は変化しない。そこで、通常は何らかの平滑化を行って個々の周波数成分の推定値の誤差を減少させる方法が一般に採用される。主な方法として、時間領域で波形を分割し、複数のピリオドグラムの平均を算出する方法がある。時間領域で波形を分割する際には、波形が時間で互いに重なりを持つように分割する、Welch のオーバーラッピング法が用いられる。また最大エントロピー法を用いてパワースペクトル密度を推定する方法もあり、その場合には (2.11) を制約条件としたエントロピー最大化の変分問題を解くこととなる。このパワースペクトル密度推定方法はバーク法として知られている。スペクトル解析に関する記述は書籍 [1] に詳しい。

脳波は、運動想起時に特定の周波数領域でエネルギーが減少する事象関連脱同期が生じるとされているため、適切な電極選定を行い、脳波のパワースペクトル密度  $PSD(\Omega)$  を推定することができれば、とある  $\Omega$  で著しくパワーが減少する様子が確認できる。

## 2.2 多変量解析

多変量解析とは多変量データを統計的に扱う手法である。扱うデータの科学的知識に基づいた特徴量抽出に加え、実データの統計的な性質を考慮した特徴量抽出も様々な分野で行われている。この章では脳波解析で用いられている多変量解析について紹介し、その有用性と限界について考察する。

### 2.2.1 Principal Component Analysis(PCA)

Principal Component Analysis(PCA) は特徴量抽出手法として幅広い分野で活用されている。簡単のため時間平均が 0 の多次元信号  $x(t) \in \mathbb{R}^D$  に対して PCA による特徴抽出を考える。PCA で



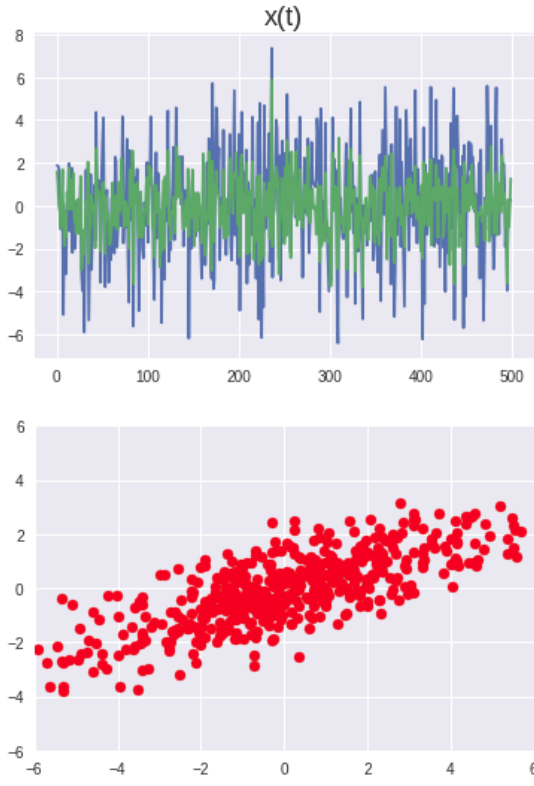


図 2.1:  $x(t)$  の波形と散布図

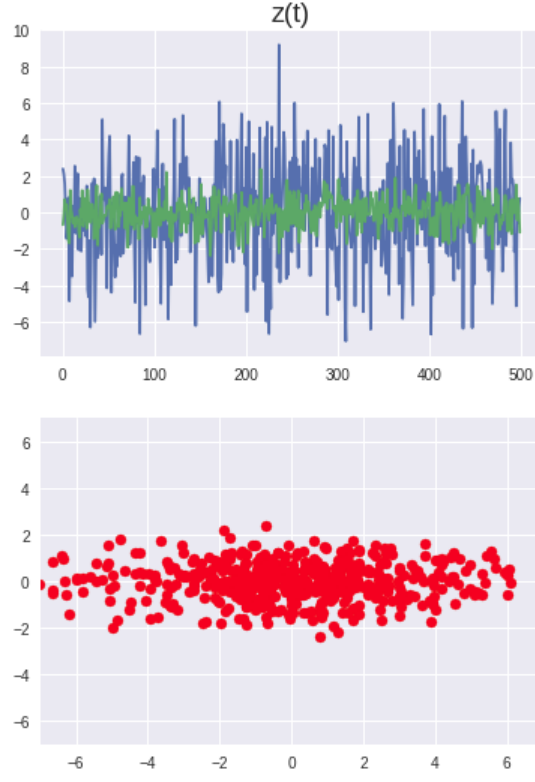


図 2.2:  $z(t)$  の波形と散布図

は変換行列  $W^T \in \mathbb{R}^{D \times D}$  を左から作用させ、特徴量

$$z(t) = W^T x(t) \in \mathbb{R}^D \quad (2.13)$$

を獲得するが、この際、 $z(t)$  の各成分が互いに無相関になるように  $W$  を決定する。 $z(t)$  が無相関となるためには、その分散共分散行列  $\Sigma_z = \mathbb{E}[z(t)z^T(t)]$  が対角行列になることが要請される。ここで  $x(t)$  の分散共分散行列を  $\Sigma_x = \mathbb{E}[x(t)x^T(t)]$  とすると、 $z(t)$  の分散共分散行列  $\Sigma_z$  について

$$\begin{aligned} \Sigma_z &= \mathbb{E}[z(t)z^T(t)] \\ &= \mathbb{E}[W^T x(t)x^T(t)W] \\ &= W^T \mathbb{E}[x(t)x^T(t)]W \\ &= W^T \Sigma_x W \end{aligned} \quad (2.14)$$

と表すことができる。(2.14) が対角行列になるような  $W$  は、 $\Sigma_x$  の固有値分解によって求まる。今、 $\Sigma_x$  の  $D$  個の固有値を  $\lambda_1 \geq \dots \geq \lambda_D$  とする。この固有値を対角成分に並べた行列を  $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_D)$  とすると、 $\Sigma_x$  の固有値分解は、ある  $U$  が存在して以下の形式となる。

$$\Sigma_x = U \Lambda U^{-1} \quad (2.15)$$

$U^{-1}$  を左から、 $U$  を右から掛けることで  $U^{-1} \Sigma_x U = \Lambda$  が得られ、 $U^{-1} = W^T$  で  $U = W$  とすれば (2.14) への要請を直ちに満たす。これは  $W^T = W^{-1}$  という条件が満たされれば良く、実数信号の分散共分散行列  $\Sigma_x$  は一般に正定値実対称行列となっており、直交行列によって固有値分解が可能であるため条件を満たす。また、このとき全ての  $i$  について固有値  $\lambda_i$  は正の値となり、固有値  $\lambda_i$  に属する固有ベクトルにデータを射影した際の分散を表す。これらの数学的な扱いやすさから PCA は非常に広く普及している。

PCA の働きを見るため  $D = 2$  とした場合の線形変換  $z(t) = W^T x(t)$  を図で確認する。図 2.1 は人工的に作成した 2 次元波形  $x(t)$  と各成分の散布図である。この  $x(t)$  に対して PCA を用いると、図 2.2 に示す  $z(t)$  が得られ、散布図から  $z(t)$  の各成分は無相関となっていることが確認できる。図 2.3 は  $x(t)$  に対して PCA を施した場合に得られる新たな直交基底を表しており、 $z(t)$  は新

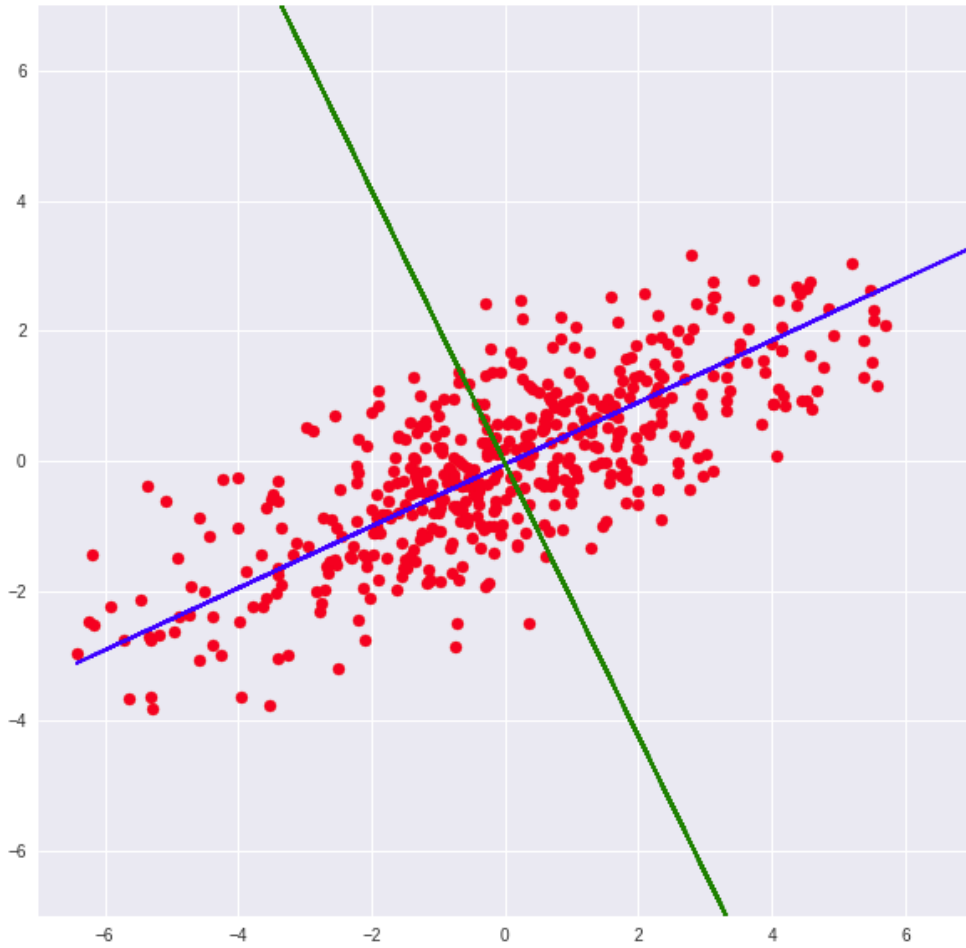


図 2.3: PCA によって得られる基底

たな直交基底に  $x(t)$  を射影したものに他ならない。PCA では基底を取り直すことで各成分が無相関な信号を獲得でき、その結果、新たに得られた信号の各成分がどのような意味を持つのかを考察しやすくなる。

しかし、応用上は単に基底を取り直すことを目的とするケースは少ない。通常は固有値分解によって求めた固有ベクトル（新たな基底）を全て利用するのではなく、値の大きな固有値に属する固有ベクトル  $w$  を  $d(< D)$  個選び、 $W_d = (w_1, \dots, w_d)$  によって変換行列を構成することで、

$$z(t) = W_d^T x(t) \in \mathbb{R}^d \quad (2.16)$$

と次元削減を行う。値の大きな固有値に属する固有ベクトルによって基底を構成することは、 $W_d$  による基底の元で、信号の分散（あるいは振幅）が最大化されることを要請することと等価である。また、射影先での分散最大化に伴い、(2.16) での変換行列  $W_d$  は、元々の信号  $x(t)$  と、 $x(t)$  を  $\mathbb{R}^D$  の部分空間  $\mathbb{R}^d$  へ射影した信号  $z(t)$  との二乗誤差を最小化する変換行列となっている。

以上から PCA は、元々の信号  $x(t)$  の情報損失を二乗誤差の意味で最小限に抑えながら、射影先で大きな変動を有し、かつ各成分が無相関となる特徴量  $z(t)$  を抽出する。しかし、脳波への応用を考える上では PCA の性質は必ずしも有効には働かない。運動想起 BCI を考える上では、脳波に含まれる全ての情報の中から識別したい身体部位に関する情報のみを抽出する必要がある。この場合、脳波信号のごく一部のみが重要である可能性があり、射影先で大きな分散を持つような信号となっているかは定かではないためである。

## 2.2.2 Independent Component Analysis(ICA)

Independent Component Analysis(ICA) は PCA を発展させた比較的新しい信号解析手法である。ICA は PCA と同様に変換行列  $W^T \in \mathbb{R}^{D \times D}$  を左から作用させ、特徴量

$$z(t) = W^T x(t) \in \mathbb{R}^D \quad (2.17)$$

を獲得するが、この際、 $z(t)$  の各成分が互いに独立になるように  $W$  を決定する。独立性は無相関性の十分条件であり、PCA に比べて  $z(t)$  により強い条件を要請する。PCA では無相関性が固有値分解という数学的によく知られた問題と関連していたが、独立性は簡単な問題への定式化は困難であるため、通常は独立性を測る目的関数を設定し、勾配法などの逐次最適化法を用いる。ICA の求解アルゴリズムを述べるのは本研究テーマから逸脱するため、主要なアルゴリズムに絞って簡単に説明する。最も普及している求解アルゴリズムは FastICA として知られており、独立性の必要条件である無相関性を要請した後、独立性を最大化することで解を得る。無相関性を満たすようにするためには前処理として PCA を用いることができる（正確には PCA を用いた中心化と白色化処理が行われる）。前処理後は独立性の最大化問題が、 $z(t)$  の各成分のエントロピー最小化問題に変換されるが、この最小化問題も容易ではない。従って更にネグエントロピーなる量を導入し、最大化問題に書き換える。ネグエントロピーは信号の非ガウス性を測る尺度であり、ガウス分布から遠いほど大きな値となる。FastICA ではネグエントロピーの最大化を各成分ごとに順次取り出していく。この時、非ガウス性が高い成分の順番に取り出されていく。独立成分分析の解説は次の文献が詳しい [2] [3]。

図 2.4 に PCA と ICA の振る舞いをトイデータを用いて比較を掲載する。図 2.4 の base vectors of ICA and PCA から分かるように ICA で得られる基底は直交するとは限らない。また、PCA の基底の大きさは等しく 1 であり正規直交基底を構築するが、ICA では正規性も持つとは限らない。

脳波への応用では、脳波計測時に混入した脳波以外の信号（筋電位、眼電位など）が脳波とは統計的に独立であると考え、脳波以外の信号成分を除去する目的で利用される。一方で運動想起 BCI において身体部位に関する脳波成分を直接抽出することは PCA 同様に難しい。筋電などの場合、脳波と独立であるという仮定は妥当であり、かつ振幅が目視可能なほど脳波に比べて大きくなる。従って、ICA によって分解された独立な成分から筋電などを見分けるのは比較的容易である。しかし、一方で身体部位に関する脳波成分がその他のあらゆる脳波と独立であるかは定かではなく、仮に独立であった場合にも分解された信号から目視によって特定することは困難であると推察される。

## 2.2.3 Blind Source Separation(BSS)

上記では PCA と ICA が次元削減として用いられることを見た。一方でこれらの手法は Blind Source Separation(BSS) 問題の解法として解釈されることも多いため、ここで簡単に述べておく。まず信号源  $s(t) \in \mathbb{R}^d$  を直接観測できない場合に、 $D$  個のセンサで  $x(t) \in \mathbb{R}^D$  という信号を観測したとする。このとき、観測信号  $x(t)$  のみから信号源  $s(t)$  を推定する問題が BSS 問題である。計測機器や環境に応じて、信号源  $s(t)$  は何らかの変換  $f(\cdot)$  を受けて観測されると考えられる。従って  $x(t)$  は

$$x(t) = f(s(t)) \quad (2.18)$$

と表記できる。このときに観測に伴う変換  $f(\cdot)$  が線形変換  $A$  であると仮定した場合、

$$x(t) = As(t) \quad (2.19)$$

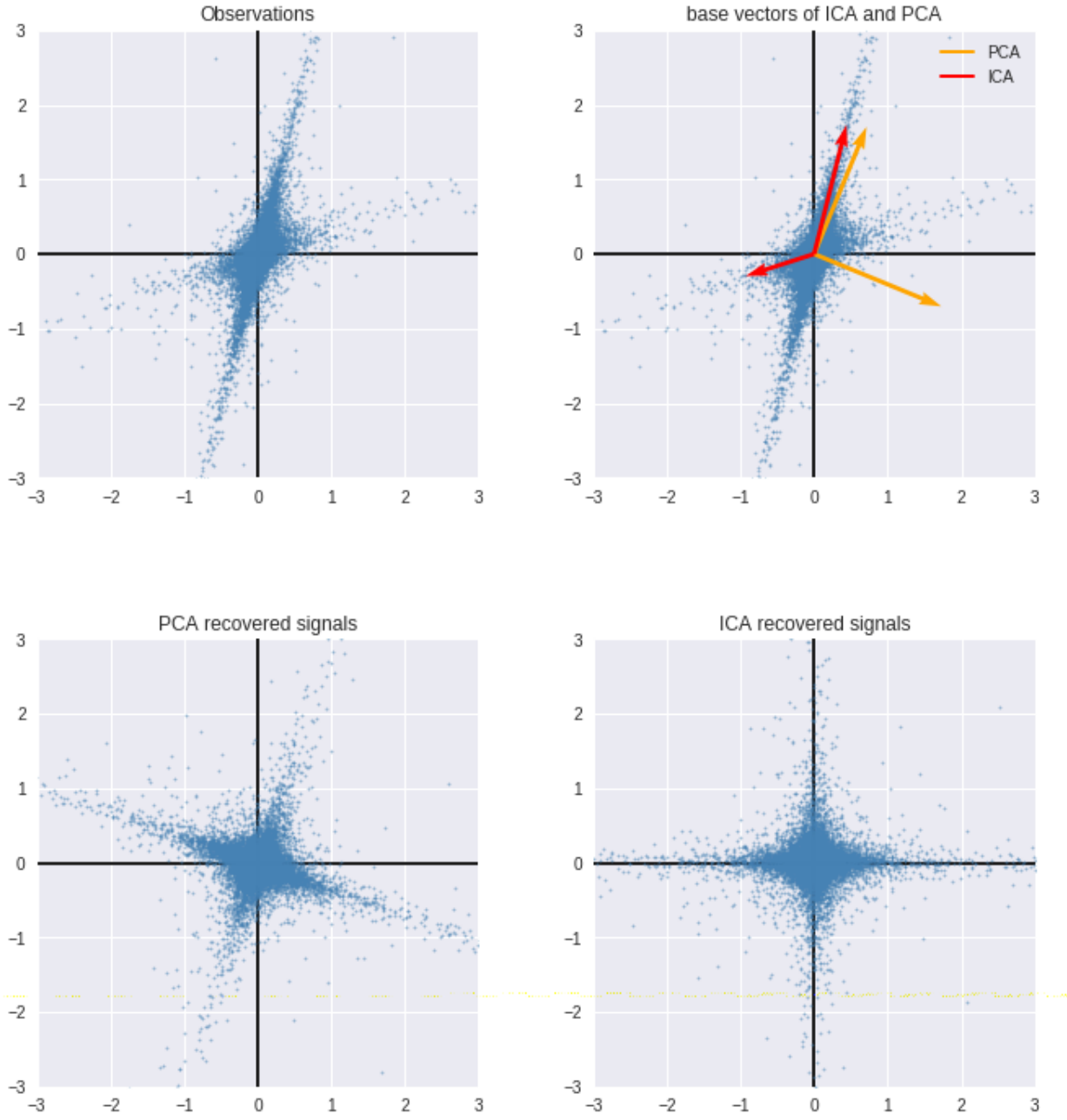


図 2.4: 無相関な基底を得る PCA と独立な基底を得る ICA の比較

と表記することができ、BSS 問題は  $x(t)$  から  $A$  と  $s(t)$  を同時に推定する問題であると見なせる。ここで仮に適当な線形変換によって、観測信号  $x(t)$  を

$$z(t) = Wx(t) \quad (2.20)$$

と変換することを考える。 $W$  を上手く選ぶことに成功すれば、 $z(t) = Wx(t) \simeq s(t)$  となることが期待できる。ここで  $D = d$ 、すなわち信号源の次元と観測信号の次元が一致している場合を考える。このとき (2.19) において、 $A$  が正則であるとし、

$$s(t) = A^{-1}x(t) \quad (2.21)$$

と表すことが可能になる。従って、(2.20) の  $W$  を  $W = A^{-1}$  とすることができれば、

$$z(t) = Wx(t) = A^{-1}x(t) = s(t) \quad (2.22)$$

と信号源を求めることが可能である。ただし、 $z(t) = s(t)$  となる  $W$  が存在するとしても、既知の  $x(t)$  に対して未知の  $W, s(t)$  を求めようとしている状況に変わりはなく、BSS 問題は基本的に不良設定問題である。また、実データでは信号源と観測信号の次元が一致しない場合が多く  $A$  は逆行

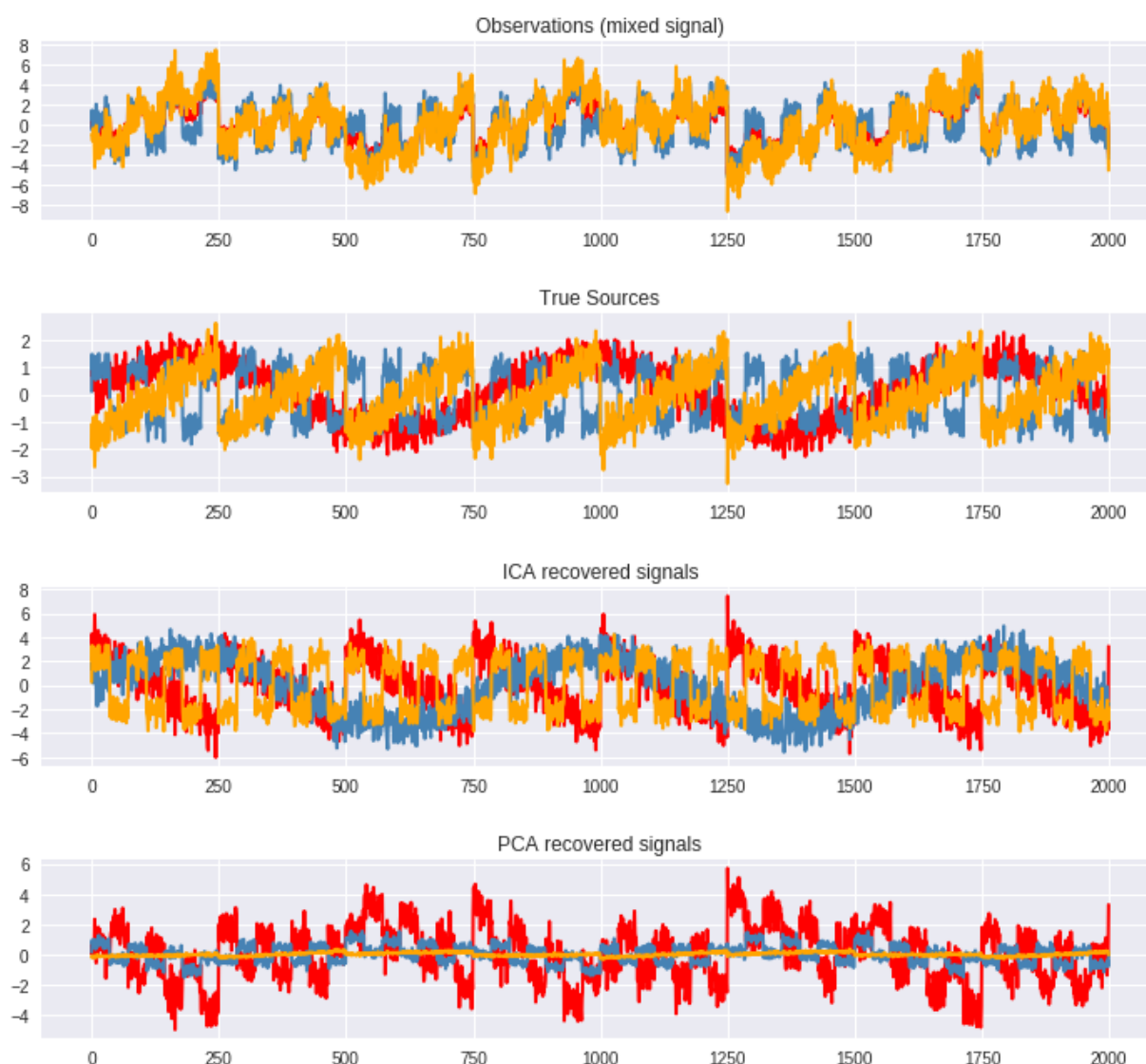


図 2.5: BSS 問題に対する PCA と ICA

列を持たないため、状況はより複雑である場合が多い。通常は BSS 問題を解くためには何らかの条件を追加するか、正則化の手法を導入する必要がある。PCA や ICA は信号源  $s(t)$  が各成分について無相関あるいは独立であると仮定することで条件式を追加し、観測信号  $x(t)$  から条件式を満たすような  $W$  と  $z(t)$  を求め、 $z(t)$  が信号源  $s(t)$  の良い近似になっていると考える BSS 問題の解法の一つである。

PCA と ICA の BSS 問題への振る舞いを確認するために、トイデータによる実験結果を図 2.5 に示す。周波数の異なる 2 つの正弦波と 1 つのノコギリ波にそれぞれガウスノイズを加算した信号源 (図 2.5 の True Sources) を準備し、適当な線形変換を施して観測信号 (図 2.5 の Observations) とする。図 2.5 の ICA recovered signals が FastICA によって推定された信号源に適当なゲインを加えたものであり、図 2.5 の PCA recovered signals が PCA によって推定された信号源に適当なゲインを加えたものである。ICA では周波数の異なる正弦波とノコギリ波を明確に分解できており、信号源に近い波形が得られていることが確認できるが、PCA では信号源と異なる信号が得られている。PCA の振る舞いは各成分を無相関にしつつ、射影先で分散を最大化するような基底を求めるためノコギリ波と位相が一致している正弦波を 1 つの成分に集約してしまっている。

トイデータによる実験では ICA が BSS 問題に対して有効に働くことが確認できるが、本来信号源がどのようなものであるかは未知であるため、信号源推定が正しく行われたかを確認するのは実データでは困難である。運動想起 BCI を想定して脳波に BSS 問題を適用する場合には、観測された信号  $x(t)$  から脳波信号の根源である  $s(t)$  を復元することを目的とするが、ICA によって推定さ

れた信号源のいずれの成分が運動想起と関連しているのかを判別するのは極めて難しい。ただし、脳波と筋電では波形が明らかに異なるため、筋電と脳波が独立であるという仮定を用いて ICA によって筋電成分を取り出すことは可能である。一般に、不良設定問題である BSS 問題に対しては何らかの仮定を置かざるを得なく、複雑な脳波に対して適切な仮定を設けることが重要な課題となる。

## 2.3 分類手法

分類問題の説明を行う。

### 2.3.1 Linear Discriminant Analysis(LDA)

Linear Discriminant Analysis(LDA) は統計分析において伝統的に用いられてきた歴史ある手法である。LDA では多次元データを部分空間で切り取り、切り取った空間で分類超平面を構築することでクラス分類を行う。分類超平面を構築する手段を与えなければ、LDA は特徴量抽出手法としても機能する。まず、多次元データ  $x \in \mathbb{R}^D$  を基底  $w \in \mathbb{R}^D$  へ射影すると、以下の式で表されるスカラー値を獲得できる。

$$z = w^T x \in \mathbb{R} \quad (2.23)$$

$z$  に対してある閾値を設定し、 $z \geq -w_0$  の場合はクラス  $C_1$  とし、そうでない場合はクラス  $C_2$  であるとする事で分類器を獲得できる。多次元データを 1 次元空間へ射影した場合には多くの情報損失が生ずるが、 $w$  の取り方を上手く調整することによって、クラス分類を行いやすい射影を選択できる。まず以下のようにクラス毎の平均ベクトル  $m_1, m_2$  を定義する。

$$m_1 = \frac{1}{|C_1|} \sum_{x \in C_1} x \quad (2.24)$$

$$m_2 = \frac{1}{|C_2|} \sum_{x \in C_2} x \quad (2.25)$$

ここに、 $|C_i|$  はクラス  $C_i$  に属するデータの数である。クラス  $C_1$  とクラス  $C_2$  の平均間の距離が射影先で大きな値となれば、異なるクラスのデータは平均的に離れて配置され、クラス分類を行いやすい射影になっていると想定できる。従って、まずは以下の距離の最大化を考慮する。

$$d = |w^T(m_1 - m_2)| \quad (2.26)$$

しかし実際には (2.26) の最大化を考慮しただけでは分類が上手く行くとは限らない (図 2.6)。射影先での各クラスのデータの分散が大きい場合には、異なるクラスのデータが重なってしまう場合が生じるからである。この問題を解決するためにはデータの分散を考慮する必要がある。まず射影先での各クラスの分散は以下で表記できる。

$$\sigma_1^2 = \sum_{x \in C_1} \{w^T(x - m_1)\}^2 \quad (2.27)$$

$$\sigma_2^2 = \sum_{x \in C_2} \{w^T(x - m_2)\}^2 \quad (2.28)$$

ここで、全データのクラス毎の分散の和を総クラス内分散として以下で定義する。

$$\sigma^2 = \sigma_1^2 + \sigma_2^2 \quad (2.29)$$

総クラス内分散 (2.29) を小さくしながらクラス間の平均の距離 (2.26) を大きくすることを考慮し LDA では以下の評価関数を用いる。

$$\begin{aligned} J(w) &= \frac{d^2}{\sigma^2} \\ &= \frac{\{w^T(m_1 - m_2)\}^2}{\sum_{x \in C_1} \{w^T(x - m_1)\}^2 + \sum_{x \in C_2} \{w^T(x - m_2)\}^2} \end{aligned} \quad (2.30)$$



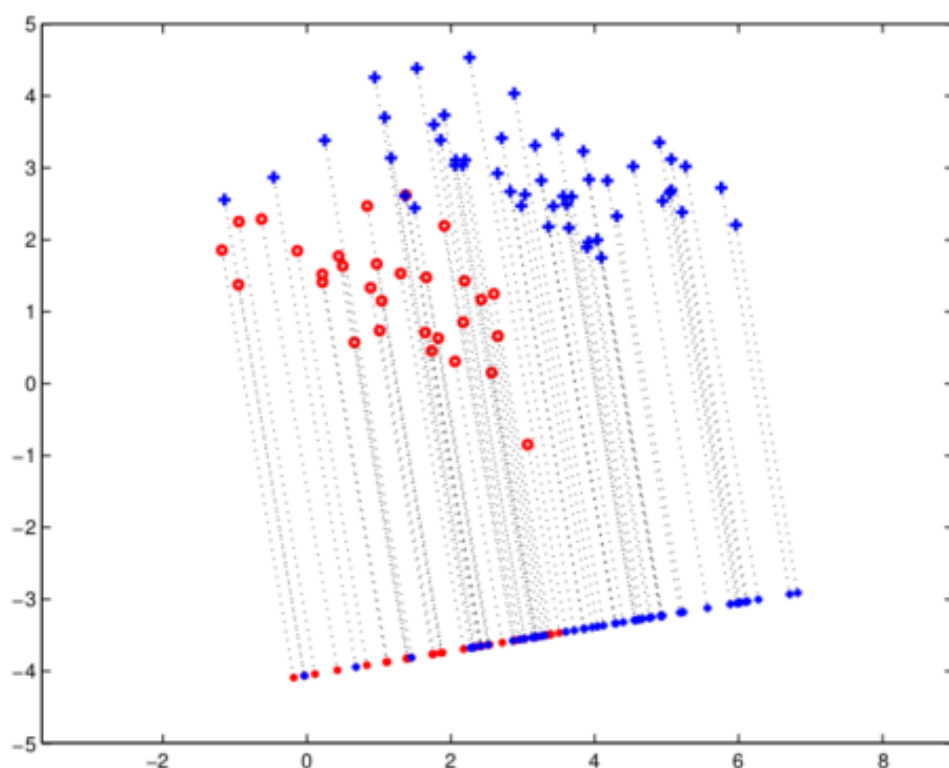


図 2.6: クラス毎の平均間の距離を射影先で最大化した判別分析

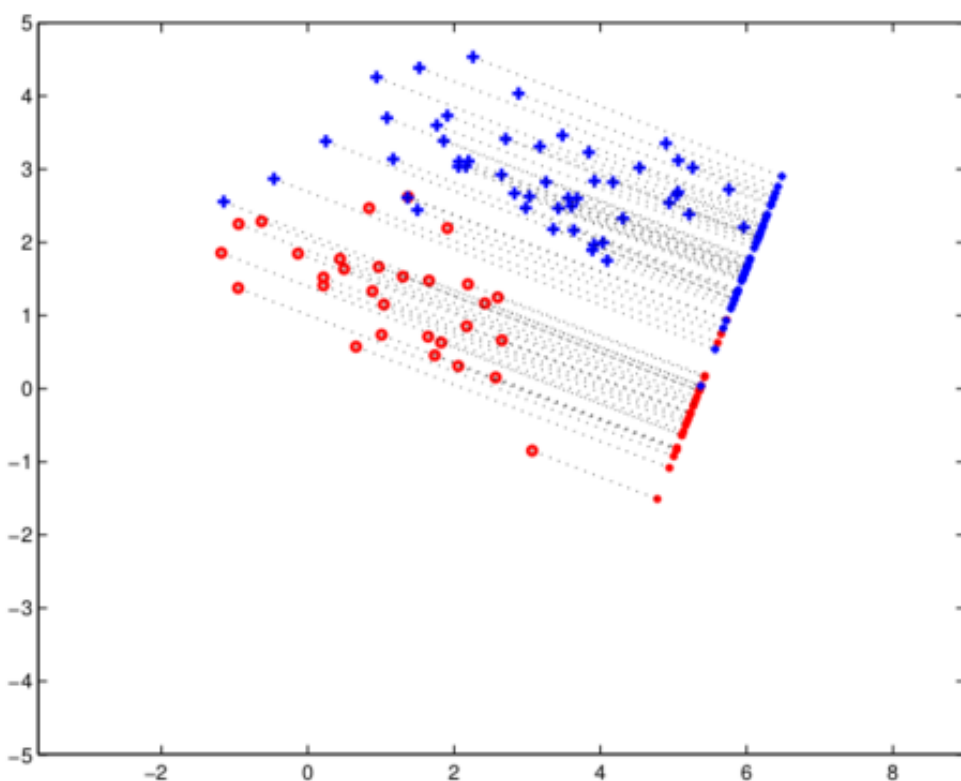


図 2.7: クラス毎の平均間の距離を射影先で最大化し、総クラス内分散を最小化した判別分析

このクラス間の平均とクラス内の分散を考慮した評価関数を用いることで、射影先でデータがクラス毎に小さくまとまり、かつ異なるクラスのデータがなるべく離れるようになる (図 2.7)。

1 次元の空間 (数直線上) でクラス毎にデータが上手く分離できた後、数直線上に閾値  $w_0 = (m_1 + m_2)/2$  を設けることで分離平面が得られる。これはクラス毎の平均値の平均値によって分離平面を設定したことに相当する。しかし、射影のされ方によってはこの閾値は適当ではない。より精密な分類を行うためには、 $z = w^T x$  がクラス毎に異なるガウス分布から生じる確率変数だと

考え、クラス  $C_i$  の条件付き確率  $p(z|C_i)$  を算出した判別基準を設け、条件付き確率が大きいクラスへ分類するなどの方法を取る。

また LDA の評価関数 (2.30) は以下のように変形が可能である。

$$\begin{aligned}
 J(w) &= \frac{\{w^T(m_1 - m_2)\}^2}{\sum_{x \in C_1} \{w^T(x - m_1)\}^2 + \sum_{x \in C_2} \{w^T(x - m_2)\}^2} \\
 &= \frac{w^T(m_1 - m_2)(m_1 - m_2)^T w}{w^T \left\{ \sum_{x \in C_1} (x - m_1)(x - m_1)^T + \sum_{x \in C_2} (x - m_2)(x - m_2)^T \right\} w} \\
 &= \frac{w^T S_B w}{w^T S_W w}
 \end{aligned} \tag{2.31}$$

ここで

$$S_B = (m_1 - m_2)(m_1 - m_2)^T \tag{2.32}$$

$$S_W = \sum_{x \in C_1} (x - m_1)(x - m_1)^T + \sum_{x \in C_2} (x - m_2)(x - m_2)^T \tag{2.33}$$

である。このとき  $S_B$  をクラス間共分散行列、 $S_W$  をクラス内共分散行列と言う。(2.31) を  $w$  に関して微分して 0 と置くことで問題は解析的に解くことができる。また問題は一般化固有値問題となり、複数の固有ベクトルを用いて多次元の特徴量を獲得することも可能である。

## 2.3.2 Support Vector Machine(SVM)

脳波の分類では Support Vector Machine(SVM) の応用例もある。基本的に SVM はマージン最大化の考えによって汎化性能の向上に成功した 2 クラス分類のための線形分類器である。まずマージン最大化という概念について説明する。マージンとは端的に述べるとデータ点と分類超平面との距離のことを表す。学習データに対してマージンを最大化することで、学習データが空間上で僅かに移動した際にも誤分類を起こしづらくすると期待できる。SVM ではこのマージン最大化によって以下の分類超平面を定める。

$$y(x) = w^T x + w_0 \tag{2.34}$$

ここに  $x$  は  $D$  次元のデータベクトルであり、 $w$  は  $D$  次元のパラメータベクトルである。 $w_0$  もスカラーパラメータであり閾値の役割を担う。分類面の役割により  $y(x)$  は  $x$  がクラス  $C_1$  に属する場合には正の値を、 $C_2$  に属する場合には負の値を取るよう学習される。ここで、(2.34) の超平面と、あるデータ点  $x_n$  との距離は以下で表される。

$$|r| = \frac{|y(x_n)|}{|w|} \tag{2.35}$$

ここで、 $x_n$  がクラス  $C_1$  に属する場合は  $t_n = 1$  とし、クラス  $C_2$  に属する場合には  $t_n = -1$  と定めた  $t_n$  を導入する。さらに  $x_n$  には分類面から最も近いデータ点のみを考慮することとし、そのときの  $|r|$  をマージンと呼び以下で表す。

$$|r|_{margin} = \min_{x_n} \frac{t_n y(x_n)}{|w|} \tag{2.36}$$

この (2.36) を最大化するようにパラメータを決定することでマージン最大化を実現することができる。従って、SVM のパラメータ決定は以下の最適化問題によって定式化される。

$$\arg \max_{w_0, w} \left( \min_{x_n} \frac{t_n y(x_n)}{|w|} \right) \tag{2.37}$$

しかしこの最適化問題において、 $w, w_0$  の大きさは本質的ではない。なぜなら  $w, w_0$  を同時に  $kw, kw_0$  と定数倍した場合にも (2.36) の値は変化しないためである。従って、 $w, w_0$  の大きさ



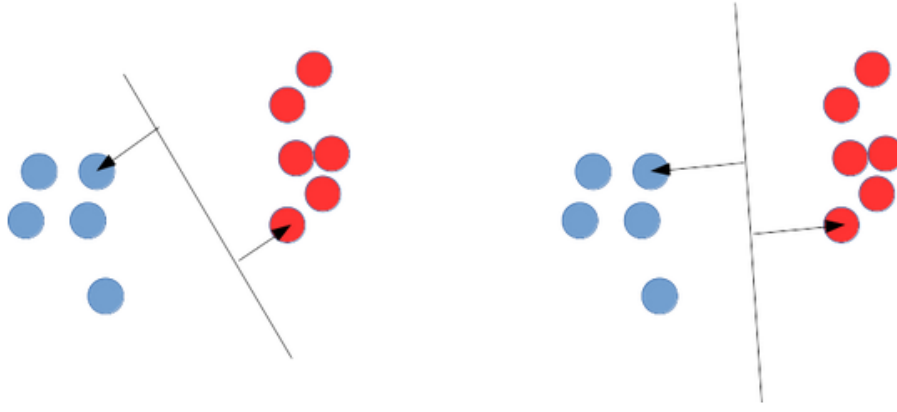


図 2.8: 分類面によってマージンの大きさが異なる様子

に関して制約を設ける必要がある。そこで分類面から最も近い  $x_n$  に関して  $t_n(w^T x + w_0) = 1$  となるような制約を  $w, w_0$  に対して要請する。この条件式に伴って、任意のデータ点において  $t_n(w^T x + w_0) \geq 1$  という制約が与えられる。最終的にマージン最大化問題 (2.37) は以下で定式化される。

$$\begin{aligned} \arg \min_{w_0, w} \quad & \frac{|w|^2}{2} \\ \text{s.t.} \quad & t_n(w^T x + w_0) \geq 1 \end{aligned} \quad (2.38)$$

目的関数の分母は、単に勾配を計算する際に約分できるというテクニックによるものである。分子の二乗は、最適化問題の解を変更せずに勾配計算などを容易に行うための変形である。図 2.8 に分類面の定め方によりマージンが異なっている様子を見ることができる。左右いずれの図も学習データに対して正しく分類が行える分類面になっているが、新規のデータに対しての分類結果が異なってくる。SVM では右図の分類面の方が優れていると考える場合に用いる手法である。ここまで線形分類を行う場合の SVM を見てきたが、一般的に線形分類器はデータ点を非線形関数  $\phi(\cdot)$  によって別の特徴空間へ写像し、特徴空間上で線形分類を行う問題へ拡張することができる。SVM では分類超平面を以下の式によって構築することに相当する。

$$y(x) = w^T \phi(x) + w_0 \quad (2.39)$$

この場合においてもこれまでと同様の議論で最適化問題を以下のように定式化できる。

$$\begin{aligned} \arg \min_{w_0, w} \quad & \frac{|w|^2}{2} \\ \text{s.t.} \quad & t_n(w^T \phi(x) + w_0) \geq 1 \end{aligned} \quad (2.40)$$

実データは線形分離不可能な場合が多いため、通常 SVM を用いる場合は上記のような非線形に拡張されたものを用いる。また実データは異なるクラスのデータが重なって分布することも多々あるため、厳密な分類を行うことは不可能な場合が多い。そういった場合に対応したソフトマージンと呼ばれる考えがあり、学習データの誤分類に対して寛容になる指標を導入する。このソフトマージンの考え方は機械学習で過学習抑制に用いられる正則化の考えと本質的には変わらない。また、制約付き最適化問題をラグランジュ法によって変形することで双対問題を獲得することができる。双対問題においては非線形変換後の空間での内積のみが必要となり、具体的な非線形変換の計算をデータ  $x$  に対して実施する必要はない。すなわち、非線形変換  $\phi(x)$  による特徴空間への写像を具体的に考える代わりに、最終的に計算の必要性がある内積  $\phi(x)^T \phi(x')$  を定義することで SVM の非線形への拡張が可能である。このような方法はカーネル法として知られており、このときに用いられる内積計算をカーネル  $k(x, x') = \phi(x)^T \phi(x')$  と呼ぶ。カーネルを定めることが特徴空間の設計を行うことに相当するが、BCI も含めた多くの応用では既に知られた優れた性質を持つカーネルを活用することがほとんどである。以下に代表的なカーネルについて記載する。

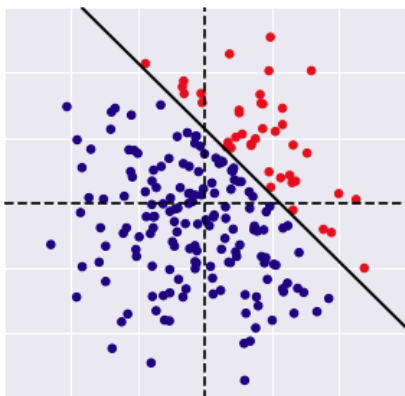


図 2.9: 線形 SVM (線形分離可能)

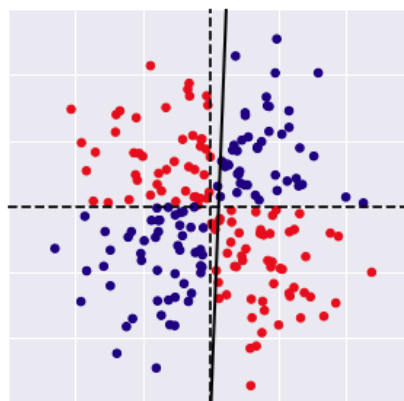


図 2.10: 線形 SVM (線形分離不可能)

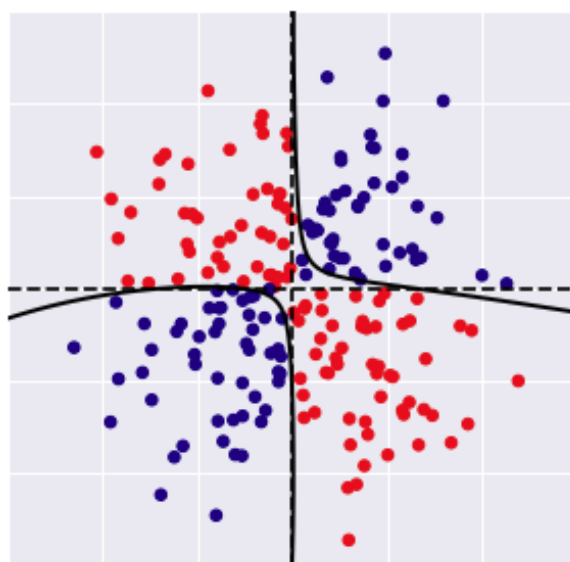


図 2.11: ガウシアンカーネル SVM (線形分離不可能)

- ・線形カーネル：非線形変換を行わないことに対応。

$$k(x, x') = x^T x'$$

- ・多項式カーネル：多項式関数による非線形変換に対応。

$$k(x, x') = (x^T x' + c)^M$$

- ・ガウス基底カーネル：特徴空間が無限次元となる。

$$k(x, x') = \exp\left(-\frac{|x-x'|^2}{2\sigma^2}\right)$$

通常、カーネルを用いる場合にはハイパーパラメータが付随する。多項式カーネルの場合はスカラーの  $c, M$ 、ガウス基底カーネルの場合はスカラーの  $\sigma$  がハイパーパラメータとなり、これらの調整次第で得られる分類面は異なる。応用上はソフトマージンカーネル SVM を用いればよく、ソフトマージンのハイパーパラメータとカーネルの設計を変えることで通常の線形 SVM の働きをさせることも可能である。図 2.9-2.11 に 2 次元データに対する SVM の分類境界を示す。図 2.10 の通り、線形 SVM では線形分離不可能な問題に対して不適切な境界を設ける。

ソフトマージンカーネル SVM の識別関数は以下で表される。

$$y(x) = \sum_{n=1}^N a_n t_n k(x, x_n) + b \quad (2.41)$$

ここで学習データの数  $N$  としている。また  $a_n$  はスカラーのパラメータであり、ラグランジュ法

で双対問題を考えた際のラグランジュ乗数である。また最適化問題は以下で定式化される。

$$\begin{aligned} \arg \max_{a_1, \dots, a_N} \quad & \sum_{n=1}^N a_n - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N a_i a_j t_i t_j k(x_i, x_j) \\ \text{s.t.} \quad & 0 \leq a_n \leq C \\ & \sum_{n=1}^N a_n t_n = 0 \end{aligned} \quad (2.42)$$

ここに  $C$  はソフトマージンのハイパーパラメータでありスカラーである。値が小さいほど  $a_n$  の取れる範囲が制限され、値が大きいくほど誤分類に寛容となる。最適化問題の解である  $a_1, \dots, a_N$  と任意の  $n$  を選んで以下の条件式に代入することで閾値  $b$  も求まる。

$$t_n \left( \sum_{m=1}^N a_m t_m k(x_n, x_m) + b \right) = 1 \quad (2.43)$$

ただし、(2.43) は無駄な計算も含まれている。 $a_n = 0$  となるような  $x_n$  に対して  $\sum_{n=1}^N a_n t_n k(x, x_n)$  の計算を実行する必要はない。 $a_n \neq 0$  となっている  $x_n$  のことをサポートベクトルと呼び、実際にはサポートベクトルのみ計算に考慮すれば良い。このことは新規のデータに対して (2.41) の計算を行うときも同様である。従って学習後に保持しておかねければならないデータはサポートベクトルのみに限定でき、実用上省メモリに貢献できる。

### 2.3.3 Logistic Regression(LR)

Logistic Regression(LR) は対数オッズ比を線形モデルで表現した分類手法である。クラス分類においてデータ  $x$  がクラス  $C_i$  に属する確率を  $p(y = C_i | x)$  と表す。この時、事象  $y = C_i$  のオッズ比は以下で表される。

$$\text{odds}(y = C_i) = \frac{p(y = C_i | x)}{1 - p(y = C_i | x)} \quad (2.44)$$

2 クラス分類においては  $\text{odds}(y = C_1) \geq 1$  であれば  $C_1$  に分類するなどの規則を設けることで分類が可能になる。ここで、対数オッズ比は以下で表される。

$$\text{logodds}(y = C_i) = \log \left\{ \frac{p(y = C_i | x)}{1 - p(y = C_i | x)} \right\} \quad (2.45)$$

(2.44) が 1 以上の値となる時、(2.45) は 0 以上の値となる。LR では多次元データ  $x \in \mathbb{R}^D$  に対して  $w \in \mathbb{R}^D$  として、線形モデル  $z = w^T x \in \mathbb{R}$  によって対数オッズ比を出力するモデルを構築する。 $y = C_1$  の対数オッズ比を線形モデルで表現すると以下で表される。

$$\log \left\{ \frac{p(y = C_1 | x)}{1 - p(y = C_1 | x)} \right\} = w^T x \quad (2.46)$$

この時、以下の式変形によってよく知られたシグモイド関数  $\sigma(z) = 1/(1 + \exp(-z))$  が導出される。

$$\begin{aligned} & \log \left\{ \frac{p(y = C_1 | x)}{1 - p(y = C_1 | x)} \right\} = w^T x \\ \iff & \frac{p(y = C_1 | x)}{1 - p(y = C_1 | x)} = \exp(w^T x) \\ \iff & \frac{1}{p(y = C_1 | x)} - 1 = \exp(-w^T x) \\ \iff & p(y = C_1 | x) = \frac{1}{1 + \exp(-w^T x)} \end{aligned} \quad (2.47)$$

シグモイド関数に関しては、

$$\sigma(-z) = 1 - \sigma(z) \quad (2.48)$$

$$\sigma'(z) = \sigma(z)(1 - \sigma(z)) \quad (2.49)$$

という性質が知られている。ここで  $y = C_1$  という事象を  $y = 1$  に対応させ、 $y = C_2$  という事象を  $y = -1$  に対応させると、(2.47) と (2.48) を用いて、

$$p(y \mid x, w) = \sigma(yw^T x) \quad (2.50)$$

とすることができる。i.i.d. を仮定して対数尤度の最尤推定を考えると以下のロジスティック損失の最小化問題に帰着される。ここで、 $N$  は訓練データ数とした。

$$\begin{aligned} \arg \max_w \sum_{i=1}^N \log \{\sigma(y_n w^T x_n)\} &= \arg \min_w \sum_{i=1}^N -\log \{\sigma(y_n w^T x_n)\} \\ &= \arg \min_w \sum_{n=1}^N \log \left\{ \frac{1}{\sigma(y_n w^T x_n)} \right\} \\ &= \arg \min_w \sum_{n=1}^N \log \{1 + \exp(-y_n w^T x_n)\} \end{aligned} \quad (2.51)$$

以上から、LR の学習はオッズ比を対数線形モデルで表現し最尤推定を行うことに相当する。導出の過程から明らかなようにシグモイド関数は非線形変換であるが、スケールの変換を行うだけで線形分離不可能な問題に対応できるわけではなく、モデルとしては単層パーセプトロンと等価である。目的関数は凸であり、学習は勾配法、ニュートン法、準ニュートン法などが用いられる。

## 3

# 従来の運動想起型 BCI

この章ではまず、BCIのために開発された解析手法である Common Spatial Pattern について紹介する。その後、前章の要素技術を含めた従来の運動想起型 BCI の構成方法について述べ、課題点を明らかにする。

## 3.1 特徴量工学

図 3.1 と図 3.2 はそれぞれ足の運動想起と手の運動想起を行った際の脳波である。いずれも 0 秒以前は以前はディスプレイを注視した状態であり、0 秒以降に運動想起を 4 秒間行っている。電極の個数は 64 個用いられており、全ての電極の波形が表示されている。足の運動想起が行われているのか手の運動想起が行われているのかを脳波の生データから識別するのは困難であることが分かる。運動想起 BCI の標準的な役割は、脳波信号から運動想起部位を識別することであるが、そのためには脳波の生データに対して何らかの処理を施し、識別に有用な特徴を見出さねばならない。

そこで、何らかの変換  $f(\cdot)$  を脳波信号  $x(t)$  に対して施すことで、特徴量  $z = f(x(t))$  の獲得を目指すのが特徴量工学である。

### 3.1.1 頭皮領域と空間フィルタ

まず初めに運動想起 BCI に必要な電極について考慮する必要がある。例えば足の動作に関する脳活動が脳波として観測される場合、頭皮上の全ての領域が対等に重要であるとは考えにくい。足の動作に関する脳の領域は頭頂部に存在するため、脳波として観測される場合、頭頂部に位置する Cz 電極が最も関係していると考えられる。図 3.4 は足の運動想起を行った際の Cz 電極によって記録された脳波である。しかし波形から足の運動想起を行っていることを判別する決定的な差異を見出すことは困難である。理由としては、脳の頭頂部での神経活動によって生じている電位が、必ずしも頭皮上の頭頂部にのみ伝搬するとは限らないからである。電位を発生させている神経活動と電位を計測している電極との間には、頭蓋骨と皮膚があり、電位の伝搬がどのように行われているかを完全に把握することは難しい。ある程度空間的に広がりを持って電位が伝搬していると考えるのが自然である。そこで計測された脳波に対して、スモールラプラシアンフィルタと呼ばれる空間フィルタが用いられることがある。ラプラシアンフィルタは一般的に 2 次元配列  $X(i, j)$  に対して以下の計算によって二次差分を計算したものである。

$$X(i, j) = -4X(i, j) + X(i-1, j) + X(i+1, j) + X(i, j-1) + X(i, j+1) \quad (3.1)$$

二次差分フィルタは画像処理では輪郭検出のために用いられる処理であり、空間的に配置された値が際立って変化する位置で大きな値を持つ配列を返す。脳波に対してラプラシアンフィルタを適用する場合は、頭皮上の電極が 2 次元的に配置されていると考え計算を行う。例として、A 電極によって計測された脳波を  $x_A(t) \in \mathbb{R}$  と表記すると、Cz 電極に対するスモールラプラシアンフィル

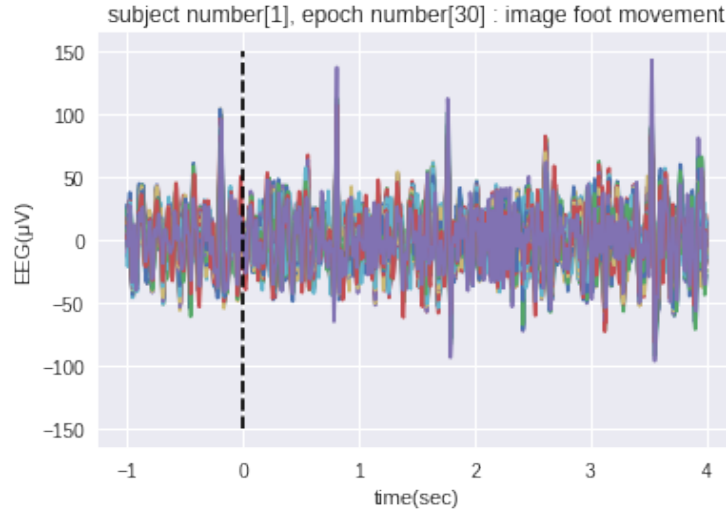


図 3.1: 0 秒～4 秒間に足の運動想起を行った際の脳波 (EEG)

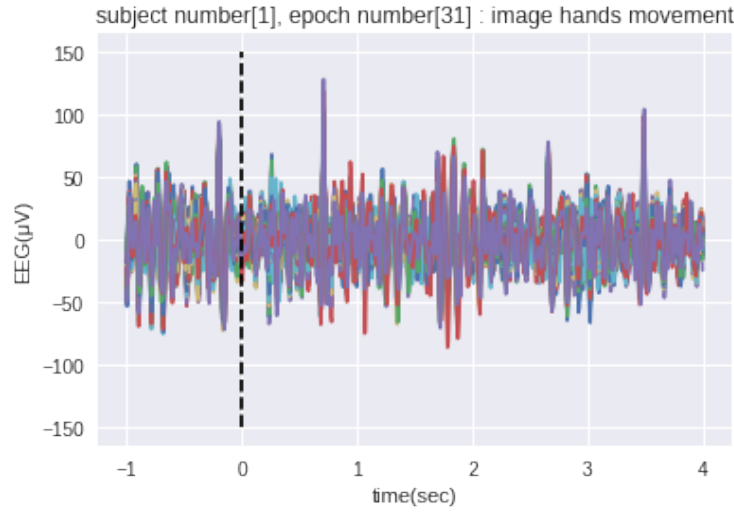


図 3.2: 0 秒～4 秒間に手の運動想起を行った際の脳波 (EEG)

タが適用された脳波  $x_{smallCz}(t)$  は以下で表される。

$$x_{smallCz}(t) = x_{Cz}(t) - \frac{1}{4}(x_{C1}(t) + x_{C2}(t) + x_{FCz}(t) + x_{CPz}(t)) \quad (3.2)$$

信号に対する符号や絶対値については、多くの場合は着目した電極の係数の符号を正とした計算が行われる。図 3.4 は Cz 電極に対してスモールラプラシアンフィルタを用いた際の脳波である。スモールラプラシアンフィルタの処理によって必ずしも有効な特徴量が得られるとは限らないため、適切な空間フィルタを獲得するために統計的なフィルタ設計方法が用いる場合もある。ここに画像処理で空間フィルタという表現を使う場合、2 次元配列に於ける計算を考えるのが一般的であるが、脳波に対して空間フィルタという表現を用いる場合は、電極に対する重み付けを行う処理全般を指す。従って、電極を  $D$  個用いて脳波  $x(t) \in \mathbb{R}^D$  を計測した場合に、 $w \in \mathbb{R}^D$  によって

$$z(t) = w^T x(t) \in \mathbb{R} \quad (3.3)$$

を計算した場合は  $w$  を空間フィルタと呼ぶ。より一般的には行列  $W = (w_1, w_2, \dots, w_d) \in \mathbb{R}^{D \times d}$  によって、

$$z(t) = W^T x(t) \in \mathbb{R}^d \quad (3.4)$$

を計算する場合にも空間フィルタを用いたと言える。この場合は、 $W$  の各列が空間フィルタ 1 つ 1 つに対応し、 $d$  個の空間フィルタについて一挙に計算を行うことが可能である。特に Independent Component Analysis(ICA) による空間フィルタの設計は既に脳波解析で一定の地位を築いている。

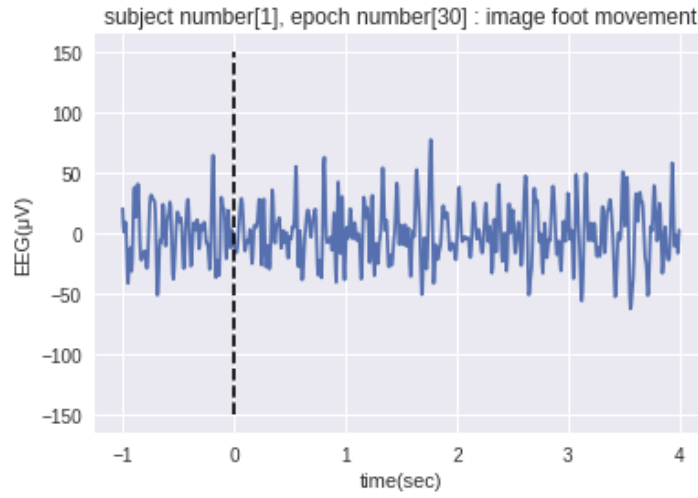


図 3.3: 0 秒～4 秒間に足の運動想起を行った際の Cz 電極の波形

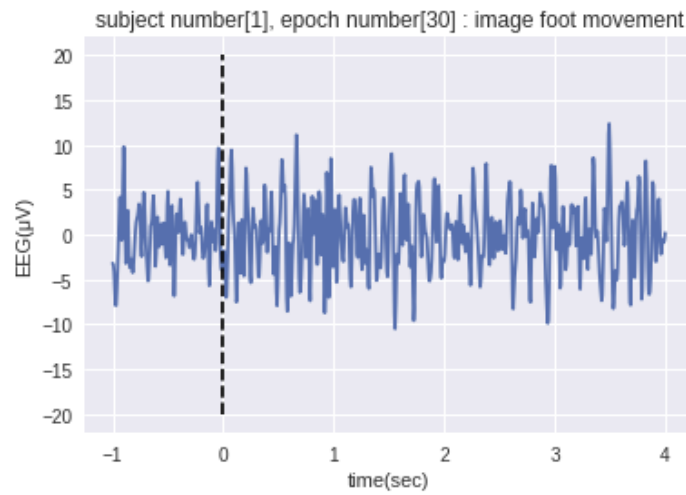


図 3.4: スモールラプラシアンフィルタを適用した際の Cz 電極の波形

また脳波解析ソフトウェアの機能の 1 つとして実装されているケースが多く、書籍 [4] では、1 章を割いて ICA の応用方法について解説がなされている。また、Common Spatial Pattern(CSP) は運動想起 BCI に対する非常に有効な空間フィルタ設計手法として考えられており、数多くの派生手法が存在するため、3.2 で詳しく述べる。

### 3.1.2 脳波と時間周波数解析

脳波には 波、 波、 波、 波などが存在するが、これらは周波数帯域で分類がなされており、それぞれの波の振幅が精神状態に応じて時間変化する。特にヒトが手や指、足等の運動およびそのイメージを行うと、 律動や 律動の ERD および ERS が惹き起こされ、ERD は運動およびそのイメージ中に、ERS は運動およびそのイメージ後に発生する [5]。従って脳波に対して時間周波数解析を行い、ERD や ERS を直接的に検知することで運動想起 BCI を構築することが可能である [6]。ERD や ERS を検知するための時間周波数解析には、最も単純なものとして短時間フーリエ変換を用いることができる (図??)。またウェーブレット変換を用いる方法やバーク法に基づく方法もあり、これらの比較についても研究報告がある [7]。更に、近年は Empirical Mode Decomposition(EMD) を用いる方法もいくつか提案されている [8,9]。

いずれの時間周波数解析を用いるとしても、その変換  $TFA(\cdot)$  は時間  $t$  の関数  $x(t)$  から、時間  $t$  と周波数  $f$  の関数  $X(t, f)$  への変換として表される。

$$X(t, f) = TFA(x(t)) \quad (3.5)$$



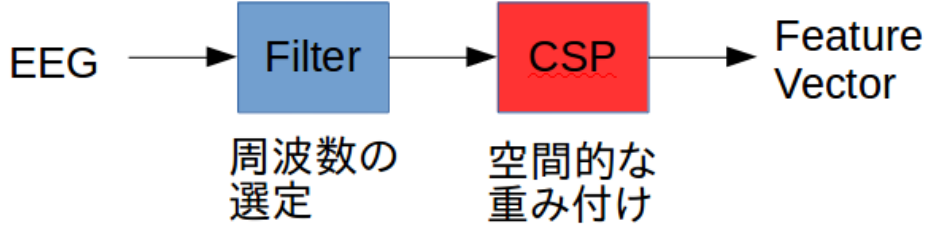


図 3.5: CSP の利用方法

この時、ERD が生じているのならば  $|X(t, f)|^2$  がある区間  $[f_{low}, f_{high}], [t_{start}, t_{end}]$  で著しく減少しているはずである。 $|X(t, f)|^2$  に顕著な変化を見出させるような変換  $TFA$  を見つけ出すことが、時間周波数領域における特徴量抽出に相当する。

## 3.2 Common Spatial Pattern とその派生手法

運動想起型 BCI において非常に活発に用いられている “Common Spatial Pattern(CSP)” は 1990 年に Zoltan [10] により提案されてから、数々の派生手法が生み出されてきた [11–17]。このセクションでは CSP の基本的な概要と、BCI に適した形で発展してきた CSP の派生手法について述べる。

### 3.2.1 脳波信号の定式化

これまで多次元の信号を  $x(t) \in \mathbb{R}^D$  と表記し、連続時間信号として扱ってきた。しかし、通常は計測された脳波はコンピュータで処理するために離散時間信号  $x_n \in \mathbb{R}^D$  に変換される。従って、以降、脳波信号を離散時間信号として取り扱う。また、運動想起時の脳波信号を計測する際には、被験者に対して定められたタイムスケジュールで運動想起を行うように指示がなされる。例として 64 個の電極を用い、サンプリング周波数 100Hz で 10 秒間の計測を行った場合、計測された運動想起 1 回分の脳波信号は  $X \in \mathbb{R}^{64 \times 1000}$  と表すことができる。従って、以降統一のため、 $M$  を電極の個数、 $N$  を計測時間点数とした場合の脳波信号を以下で定義する

$$X = (x_1, \dots, x_N) \in \mathbb{R}^{M \times N} \quad (3.6)$$

運動想起を  $K$  回行った場合には、 $K$  個の  $X$  が得られる。通常は運動想起時の脳波を数個から数十個集め、統計的な指標を元に有用な特徴量を抽出する。CSP は運動想起 BCI に対して極めて有効に働くとされている特徴量抽出手法である。

### 3.2.2 Common Spatial Pattern(CSP)

CSP は複数の電極で計測されることが一般的である脳波に対して、適切な電極の重み付けを行うことで、重要な頭皮領域の脳波を取り出す手法である。CSP を脳波に用いる際は、脳波信号  $X$  を直接扱うのではなく、何らかの前処理を施した信号  $\hat{X} = \mathcal{H}(X)$  を用いる。通常、 $\mathcal{H}$  には、運動想起に関連のある周波数帯域のみを通過させるバンドパスフィルタを用いる (図 3.5)。バンドパスフィルタ通過後の脳波信号を以下のように表記する。

$$\hat{X} = (\hat{x}_1, \dots, \hat{x}_N) \in \mathbb{R}^{M \times N} \quad (3.7)$$

$\hat{x}_i \in \mathbb{R}^M$  における添字  $i$  はサンプル時刻の添え字である。CSP では、新たな基底  $w \in \mathbb{R}^M$  にバンドパスフィルタ通過後の脳波信号を射影し、スカラー時間信号である  $z = w^T \hat{X} \in \mathbb{R}^N$  を抽出す



る。この時の  $w$  の決め方を以下に記す。

まず  $\hat{X}$  を基底  $w$  に射影した際の時間分散  $\sigma^2(\hat{X}, w)$  を以下で定義する。

$$\sigma^2(\hat{X}, w) = \frac{1}{N} \sum_{i=1}^N \left| w^T \left( \hat{x}_i - \frac{1}{N} \sum_{j=1}^N \hat{x}_j \right) \right|^2 \quad (3.8)$$

ここで計測された複数個の  $\hat{X}$  は必ず集合  $C_1$  か  $C_2$  のいずれか一方に属するとし、 $C_1 \cap C_2 = \phi$  であるとする。CSP では、ベクトル  $w \in \mathbb{R}^M$  を新たな基底とした電極空間において、一方のクラスに属する信号  $\hat{X} \in C_d$  についての時間分散 (3.8) が最大となるように、 $w$  を決める。これは以下の最大化問題によって定式化される。

$$\begin{aligned} \max_w \quad & \mathbb{E}_{X \in C_1} [\sigma^2(\hat{X}, w)] \\ \text{s.t.} \quad & \sum_{d=1,2} \mathbb{E}_{X \in C_d} [\sigma^2(\hat{X}, w)] = 1 \end{aligned} \quad (3.9)$$

最大化問題 (3.9) を解いて得られる  $w_{csp}$  は、クラス  $C_1$  に属する脳波の分散を最大化するような基底である。一方で、制約条件によって 2 つのクラスの分散の和が 1 であるとされているため、自動的にクラス  $C_2$  に属する脳波の分散を最小化する基底ともなる。すなわち  $w_{csp}$  によって得られるスカラー信号は一方のクラスの信号のみを増幅させ、他方を減衰させる働きをする。(3.9) は更に以下で定式化することができる。

$$\begin{aligned} \max_w \quad & w^T \Sigma_1 w \\ \text{s.t.} \quad & w^T (\Sigma_1 + \Sigma_2) w = 1 \end{aligned} \quad (3.10)$$

ここで

$$\Sigma_i = \mathbb{E}_{X \in C_i} \left[ \frac{\hat{X} \hat{X}^T}{\text{tr}(\hat{X} \hat{X}^T)} \right] \quad (3.11)$$

である。この問題の解はラグランジュ法によって解析的に求めることが可能である。簡単な計算によって (3.10) は以下の一般化固有値問題に帰着される [15]。

$$\Sigma_1 w = \lambda (\Sigma_1 + \Sigma_2) w \quad (3.12)$$

この一般化固有値問題は右辺の  $\Sigma_1 + \Sigma_2$  が正則であれば、その逆行列を両辺左から掛けることで普通の固有値問題に変形できる。

一般化固有値問題が解けた時の  $M$  個の固有値を  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_M$  とする。また、 $\lambda_i$  に属する固有ベクトルを  $w^{(i)}$  とする。このとき最適化問題の解に相当するのは  $w^{(1)}$  であり、 $C_1$  に属する脳波を増幅し、 $C_2$  に属する脳波を減衰させたスカラー信号を獲得できる。一方で  $w^{(M)}$  は  $C_1$  に属する脳波を減衰し、 $C_2$  に属する脳波を増幅させたスカラー信号を獲得できる。従って通常は、 $w^{(1)}, w^{(M)}$  の固有ベクトルを対で用いる。同様に  $w^{(2)}, w^{(M-1)}$  を対で取り出して 4 つの固有ベクトルを用いることもできる。一般に  $2m (< M)$  個の固有ベクトルを使うこととすれば、特徴量  $\sigma^2(\hat{X}, w^{(i)})$  を  $2m$  個準備することができる。ここで単に  $z = w^{(i)} \hat{X}$  を用いないのは、CSP によって得られる特徴量は振幅に大きな違いがあるためである。通常は CSP によって獲得される特徴量  $y$  は以下の形式となる。

$$y = (\sigma^2(\hat{X}, w^{(1)}), \dots, \sigma^2(\hat{X}, w^{(m)}), \dots, \sigma^2(\hat{X}, w^{(M-m+1)}), \dots, \sigma^2(\hat{X}, w^{(M)}))^T \quad (3.13)$$

あるいは  $y = (y_1, \dots, y_{2m})$  に対して以下のような変換を行った特徴量  $f$  を使う。

$$\begin{aligned} f_i &= \log \left( \frac{y_i}{\sum_{i=1}^{2m} y_i} \right) \\ f &= (f_1, \dots, f_{2m})^T \end{aligned} \quad (3.14)$$

また  $m = 1, 2$  が多くの CSP の応用研究で使われており、 $m$  を増やしても BCI の性能向上には繋がらないことが報告されている [12]。

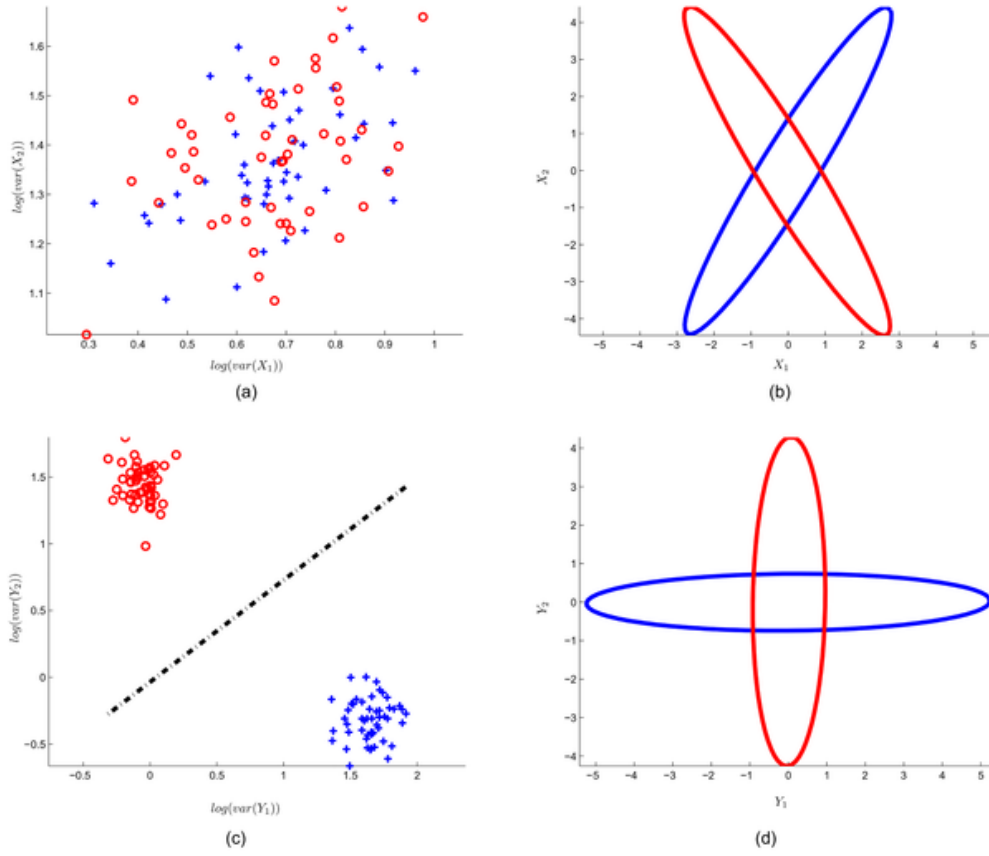


図 3.6: 林檎の図

### 3.2.3 Common Sparse Spatio Spectral Pattern(CSSSP)

CSP ではバンドパスフィルタ  $\mathcal{H}$  通過後の信号を扱ったため、バンドパスフィルタの設計を終えてからでなければ CSP の問題を解くことができない。従って、あるバンドパスフィルタを通過した脳波に対して、最適化問題を解いているに過ぎなく、バンドパスフィルタ自体が適切であったかに関しては一切述べられない。このことによって生じる問題点に関しては、更に詳しく後述する。CSSSP の主なモチベーションはバンドパスフィルタの設計を CSP の問題の中に取り込むことである [14]。

まず脳波信号  $X = [x_1, \dots, x_N] \in \mathbb{R}^{M \times N}$  に対して、観測時間点を  $\tau > 0$  だけずらした  $X_\tau = (x_{1+\tau}, \dots, x_{N+\tau}) \in \mathbb{R}^{M \times N}$  を考える。ここで  $\tau = 1, \dots, T$  である。更にクラス  $C_d$  に属する  $X$  に関して各  $\tau$  毎に、

$$\Sigma_d^\tau = \mathbb{E}_{X \in C_d} [X X_\tau^T + X_\tau X^T] \quad (3.15)$$

を定義する。ただし  $\tau = 0$  のとき  $\Sigma_d^0 = \mathbb{E}_{X \in C_d} [X X^T]$  とする。更に、 $T$  個の要素を持つ係数ベクトル  $b = (b_1, b_2, \dots, b_T)^T$  とする。ここで CSSSP の最適化問題は以下で表現される。

$$\begin{aligned} \max_{b_2, \dots, b_T} \max_w \quad & w^T \left\{ \sum_{\tau=0}^{T-1} \left( \sum_{j=1}^{T-\tau} b_j b_{j+\tau} \right) \Sigma_1^\tau \right\} w - \frac{\alpha}{T} |b|_1 \\ \text{subject to} \quad & w^T \left[ \sum_{\tau=0}^{T-1} \left\{ \sum_{j=1}^{T-\tau} b_j b_{j+\tau} \right\} (\Sigma_1^\tau + \Sigma_2^\tau) \right] w = 1 \end{aligned} \quad (3.16)$$

この最適化問題の目的関数第 1 項は  $b$  を FIR フィルタ  $f$  の係数ベクトルとした場合に、 $f$  を通過後の脳波  $X$  に対して CSP の最適化問題を適用していることに相当する。従って CSSSP では空間重みベクトル  $w \in \mathbb{R}^M$  と次数が  $T$  の FIR フィルタのフィルタ係数  $b$  を同時に評価することが可能である (図??)。目的関数における第二項は正則化項であり、 $\alpha$  はハイパーパラメータである。この

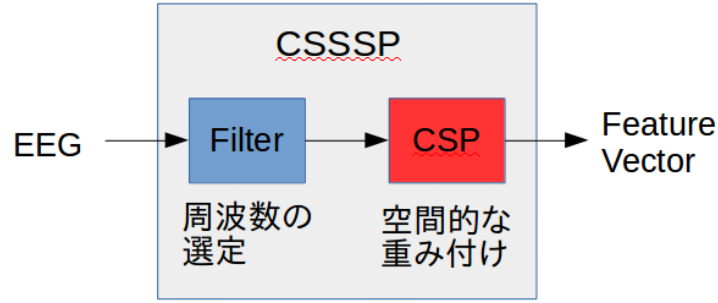


図 3.7: CSSSP の働き

正則化によって FIR フィルタの系列ベクトルに対してスパース性が要請され、フィルタリングに用いられる時系列点数は必要最小限に抑えられる。このことにより FIR フィルタが必要以上に複雑になることが避けられる。解析的に解くことは困難であるため、勾配法あるいはニュートン法を用いた直線探索によって数値的に解を得る。

### 3.2.4 Filter Bank Common Spatial Pattern(FBCSP)

CSSSP は FIR フィルタの設計を CSP の最適化問題に含むことで、バンドパスフィルタと CSP の同時最適化の定式化に成功した。しかし、CSSSP によって求まる特徴量はフィルタ係数  $b$  を持つ FIR フィルタを通過し、変換行列  $W^T$  の CSP によって変換された脳波信号であり、ある 1 つの周波数領域のみしか考慮されていないこととなる。CSSSP を解いて得られるフィルタ係数  $b$  は最適化問題の解であるが、必ずしもそのフィルタ係数のみを用いることが最良であるとは言い難い。なぜなら、運動想起を行う際にはその身体部位に応じて特徴的な変動を持つ周波数領域は異なっているためである。例えば左手の運動想起時と下肢の運動想起時では重要な周波数帯域が異なり、CSSSP の解がいずれかの情報を失ってしまっている可能性が考えられる。FBCSP はこのような問題の解決を図る手法である [13]。

まず、FBCSP ではバンドパスフィルタバンク  $\{\mathcal{H}_1, \dots, \mathcal{H}_K\}$  を定義する。このバンドパスフィルタバンクは 1 つ 1 つのフィルタ  $\mathcal{H}_k$  が異なる周波数領域を通過させるように設計される。各バンドパスフィルタを通過した  $\hat{X}_k = \mathcal{H}_k(X)$  に対して、それぞれ CSP の問題を解くことで各周波数領域における空間的な特徴量を獲得することが可能となる。 $\hat{X}_k$  に対して CSP によって取り出した特徴量を  $f_k$  とすれば、最終的な特徴ベクトル  $f$  は

$$f = (f_1^T, \dots, f_K^T)^T \quad (3.17)$$

であり、単に各周波数帯域の CSP による特徴量を連結したベクトルとなる。ただし CSSSP とは異なり、CSP とバンドパスフィルタの同時最適化を行ってはいない。従って、特徴ベクトル  $f$  のいずれの要素が重要であるか、すなわちいずれの周波数帯域が重要であるかについての最適化は行われておらず、あくまで複数の周波数帯域においてそれぞれの最適な電極の重み付けを CSP によって獲得したにすぎない。そのため、FBCSP によって得られた特徴量から更に特徴量を抽出する手法が別途用いられる図 3.7。また、FBCSP 後の特徴量の選定を行う方法として相互情報量を用いた手法が提案されている [18]。

## 3.3 従来手法のまとめ

運動想起型 BCI は、様々な特徴量抽出手法と分類手法を適宜組み合わせで構築されている。数多くの手法が以下に示す脳波に関する知見を意識している。

- 運動想起する身体部位に応じて、強く反応する頭皮領域は異なる。

- 事象関連脱同期は、特定の周波数領域に生ずる。
- 個人差や計測環境の影響を受けやすい。

空間フィルタやバンドパスフィルタあるいは時間周波数解析を用いること、また複数の分類手法を比較しなければならないことの理由が上記の脳波の性質に集約されている。このセクションでは、多数存在する運動想起型 BCI の構築方法に関して、統一的な視点に立って概観する。その後、従来手法全般に対する問題点を提示し、このチャプターを終える。

### 3.3.1 特徴量の観点からの従来手法

運動想起を 1 回行った際に計測された脳波を以下で表記する。

$$X = (x_1, \dots, x_N) \in \mathbb{R}^{M \times N} \quad (3.18)$$

ここに  $M$  は電極の個数、 $N$  は計測時間点数である。空間フィルタは  $W \in \mathbb{R}^{M \times M}$  によって脳波を  $W^T X \in \mathbb{R}^{M \times N}$  と変換する。通常は更に電極の軸に関して次元削減を行うが、ここでは単に線形変換とする（必要であれば、後に削除すれば良い）。時間周波数解析  $h(\cdot)$  は時間的に局在する基底関数を  $K$  個準備した場合、各時間  $n$  において信号を基底関数の重ね合わせで表現した際の係数  $h(X) \in \mathbb{R}^{M \times N \times K}$  へと変換する。空間フィルタと時間周波数解析の両方を用いる場合には、以下の形式で表される特徴量を抽出する。

$$Z = h(W^T X) \in \mathbb{R}^{M \times N \times K} \quad (3.19)$$

一方でフィルタバンクを用いる FBCSP などは以下の形式で表すことができる。

$$\mathcal{H} = \{\mathcal{H}_1, \dots, \mathcal{H}_K\} \quad (3.20)$$

$$\hat{X}_k = \mathcal{H}_k(X) \in \mathbb{R}^{M \times N} \quad (3.21)$$

$$Z_k = W_k^T \mathcal{H}_k(\hat{X}_k) \in \mathbb{R}^{M \times N} \quad (3.22)$$

$$Z = (Z_1, \dots, Z_K) \in \mathbb{R}^{M \times N \times K} \quad (3.23)$$

ここに  $\mathcal{H}$  はフィルタバンクであり、 $K$  はフィルタの個数に相当する。いずれにしても、抽出される特徴量は  $Z \in \mathbb{R}^{M \times N \times K}$  という 3 階テンソル (電極  $\times$  時間  $\times$  周波数) である。

A. S. Aghaei [19] も同様のことに言及しており、脳波  $X$  から  $Z \in \mathbb{R}^{M \times N \times K}$  を獲得する様々な手法について比較している。更に  $Z$  が得られた後の次元削減および運動想起部位を出力する分類器も含め、一連の手法に関して、電極軸と周波数軸に対する双線形変換によるアプローチを提案している (図 3.8(b,c))。図中で “space” と表現されている単語は本論文中では電極に相当する。 $\Omega$  は運動想起部位であり、 $\hat{\Omega}$  が BCI の予測出力である。

### 3.3.2 構成の観点からの従来手法

次に典型的な運動想起型 BCI の構成について述べる。典型的な運動想起 BCI は、運動想起に関連している脳波を取り出すための前処理  $\mathcal{H}(\cdot)$  によって脳波の生データから  $\hat{X}$  を獲得することが一般的である。

$$\hat{X} = \mathcal{H}(X) \quad (3.24)$$

次に、運動想起に関連している電極を選定するための空間フィルタ  $f(\cdot)$  を適用する。

$$Z = f(\hat{X}) \quad (3.25)$$

続いて  $Z$  に対して、運動想起部位  $Y$  を出力する分類器  $g(\cdot)$  を準備することで、運動想起 BCI が構成されている。

$$Y = g(Z) \quad (3.26)$$

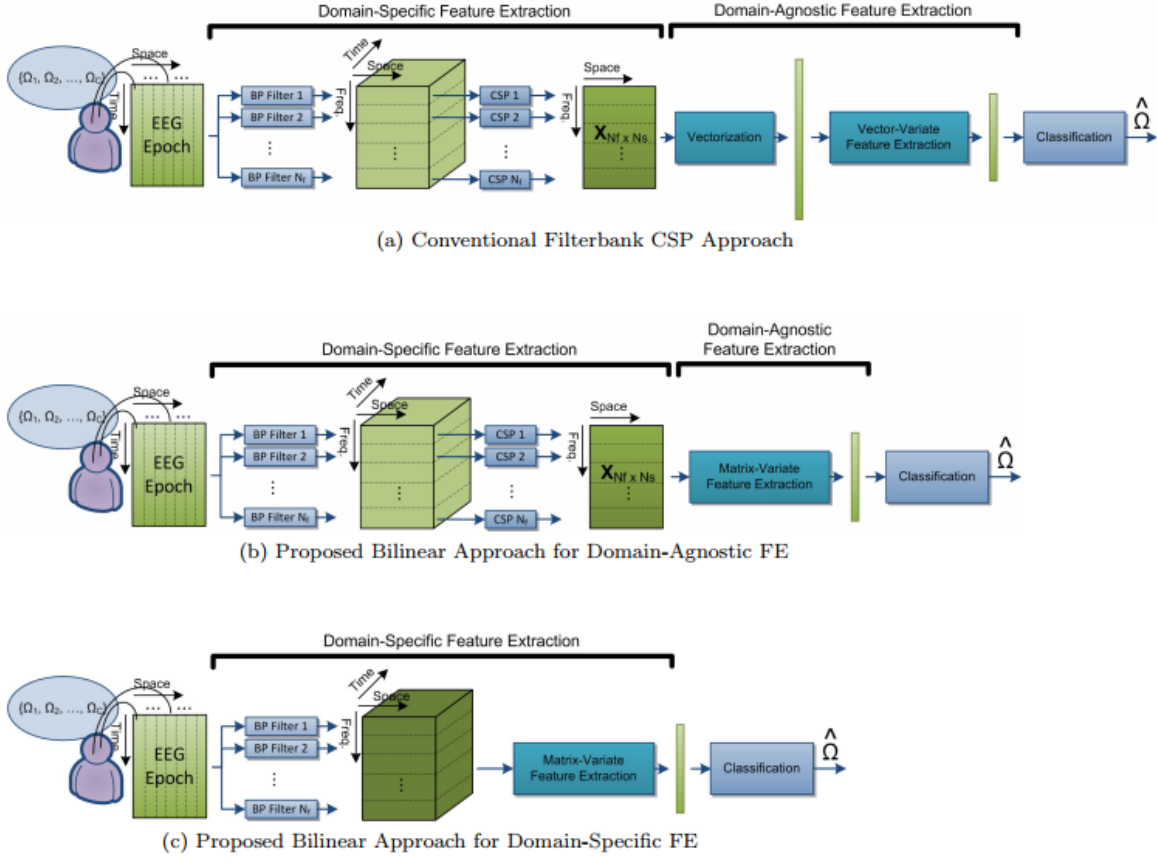


図 3.8: 脳波を 3 階テンソルとした手法の模式図 (参照 [19])

従って、BCI は脳波  $X$  を引数とした合成関数という形式を取る。

$$Y = (g \circ f \circ \mathcal{H})(X) \quad (3.27)$$

典型的な CSP を用いた BCI では  $\mathcal{H}$  をバンドパスフィルタ、 $f$  を CSP、 $g$  を LDA や SVM によって個別に構成する。ここで  $f^*(\cdot) = (f \circ \mathcal{H})(\cdot)$  として、バンドパスフィルタと CSP を同時に構成することを考えれば CSSSP による BCI になり、 $\mathcal{H}$  をフィルタバンクにすることで FBCSP による BCI となる。一方で時間周波数解析に基づく BCI では変換 (3.5) を  $h(\cdot)$  として、

$$Y = (g \circ h \circ f \circ \mathcal{H})(X) \quad (3.28)$$

という形式で表せる。この時、 $\mathcal{H}$  や  $f$  は ERD を検出するための神経科学的な知見に基づいた設計がなされる場合もあれば、機械学習の手法が用いられる場合もある。更に時間周波数解析によって得られるパワースペクトログラムに対して非負値行列分解などを用いて特徴量を抽出する試みもある [20, 21]。この場合も行列分解による変換を  $a(\cdot)$  と置けば

$$Y = (g \circ a \circ h \circ f \circ \mathcal{H})(X) \quad (3.29)$$

と表され、形式上は合成関数である。それぞれの関数の役割を明示しなければ、BCI は単に以下の合成関数である。

$$Y = (f_K \circ \dots \circ f_1)(X) \quad (3.30)$$

BCI を合成関数 (3.30) を出発点にして見ると、従来の BCI 構築手法は、合成関数に含まれる関数の数を明確にし、それぞれに役割を付与し、与えた役割を担うような調整が個々に行われていると見なせる。ただし、関数  $f_i$  を設計するためには  $f_{i-1}$  の設計が終了していなければならない。

## 4

# 提案手法

この章では初めに提案手法の大部分を占めるニューラルネットワークについて簡単に述べる。その後、ニューラルネットワークを用いて運動想起型 BCI を構築するアイデアと、その具体的な提案モデルについて解説する。

### 4.1 ニューラルネットワーク

ニューラルネットワークは近年、“深層学習”あるいは“ディープラーニング”の名の元に、再注目を浴びている学習モデルである。ニューラルネットワークは元々脳の神経活動を模倣した数理モデルとして提案されたが、現在では学習の力学特性や統計物理学の知見、またはヒューリスティクスによる発展が著しい。従って、本論文では単にある特定の形式で表される数理モデルとしてニューラルネットワークを扱う。

#### 4.1.1 ニューラルネットワークに用いられる層

##### Linear 層

ニューラルネットワークで最も基本的な層は Linear 層である。Linear 層はパラメータとして  $W \in \mathbb{R}^{d \times D}$  と  $b \in \mathbb{R}^d$  を有した以下の関数である。

$$L(x, W) = Wx + b \quad (4.1)$$

入力  $x = (x_1, \dots, x_D)^T$  として、出力  $y = (y_1, \dots, y_d)^T$  を以下で表すことができる。

$$y_j = \sum_{i=1}^D w_{j,i} x_i + b_j \quad (4.2)$$

ここに  $w_{j,i}$  は行列  $W$  の  $(j, i)$  成分、 $b_j$  はベクトル  $b$  の  $j$  番目の成分である。

##### Bilinear 層

Bilinear 層は入力として  $x^{(1)} \in \mathbb{R}^{D_1}$  と  $x^{(2)} \in \mathbb{R}^{D_2}$  の 2 つを受け取り、1 つの出力  $y \in \mathbb{R}^d$  を返す。

$$y_k = \sum_{j=1}^{D_1} \sum_{i=1}^{D_2} w_{j,i,k} x_j^{(1)} x_i^{(2)} + b_k \quad (4.3)$$

ここで  $n$  階のテンソル  $A \in \mathbb{R}^{D_1, \dots, D_n}$  の  $(d_1, \dots, d_n)$  成分を  $a_{d_1, \dots, d_n}$  と表記している。パラメータは  $W \in \mathbb{R}^{D_1 \times D_2 \times K}$  と  $b \in \mathbb{R}^K$  である。

## Convolution 層

Convolution 層はパラメータとして  $H \in \mathbb{R}^{P \times Q \times C \times L}$  と  $B \in \mathbb{R}^{I \times J \times L}$  を有し、入力を  $X \in \mathbb{R}^{I \times J \times C}$  として、出力  $Y \in \mathbb{R}^{I' \times J' \times L}$  は以下で表される。

$$y_{i,j,l} = \sum_{c=1}^C \sum_{p=0}^{P-1} \sum_{q=0}^{Q-1} x_{i+p,j+q,c} h_{p,q,c,l} + b_{i,j,l} \quad (4.4)$$

ここでも  $n$  階のテンソル  $A \in \mathbb{R}^{D_1, \dots, D_n}$  の  $(d_1, \dots, d_n)$  成分を  $a_{d_1, \dots, d_n}$  と表記している。出力テンソルの大きさを決める  $I', J'$  は  $I, J$  と  $P, Q$  に依存し、それぞれ  $I' = P - I + 1$  と  $J' = Q - J + 1$  となる。しかし、Convolution 層は実応用では stride や padding、dilation などのテクニックによって更に複雑な動作をする。これらのテクニックはそれぞれハイパーパラメータを持ち、 $I', J'$  はその全てに依存する。

## Recurrent 層

Recurrent 層は入力  $X = (x_1, \dots, x_T) \in \mathbb{R}^{D \times T}$ 、出力  $Y = (y_1, \dots, y_T) \in \mathbb{R}^{d \times T}$  として以下で表される。

$$y_t = \tanh(W_{in}x_t + W_{out}y_{t-1} + b) \quad (4.5)$$

ここにパラメータは  $W_{in} \in \mathbb{R}^{d \times D}$  と  $W_{out} \in \mathbb{R}^{d \times d}$  ならびに、 $b \in \mathbb{R}^d$  である。元々系列データへの応用のために考案されたため、 $D$  次元のベクトル  $x_t \in \mathbb{R}^D$  が時間変化していくような場合を想定して (4.5) と定式化される。しかし、実際には入力は画像のような静的なデータでも構わなく、画像を左から右に走査するような働きを担う。 $\tanh(\cdot)$  は双曲線正接関数であるが、ニューラルネットワークの活性化関数として広く持ちいられている。

## LSTM 層

LSTM は Long Short-Term Memory の略であり、LSTM 層は Recurrent 層が長期的な系列データを上手く学習できない問題を解決した。LSTM 層は入力  $X = (x_1, \dots, x_T) \in \mathbb{R}^{D \times T}$ 、出力  $Y = (y_1, \dots, y_T) \in \mathbb{R}^{d \times T}$  として以下で表される。

$$i_t = \sigma(W_{ii}x_t + W_{hi}h_{t-1} + b_i) \quad (4.6)$$

$$f_t = \sigma(W_{if}x_t + W_{hf}h_{t-1} + b_f) \quad (4.7)$$

$$o_t = \sigma(W_{io}x_t + W_{ho}h_{t-1} + b_o) \quad (4.8)$$

$$g_t = \tanh(W_{ig}x_t + W_{hg}h_{t-1} + b_g) \quad (4.9)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \quad (4.10)$$

$$y_t = o_t \odot \tanh(c_t) \quad (4.11)$$

ここに  $\odot$  は同じ次元のベクトルの要素ごとの積である。パラメータは  $W_{ii}, W_{if}, W_{io}, W_{ig} \in \mathbb{R}^{d \times D}$  と、 $W_{hi}, W_{hf}, W_{ho}, W_{hg} \in \mathbb{R}^{d \times d}$  および、 $b_i, b_f, b_o, b_g \in \mathbb{R}^d$  である。Recurrent 層同様に、系列データを想定した定式化がなされているが、実際には入力が画像でも構わない。 $\sigma(\cdot)$  はシグモイド関数であり (2.47) と同じものである。ここでのシグモイド関数は、値域が  $(0, 1)$  であるために用いられている。

$i_t, f_t, o_t, g_t$  は現在の入力  $x_t$  と過去の出力  $y_{t-1}$  を引数に取り、それぞれ個別のパラメータを用いて値が算出される。 $c_t$  は Memory Cell と呼ばれ、過去の情報を層の内部に記憶する役割を担っている。 $f_t \in (0, 1)^d$  を乗じることで、過去の情報  $c_{t-1}$  を一定の割合のみ保持し（あるいは忘却し）、 $i_t$  に対して  $g_t \in (-1, 1)^d$  を乗じた値を新たに加算して保持する。 $y_t$  は内部情報である  $c_t$  を  $(-1, 1)^d$  にスケールしたベクトルと  $o_t$  との要素積によって算出される。すなわち、 $o_t$  の値が内部情報  $c_t$  に応じて大きさと正負が調整されて LSTM 層の出力となる。

## 4.1.2 活性化関数

ここではパラメータを持たない変換について述べる。現在は数多くの活性化関数が利用されており、ここでまとめて扱う。

### シグモイド関数

シグモイド関数  $\sigma(x)$  は以下で定義される。

$$\sigma(z) = \frac{1}{1 + \exp(-z)} \quad (4.12)$$

ニューラルネットワークは当初、神経活動の数値モデルとしての一面を有しており、入力値がある閾値を超えると発火するというモデルを表現するためにステップ関数が利用されていた。シグモイド関数は、ステップ関数を微分可能な形で表現することを動機に考案された。

### 双曲線正接関数

双曲線正接関数  $\tanh(z)$  は以下で定義される。

$$\tanh(z) = \frac{\exp(z) - \exp(-z)}{\exp(z) + \exp(-z)} \quad (4.13)$$

先に述べたシグモイド関数とは  $\sigma(z) = (\tanh(x/2) + 1)/2$  という関係にあり、負の値を取ることもできるため状況に応じてシグモイド関数の代わりに用いられる。

### ランブ関数

ランブ関数はニューラルネットワークでは Rectified Linear Unit(ReLU) 関数と呼ばれ、以下で表される。

$$\text{ReLU}(z) = \max(z, 0) \quad (4.14)$$

入力が正で  $-1$  の勾配を持ち、その他の場合は勾配は  $0$  となる。この性質がニューラルネットワークの勾配消失問題を大きく改善したとされる。

### leaklyReLU 関数

ReLU 関数が入力が負の場合にも値を有するように提案された。

$$\text{leaklyReLU}(z) = \max(z, 0) + \alpha \min(z, 0) \quad (4.15)$$

入力が正で  $-1$  の勾配を持ち、その他の場合は勾配は  $\alpha$  でハイパーパラメータである。

### elu 関数

ReLU 関数を全域で微分可能にするように調整された関数。

$$\text{elu}(z) = \max(z, 0) + \min(\alpha(\exp(z) - 1), 0) \quad (4.16)$$

図 4.1 はここまでに述べた 5 つの活性化関数の入出力を表した図である。どのような場面でどの活性化関数が有効であるか、理論的な根拠は確認されておらず、試行錯誤的に決定される。

### ソフトマックス関数

ソフトマックス関数は一般的に、分類問題におけるニューラルネットワークの出力層に用いられる。ベクトルの入力を受け取りベクトルの出力を行うが、以下に要素毎の計算を示す。

$$\text{softmax}_i(z) = \frac{\exp(z_i)}{\sum_j \exp(z_j)} \quad (4.17)$$



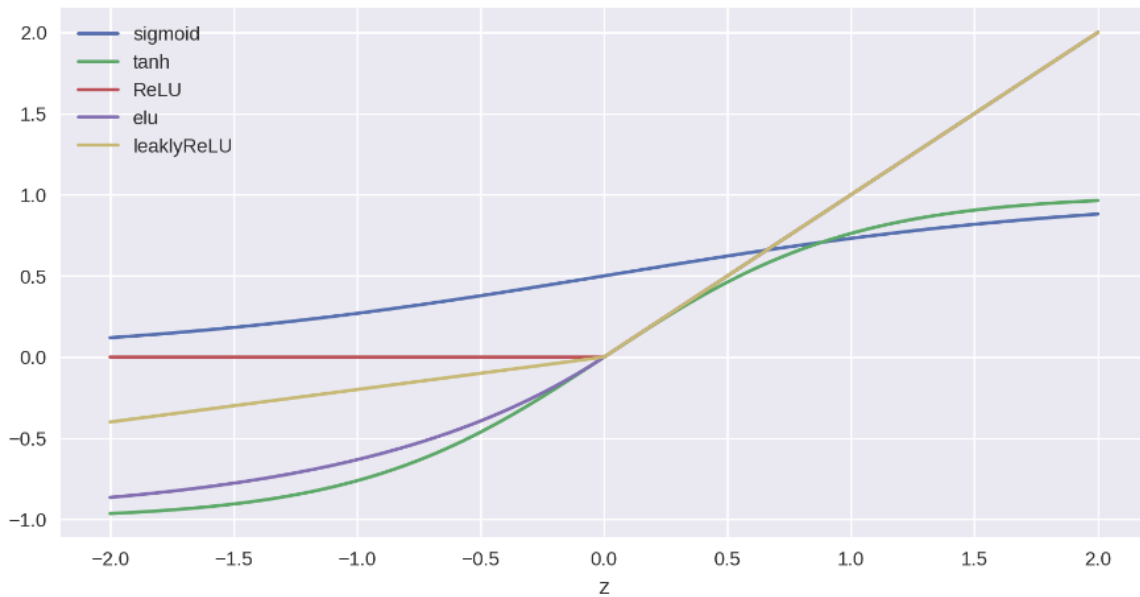


図 4.1: 5 つの活性化関数

ここで、 $z_i$  はベクトル  $z$  の  $i$  番目の要素である。定義から明らかに、値域は  $(0, 1)$  に制限され、出力の要素の総和は 1 である。

### 4.1.3 過学習抑制手法

#### ドロップアウト

ドロップアウトは過学習抑制のために考案された手法であり、以下で表される。

$$\text{dropout}(z, p) = z \odot b \quad (4.18)$$

$$b \sim \text{Bern}(p) \quad (4.19)$$

$\text{Bern}(p)$  は  $p$  の確率で 1 を出力し、 $1 - p$  の確率で 0 を出力するベルヌーイ分布である。従って、ドロップアウトは入力  $z$  の各要素は  $1 - p$  の確率で値を 0 に変更する。ドロップアウトは入力が行われる度に毎回変更されるため、学習時にはドロップアウトされなかった一部の要素に関わるパラメータのみが更新される。テスト時にはドロップアウトを行わず、代わりに  $zp$  を出力する。あるいはドロップアウトを用いたまま複数回サンプリングを行い、加算平均を出力とする。

パラメータを持つ全ての層の後にドロップアウトを適用した場合は、ニューラルネットワークの学習がある仮定の下での変分ベイズ学習に厳密に一致する。また、テスト時に複数回のサンプリングによる出力を行った場合、ベイズ予測分布に一致する [22]。ただし、ベイズの解釈としては出力が確率的に 0 になるのではなく、学習パラメータが確率的に 0 になった結果、出力が 0 になると考えなければならない。

#### バッチ正規化

# 5

## 実験方法

この章では実験に用いたデータについて説明し、その後、データを用いた実験に対してどのような評価を行うのかを明示する。

## 6

# 結果と考察

この章では従来手法と提案手法の実験結果をそれぞれ示す。結果に対して考察と本研究の結論を述べ、更に今後考えうる改善案などについて簡単に述べる。

# Appendix

## A.1 セクション

### A.1.1 サブセクション

# 謝辭

... ..

## 参考文献

- [1] 日野幹雄. スペクトル解析 (統計ライブラリー). 朝倉書店, 1977.
- [2] Hiromu Gotanda, Takaaki Ishibashi, Nobuo Iwasaki, and Katsuhiro Inoue. 独立成分分析の基礎と応用. Technical Report 1743, 数理解析研究所, 2011.
- [3] 村田昇. 入門 独立性分析. 東京電機大学出版, 2004.
- [4] 開一夫, 金山範明, 河内山隆紀, 松本敦宮, 腰誠. 脳波解析入門:EEGLAB と SPM を使いこなす. 東京大学出版会, 2016.
- [5] G. Pfurtscheller and F. H. L. da Silva. Event-related eeg/meg synchronization and desynchronization: Basic principles. *Clinical Neurophysiology*, Vol. 110, No. 11, pp. 1842–1857, 1999.
- [6] G. Pfurtscheller and T. Solis-Escalante. Could the beta rebound in the eeg be suitable to realize a “ brain switch ” ? *Clinical Neurophysiology*, Vol. 120, No. 1, pp. 24–29, 2009.
- [7] Kenji Kato, Kensho Takahashi, Nobuaki Mizuguchi, and Junichi Ushiba. Online detection of amplitude modulation of motor-related eeg desynchronization using a lock-in amplifier: Comparison with a fast fourier transform, a continuous wavelet transform, and an autoregressive algorithm. *Journal of Neuroscience Methods*, Vol. 293, pp. 289–298, 2018.
- [8] Guo Xiaojing, Wu Xiaopei, and Zhang Dexiang. Motor imagery eeg detection by empirical mode decomposition. In *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, pp. 2619–2622, June 2008.
- [9] SachinTaran, VarunBajaj, DheerajSharma, SiulySiuly, and A.Sengur. Features based on analytic imf for classifying motor imagery eeg signals in bci applications. *Measurement*, Vol. 116, pp. 68–76, 2018.
- [10] Zoltan J. Koles, Michael S. Lazar, and Steven Z. Zhou. Spatial patterns underlying population differences in the background eeg. *Brain Topography*, Vol. 2, pp. 275–284, 1990.
- [11] G. Pfurtscheller, C. Guger, and H. Ramoser. International work-conference on artificial neural networks. In *Engineering Applications of Bio-Inspired Artificial Neural Networks*, 1999.
- [12] Jyoti Singh Kirara and R. K. Agrawala and. Optimal spatio-spectral variable size subbands filter formotor imagery brain computer interface. *Procedia Computer Science*, Vol. 84, pp. 14–21, 2016.
- [13] Kai Keng Ang, Zheng Yang Chin, Haihong Zhang, and Cuntai Guan. Filter bank common spatial pattern (fbcsp) in brain-computer interface. *IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING*, Vol. XX, No. Y, pp. 2390–2397, 2008.
- [14] Guido Dornhege, Benjamin Blankertz, Matthias Krauledat, Florian Losch, Gabriel Curio, and Klaus-Robert Muller. Combined optimization of spatial and temporal filters for improving brain-computer interfacing. *IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING*, Vol. XX, No. Y, 2006.
- [15] B. Blankertz, M. Kawanabe, R. Tomioka, F. Hohlefeld, V. Nikulin, and K. R. M. In-variant common spatial patterns: Alleviating nonstationarities in brain-computer interfacing. *Advances in Neural Information Processing Systems*, Vol. 20, pp. 113–120, 2008.

- [16] N. Behrooz, B. Reza, and J. M. Zolghadri. An efficient hybrid linear and kernel csp approach for eeg feature extraction. *Neurocomputing*, Vol. 73, pp. 432–437, 2009.
- [17] F. Lotte and C. Guan. Regularizing common spatial patterns to improve bci designs: Unified theory and new algorithms. *IEEE Transactions on Biomedical Engineering*, Vol. 58, No. 2, pp. 355–362, 2011.
- [18] Kai Keng Ang, Zheng Yang Chin, Chuanchu Wang, , Cuntai Guan, and Haihong Zhang. Filter bank common spatial pattern algorithm on bci competition iv datasets 2a and 2b. *frontiers in NEUROSCIENCE*, Vol. 29, pp. 2390–2397, March 2012.
- [19] A. S. Aghaei, M. S. Mahanta, and K. N. Plataniotis. Separable common spatio-spectral patterns for motor imagery bci systems. *IEEE Transactions on Biomedical Engineering*, Vol. 63, No. 1, pp. 15–29, Jan 2016.
- [20] M. Sakai. Kernel nonnegative matrix factorization with constraint increasing the discriminability of two classes for the eeg feature extraction. In *2013 International Conference on Signal-Image Technology Internet-Based Systems*, pp. 966–970, Dec 2013.
- [21] Hyekyoung Lee, Andrzej Cichocki, and Seungjin Choi. Kernel nonnegative matrix factorization for spectral EEG feature extraction. *Neurocomputing*, Vol. 72, No. 13-15, pp. 3182–3190, 2009.
- [22] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: representing model uncertainty in deep learning. In *ICML'16 Proceedings of the 33rd International Conference on International Conference on Machine Learning*, Vol. 48, pp. 1050–1059, June 2016.