第 1 章

能動学習の理論

1.1　能動的な学習とは何か？

ここでは「能動学習（active learning)」の基本的な概念を説明し，本章で解説する機械学習における能動学習について概観したい．学習という用語の意味を，データを用いることにより，システムの持つパラメータをある規準に従って最適化していくプロセスだと定めることにしよう．この定義は，本章で述べるような確率的データによる機械学習にもあてはまるし，学習に用いられるシステムが脳であり，脳の神経細胞を結ぶ結合状態がシステムパラメータだと考えれば，我々の脳における学習にもあてはまるであろう．

データから知識を獲得する機械学習の理論では，与えられたデータに対してシステム設計者が適切なモデルを設定し，問題に沿った規準（目的関数）を最適化するように学習を行う，という枠組みで議論が進められることが多い．ここで着目して欲しいのは，「あらかじめ与えられたデータ」を学習の出発点としている点である．一方，我々人間が通常行っている学習を振り返ると，受動的に与えられた情報だけを使って知識を獲得しているわけではない．小さな子供は「あれは何？」と周囲に質問をさかんに発し，また自ら試行錯誤を繰り返しながら成長していく．抽象的に言えば，このような学習の方法は，自発的にデータを収集していくという意味において「能動的」な学習だと考えることが出来る．

機械でも能動的に学習すればより効果的な学習が行えるに違いない — この発想が能動学習の原点である．すなわち，機械学習における能動学習とは，学習データがあらかじめ受動的に与えられていると考えず，学習に都合のよいデータの採取法を機械自らに設計させようという方法論である．このような能動学習の方法論は，機械が実世界の知識を獲得する場合に特に重要となる．例えば，チームを組んでサッカーの試合を行うロボットの行動戦略を作ることを考えてみよう．このためには多様な局面に対して適切に行動するようにロボットをプログラムする必要があるが，受動的なデータによってロボットを学習させようとすると，設計者は，考えられる限りのゲームの局面とそれに対する行動パターンをデータ化してロボットに与えなければならない．この場合，強い戦略を獲得するためにどのようなデータを与えればよいのかを決めることは難しく，むしろロボットが実際に試合を行い，時には試してみたい局面を故意に作り出し，試行錯誤の中でよりよい戦略を学習していく方がよさそうである．このように，限ら

れた状況下で定まったタスクを実行する機械から，複雑な実世界の知識を柔軟に獲得していく機械へと要請が移るにつれて，機械が主体的にデータを収集する能動学習の重要度が増してきている．

あとで詳しく述べるように，最適なデータ採取法という観点から見ると，能動学習のひとつの形態は「最適実験計画」と呼ばれる統計学の一分野の中で古くから研究されてきた．しかしながら，それらの研究は，機械を線形回帰モデルなど単純なモデルに限定し，データを採取する入力点を理論的に最適設計することに主眼が置かれていた．一方「能動学習」という呼び方には，データの採取と機械のパラメータ最適化を逐次的に行っていくという意味合いが込められている．このようなデータとパラメータの逐次的な最適化が活発に議論されるようになった理由のひとつには，計算機の発達によりデータを収集しながら実環境下で学習することが現実的になってきたことが挙げられよう．

本章では，このような能動学習の理論的な枠組みについて述べる．一口に効果的なデータの採取法といってもその実現の仕方はさまざまであり，機械が解こうとしている問題の種類によってもアプローチが異なる．本章では，入力から出力への入出力関係をニューラルネットなどの非線形回帰モデルを用いて学習する問題を対象とし，最終的な関数の推定精度ないしは汎化誤差を最適化するように，パラメータの最適化とデータ採取点の最適化を交互に行っていく方法を中心に述べる．

1.2　確率的なデータからの入出力関係の学習

まず準備として，入力 $\boldsymbol{x}$ から出力 y への入出力関係を，有限個の確率的なデータから推定する問題を理論的に定式化する．このような枠組みは，入力である文字画像から文字コードを出力する文字認識の問題や，現在の経済指標を入力することにより次期の経済指標を出力する時系列予測の問題など，様々な現実の設定を含んでいる．

1.2.1　確率的な動作をするシステム

L 次元の入力ベクトル x を与えると M 次元の出力ベクトル y を返すシステムを考えよう．現実に遭遇する多くの問題では，入出力関係が必ずしも決定論的に $y = \boldsymbol{f}(x)$ という関数関係で記述できるわけではなく，確率的な要素を含んでいることも多い．経済指標の予測などでは，過去の経済指標によって次期の値が完全に決定されるとは考え難いし，文字認識の問題でも，手書き数字の「1」と「7」の識別では，書き手の癖などによってほとんど同じ画像が異なる数を表していることもありえる．そこで決定論的な関数で表現できない部分は確率的要素として扱うことにし，ある入力 x が与えられたときの出力 y の条件付き確率によって，学習対象である真のシステムの確率的な入出力関係を表す．この条件付確率の密度関数を

$$p(y|x) \tag{1.1}$$

とおく．このシステムに n 個の入力データ $\{x_i\}_{i=1}^n$ を与えると，$\{y_i\}_{i=1}^n$ が (1.1) 式の条件付き確率に従って発生するが，今後の議論では異なる i に対して $\{y_i\}_{i=1}^n$ は互いに独立である

と仮定する．さらに，システムは，通常の状態ではある一定の環境に置かれており，決まった確率 Q に従う入力ベクトルを受け取っていると考えよう．以下，入力確率 Q の密度関数を $q(x)$ で表す．

本章で述べる能動学習の枠組みでは，学習者が与えたどんな入力データに対しても，(1.1) 式の確率的ルールに従って出力データを返答するシステムを想定する．以下では話をさらに簡単にするために，この確率的ルールが，決定論的な関数 $\varphi_o(x)$ と条件付き確率 $r(y|\boldsymbol{s})$ を用いて

$$p(y|x) = r(y|\boldsymbol{\varphi}_o(x))$$

と表されると仮定する．ここで，確率的要素 $r(y|\boldsymbol{s})$ はノイズや揺らぎなどの不確定要因を表現している．以下で二つの例を示そう．

例１：加法的ガウスノイズ　入力ベクトル x に対して，決定論的な出力 $\boldsymbol{\varphi}(x)$ に，平均 0 分散共分散行列 $\sigma^2 I_M$ の独立なガウスノイズ $\boldsymbol{z}$ が加わったものが y として観測されるとしよう．このとき，

$$y = \boldsymbol{\varphi}(x) + \boldsymbol{z}$$

と表される．条件付き確率密度関数で書けば，

$$p(y|x) = \frac{1}{(2\pi\sigma^2)^{M/2}} \exp\Big\{-\frac{1}{2\sigma^2}\|y - \boldsymbol{\varphi}(x)\|^2\Big\}$$

となる．

例２：２クラスの識別問題　パターンを２つのクラスに識別する問題では，出力 y が $\{1, 0\}$ に値を取るとして

$$r(y|s) = \frac{e^{ys}}{1 + e^s}$$

という確率的要素を導入し，

$$p(y|x) = r(y|\varphi(x)) = \frac{e^{y\varphi(x)}}{1 + e^{\varphi(x)}}$$

というルールを考えることがよく行われる．これは，$y = 1$ となる確率が $1/(1 + e^{-\varphi(x)})$ となるモデルである．関数 $1/(1 + e^{-s})$ はロジスティック関数と呼ばれ図**??**のようなグラフを持つ．

1.2.2　入出力関係の学習

確率的なシステムの背後にある関数関係 $\boldsymbol{\varphi}_o(x)$ を学習するために，ここではパラメトリックな方法論を取る．すなわち，真の関数関係 $\boldsymbol{\varphi}_o(x)$ の候補となる関数族 $\{\boldsymbol{\varphi}(x;\boldsymbol{\theta}) \mid \boldsymbol{\theta} \in \Theta\}$（$\boldsymbol{\theta} \in \Theta$ は d 次元のパラメータベクトル）と，真のシステムから得られた学習データ $\{(x_i, y_i)\}_{i=1}^n$ を用意し，真の関数関係を精度よく推定できるように，学習データを用いてパラメータ $\boldsymbol{\theta}$ を最適化する．

まず，関数族に関して考えよう．本節ではこのようなパラメトリックな関数のことを学習機械と考える．学習機械の設定の仕方はシステムの設計者に任されている．問題の本質を汲み上げて，なるべく高い性能の機械が出来るようにモデル化するのがよい．問題の構造に関する十分な知識がある場合には，それを適切に表現するパラメトリックな関数族を設定できることもある．しかしながら，例えば文字認識の問題において，濃淡画像や特徴ベクトルの空間から文字コードへの写像をよく表すモデルを我々が想像することはほとんど不可能である．そのため，問題の構造を直接モデル化するのではなく，さまざなま関数の近似が可能な汎用関数系を用意して，それによって未知なる真の関数関係を学習しようとする方針もありえる．後者のような立場で選択される関数族として，多層パーセプトロンや Radial Basis Funcitons (RBF) に代表されるニューラルネットワークがある．後に述べる応用ではニューラルネットを用いた例を述べる．

次に学習データの採取法であるが，これには入力データを真のシステムに与えて出力データを観測する必要がある．このとき，システムに与える入力データ $\{x_i\}_{i=1}^n$ は，必ずしも真のシステムが通常置かれている環境の入力分布 Q から発生したものを用いる必要はない．この入力データをどのように設計すればよいかが能動学習の問題となる．その具体的な設計法は後で説明するとして，ここではある定まった入力データ $X_n = \{x_i\}_{i=1}^n$ に対して出力データ $Y_n = \{y_i\}_{i=1}^n$ が観測され，学習データ $\{(x^{(i)}, y^{(i)})\}_{i=1}^n$ が得られたとする．

やや天下り的であるが，以降入力データ $X_n = \{x_i\}_{i=1}^n$ の性質として，密度関数 $u(x)$ を持つある確率分布 U が存在して，x の U に関する任意の可積分関数 $g(x)$ のサンプル平均が，大数の法則

$$\frac{1}{n}\sum_{i=1}^{n} g(x_i) \quad \longrightarrow \quad \int g(x)u(x)dx \qquad (n \to \infty) \tag{1.2}$$

を満たすことを仮定する．また，学習データ $\{(x_i, y_i)\}_{i=1}^n$ に対しては可積分関数 $h(x, y)$ のサンプル平均が

$$\frac{1}{n}\sum_{i=1}^{n} h(x_i, y_i) \quad \longrightarrow \quad \int\int h(x, y)r(y|\varphi_o(x))u(x)dydx \qquad (n \to \infty) \tag{1.3}$$

と収束することを仮定する．

学習機械と学習データを用いて真の関数関係の学習を行うためには，学習の目的を定義する目的関数を決める必要がある．そのために，M 次元ベクトル y と $\boldsymbol{s}$ の近さを測る**損失関数** (loss function) $\ell(y, \boldsymbol{s})$ を用意し，与えられたデータに対する**経験損失関数** (empirical loss function) $L_n(\boldsymbol{\theta})$ を

$$L_n(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, \boldsymbol{\varphi}(x_i; \boldsymbol{\theta})) \tag{1.4}$$

により定義する．この $L_n(\boldsymbol{\theta})$ を最小にすることを学習の目的と考え，その最小値をとるパラメータを $\widehat{\boldsymbol{\theta}}$ と書く．損失関数 $\ell(y, \boldsymbol{s})$ の満たすべき条件として，任意の $\boldsymbol{s}_1$, $\boldsymbol{s}_2$ に対して

$$\int \ell(y, \boldsymbol{s}_2)r(y|\boldsymbol{s}_1)dy \geq \int \ell(y, \boldsymbol{s}_1)r(y|\boldsymbol{s}_1)dy \tag{1.5}$$

が成り立つことを要請しておく．

(1.3) 式の仮定のもと，データ数 n が大きいとき，$L_n(\boldsymbol{\theta})$ は

$$L_\infty(\boldsymbol{\theta}) = \int\int \ell(y, \boldsymbol{\varphi}(x;\boldsymbol{\theta}))r(y|\boldsymbol{\varphi}_o(x))u(x)dydx \tag{1.6}$$

の近似となる．従って，経験損失関数の最小化は，近似的に (1.6) 式を最小化する $\boldsymbol{\theta}$ を探していることになる．

損失関数の代表的な選び方として，統計学でよく用いられる**最尤法** (maximum likelihood method) がある．これは，損失関数として負の対数尤度関数

$$\ell(y, \boldsymbol{s}) = -\log r(y|\boldsymbol{s})$$

を取る方法である．また，２乗誤差 $\ell(y, \boldsymbol{s}) = \|y - \boldsymbol{s}\|^2$ もよく用いられる．

1.2.1 節で紹介した２つの例に関して最尤法がどのような目的関数を与えるか調べておこう．例 1 の加法的ガウスノイズの場合には，簡単な計算により

$$L_n(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{2\sigma^2}\|y_i - \boldsymbol{\varphi}(x_i;\boldsymbol{\theta})\|^2 + \text{ 定数}$$

となり，最小２乗誤差の規準と一致する．また例２の２クラス識別問題では，関数 $\varphi(x;\boldsymbol{\theta})$ のもとで $y = 1$ となる確率を $p(x;\boldsymbol{\theta}) = 1/(1 + e^{-\varphi(x;\boldsymbol{\theta})})$ で表すとき，

$$L_n(\boldsymbol{\theta}) = -\frac{1}{n}\sum_{i=1}^{n}\big\{y_i \log p(x_i;\boldsymbol{\theta}) + (1 - y_i)\log(1 - p(x_i;\boldsymbol{\theta}))\big\}$$

が得られる．この目的関数をクロスエントロピーと呼ぶことがある．

1.2.3 学習機械の汎化能力

学習機械 $\boldsymbol{\varphi}(x;\boldsymbol{\theta})$ に対して，その期待損失を

$$K(\boldsymbol{\theta}) = \int\int \ell(y, \boldsymbol{\varphi}(x;\boldsymbol{\theta}))r(y|\boldsymbol{\varphi}_o(x))q(x)dydx \tag{1.7}$$

により定義する．$K(\boldsymbol{\theta})$ は，学習機械が定環境における入力分布 Q から入力ベクトルを受け取ったときに，真のシステムの出力 y を $\boldsymbol{\varphi}(x;\boldsymbol{\theta})$ によって予測した場合の損失の平均値を表している．

経験損失関数 $L_n(\boldsymbol{\theta})$ を最小にするパラメータを $\widehat{\boldsymbol{\theta}}$ として，その期待損失 $K(\widehat{\boldsymbol{\theta}})$ を考えよう．これは有限個のデータによる学習からどれぐらい真の関数を忠実に再現できたかを示しており，得られた機械の**汎化誤差** (generalization error)，もしくは**予測誤差** (prediction error) と呼ばれる．

学習によって得られたパラメータ $\widehat{\boldsymbol{\theta}}$ は学習データの関数であるから，確率的なばらつきを持つ確率変数である．いま学習データの入力点 $X_n = \{x_i\}_{i=1}^n$ を固定して考え，X_n を条件としたときの出力データ $Y_n = \{y_i\}_{i=1}^n$ による汎化誤差の期待値

$$R(X_n) = E_{Y_n}\big[K(\widehat{\boldsymbol{\theta}})|X_n\big] \tag{1.8}$$

を考える．本章における能動学習の目的は $R(X_n)$ を最小にする X_n を探すことである．

1.3　能動学習の方法 – 汎化誤差を最小にするデータ採取点

1.3.1　漸近理論による汎化誤差の期待値の推定

汎化誤差の期待値 (1.8) 式は未知の関数 $\boldsymbol{\varphi}_o(x)$ を用いて定義されているため，これを直接計算することはできず，何らかの方法により推定することが必要となる．

汎化誤差を推定する方法には，大きく分けて，データ数 n が大きいときの推定量の統計的性質を記述する統計的漸近理論を用いる方法や，学習データを学習に使うデータとテストデータに分けて検証を行うクロス・バリデーション（cross-validation, [1]）などのアプローチがある．クロス・バリデーションでは，推定量が陽に計算できる単純なモデルを除くと，実際に学習を行ってみないと汎化誤差の推定が行えないため，採取するデータ点に $R(X_n)$ がどう依存するかを事前に明示的な形で与えるのが難しい．そこで以下では漸近理論を用いた方法を考察していく．

(1.6) 式の経験損失の極限 $L_\infty(\boldsymbol{\theta})$ を最小にするパラメータを $\boldsymbol{\theta}_o$ とおく．このパラメータは，学習機械のなかで，学習データを与える入力分布 U のもとで真の入出力関係 $\boldsymbol{\varphi}_o(x)$ を最もよく近似する機械を与える．微分可能性などの適当な条件のもと，$\boldsymbol{\theta}_o$ は

$$\frac{\partial L_\infty(\boldsymbol{\theta}_o)}{\partial \boldsymbol{\theta}} = 0 \tag{1.9}$$

を満足する．真の入出力関係がもともと有している期待損失

$$\int\int \ell(y, \boldsymbol{\varphi}_o(x)) r(y|\boldsymbol{\varphi}_o(x)) q(x) dy dx$$

を K_* で表すことにすると，(1.8) 式は以下のように分解される．

$$R(X_n) = K_* + \{K(\boldsymbol{\theta}_o) - K_*\} + E_{Y_n}[K(\widehat{\boldsymbol{\theta}}) - K(\boldsymbol{\theta}_o)]. \tag{1.10}$$

右辺第 1 項はシステムは学習機械や学習データには依存しない．第 2 項は，入力分布 U のもとで，最適な関数が真の入出力関係からどれぐらいずれているかを表している．学習データのばらつきに依存するのは第 3 項のみであり，データから推定された機械が最もよい機械からどれほどばらつくかを表している（図**??**）．

いま，システム設計者が学習機械として十分に表現能力の高い関数族を設定したと仮定し，第 3 項に比べて第 2 項が無視できるほど小さい場合を考察の対象としてみよう．そこで理論的な単純化として，真の入出力関係が $\boldsymbol{\theta}_o$ で与えられる，すなわち

$$\boldsymbol{\varphi}(x; \boldsymbol{\theta}_o) = \boldsymbol{\varphi}_o(x) \tag{1.11}$$

を仮定する．このとき (1.10) 式の第 2 項は，(1.5) 式の仮定より，最小値である 0 をとる．さらに 1.3 節では，任意の x に対し

$$\frac{\partial}{\partial \boldsymbol{\theta}} \int \ell(y, \boldsymbol{\varphi}(x; \boldsymbol{\theta})) r(y|\boldsymbol{\varphi}_o(x)) dy|_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} = 0 \tag{1.12}$$

が成り立つことを仮定する．簡単な計算により，損失関数 $\ell(y, \boldsymbol{s})$ が 負の対数尤度 $-\log r(y|\boldsymbol{s})$ であれば (1.12) 式が成り立つことがわかる．また，条件付密度関数 $r(y|\boldsymbol{s})$ において $\boldsymbol{s}$ が y の平均値を表しており（$\int yr(y, \boldsymbol{s})dy = \boldsymbol{s}$ ），かつ損失関数が 2 乗誤差 $\ell(y, \boldsymbol{s}) = \|y - \boldsymbol{s}\|^2$ であるならば，やはり (1.12) 式を満足する．

(1.11) 式の仮定のもと，汎化誤差の期待値は

$$R(X_n) = K_* + E_{Y_n}[K(\widehat{\boldsymbol{\theta}}) - K(\boldsymbol{\theta}_o)]$$

となるが，これを $\boldsymbol{\theta}_o$ のまわりで $\widehat{\boldsymbol{\theta}}$ に関して Taylor 展開すると

$$R(X_n) = K_* + \frac{1}{2}\sum_{a,b=1}^{d} \frac{\partial^2 K(\boldsymbol{\theta}_o)}{\partial\theta^a \partial\theta^b} E_{Y_n}[(\hat{\theta}^a - \theta_o^a)(\hat{\theta}^b - \theta_o^b)] + O(E_{Y_n}\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}_o\|^3) \quad (1.13)$$

が得られる．ここで (1.12) 式を用いた．

$n \to \infty$ の時に $E_{Y_n}[(\hat{\theta}^a - \theta_o^a)(\hat{\theta}^b - \theta_o^b)]$ が収束する値を漸近理論（[2], Section 6）により知ることができる．半正定値行列 $G(\boldsymbol{\theta}; x)$, $H(\boldsymbol{\theta}; x)$ を

$$G(\boldsymbol{\theta}; x)_{ab} = \int \frac{\partial\ell(y, \boldsymbol{\varphi}(x; \boldsymbol{\theta}))}{\partial\theta^a} \frac{\partial\ell(y, \boldsymbol{\varphi}(x; \boldsymbol{\theta}))}{\partial\theta^b} r(y|\boldsymbol{\varphi}(x; \boldsymbol{\theta}))dy$$
$$H(\boldsymbol{\theta}; x)_{ab} = \int \frac{\partial^2\ell(y, \boldsymbol{\varphi}(x; \boldsymbol{\theta}))}{\partial\theta^a \partial\theta^b} r(y|\boldsymbol{\varphi}(x; \boldsymbol{\theta}))dy$$

により定義し，入力データ X_n に対する情報行列 $G_n(\boldsymbol{\theta}; X_n)$, $H_n(\boldsymbol{\theta}; X_n)$ を

$$G_n(\boldsymbol{\theta}; X_n) = \frac{1}{n}\sum_{i=1}^{n} G(\boldsymbol{\theta}; x_i), \qquad H_n(\boldsymbol{\theta}; X_n) = \frac{1}{n}\sum_{i=1}^{n} H(\boldsymbol{\theta}; x_i) \quad (1.14)$$

により定める．$G_n(\boldsymbol{\theta}_o; X_n)$ は正定値であると仮定し，情報行列 $J_n(\boldsymbol{\theta}_o; X_n)$ を

$$J_n(\boldsymbol{\theta}_o; X_n) = H_n(\boldsymbol{\theta}_o; X_n) G_n^{-1}(\boldsymbol{\theta}_o; X_n) H_n(\boldsymbol{\theta}_o; X_n)$$

により定義する．(1.2) 式の仮定により，$n \to \infty$ における $G_n(\boldsymbol{\theta}_o; X_n)$, $H_n(\boldsymbol{\theta}_o; X_n)$ の極限をそれぞれ $G(\boldsymbol{\theta}_o)$, $H(\boldsymbol{\theta}_o)$ とおくと，$J_n(\boldsymbol{\theta}_o; X_n)$ は $n \to \infty$ のとき

$$J_n(\boldsymbol{\theta}_o; X_n) \to J(\boldsymbol{\theta}_o) = H(\boldsymbol{\theta}_o) G^{-1}(\boldsymbol{\theta}) H(\boldsymbol{\theta}_o)$$

と収束する．このとき，適当な正則条件のもとで，

$$nE_{Y_n}[(\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}_o)(\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}_o)^T] \quad \longrightarrow \quad J^{-1}(\boldsymbol{\theta}_o)$$

と収束することが示される．さらに対称行列 $F(\boldsymbol{\theta})$ を

$$F_{ab}(\boldsymbol{\theta}) = \frac{\partial^2 K(\boldsymbol{\theta})}{\partial\theta^a \partial\theta^b} = \int H(\boldsymbol{\theta}_o; x)_{ab} q(x)dx$$

により定義すると，データ数 n が大きいとき，(1.13) 式より

$$R(X_n) = K_* + \frac{1}{2n}\mathrm{Tr}\big[F(\boldsymbol{\theta}_o) J^{-1}(\boldsymbol{\theta}_o)\big] + o_p(\tfrac{1}{n})$$

となるので，X_n を用いて

$$R(X_n) \approx K_* + \frac{1}{2n}\mathrm{Tr}\big[F(\boldsymbol{\theta}_o)J_n^{-1}(\boldsymbol{\theta}_o; X_n)\big] \tag{1.15}$$

と近似できる．以上により，汎化誤差の期待値を最小にする能動学習の規準

$$\mathrm{Tr}\big[F(\boldsymbol{\theta}_o)J_n^{-1}(\boldsymbol{\theta}_o; X_n)\big] \tag{1.16}$$

が得られた．

システムが置かれる環境から与えられたデータを学習に使う場合，X_n は Q からの独立なサンプルである．このような学習は能動学習に対して**受動学習**と呼ばれる．能動学習を行うことにより，受動学習よりも汎化誤差を小さくすることが期待される．大数の法則により，受動学習の場合には，$G(\boldsymbol{\theta}_o) = E_Q[G(\boldsymbol{\theta}_o; x)]$, $H(\boldsymbol{\theta}_o) = E_Q[H(\boldsymbol{\theta}_o; x)]$ となる．特に $\ell(y, \boldsymbol{s}) = -\log r(y|\boldsymbol{s})$ の場合には，$\int \frac{\partial r(y|s)}{\partial s} dy = 0$, $\int \frac{\partial^2 r(y|s)}{\partial s \partial s} dy = 0$ を用いることにより，

$$G(\boldsymbol{\theta}; x) = H(\boldsymbol{\theta}; x)$$

が得られるので，$J(\boldsymbol{\theta}) = F(\boldsymbol{\theta}) = G(\boldsymbol{\theta}) = H(\boldsymbol{\theta})$ が成り立つ．すると，パラメータ $\boldsymbol{\theta}$ の次元を d とするとき

$$R(X_n) \approx K_* + \frac{d}{2n}$$

となることがわかる．能動学習の効果の理論値は，$\mathrm{Tr}[F(\boldsymbol{\theta}_o)J_n(\boldsymbol{\theta}_o; X_n)^{-1}]$ が d よりどれだけ小さくなるかによって決定される．

能動学習の規準 (1.16) 式は，未知パラメータ $\boldsymbol{\theta}_o$，言い換えれば真の関数 $\boldsymbol{\varphi}_o(x)$ に依存している．そこで，これを推定量 $\widehat{\boldsymbol{\theta}}$ で置き換える必要が生じるのであるが，それは後に回し，手始めに (1.16) 式が $\boldsymbol{\theta}_o$ に依存しない場合を考察してみよう．これは関数族 $\boldsymbol{\varphi}(x; \boldsymbol{\theta})$ が $\boldsymbol{\theta}$ に関して線形の場合である．

1.3.2　汎化誤差を小さくする能動学習 – 線形の場合 –

ここでは簡単のため出力次元 M が 1 の場合に話を限る．学習機械が，K 個の固定された関数 $\psi_k(x)$ とパラメータ v_k $(1 \leq k \leq K)$ とを用いて，

$$\varphi(x; \boldsymbol{v}) = \sum_{k=1}^{K} v_k \psi_k(x) = \boldsymbol{v}^T \boldsymbol{\psi}(x)$$

($\boldsymbol{v} = (v_1, \ldots, v_K)^T$, $\boldsymbol{\psi}(x) = (\psi_1(x), \ldots, \psi_K(x))^T$) という線形の関数族で定義されているとする．さらに，損失関数は 2 乗誤差 $\ell(y; s) = \frac{1}{2}\|y - s\|^2$ であり，条件付き確率密度 $r(y|s)$ は平均 0 分散 σ^2 のある確率密度関数 $g(s)$ を用いて $r(y|s) = g(y - s)$ と表すことができると仮定する．このとき簡単な計算により，

$$F(\boldsymbol{\theta}_o) = E_Q[\boldsymbol{\psi}(x)\boldsymbol{\psi}(x)^T], \qquad J_n(\boldsymbol{\theta}_o; X_n) = \tfrac{1}{\sigma^2}\tilde{J}_n(X_n) = \tfrac{1}{\sigma^2}\overline{\boldsymbol{\psi}(x)\boldsymbol{\psi}(x)^T}$$

となり，これらは $\boldsymbol{\theta}_o$ に依存しないことがわかる．ここで，$\tilde{J}_n(X_n)$ は

$$\tilde{J}_n(X_n) = \overline{\boldsymbol{\psi}(x)\boldsymbol{\psi}(x)^T} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{\psi}(x_i)\boldsymbol{\psi}(x_i)^T$$

で定義される，学習データの入力点の標本相関行列であり，デザイン行列と呼ばれることもある．さらに，以上の仮定のもとでは，(1.15) 式は，$n \to \infty$ における近似ではなく，$R(X_n)$ を厳密に与える式であることが簡単な計算によりわかる．

(1.16) 式は X_n のみの関数であり，学習を行う以前に最適化を行うことが可能である．実際，入力分布 Q が特殊な場合に，多項式などいくつかの関数系に関して最適なデータ採取点が知られている ([3], Section 2.3, 2.4)．以下に三角関数の場合の最適データ採取点を示しておこう．

入力の空間として閉区間 $[0, 2\pi]$ を考え，自然数 H と $K = 2H + 1$ に対して，三角関数系

$$\psi_1(x) = 1, \quad \psi_{2j}(x) = \sqrt{2}\cos(jx), \quad \psi_{2j+1}(x) = \sqrt{2}\sin(jx), \quad (1 \leq j \leq H)$$

を設定する．入力空間上の分布 Q として区間 $[0, 2\pi]$ 上の一様分布を定めると，$\{\psi_k(x) \mid k = 1, \ldots, K\}$ が $L^2(Q)$ の正規直交系となることは容易にわかる．すると，$F(\boldsymbol{\theta}) = I_K$ （K 次単位行列）であり，汎化誤差最小の学習データは

$$\mathrm{Tr}[\tilde{J}_n(X_n)^{-1}] \tag{1.17}$$

を最小にする入力点で取ればよいことがわかる．実は次の定理が示すように，等間隔に取った X_n がこの条件を満たしている．

定理 1 (1.17) 式が最小となるのは $\tilde{J}_n = I_K$ の場合である．また，$n \geq K$ のとき，入力点

$$x_i = \frac{i-1}{n}2\pi, \qquad i = 1, \ldots, n$$

はこの条件を満足する．

1.3.3 汎化誤差を小さくする能動学習 – 一般の場合 –

1.3.2 節で見た線形の場合とは異なり，一般には能動学習の規準式 (1.16) は未知パラメータ $\boldsymbol{\theta}_o$ に依存している．そこで，これを推定量 $\widehat{\boldsymbol{\theta}}$ で置き換え，

$$\mathrm{Tr}[F(\widehat{\boldsymbol{\theta}})J_n(\widehat{\boldsymbol{\theta}}; X_n)^{-1}] \tag{1.18}$$

を考えるとよい．しかしながら，推定量 $\widehat{\boldsymbol{\theta}}$ はデータを決めないと計算できないので，比較的少数の学習データによって求められた推定量から出発し，新たな学習データ採取点の最適化と推定量の更新とを交互に行っていくシーケンシャルな学習が自然と必要になる（図**??**）．

これをまとめると以下のような能動学習の手順が構成できる．

[シーケンシャルな能動学習]

1. 初期学習データ D_{N_0} を用意する.

2. D_{N_0} を用いて初期推定量 $\widehat{\boldsymbol{\theta}}_{N_0}$ を計算する.

3. $n := N_0 + 1$ とおく.

4. $X_n = X_{n-1} \cup \{x^{(n)}\}$ として，次式を最小にする $x^{(n)}$ を算出する.

$$\mathrm{Tr}\left[F(\widehat{\boldsymbol{\theta}}_{n-1})J_n(\widehat{\boldsymbol{\theta}}_{n-1}; X_n)^{-1}\right] \tag{1.19}$$

5. システムに $x^{(n)}$ を入力し，それに対する出力 $y^{(n)}$ を観測する.

6. 学習データを $D_n := D_{n-1} \cup \{(x^{(n)}, y^{(n)})\}$ により更新する.

7. 学習データ D_n を用いて推定量 $\widehat{\boldsymbol{\theta}}_n$ を計算する.

8. $n := n + 1$ とおく.

9. $n > N$ ならば終了．そうでないならば **4** へ行く.

ここではデータ点を 1 個ずつ取るような方法を述べたが，一度に複数個のデータ点をとるようにしてもよい.

さて，ここで述べたシーケンシャルな能動学習は最も基本的なものであるが，応用される問題や使われる関数系によってはいくつかの問題点も含んでいる．まず第 1 の問題は，パラメータ学習の困難さに関する点である．非線形な関数系を学習機械に用いた場合には，パラメータ $\boldsymbol{\theta}$ の最適化に最急降下法や Newton 法といった非線形最適化手法を用いることになる．一方，能動学習により得られる学習データは，最適パラメータが正確に求められれば汎化誤差の期待値を最小にするが，逆に経験損失関数の形状を複雑にしてしまい，パラメータ $\boldsymbol{\theta}$ の最適化を困難にする可能性がある．実際，汎化誤差最小の規準による最適データ採取点を求めることにより，同じ点が繰り返し選ばれ，パラメータの最適化が困難になる例が，Fukumizu ([4]) に示されている.

最適な入力点が同じような点を選ぶ現象は，理論的には次のように理解することが出来る．規準式 (1.16) は，対称行列 $G_n(\boldsymbol{\theta}_o; X_n)$ と $H_n(\boldsymbol{\theta}_o; X_n)$ の関数と見なせる．これらは合わせて $d(d-1)$ 次元のベクトルであるが，(1.14) 式からわかるように，任意のベクトルが集合 $\Delta = \{(G(\boldsymbol{\theta}_o; x), H(\boldsymbol{\theta}_o; x)) \mid x \in X_n\}$ の点の凸結合として表現できる．付録で述べた Carathéodory の定理に従うと，これは高々 $d(d-1)+1$ 個の点の凸結合として表現できる．従って，もし最適なデータ採取点が存在すれば，それらは高々 $d(d-1)+1$ 個の点をある割合で繰り返し取ることにより近似できる．この結果として，バリエーションの少ない入力点が選ばれる可能性が生じる.

第 2 の問題点は，最適化に伴う計算コストの問題である．非線形関数を学習機械として用いる場合には，機械のパラメータ $\boldsymbol{\theta}$ を数値的最適化により求める必要があるが，それに加えて新たなデータ採取点も数値的最適化で求めなくてはならない．シーケンシャルな能動学習においては，これをデータ採取ごとに繰り返し行うので，情報行列に対する多くの演算を必要とし計算コストは非常に高くなる.

以上のような問題点に対処するための工夫のひとつとして，以下では確率的な能動学習につ

いて述べる．

1.3.4 確率的な能動学習

前節で述べた問題から，(1.16) 式の規準を厳密に最小化するのではなく，確率的にばらつきを残した入力点を選ぶ方法が考えられている．Fukumizu ([4]) で提案されている多点探索を用いた方法を以下に述べる．この方法は，特に学習の初期段階でのパラメータ最適化の失敗を防ぐため，学習の初期においては得られるデータ採取点のばらつきを大きくし，徐々に真に最適な採取点が見つかりやすくする．以下で T_n は n に関して単調増加な自然数列である．

[確率的な能動学習（多点探索による方法）]

1. 初期学習データ D_{N_0} を用意する．
2. D_{N_0} を用いて初期推定量 $\widehat{\boldsymbol{\theta}}_{N_0}$ を計算する．
3. $n := N_0 + 1$ とおく．
4. T_n 個の候補点 $x_{<1>}, \ldots, x_{<T_n>}$ を発生させる．
5. 次式の最小化問題の解 $x_{<j>}$ を新しい入力点 $x^{(n)}$ とする．

$$\min_{j=1,\ldots,T_n} \mathrm{Tr}\left[F(\widehat{\boldsymbol{\theta}}_{n-1})J_n(\widehat{\boldsymbol{\theta}}_{n-1}; X_{n-1} \cup \{x_{<j>}\})^{-1}\right]$$

6. システムに $x^{(n)}$ を入力し，それに対する出力 $y^{(n)}$ を観測する．
7. 学習データを $D_n := D_{n-1} \cup \{(x^{(n)}, y^{(n)})\}$ により更新する．
8. 学習データ D_n を用いて推定量 $\widehat{\boldsymbol{\theta}}_n$ を計算する．
9. $n := n + 1$ とおく．
10. $n > N$ ならば終了．そうでないならば **4** へ行く．

もし候補点を入力分布 Q から発生させたとすると，この学習は初期においては受動的学習に近い学習を行ない徐々に最適に近いデータを採取していくことになる．これにより，実際に学習データ採取点のばらつきが大きくなることが実験的にも確認されている（Fukumizu [4]）．また，この方法は計算コストの点でも利点が大きい．一般には，(1.19) 式の最適化には数値的な非線形最適化の手法を用いる必要があるが，上の多点探索を用いるとその必要はない．

ばらつきを残した能動学習と考えられる他の例として，Cohn([5]) の提案した方法がある．この方法は，(1.19) 式の情報行列の計算を省略するために，定環境下での入力分布 Q に従う参照点 x_r をデータ点選択時ごとにひとつ取り，$F(\boldsymbol{\theta})$ のかわりに $H(\boldsymbol{\theta}; x_r)$ を用いて

$$\min_{x^{(n)}} \mathrm{Tr}[H(\widehat{\boldsymbol{\theta}}_{n-1}; x_r)J_n(\widehat{\boldsymbol{\theta}}_{n-1}; X_{n-1} \cup \{x^{(n)}\})^{-1}]$$

を達成する $x^{(n)}$ を次の入力点として選択するものである．これは本来 $F(\widehat{\boldsymbol{\theta}})$ の積分計算を省略することを目的として提案された手法であるが，Q からの参照点を取ることにより，データ採取点にバリエーションを持たせる働きもあると考えられる．

1.3.5　その他の規準による最適データ採取点探索

今まで汎化誤差最小を規準とした能動学習を述べてきたが，汎化誤差を定義するためには，実環境下における入力分布 Q が必要であった．しかしながら，この分布を学習時に知ることができない場合もあるため，汎化誤差とは異なる規準を考えることも重要である．統計学の最適実験計画や，Response Surface Methodology ([6],[7]) と呼ばれる回帰分析の方法論においては，さまざまな規準によるデータ採取点の最適化が研究されてきた．ここではその中で，主に最適実験計画で議論されてきた Minmax 規準，D-optimality，A-optimality などを紹介する．

本節では，以下の仮定のもとで問題を考えることにする．

1. 真の入出力関係は設定したモデルに含まれており，$\boldsymbol{\varphi}(x;\boldsymbol{\theta}_o) = \boldsymbol{\varphi}(x)$ を満たす．
2. (1.12) 式が成立する．すなわち，任意の x について

$$\frac{\partial}{\partial \boldsymbol{\theta}} \int \ell(y, \boldsymbol{\varphi}(x;\boldsymbol{\theta}_o)) r(y|\boldsymbol{\varphi}_o(x)) dy = 0.$$

3. ある正数 c があって $H(\boldsymbol{\theta}_o; x) = cG(\boldsymbol{\theta}_o; x)$ が成り立つ．

上の仮定のうち，2, 3 は，次の (A), (B) のうちどちらか一方が満足されれば成立する．

(A) 損失関数が負の対数尤度：　$\ell(y, \boldsymbol{s}) = -\log r(y|\boldsymbol{s})$
(B) 損失関数が 2 乗誤差 $\ell(y, \boldsymbol{s}) = \frac{1}{2}\|y - \boldsymbol{s}\|^2$ で，$r(y|\boldsymbol{s})$ は，平均 0 分散共分散行列 $\sigma^2 I_M$ なる確率密度関数 $g(\boldsymbol{s})$ を用いて $r(y|\boldsymbol{s}) = g(y - \boldsymbol{s})$ と書ける．

この事実をチェックするのはそれほど難しくないので，ここでは省略するが，上の (A),(B) は特によく用いられる重要なケースである．

Minmax 規準

経験損失最小によって得られた学習機械 $\boldsymbol{\varphi}(x;\widehat{\boldsymbol{\theta}})$ に対して，各 x における推定の誤差を表す量として

$$d(x; X_n) = E_{Y_n}\Big[\int \ell(y, \boldsymbol{\varphi}(x;\widehat{\boldsymbol{\theta}})) r(y|\boldsymbol{\varphi}(x;\boldsymbol{\theta}_o)) dy\Big] - \int \ell(y, \boldsymbol{\varphi}(x;\boldsymbol{\theta}_o)) r(y|\boldsymbol{\varphi}(x;\boldsymbol{\theta}_o)) dy$$

を考える．上記 (B) のケースでは，$d(x; X_n)$ は定数倍を除いて点 x における誤差分散の期待値 $E_{Y_n}\big[\|\boldsymbol{\varphi}(x;\widehat{\boldsymbol{\theta}}) - \boldsymbol{\varphi}(x;\boldsymbol{\theta}_o)\|^2\big]$ に一致する．**Minmax 規準**とは，学習データの入力点 X_n を

$$\min_{X_n} \max_{x} d(x; X_n) \tag{1.20}$$

という minmax 問題の解として求めるものである．すなわち，最悪の誤差をなるべく小さくしようとする学習データ設計法である．

(1.20) 式の形では右辺が X_n にどのように依存しているかわかりにくいので，漸近展開ないしは線形近似を行ってみよう．(1.12) 式を用いると，Taylor 展開により

$$d(x;X_n) \approx \frac{1}{2}\sum_{a,b=1}^{d} H(\boldsymbol{\theta}_o;x)_{ab} E_{Y_n}[(\widehat{\theta}^a - \theta_o^a)(\widehat{\theta}^b - \theta_o^b)]$$

となるが，1.3.1 節でも述べたように，仮定の 2，3 のもとでは，n が大きいとき

$$E_{Y_n}[(\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}_o)(\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}_o)^T] \approx \tfrac{1}{n} J_n(\boldsymbol{\theta}_o)^{-1} = \tfrac{1}{cn} H_n(\boldsymbol{\theta}_o;X_n)^{-1}$$

と近似できる．したがって，

$$d(x;X_n) \approx \frac{1}{2nc}\mathrm{Tr}[H(\boldsymbol{\theta}_o;x) H_n(\boldsymbol{\theta}_o;X_n)^{-1}] \tag{1.21}$$

である．そこで，線形化された minmax 規準として，

$$\tilde{d}(x;X_n) = \mathrm{Tr}[H(\boldsymbol{\theta}_o;x) H_n(\boldsymbol{\theta}_o;X_n)^{-1}] \tag{1.22}$$

に対して

$$\min_{X_n} \max_{x} \tilde{d}(x;X_n) \tag{1.23}$$

を考えることができる．(1.23) 式は未知パラメータ $\boldsymbol{\theta}_o$ を含んでいるが，汎化誤差最小規準と同様，シーケンシャルな学習の規準として利用できる．また $\boldsymbol{\theta}_o$ に依存しない線形の場合には一度に X_n の最適化が可能である．

D-optimality

情報行列 J_n が大きいことが推定精度を上げると考えられるので，情報行列の行列式の大きさを能動学習の規準として採用することにして，

$$\det J_n(\boldsymbol{\theta}_o;X_n) \tag{1.24}$$

を最大にする入力点を考える．このような規準を **D-optimality** と呼ぶ．

A-optimality

$$\mathrm{Tr}[J_n(\boldsymbol{\theta}_o;X_n)^{-1}] \tag{1.25}$$

を最小にする規準を **A-optimality** と呼ぶ．これは，漸近的にはパラメータの平均 2 乗誤差 $\frac{1}{m}\sum_{a=1}^{m} E_{Y_n}[(\hat{\theta}^a - \theta_o^a)^2]$ を最小にするような規準と考えることができる．

誤差分散に基づく逐次的 D-optimality

D-optimality において，ケース (B) の場合を考える．このとき，$J_n(\boldsymbol{\theta}_o;X_n) = cH_n(\boldsymbol{\theta}_o;X_n)$ であるから，$\det H_n(\boldsymbol{\theta}_o;X_n)$ を最大化すればよいが，これを逐次的に実行することを考える．

以降簡単のため，情報行列 $H_n(\boldsymbol{\theta}_o;X_n)$ を H_n と略する．すると，

$$H(\boldsymbol{\theta}_o;x) = \frac{\partial \boldsymbol{\varphi}(x;\boldsymbol{\theta}_o)}{\partial \boldsymbol{\theta}}^T \frac{\partial \boldsymbol{\varphi}(x;\boldsymbol{\theta}_o)}{\partial \boldsymbol{\theta}}$$

($\frac{\partial \boldsymbol{\varphi}(x;\boldsymbol{\theta}_o)}{\partial \boldsymbol{\theta}}$ は $M \times d$ 行列としている）であるから，H_{n+1} は H_n を用いて，

$$H_{n+1} = \tfrac{n}{n+1}\Big(H_n + \frac{1}{n}\frac{\partial \boldsymbol{\varphi}(x^{(n+1)};\boldsymbol{\theta}_o)}{\partial \boldsymbol{\theta}}^T \frac{\partial \boldsymbol{\varphi}(x^{(n+1)};\boldsymbol{\theta}_o)}{\partial \boldsymbol{\theta}}\Big)$$

と書くことができる．したがって，付録の補題 1 より

$$\det H_{n+1} = \big(\tfrac{n}{n+1}\big)^d \det H_n \det\Big(I_M + \frac{1}{n}\frac{\partial \boldsymbol{\varphi}(x^{(n+1)};\boldsymbol{\theta}_o)}{\partial \boldsymbol{\theta}} H_n^{-1} \frac{\partial \boldsymbol{\varphi}(x^{(n+1)};\boldsymbol{\theta}_o)}{\partial \boldsymbol{\theta}}^T\Big)$$

を得る．X_n が既に決まっているときに $\det H_{n+1}$ を最大化するには，

$$\det\Big(I_M + \frac{1}{n}\frac{\partial \boldsymbol{\varphi}(x^{(n+1)};\boldsymbol{\theta}_o)}{\partial \boldsymbol{\theta}} H_n^{-1} \frac{\partial \boldsymbol{\varphi}(x^{(n+1)};\boldsymbol{\theta}_o)}{\partial \boldsymbol{\theta}}^T\Big) \tag{1.26}$$

を最大にする $x^{(n+1)}$ を選択すればよい．出力次元 M が 1 の場合には，これは $\tilde{d}(x;X_n)$ を最大にする点をデータ採取点にすることを意味している．出力が多次元でも，n が十分大きいときに任意の行列 B に対して

$$\mathrm{Tr}[\tfrac{1}{n}B] = \log\det\exp\big(\tfrac{1}{n}B\big) \approx \log\det[I_M + \tfrac{1}{n}B]$$

という近似が成り立つことに注意すると，(1.26) 式の log は

$$\mathrm{Tr}\Big[\frac{1}{n}\frac{\partial \boldsymbol{\varphi}(x^{(n+1)};\boldsymbol{\theta}_o)}{\partial \boldsymbol{\theta}} H_n^{-1} \frac{\partial \boldsymbol{\varphi}(x^{(n+1)};\boldsymbol{\theta}_o)}{\partial \boldsymbol{\theta}}^T\Big] = \frac{1}{n}\tilde{d}(x^{(n+1)};X_n)$$

により近似される．以上により，誤差分散の期待値が最も大きくなる点を次のデータ採取点としていけば，近似的にその都度 $\det J_n(\boldsymbol{\theta}_o; X_n \cup \{x^{(n+1)}\})$ を最大にするように $x^{(n+1)}$ を採取していることになる．Fedorov ([3], Section 2.5, 4.2) には，$M = 1$ の場合に，この手続きによって得られるデータ点が，$n \to \infty$ のとき D-optimal であることが示されている．

誤差分散 $d(x;X_n)$ を推定するためには，ブートストラップ法 ([8]) を用いることが可能である．Kindermann ら ([9]) は，ここで述べた誤差分散をブートストラップ法で推定し，それを最大にする点を探索する能動学習法を提案している．

ここでは逐次的に誤差分散を小さくするデータ点が D-optimality と関連する場合を議論したが，実は D-optimality と $\tilde{d}(x;X_n)$ の minmax 規準にはさらに密接な関係がある．それは，比較的弱い条件下で，D-optimality と minmax 規準により与えられるデータ点が完全に一致するという，Kiefer-Wolfovitz の同値性定理である．これについては省略するが，文献 [10] に初等的で明快な証明があるので，詳しくはそれを参照して欲しい．

1.4　能動学習とモデル選択

ここまで能動学習の規準を導く際には，学習機械として十分に能力の高い関数族を設定し，真の入出力関係が学習機械によって実現可能であると仮定した．現実にはこのような仮定が成り立つとは限らないが，実は今まで述べた能動学習が効果を発揮するには，この仮定は非常に本質的である．本節ではこの仮定が能動学習に及ぼす影響を説明し，能動学習を効果的に働かせるためのモデル選択について述べる．

1.4.1 不適合なモデルのもとでの能動学習の悪影響

設定したモデルに真の入出力関係が含まれていない場合に能動学習がどのような結果をもたらすかを簡単な例によって考察する．区間 $[-1,1]$ 上の関数を学習する問題を考え，1 次関数の族 $\mathcal{F} = \{ax + b \mid a, b \in \mathbb{R}, x \in [0,1]\}$ を学習機械として設定する．結果を評価するための入力分布 Q は $[-1,1]$ 上の一様分布とし，損失関数は 2 乗誤差 $\ell(y,s) = (y-s)^2$ とする．Q に関する 2 乗可積分関数の空間 $L^2(Q)$ の正規直交系

$$h_1(x) = 1, \qquad h_2(x) = \sqrt{3}x, \qquad h_3(x) = \tfrac{\sqrt{5}}{2}(3x^2 - 1)$$

を用いると，$\mathcal{F} = \{\theta_1 h_1(x) + \theta_2 h_2(x) \mid \theta_1, \theta_2 \in \mathbb{R}\}$ である．真の入出力関係はこのモデルから少しずれており，

$$\varphi_o(x) = \lambda h_3(x)$$

という 2 次関数に，平均 0 分散 σ^2 のガウスノイズが加わったものと仮定する．

この設定のもとで経験損失最小（最小 2 乗誤差）による推定の汎化誤差を考える．今の場合 $F(\boldsymbol{\theta}_o) = 2I_2$ となるので，汎化誤差最小規準は

$$\min_{X_n} \mathrm{Tr}\left[\begin{pmatrix} 1 & \sqrt{3}\overline{x} \\ \sqrt{3}\overline{x} & 3\overline{x^2} \end{pmatrix}^{-1}\right] = \min_{X_n} \frac{3\overline{x^2} + 1}{3\overline{x^2} - 3(\overline{x})^2} \tag{1.27}$$

になる．ここで $\overline{x}$, $\overline{x^2}$ はそれぞれ x, x^2 のサンプル平均である．(1.27) 式を最小にするには，$|\overline{x}|$ を小さく $\overline{x^2}$ を大きくするのがよいので，$x = 1$ と $x = -1$ にデータ採取点を半分ずつおくのが最適となる．

ところが，$\lambda \neq 0$ のとき，すなわち真の入出力関係がモデルに属していないときに，このデータ採取法がよくないことは図?? を見れば明らかである．実際，データ数が無限大になったとき最小 2 乗誤差推定量 $\widehat{\boldsymbol{\theta}}$ は $\boldsymbol{\theta}^* = (\sqrt{5}\lambda, 0)$ に近づくので，汎化誤差の期待値は，真の入出力関係と関数族のずれが主要項となり，

$$R(X_n) = \sigma^2 + 3\lambda^2 + O(1/n) \tag{1.28}$$

で与えられることが簡単な計算によりわかる．

一方，受動的な学習においては，最適なパラメータ $\boldsymbol{\theta}^{**}$ は

$$\int_{-1}^{1} (\theta_1 h_1(x) + \theta_2 h_2(x) - \lambda h_3(x))^2 dx$$

を最小にするものとして与えられる．$h_a(x)$ の正規直交性より $\theta_1^{**} = \theta_2^{**} = 0$ が得られ，

$$R(X_n) = \sigma^2 + \lambda^2 + O(1/n) \tag{1.29}$$

となる．(1.28),(1.29) 式より，データ数 n がある程度大きいと受動学習のほうが汎化誤差が小さくなることがわかる．これは，真の入出力関係が設定したモデルに含まれるという前提が成り立たないために，その前提で導かれた最適学習データが，真の入出力関係とモデルとの距離を大きくしてしまった結果である．

1.4.2　モデル選択を組み合わせた能動学習

モデルの不適合によって引き起こされる能動学習の悪影響を防ぐには，（1）モデルが不適合な場合に汎化誤差を推定して，それを最小化するデータ点を探す，あるいは（2）モデルが適合するようにモデル選択を念入りに行う，の 2 種類の方針が考えられる．第 1 の方針では，モデルが一致していない場合に汎化誤差の推定が容易でないことが問題となる．未知である真の入出力関係とモデルとの距離を推定する必要が生じるため，1.3.1 節で論じた漸近的な方法は適用できない．また，汎化誤差を推定するには Cross Validation や Bootstrap ([8]) などの方法も考えられるが，汎化誤差の推定値が入力データ点にどのように依存するかを知るのが難しいくなる[*1]．そこで，後者の方針にあるように，十分大きいモデルを用意して，真の入出力関係が設定したモデルにほぼ含まれているような状況を作り出すのがよいと考えられる．

このようなモデル選択は，汎化誤差を最小にするために行う通常のモデル選択とは目的を異にしていることに注意しておく．モデル選択には，赤池情報量規準（AIC）や最小記述長原理（MDL）をはじめ様々な方法があるが，これらは経験損失にモデルの複雑度を表すペナルティ項を加えたものを最小化する方法である．小さすぎるモデルでは経験損失が大きくなり，大きすぎるモデルではペナルティが大きくなることにより，適切なモデルが選択されるようになっている．しかしながら，能動学習に対してこの方針をそのまま用いるのは適当ではない．実際，能動学習では汎化誤差を評価するための入力分布と学習データとして用いられるデータの分布が違ってもよいので，1.4.1 節の例でみたように，小さすぎるモデルを選んでも経験損失が大きくならず，そういうモデルが選択される可能性がある．これは能動学習に大きな悪影響を及ぼす．

従って，能動学習を行う際には十分大きいモデルを用い，汎化誤差を小さくする役割はデータ点選択に任せるのがよいと考えられる．実際，実数直線上の多項式近似の問題では，データ採取点を上手く設計すると，理論的にはモデルサイズによらず汎化誤差の期待値が一定の値まで下げられることが示されている ([12])．Paass et al.([13]) や Kindermann et al. ([9]) では，能動学習時にモデル選択を行って徐々にモデルの構造を複雑にしていく方法を提案している．また，Fukumizu([4]) では，ニューラルネットモデルの能動学習において，大きいモデルから出発して必要な場合にモデルを小さくする方法を提案している．

1.5　ニューラルネットの能動学習

ここでは学習機械として 3 層パーセプトロン ([14]) タイプのニューラルネットを考え，その能動学習において生じる特別な状況を論じる．3 層パーセプトロンは，次式の関数系 $\boldsymbol{\varphi}(x;\boldsymbol{\theta})$

[*1] Box & Draper ([11]) では，真の関数とモデルとのずれを推定しそれを最小化するような入力点設計法を議論している．しかし，彼らの議論は，真の関数がある線形の関数族に属しておりモデルがその部分族である場合を前提としており，適用範囲は限られると思われる．

として定義される．

$$\boldsymbol{\varphi}(x;\boldsymbol{\theta}) = \sum_{j=1}^{H} \boldsymbol{v}_j \, s(\boldsymbol{w}_j^T x + \zeta_j) + \boldsymbol{\eta}.$$

ここで，$\boldsymbol{v}_j,\ \boldsymbol{\eta} \in \mathbb{R}^M,\ \boldsymbol{w}_j \in \mathbb{R}^L,\ \zeta_j \in \mathbb{R}\ (1 \leq j \leq H)$ はパラメータであり，$\boldsymbol{\theta} = (\boldsymbol{v}_1^T, \zeta_1, \ldots, \boldsymbol{v}_H^T, \zeta_H, \boldsymbol{v}_1^T, \ldots, \boldsymbol{v}_H^T, \boldsymbol{\eta}^T)^T$ はパラメータ全体をあらわすベクトルである．また，$s(t)$ は一般に単調飽和型の 1 変数非線形関数であり，ロジスティック関数 $s(t) = 1/(1+e^{-t})$ や $s(t) = \tanh(t)$ などがよく使われる．このモデルは図**??**で表されるようなグラフィカルな表現を持ち，H は中間素子の個数を表している．

ニューラルネットに対しても 2 節で述べた学習の一般的枠組みが利用できるが，能動学習やモデル選択を考えるとき特別な事情が現れる．それは，ニューラルネットモデルにおけるパラメータの識別不能性と呼ばれる問題である．そこで，識別不能性について以下で説明しよう．簡単のため，出力が 1 次元で中間素子が 2 個のモデル

$$\varphi(x;\boldsymbol{\theta}) = v_1 \, s(\boldsymbol{w}_1^T x + \zeta_1) + v_2 \, s(\boldsymbol{w}_2^T x + \zeta_2) + \eta$$

を用いて説明する．中間素子関数 $s(t)$ は $\tanh(t)$ であるとする．真の関数 $\varphi_o(x)$ がモデルに含まれているとし，これが中間素子 1 個で表現可能な関数

$$\varphi_o(x) = v_0 \, s(\boldsymbol{w}_0^T x + \zeta_0) + \eta_0$$

であったと仮定しよう．真の関数を実現するパラメータ $\boldsymbol{\theta}$ を考えると，2 つの中間素子の内部パラメータを等しくおいた

$$\{\boldsymbol{\theta} \mid (\boldsymbol{w}_1^T, \zeta_1) = (\boldsymbol{w}_2^T, \zeta_2) = (\boldsymbol{w}_0^T, \zeta_0), v_1 + v_2 = v_0, \eta = \eta_0\}$$

という集合内のパラメータは，すべて φ_0 と同一の関数を定める．重要なのは，この集合が高次元集合（この場合は直線）となっている点である．また，$\{\boldsymbol{\theta} \mid (\boldsymbol{w}_1^T, \zeta_1) = (\boldsymbol{w}_0^T, \zeta_0), v_1 = v_0, \boldsymbol{w}_2 = 0, v_2 s(\zeta_2) + \eta = \eta_0\}$, $\{\boldsymbol{\theta} \mid (\boldsymbol{w}_1^T, \zeta_1) = (\boldsymbol{w}_0^T, \zeta_0), v_1 = v_0, \eta = \eta_0, v_2 = 0\}$ で定義される 2 つの集合も，$\varphi_0(x)$ と同一の関数を定めるが (図**??**参照)，パラメータ空間の中で前者は 2 次元の曲面，後者は $(L+1)$ 次元のアフィン平面である．以上により，2 個の中間素子を持つモデルの中で，1 個の中間素子で実現可能な関数を定めるパラメータ集合は高次元集合を成していることが確認できる．このように，同一の関数を定めるパラメータが一意的ではなく連続集合をなしているような状況を，パラメータが**識別不能**であるという．

中間素子の非線形関数が tanh やロジスティック関数である場合には，3 層パーセプトロンのパラメータが識別不能になるための条件は以下のように与えられる．詳細は [15], [16] をご覧頂きたい．

定理 2 中間素子関数を tanh とする．中間素子を H 個持つ 3 層パーセプトロンモデルにおいて，パラメータが識別不能であることと，そのパラメータが定める関数が $H-1$ 個以下の中間素子を持つ 3 層パーセプトロンで実現可能であることは同値である．さらにこれは，

[1] ある j が存在して，$\boldsymbol{w}_j = 0$

[2] ある j が存在して，$\boldsymbol{v}_j = 0$

[3] ある $j_1 \neq j_2$ が存在して，$(\boldsymbol{w}_{j_1}^T, \zeta_{j_1}) = \pm(\boldsymbol{w}_{j_2}^T, \zeta_{j_2})$

のいずれかが成り立つことと同値である．

真のパラメータが識別不能な状況で関数の学習を行うと何が起こるであろうか？ 1.3 節では真のパラメータ $\boldsymbol{\theta}_o$ を仮定して話を進めたが，今の状況ではそれが一意的ではないので $\widehat{\boldsymbol{\theta}}$ が $\boldsymbol{\theta}_o$ の近傍にあるという考えは適用できない．したがって，Taylor 展開や，$E_{Y_n}[(\widehat{\boldsymbol{\theta}}-\boldsymbol{\theta}_o)(\widehat{\boldsymbol{\theta}}-\boldsymbol{\theta}_o)^T]$ の漸近展開などを使うことができず，それに基づいて導かれた規準 (1.16) 式はそのまま適用することができない．このような識別不能性は，ニューラルネットに限らず，複雑な構造を持つさまざまなモデルが持つ問題である．この話題については [18] が詳しく論じている．

このような特異な現象は情報行列の退化としても捉えることができる．真のパラメータ集合の中の一点 $\boldsymbol{\theta}_o$ を固定して，そこでの情報行列 $G(\boldsymbol{\theta}_o)$ あるいは $H(\boldsymbol{\theta}_o)$ を考える．このとき，例でみたように $\boldsymbol{\theta}_o$ を含んだある方向に関して関数 $\varphi(x;\boldsymbol{\theta})$ は一定になる．この方向ベクトルを $\boldsymbol{u}$ と置くと，方向微分 $\boldsymbol{u}^T \frac{\partial \ell(y,\varphi(x;\boldsymbol{\theta}))}{\partial \boldsymbol{\theta}}$ は 0 になり，

$$\boldsymbol{u}^T \frac{\partial \ell(y,\varphi(x;\boldsymbol{\theta}))}{\partial \boldsymbol{\theta}} \frac{\partial \ell(y,\varphi(x;\boldsymbol{\theta}))}{\partial \boldsymbol{\theta}} \boldsymbol{u} = \boldsymbol{u}^T \frac{\partial^2 \ell(y,\varphi(x;\boldsymbol{\theta}))}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}} \boldsymbol{u} = 0$$

が成り立つので，$G(\boldsymbol{\theta}_o)$ や $H(\boldsymbol{\theta}_o)$ は退化している．このことからも，真のパラメータが識別不能な状況では，情報行列の逆行列や行列式を用いた能動学習の規準は用いることが出来ないことが確認できる．逆に，情報行列が逆行列を持たないのは，定理 2 の 3 条件のいずれかを満たす場合に限られることが証明されている ([17])．したがって，問題になるのは真の関数が少ない中間素子で表される場合である．現実の問題では，真の関数がモデルに属しており，しかもより小さい中間素子数で完全に実現されることはあり得ないかもしれないが，真の関数を実現するのに冗長に近い中間素子が存在すると，情報行列は 0 に近い固有値を有し，能動学習は安定しないことが予想される．

1.4 節で，能動学習を用いる際には，真の入出力関係がモデルによって実現可能なぐらいにモデルを大きく設定する必要があることを述べた．しかし，本節の考察によれば，ニューラルネットではモデルが大きすぎると能動学習がうまく適用できない．したがって，ニューラルネットの能動学習では必要かつ十分なモデルサイズを選択する必要がある．そのためのひとつの手法として，Fukumizu ([4]) は最初に十分大きいモデルから出発して，次のような中間素子削除付きの学習を行うことを提案している．ここでは，中間素子の関数を tanh とし，第 j 中間素子の出力値を $\hat{s}_j = s(\hat{\boldsymbol{w}}_j^T x + \hat{\zeta}_j)$ により省略している．また，T は適当な自然数，A は適当な正数である．

[中間素子削除付きの学習則]

1. $t := 1$.

2. $(x^{(t \bmod n)}, y^{(t \bmod n)})$ に対して $\widehat{\boldsymbol{\theta}}$ を更新する．

3. もし $t \bmod T = 0$ ならば，次の手続きを行う．

(a) もし $\|\hat{\boldsymbol{v}}_j\|^2 \int (\hat{s}_j - s(\hat{\zeta}_j))^2 q(x)dx < \dfrac{A}{n}$ ならば，第 j 中間素子を削除し $\boldsymbol{\eta} \mapsto \boldsymbol{\eta} + \boldsymbol{v}_j s(\zeta_j)$ と変更する．

(b) もし $\|\hat{\boldsymbol{v}}_j\|^2 \int (\hat{s}_j)^2 q(x)dx < \dfrac{A}{n}$, ならば，第 j 中間素子を削除する．

(c) もし $j_1 \neq j_2$ に対し $\|\hat{\boldsymbol{v}}_{j_2}\|^2 \int (\hat{s}_{j_2} \mp \hat{s}_{j_1})^2 q(x)dx < \dfrac{A}{n}$ ならば，第 j_2 中間素子を削除し $\boldsymbol{v}_{j_1} \mapsto \boldsymbol{v}_{j_1} \pm \boldsymbol{v}_{j_2}$ と変更する．

4. $t := t + 1$.

5. もし $t > t_{MAX}$ ならば終了する．そうでなければ **2** へ行く．

(a)–(c) の 4 つの判定式は，定理 2 の [1]–[3] に対応するものである．能動学習でパラメータを最適化する際に，上のような手法により冗長な中間素子を削除していき，情報行列が退化しないようにすることが重要となる．図**??**は，多点探索を用いた確率的な能動学習に中間素子の削減を組み合わせた場合の，汎化誤差と中間素子数の推移の一例を示している．

1.6 能動学習の応用例

最適実験計画という観点では 1950 年代から活発に理論的研究が行われているものの，入出力関係を表す回帰関数を学習する現実の問題に対して，汎化誤差を小さくするための最適データ点を実験を行いながら選んだ応用例は少ない．これは，実際の実験にかかるコストから，試行錯誤的なテストが難しいことが一因であろうと考えられる．また，従来の実験により知られている関数関係から人工的にデータを発生させ，ニューラルネットによる能動学習の実験を行った例として，Bayes 的な立場から導いた規準に従って焼夷弾の効果の分類問題を学習させた Belue et al. ([19]) などがある．また Fukumizu([4]) は色の表現を RGB から YMC に変換する関数を確率的な能動学習によって学習させた例を示している．以下では，後者の例についてやや詳しく述べる．

カラープリンタなどのカラー印刷を行う機械では，CMY（シアン，マゼンタ，イエロー）表色形でインクの色を指定することが多い．一方，作られた印刷の物理的な色は RGB（レッド，グリーン，ブルー）によって表現される．CMY と RGB の変換関数は理論的には得られているが，所望の RGB を得るために機械に与える CMY の設定値は，印刷機械の個々の特性などにより完全に理論値と一致するわけではない．そこで，特定の印刷機械に対し，ある RGB の値を持つ色を発生させるための CMY の設定値を求める問題は，カラー印刷において重要となる．この RGB から CMY への関数関係を 3 層パーセプトロンで学習する問題に能動学習の手法を用いる実験を行った．実験に際しては，実際の印刷機械からの測定を行わず，CMY から RGB への変換関数として知られている Neugebauer 方程式 ([20]) を数値的に逆に解き，それに観測ノイズとして小さい分散を持つガウスノイズを加えたものを真の入出力関係として用いた．Neugebauer 方程式は特にオフセット印刷では現実の色変換を非常によい精度で近似しているので，この入出力関係がよく学習できれば，実機においてもよい学習精度が期待できる．

実験に用いた学習機械はロジスティック関数を中間素子関数として持つ 3 層パーセプトロ

ンで，中間素子は 7 個から初め，1.5 節で述べた方法により冗長な中間素子があれば削除していった．能動学習は 1.3.4 節で述べた，多点探索による確率的な能動学習を用いた．図**??**の右のグラフは，学習データの個数を徐々に増加させていったときの，３０回の試行に対する汎化誤差の平均値を表している．汎化誤差を測るための入力分布 Q には $[0,1]^3$ 上の一様分布を用いている．このグラフからわかるように，受動的な学習と比較して能動学習法による汎化誤差の減少がみられ，能動学習が有効に機能していることが確認できる．

1.7　おわりに

本章を締めくくるにあたって，ここで取り上げられなかった話題について簡単に触れる．また，能動学習に関するさまざまな話題を扱った特集が文献 [21] にあるので，そちらも合わせてみていただくとよい．

最適データ点設計の問題においては，本章で述べたような経験損失最小化による学習のほかに Bayes 推定に基づいた方法も展開されている．1.3.5 節で述べた D-optimality, A-optimality などに対して，それぞれ Bayes 推定の枠組みにおける対応物が考察されている．Bayes 的な最適実験計画に関しては文献 [22] に詳しい解説がある．

本章で述べてきた能動学習の枠組みでは，学習のターゲットは時間的に不変な静的システムであり，かつ目的は学習の結果得られた推定量の精度であった．こういう簡単な場合を考えることによって理論的な展開が可能となり，それに基づくさまざまな手法が導き出された．しかしながら「能動的な学習」という言葉に対して，今まで述べた枠組みは限定的過ぎると感じられた方も多いであろう．実際，学習者が戦略を最適化する方法論には本章で述べたものと異なる枠組みも存在する．そのような例として Bandit 問題とマルコフ決定プロセスの学習についてごく簡単に紹介しておく．

Two-armed bandit とはアームがふたつ付いたスロットマシンのことである．ふたつのアームを引くとそれぞれ確率 p_1, p_2 で 1 ドル賞金がもらえる機械を想定する．このとき，p_1, p_2 が未知として，n 回アームを引いた際に賞金の期待値を最大化するような戦略を考えるのが Two-armed bandit 問題である．もちろん確率 p_1, p_2 が十分な精度で推定されていれば，確率の大きいアームを引いたほうがよいのはあきらかであるが，精度よく推定するためには多くの試行錯誤を必要とする．この問題では，単に推定精度を高めるのではなく，賞金の総和を多くすることが目的である点が，これまで述べた設定と本質的に異なっている．しかしながら，目的関数を最大にするために戦略を立てながら試行を行っていく点では，能動学習の一種と考えられる．全体の試行を２ステージに分けて，初期には確率の推定を行い後に確率の高いアームによって賞金を稼ぐ戦略など，さまざまな研究が数多くなされている．詳しくは文献 [23] およびその中の文献リストを見てほしい．

Bandit 問題では，システムは時間的に不変で，時間的に独立に確率 p_j に従う値を返すと考える場合が多いが，これを動的なシステムに拡張するとどうなるであろうか？その一つの定式化が強化学習あるいはマルコフ決定プロセスの学習と呼ばれているものである．時刻 t において状態 $S_t \in \mathcal{S}$ を持つシステムがあるとしよう．学習者はこの系に対して動作 $A_t \in \mathcal{A}$ を施す

ことができ，現在の状態 S_t と動作 A_t に基づく報酬 $r_t \in \mathbb{R}$ が得られる．また，システムの状態も S_t, A_t に基づいて次の状態 S_{t+1} に遷移する．一般にこの遷移は確率的であるが，現在の状態および動作のみによって次の状態が確率的に決まるので，マルコフ的であると言われる．状態遷移の確率や報酬を決定するルールは未知であると仮定する．このような問題設定のもと，現在の状態 S_t から動作 A_t を決める戦略 $\pi : \mathcal{S} \to \mathcal{A}$ の中で，期待される報酬の和を最も大きくするものを学習するのが強化学習である．このような問題に対して，TD-learning, Q-learning と呼ばれる有効な学習方法が提案されている．これらの学習では，現在得られている情報から報酬を大きくするような行動を決める一方，確率的なルールを推定するために色々な状態を探索的に現出させることが必要となる．強化学習に関しては，Barto, Sutton, Watkins による解説 ([24]) や Sutton, Barto による教科書 ([25]) を見てほしい．

さて，本章では主として理論的な基礎がきちんとした学習の話題を取り扱ったが，「能動学習」という言葉は，本来もっと広い枠組みで捉えることが可能だと思われる．たとえば，ロボットの行動学習をする際に，情報をよりくわしく取りたい場所を細かく調べる戦略などは，まさに能動的な学習そのものである．また，より詳しく見たい場所に視点を制御するアクティブ・ヴィジョンの技術 ([26]) は，視覚系の情報を用いた学習システムにおける能動的学習の基盤といえるであろう．本章で述べたような能動学習の理論がさらにその枠組みを広げ，ロボットの学習などのより実世界的な学習問題の基礎付けへと発展していくことを期待している．

第 2 章

参考文献

[1] M. Stone. Cross-validatory choice and assessment of statistical predictions. *J. Royal Stat. Soc.*, 36:111–133, 1974.

[2] E.L. Lehmann", *Theory of point estimation*. John Wiley & Sons, 1983.

[3] V.V. Fedorov. *Theory of Optimal Experiments*. Academic Press, New York, 1972.

[4] K. Fukumizu. Statistical active learning in multilayer perceptrons. *IEEE Trans. Neural Networks*, 11(1):17–26, 2000.

[5] D.A. Cohn. Neural network exploration using optimal experiment design. In Jack D. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems*, volume 6, pages 679–686. Morgan Kaufmann, 1994.

[6] W. J. Hill and W. G. Hunter. A review of response surface methodology: A literature survey. *Technometrics*, 8(4):571–590, 1966.

[7] A. I. Khuri, R. H. Myers, and Jr. W. H. Carter. Response surface methodology: 1966-1988. *Technometrics*, 31(2):137–157, 1989.

[8] B. Efron and R. Tibshirani. *An Introduction to the Bootstrap*. Chapman and Hall, New York, 1993.

[9] J. Kindermann, G. Paass, and F. Weber. Query construction for neural networks using the bootstrap. In *Proc. Intern. Conf. Artificial Neural Networks 95*, pages 135–140, 1995.

[10] J. Kiefer and J. Wolfowitz. The equivalence of two extremum problems. *Canadian J. Math.*, 12:363–366, 1960.

[11] G.E.P. Box and N.R. Draper. A basis for the selection of a response surface desing. *J. American Stat. Assoc.*, 54:622–654, 1959.

[12] 福水健次, 渡邊澄夫. 多項式近似における学習データの最適設計と予測誤差. **電子情報通信学会論文誌** *A*, J79-A(5):1100–1108, 1996.

[13] G. Paass and J. Kindermann. Bayesian query construction for neural network models. In G. Tesauro, D. Touretzky, and T. Leen, editors, *Advances in Neural Information Processing Systems*, volume 7, pages 443–450. The MIT Press, 1995.

[14] D.E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. In D. E. Rumelhart, J. L. McClelland, and the PDP Research Group, editors, *Parallel Distributed Processing*, volume 1, pages 318–362. MIT Press, Cambridge, 1986.

[15] H. J. Sussmann. Uniqueness of the weights for minimal feedforward nets with a given input-output map. *Neural Networks*, 5:589–593, 1992.

[16] K. Fukumizu and S. Amari. Local minima and plateaus in hierarchical structures of multilayer perceptrons. *Neural Networks*, 13(3):317–327, 2000.

[17] K. Fukumizu. A regularity condition of the information matrix of a multilayer perceptron network. *Neural Networks*, 9(5):871–879, 1996.

[18] 福水健次，栗木哲，竹内啓，赤平昌文．特異モデルの統計学．岩波書店，2004.

[19] L.M. Belue, Jr.K.W. Bauer, and D.W. Ruck. Selecting optimal experiments for multiple output multilayer perceptrons. *Neural Computation*, 9:161–183, 1997.

[20] 日本色彩学会（編）. **新編色彩科学ハンドブック（第２版）**. 東京大学出版会, 1998.

[21] 中村篤祥（編）特集 能動学習. **情報処理**, 38(7):557–588, 1997.

[22] K. Chaloner and I. Verdinelli. Bayesian experimental design: A review. *Statistical Science*, 10(3):273–304, 1995.

[23] D.A. Berry and B. Fristedt. *Bandit Problems: Sequential Allocation of Experiments*. Chapman and Hall, 1985.

[24] A.G. Barto, R.S. Sutton, and C.J.C.H. Watkins. Learning and sequential decision making. In M. Gabriel and J.W. Moore, editors, *Learning and Computational Neuroscience: Foundations of Adaptive Networks*, pages 539–602. 1990.

[25] R.S. Sutton and A.G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

[26] A. Blake and A. Yuille, editors. *Active Vision*. MIT Press, 1992.

[27] 福島雅夫. **非線形最適化の基礎**. 朝倉書店, 2001.

.1 Carathéodory の定理

定理 3　Δ を $\mathbb{R}^d$ 内の集合とする．$\boldsymbol{z}$ が Δ の点の凸結合であるとき，$\boldsymbol{z}$ はたかだか $d+1$ 個の Δ の点の凸結合として表される．すなわち，$\sum_{i=1}^{d+1} p_i = 1$ を満たす $p_i \geq 0$ と $\boldsymbol{z}_i \in \Delta$ $(1 \leq i \leq d+1)$ が存在して，$\boldsymbol{z} = \sum_{i=1}^{d+1} p_i \boldsymbol{z}_i$ と書ける．

証明は例えば [27] を見よ．

.2 行列式に関する関係式

補題 1 正方行列 A, D が正則であるとき，正方行列 $H = \left(\begin{smallmatrix} A & B \\ C & D \end{smallmatrix}\right)$ に対して，

$$|H| = |A||D - CA^{-1}B| = |D||A - BD^{-1}C|$$

が成り立つ.