■S3 群（脳・知能・人間）－３編（人工知能と学習）

3 章 機械学習

（章主任：山本章博，岩沼宏司）
（執筆者：山本章博，岩沼宏治）［2019 年 1 月 受領］

■概要■

近年，深層学習に代表される神経回路網の機械学習が脚光を浴びているが，機械学習の枠組みや諸原理，その基盤は非常に幅広く多様である．人工知能においては伝統的に知識の記号的表現が重視されてきたため，機械学習の研究では，概念の学習と知識獲得のための学習プログラムの構築が目標とされてきた．しかし，21 世紀に入り，機械学習の位置づけと方向は大きく様変わりした．すなわち，コンピュータの高性能化とネットワークの普及に伴い，機械学習は大規模データから動的にかつ適切に知識獲得する手法であるとの意識が高まり，新たな展開を見せている．大規模データを対象とした機械学習においては，確率的手法が不可欠であり，また記号的手法と確率的手法を融合する研究が活発化している．特に深層学習においては，結果の可読性が重要視されるようになり，記号的手法との融合が期待されている．本章では，人工知能の分野において研究されてきた機械学習について，主に記号的手法の結果を，21 世紀の展開を踏まえて解説を行う．

【本章の構成】

本章では，3-1 節で概説を述べた後，3-2 節では機械学習のフレームワーク（背景）に関して，計算論的学習，統計的学習と情報論的学習について基本的な解説を行う．3-3 節は機械学習のアプローチに関する解説であり，説明による学習，強化学習，能動学習，及び機械学習と知識発見などについて基本的な解説を行う．3-4 節では対象と表現手法に関する解説を行う．識別関数の学習，形式言語の学習，決定木の学習，更に帰納論理プログラミングについて基本的な解説を行う．3-5 節は評価の枠組みに関する解説であり，極限同定，ブースティングと PAC 理論，EXACT 学習及びオンライン学習などについて解説する．

■S3 群－3 編－3 章

3-1 概　説

（執筆者：山本章博）［2019 年 1 月　受領］

学習が人間や動物の知的行為の一部であることは誰もが認めるであろう．元々，機械学習システムとは，学習という行為と機能的に共通点を持つ機械システム（コンピュータプログラム）を指している．最初から学習を模倣することを目指して設計されたシステムの場合と，知的振る舞いを目指して設計されたシステム（の一部）が学習と類似している場合のいずれもに対して用いられる．

本章では，機械学習システムがコンピュータプログラムである場合を対象にする．学習するプログラムを単純かつ一般的に定義したものの一つに Mitchell [1] によるものがある．

> コンピュータプログラムが，タスク T について，経験 E から，パフォーマンスの測度 P のもとで学習するとは，タスク T におけるそのプログラムのパフォーマンスが，それを測度 P によって計測したときに，E を与えることで向上することをいう．

Simon [2] も同様の定義を与えている．また，機械学習の理論的分析においては，プログラムをアルゴリズムに抽象化することがある．この定義に従って，チェッカーの手を学習するプログラムを考えると，そのタスクはチェッカーで対戦することであり，経験とは自分自身との対戦の盤面と手，パフォーマンスとは，対戦に勝つ確率や時間などである．また，かな漢字変換システムの学習機能におけるタスクとは，かな文字列を漢字かな混じりの文字列に変換すること，経験とはそれまでの変換履歴であり，そしてパフォーマンスとしては意図した変換が得られるまでに出現する候補の数などが考えられる．

タスク，経験，パフォーマンスを具体的に数理化すると，Gold [3] や Rivest らが定義したような機械学習の理論的枠組みが得られる．つまり，タスクを規則性の獲得と獲得した規則性による行動に分解すれば，規則性の獲得が学習の主要部となる．規則性の獲得のためには，対象となる規則性の定義と規則性の表現方法を定義しておかなければならない．更に．機械学習プログラムが得る経験も，背景知識と観測に分解することもある．そして，パフォーマンスとは，学習による結果の向上の度合いに加えて，結果の正当性と計算量を勘案することになる．これらの項目を具体的に設定すれば，考察の対象となる学習モデルが得られる．例えば，記号列データからの機械学習を考察する際に，規則性として正則言語を学習の対象とすれば，有限オートマトンが個々の正則言語を表現する．経験として，ある文字列が目標の正則言語に入っているか否かを表すデータの系列，学習プログラムのパフォーマンスの一つである結果の正当性の保証としては様々な学習可能性基準（極限同定 [3], PAC 学習可能性 [4] など）を選ぶことになる．

コンピュータ開発が始まった当時は, Rosenblatt [5] のパーセプトロンを初めとする素子レベルで機械学習を実現するシステムが研究された．これらの研究は，パターン認識における識別関数の構成や意思決定理論へと展開される一方，制御理論での研究も行われた．一方で，Gold [3] は計算可能性理論を用いた機械学習の数学的研究を開始した．計算可能性という概念が「計算」という行為のモデル化によって確立されたのと同様に，学習可能性を「学習」という行為のモデル化によって構成することを目指すものである．

学習を計算と比較すると次のような明確な違いがある．すなわち，計算とは入力に対して出

力を求める行為であるのに対して，学習は入力や出力以外にも経験を考慮する．経験はシステムの外にある（人間などの）別のシステムからデータとして与えられるものであれば，その別システムの持つ意図や知識を経験という形で受け取っていることになる．学習システムを受信者，経験を与える別システムを送信者，経験のためのデータを信号と解釈すれば，学習は情報伝達という側面を持っており，情報理論の成果を利用することが可能となる．

人工知能において知識の記号的表現が重視されるにつれ，すなわち経験的知識を記号によって記述する方式が隆盛となるなかで，概念の学習と知識獲得のための機械学習プログラムを構築することが目指され，多くの機械学習システムが，関数型プログラミング言語や論理プログラミング言語を用いて提案された．

21 世紀に入り，機械学習の位置づけと方向は大きく様変わりした．すなわち，コンピュータの高性能化により，サポートベクトルマシンのように従来は実行が難しかった技術が利用可能になり．更に GPU の利用など新たな計算手法の開発により深層学習のように神経回路網を複雑化した学習モデルが実用可能な水準に到達した．機械学習に用いるデータについても，ネットワークとセンシング技術の発達と普及に伴い，かつてないほど大規模なものが利用可能となった．知識の記号表現を基礎とした人工知能においては，その知識の獲得方法と適切さに関する尺度を持つことができない状態であったが，機械学習は大規模データから動的にかつ適切に知識獲得する手法であるとの意識から，新たな展開の段階に入っている．大規模データを対象とした機械学習においては，確率的手法が不可欠であり，ベイズ推論や確率文脈自由文法などの確率的手法が注目されるようになり，記号的手法と確率的手法を融合する研究が活発化している（例えば参考文献 6)）．神経回路網による学習でも，学習によって得られる結果を人間にとって理解可能な知識に還元するための手法の研究が開始されている．

計算機科学以外の諸科学やコンピュータの実用においても，データ入力機器の発達に伴い，データの収集と蓄積が容易になり，データの大規模化が進んだ．そこで，従来は統計学が担っていたデータの分析と予測の手段としての役割，すなわちデータマイニングや知識発見の一端を機械学習が担うことになり，生物学，特に生命科学や化学，あるいは文科系に分類される諸科学においても機械学習の応用が行なわれるようになった．このような機械学習の役割は，上述の定義からは外れているように見えるが，表に示すように，例示からの学習の枠組を設定するために必要な項目群は，統計的推論を構成するために必要な項目群と類似している．

例示からの機械学習	統計的推論
対　象	確率分布
対象の表現	確率分布を決定するパラメータ
例　示	サンプリング
学習アルゴリズム	パラメータ推定アルゴリズム
学習の正当性	推定の基準（一致性，普遍性など）

機械学習の理論面では，計算可能性理論，計算量理論，形式言語理論，数理論理学，統計学，情報理論をはじめとする計算機科学の諸分野の他，数学や物理学の理論を用いた理論が整備されている [7)～10)]．

本章は，機械学習に関する項目を，いくつかの視点から分類することで構成されている．

背景：計算論的学習，統計的学習，情報理論的学習
アプローチ：説明による学習，強化学習，能動学習，機械学習と知識発見
対象：関数の学習，形式言語の学習，決定木，帰納論理プログラミング
枠組み：極限同定，PAC 学習，EXACT 学習，オンライン学習，Query by Committee

統計的学習，情報理論的学習の詳細については，他章も参照いただきたい．本章で扱うような機械学習研究の最新動向を知る方法としては，以下にあげる国際会議の予稿集をご覧いただくことが最善であろう．

International Conference on Machine Learning（ML）
International Conference on Computational Learning Theory（COLT）
International Conference on Algorithmic Learning Theory（ALT）
European Conference on Machine Learning（ECML）
International Conference on Inductive Logic Programming（ILP）
International Colloquium on Grammatical Inference（ICGI）

■参考文献

1) T.M. Mitchell：“Machine Learning,” McGraw-Hill, 1997.
2) H.A. Simon：“Why Should Machines Learn?,” in R.S. Michalski, J.G. Carbonell, and T.M. Mitchell (eds.)：“Machine Learning: An Artificial Intelligence Approach,” Morgan Kaufmann, 1983.
3) M.E. Gold：“Language Identfication in the Limit,” Information and Control, vol.10, pp.447-474, 1967.
4) L.G. Valiant：“A Theory of the Learnable,” Commun. ACM, vol.27, no.11, pp.1134-1142, 1984.
5) F. Rosenblatt：“The Perceptron: A Probabilistic Model for Information Storage and Organization on the Brain,” Psychological Review, vol.65, pp.386-407, 1958.
6) S. Russel and P. Novig：“Artificial Intelligence: A modern approach (second edition),” Prentice Hall, 2003.
7) S. Jain, D. Osherson, J.S. Royer, and A. Sharma：“Systems That Learn—An introduction to Learning Theory (second edition),” MIT Press, 1999.
8) 榊原康文，小林　聡，横森　貴：“計算論的学習,” 培風館, 2001.
9) C.M. Bishop：“Pattern Recognition and Machine Learning,” Springer-Verlag, 2006.
10) 渡辺澄夫，萩原克幸，赤穂昭太郎，本村陽一，福水健次，岡田真人，青柳美輝：“学習システムの理論と実現,” 森北出版, 2005.

■S3 群－3 編－3 章

3-2 フレームワーク

3-2-1 計算論的学習

3-2-2 統計的学習

（執筆者：杉山　将）［2016 年 1 月 受領］

統計的学習（Statistical Learning）とは，データの統計的な性質を利用する機械学習の枠組みである．ここでは特に，入力 x と出力 y の組から成る n 個の訓練データ $D = \{x_i, y_i\}_{i=1}^{n}$ を用いて，その背後に潜んでいる入出力関係を学習する教師付き学習（Supervised Learning）と呼ばれる問題を考える．通常，訓練データは同時確率分布 $P(x,y)$ に独立同一分布（Independent And Identically Distributed : i.i.d.）に従って生成されていると仮定する．教師付き学習のほかには，出力のない入力だけが訓練データとして与えられる場合の学習問題を議論する教師なし学習（Unsupervised Learning），教師付きと教師なしの間に位置する学習問題を議論する半教師付き学習（Semi-supervised Learning），マルコフ決定過程（Markov Decision Process）と呼ばれる環境での学習問題を議論する強化学習（Reinforcement Learning）などがある．

教師付き学習において入出力関係をうまく学習することができれば，学習していない入力 x に対する出力 y を予測できるようになる．すなわち，学習機械は未知の状況に適応できる汎化能力（Generalization Ability）を持つ．できるだけ少ない訓練データから最高の汎化能力を獲得することが教師付き学習の目標である．y が実数値をとる場合は回帰（Regression）と呼ばれ，y が離散値をとる場合は分類（Classification）と呼ばれる．また，y の相対的な大小関係を推定する順序回帰（Ordinal Regression）という問題もある．

入出力関係の学習には，パラメータ（Parameter）θ を含む入出力モデル（Model）$y = f(x; \theta)$ を用いる．これは，入力 x を「識別」するために出力 y の事後分布 $P(y \mid x)$ をモデル化・学習していることに相当するため，識別モデル（Discriminative Model）学習法と呼ばれる．また，パラメータが有限次元のときはパラメトリック法（Parametric Method）と呼ばれ，パラメトリックモデルを用いない場合や無限次元のパラメータを含むモデルを用いる場合はノンパラメトリック法（Non-parametric Method）と呼ばれる．教師付き学習においては，パラメータを最適に決定するための学習法（Learning Method）の研究，モデルを最適に決定するためのモデル選択（Model Selection）の研究，訓練データの入力 $\{x_i\}_{i=1}^{n}$ を最適に決定するために能動学習（Active Learning）の研究，多次元の入力 x のうち出力 y を予測するのに役立つ部分集合/低次元表現を見つける特徴選択/特徴抽出（Feature Selection/Feature Extraction）などの研究が盛んに行われている．

学習法の研究には，大きく分けて 2 つの流派がある．一つは頻度主義に基づくアプローチ（Frequentist Approach）[2)～4)] であり，もう一つはベイズ主義に基づくアプローチ（Bayesian Approach）[5),6)] である．機械学習における頻度主義とベイズ主義は，確率論における古典統計学とベイズ統計学との関係に対応している．頻度主義的アプローチでは，適当な損失関数（Loss Function）のもとでモデルのパラメータを訓練データに適合させる経験リスク最小化原理（Empirical Risk Minimization Principle）に基づいて学習が行われる．

$$\min_{\theta} \sum_{i=1}^{n} \ell(x_i, y_i, f(x_i; \theta))$$

ここで $\ell(x, y, y')$ は，入力 x における出力 y を y' と予測したときの損失関数である．回帰問題においては，二乗損失（Squared Loss），フーバー損失（Huber Loss），絶対値損失（Absolute Loss）などがよく用いらる．これらの損失は凸関数（Convex Function）であるため，大域的最適解（Global Optimal Solution）を容易に求めることができる．また，フーバー損失や絶対値損失は外れ値（Outlier）に対してロバスト（Robust）であるため，実用的価値が高い．一方，分類問題では誤識別率を与える 0/1 損失（Zero-one Loss）を使うのが自然である．しかし，0/1 損失は非凸関数（Non-convex Function）であるため，その凸近似（Convex Approximation）であるヒンジ損失（Hinge Loss），ロジスティック損失（Logistic Loss），指数損失（Exponential Loss）などがよく用いられる．

経験リスク最小化原理は，統計学における最尤推定法（Maximum Likelihood Estimation）に対応している．したがって，訓練データ数が少ない場合には，経験リスクの最小化によって学習結果がノイズの重畳した訓練データに過適合（Overfit）してしまうことがある．過適合を避けるために，正則化（Regularization）がよく用いられる．

$$\min_{\theta} \sum_{i=1}^{n} \ell(x_i, y_i, f(x_i; \theta)) + \lambda R(\theta)$$

ここで，$\lambda\,(>0)$ は正則化パラメータ（Regularization Parameter）と呼ばれ，正則化の強さをコントロールする．$R(\theta)$ は正則化汎関数（Regularization Functional）と呼ばれ，ℓ_2-ノルムや ℓ_1-ノルムがよく用いられる．ℓ_1-ノルムを正則化汎関数として用いれば，解が疎（Sparse）になることが知られている．

一方，ベイズアプローチでは，パラメータの事前分布（Prior Distribution）$P(\theta)$ を決めたもとで，データの尤度（Likelihood）$P(D\,|\,\theta)$ に基づいてパラメータの事後分布（Posterior Distribution）$P(\theta\,|\,D)$ を計算する．

$$P(\theta|D) = \frac{P(D|\theta)P(\theta)}{\int P(D|\theta)P(\theta)d\theta}$$

この式には未知の量が含まれないことから，ベイズアプローチは「学習」というよりも，むしろ「計算」であることが分かる．実際，ベイズ学習研究の主眼はいかに効率良く事後分布を計算するかにある．事後分布は事前分布の選び方に依存するため，ベイズアプローチでは学習結果を主観的にコントロールできる．これにより，訓練データ数が少ない場合でも良い学習結果が得られることがある．しかし，学習結果に主観を持ち込むことに対して否定的な意見もある．一方，最大事後確率推定法（Maximum a Posteriori Estimation；MAP 推定法）と呼ばれる，事後分布を最頻値で近似するベイズ学習法は，頻度主義における最尤推定法を正則化したものと本質的に等価である．このことから，ベイズ主義の主観性に対する批判は，工学的にはそれほど重要ではないと考えられる．実用上は，計算に都合の良い共役事前分布（Conjugate Prior）を選ぶことが多い．

事後分布 $P(y\,|\,x)$ のモデル化・推定に相当する上記の識別モデル学習法に対して，生成モデル（Generative Model）学習法と呼ばれるアプローチもある．これは，事後分布 $P(y|x)$ がデータを「生成」している分布 $P(y|x)$ に比例することを用いて，$P(y|x)$ をモデル化・推定するアプローチである．生成モデル学習の枠組みでも，パラメトリック法・ノンパラメトリック法，

及び，頻度主義的学習法，ベイズ学習法が用いられる．

ところで，$P(x, y)$ が分かれば次式により $P(y|x)$ を求めることができる．

$$P(y|x) = \frac{P(x,y)}{\int P(x,y)dy}$$

しかし，逆に $P(y|x)$が分かったとしても一般に $P(x, y)$ を求めることはできない．したがって，$P(x, y)$ を推定する問題の方が $P(y|x)$ を推定する問題よりも難しいと考えられる．この考え方に基づいて，生成モデルを学習するアプローチよりも識別モデルを学習するアプローチの方がふさわしいと主張する学派がある [4]．一方，生成モデル学習の枠組みでは，データの生成分布に関する先見的知識を有効に活用することができるという利点がある．また，2 つのアプローチが一致する場合もあり [1]，生成モデル学習と識別モデル学習どちらが良いかは状況に依存する．

■参考文献

1) R.O. Duda, P.E. Hart, and D.G. Stor：“Pattern Classification,” Wiley, New York, 2001.
2) T. Hastie, R. Tibshirani, and J. Friedman：“The Elements of Statistical Learning: Data Mining, Inference, and Prediction,” Springer, New York, 2001.
3) B. Schälkopf and A.J. Smola：“Learning with Kernels,” MIT Press, Cambridge, MA, 2002.
4) V.N. Vapnik：“Statistical Learning Theory,” Wiley, New York, 1998.
5) 元田　浩，栗田多喜夫，樋口知之，松本裕治，村田　昇(編)：“パターン認識と機械学習（上）：ベイズ理論による統計的予測,” シュプリンガー・ジャパン，東京, 2007.
6) 元田　浩，栗田多喜夫，樋口知之，松本裕治，村田　昇(編)：“パターン認識と機械学習（下）：ベイズ理論による統計的予測,” シュプリンガー・ジャパン，東京, 2008.

3-2-3 情報論的学習

（執筆者：山西健司）［2016 年 7 月 受領］

「学習とは，データを最も圧縮できる表現を抽出することである．」 情報論的学習理論とは，機械学習をこのようなデータ圧縮の立場から捉える理論である [8),9)]．つまり，最も短い符号長でデータを圧縮できるモデルこそが，データに内在する知識の最も適した表現であるという原理に立脚している．このような原理を記述長最小（Minimum Description Length：MDL）原理 [2)～4)] と呼ぶ.

(1)　情報・符号・確率

X を定義域として，$\{0,1\}^*$ を 2 元系列の集合とする．長さ n のデータ列 $x^n = x_1, \cdots, x_n \in X$ が与えられたもとで，これを 2 元符号語に変換する写像 $\phi: X^n \to \{0,1\}^*$ を符号化と呼ぶ．特に，$x^n \neq y^n$ に対して，$\phi(x^n)$ がどの $\phi(y^n)$ とも語頭が一致しない性質を持つ符号化のことを語頭符号化と呼ぶ．L が語頭符号長関数であることの必要十分条件は，クラフトの不等式

$$\sum_{x^n \in X^n} 2^{-L(x^n)} \leq 1$$

を満たすことである [1]．x^n の発生確率分布 $p\,(X^n)$ が既知の場合には，$L(x^n) = -\log p(x^n)$とすることにより，$L(x^n)$はクラフトの不等式を満たすから，そのような語頭符号化が存在する．また，そのような語頭符号化は最短の平均符号長を達成するという意味で最適である [1]．

(2)　一括学習とモデル選択原理

データの発生確率分布が未知であるが，あるパラメトリックなクラス $\mathcal{P}_k = \{P(x^n;\theta) : \theta \in \mathbf{R}^k\}$

に属していると仮定する．これを用いてデータ列 x^n をできるだけ短い符号長で語頭符号化したい．このときの符号長関数の良し悪しの規準として，以下のようなミニマックスリグレットを採用する．

$$\min_L \max_{x^n} \left\{ L(x^n) - \min_\theta(-\log p(x^n;\theta)) \right\} \tag{2・3・1}$$

式(2・3・1)の最小を達成する最適な語頭符号長関数は，以下の正規化最尤符号長により達成される[4]．

$$SC(x^n;\mathcal{P}_k) = -\log p(x^n;\hat{\theta}(x^n)) + \log \sum_{y^n} p(y^n;\hat{\theta}(y^n)) \tag{2・3・2}$$

ここに，$\hat{\theta}$ は θ の x^n からの最尤推定量，つまり，$\hat{\theta} = \arg\max_\theta p(x^n;\theta)$ である．符号長(2・3・2)のことを x^n の $\mathcal{P}_k$ に対する確率的コンプレキシティ（Stochastic Complexity）と呼び，第二項を $\mathcal{P}_k$ のパラメトリックコンプレキシティと呼ぶ．パラメトリックコンプレキシティについては，最尤推定量について中心極限定理が成立する正則条件のもとで以下の近似が成立する[4]．

$$\log \sum_{y^n} p(y^n;\hat{\theta}(y^n)) = \frac{k}{2}\log\frac{n}{2\pi} + \log\int\sqrt{|I(\theta)|}d\theta + o(1) \tag{2・3・3}$$

ここに，$I(\theta)$ は (i,j) 成分を $I_{ij}(\theta) = \lim_{n\to\infty}(1/n)E_\theta[-\partial^2 \log p(x^n;\theta)/\partial\theta_i\partial\theta_j]$ とする Fisher 情報行列であり，$o(1)$ は x^n に関して一様に $\lim_{n\to\infty} o(1) = 0$ となる量である．与えられた x^n に対して，最も適切なモデルの複雑さ（パラメータの次元 k など）を決定するための戦略として，確率的コンプレキシティ(2・3・2)を規準としてこれ最小化する k を選ぶ戦略を考えることができる．この規準を MDL（Minimum Description Length）規準と呼ぶ[2)~4)]．これは一括型の知識表現の学習の基本アルゴリズムを与える．MDL 規準で求められたモデルについては，一致性があること（漸近的に真のモデルに収束すること）[4]，やその収束速度が確率的近似学習の意味において迅速であることが知られている[5]．また，任意の語頭符号長関数 L に対して，測度ゼロの θ の集合を除いて，中心極限定理の成立する条件の下で，任意の $\epsilon > 0$ に対して，以下の不等式が成り立つ[3]．

$$E_\theta[L(x^n)] \geq E_\theta[-\log p(x^n;\theta)] + \frac{k-\epsilon}{2}\log n \tag{2・3・4}$$

ここで，E_θ は $p(x^n;\theta)$ に関してとられる期待値を表す．

(3)　逐次学習と逐次符号化

データが $x_1, x_2, \cdots, x_t, \cdots$ と逐次的に与えられるときに，各時刻 t でデータが発生する確率分布を $x^{t-1} = x_1, \cdots, x_{t-1}$ に基づいて予測する問題を考える．これを逐次的確率予測の問題と呼ぶ．逐次的確率予測アルゴリズムが各時刻において出力する予測分布を $p(X|x^{t-1})$ と書き，真の結果 x_t が与えられたときに，その予測損失を対数損失 $-\log p(x_t|x^{t-1})$ で測る．x^n まで観測されたとき，累積予測損失 $\sum_{t=1}^n(-\log p(x_t|x^{t-1}))$ が小さければ小さいほどアルゴリズムは良いとする．累積予測損失は逐次的に符号化を行った時の総符号長に相当するので，逐次的確率予測の問題は，逐次的なデータ圧縮の問題に帰着できる．確率モデルのクラス $\mathcal{P}_k = \{p(X;\theta)\}$ を用いる逐次的予測アルゴリズムとしては，例えば，最尤予測，ベイズ予測，逐次的正規化最尤予測などがある．それぞれの予測分布は以下で与えられる．

$$p(X|x^{t-1}) = p(X;\hat{\theta}(x^{t-1})) \quad \text{（最尤予測分布）}$$

$$p(X|x^{t-1}) \quad = \quad \int p(X;\theta)p(\theta|x^{t-1})d\theta \quad \text{（ベイズ予測分布）}$$

$$p(X|x^{t-1}) \quad = \quad p(X;\hat{\theta}(X\cdot x^{t-1}))/\sum_{X'} p(X';\hat{\theta}(X'\cdot x^{t-1})) \quad \text{（逐次的正規化最尤予測）}.$$

ここで，$\hat{\theta}(x^{t-1})$ は x^{t-1} からの θ の最尤推定量，$p(\theta|x^{t-1})$ は θ の事後分布とする．いずれの予測についても，累積予測損失としての総符号長は式(2･3･4)の下限を漸近的に達成する [8),9)]．この意味でこれらの逐次的確率予測のアルゴリズムは最適性を持っている．

特に，最尤予測を用いたときの累積予測損失

$$PSC(x^n;\mathcal{P}_k) = \sum_{t=1}^{n}(-\log p(x_t;\hat{\theta}(x^{t-1})) \qquad (2\cdot3\cdot5)$$

を予測的確率的コンプレキシティと呼ぶ．式(2･3･5)を最小化する k を最適なモデルとして選ぶモデル選択原理を予測的 MDL 原理と呼ぶ．

(4)　拡張型確率的コンプレキシティ

確率的コンプレキシティの概念を一般の関数クラス及び一般の損失関数を用いる統計的決定理論の枠組みに拡張したものが，拡張的コンプレキシティである．これは $\mathcal{F}=\{f(x)\}$ を関数のクラス，$\mu(f)$ を $\mathcal{F}$ 上の測度，$L(y,f(x))$ をデータ x から y を $f(x)$ で予測したときの損失関数，$\lambda>0$ として，データ列 $D_n=D_1$, …, D_n $D_i=(x_i,y_i)$ に対して以下で与えられる量である [6)]．

$$ESC(D^n;\mathcal{F}) = -\frac{1}{\lambda}\log\exp\left(-\lambda\sum_{t=1}^{n}L(y_t,f(x_t))\right)\mu(df) \qquad (2\cdot3\cdot6)$$

確率的コンプレキシティから確率モデルのモデル選択規準や逐次確率予測アルゴリズムが導出されたように，式(2･3･6)を基にして，一般の関数の一括学習，及び一般の損失関数に関する逐次予測アルゴリズムを導出できる．その際，λ の与え方が損失関数の選び方に依存する．そうして得られたアルゴリズムはある損失関数のクラスについては，一括学習における迅速な収束性や逐次予測における最適性が示されている [6),8)]．

(5)　動的モデル選択

通常の一括学習におけるモデル選択は，データに定常性を仮定している．一方で，モデルは時刻ともに変化しうるといった非定常の場合を考えるのが現実的である．そのような場合に対応して，データ列から最良なモデル列を選択することを問題にする．これを動的モデル選択の問題と呼ぶ．これに対しても MDL 原理の立場から，モデル列とデータ列の双方を含めた記述長を最小にするようなモデル列を求める戦略が得られる．このときのモデル列選択規準を動的モデル選択規準と呼ぶ．データ列 $x_n=x_1$, …, x_n に対して，各時刻 t でモデル M_t を用いた予測分布（最尤予測，ベイズ予測，逐次的正規化最尤予測のいずれか）に対する x_t の語頭符号長を $-\log p(x_t|x^{t-1},M_t)$ とする．また，M_t 自体も確率的に遷移するとして，その予測分布を $\hat{p}(M_t|M^{t-1})$ とする．このとき，動的モデル選択規準は以下のように与えられる．

$$DSC(x^n;M_1,\ldots,M_n) = \sum_{t=1}^{n}(-\log p(x_t|x^{t-1};M_t)) + \sum_{t=1}^{n}(-\log\hat{p}(M_t|M^{t-1})) \qquad (2\cdot3\cdot7)$$

式(2･3･7)の規準を最小化するモデル列 M_1, …, M_t が最良なモデル列として与えられる．これは動的計画法を用いて計算することができる．その際のモデルの変化点を調べることにより，モデル変化検知が可能となる．モデル変化検知には一括型の手法と逐次型の手法が存在する．これらはデータマイニングにおける異常検知や潜在的構造変化検知に広く展開されている [9)]．

(6)　完全変数化 MDL 原理

機械学習では，潜在変数 Z を伴う確率モデル $p\,(X, Z;\,\theta, M)$を扱う場合が多い．θ は実数値パラメータ，M はクラスの複雑さを示すモデルである．これを以下，潜在変数モデルと呼ぶ．顕在変数 X は周辺分布 $p\,(X;\,\theta, M) = \Sigma_Z\, p\,(X, Z;\,\theta, M)$ に従って生成されているとする．与えられたデータ列 x^n からモデル M を推定するのに，直接 $p\,(X;\,\theta, M)$ に対してモデル選択を適用することは一般に困難である．何故なら，潜在変数を伴うモデルでは，パラメータと確率分布の対応関係が必ずしも 1 対 1 に対応しないことがあり，その際には最尤推定の中心極限定理が成り立たないので，式(2･3･3)のような漸近的近似式を用いてモデル選択を行うことはもはやできないからである．そこで，潜在変数も含めてモデル選択の対象としてモデル選択を行う方法が存在する．これが完全変数化 MDL 原理である．これは，z_i を x_i に対応する潜在変数の値として，$z^n = z_1, \cdots, z^n$ と書くとき，完全変数化 MDL 規準を

$$LSC(x^n, z^n; M) = -\log p(x^n, z^n; \hat{\theta}(x^n, z^n), M) + log \sum_{y^n} \sum_{w^n} p(y^n, w^n; \hat{\theta}(y^n, w^n) : M) \tag{2･3･8}$$

として定め，式(2･3･8)を最小化するような M のみならず z^n を求める戦略である[9]．これにより，潜在変数モデルに対しても MDL 原理に基づいてモデル選択が可能となる．この方法は，混合モデル，関係データモデル，トピックモデル，非負値行列因子分解モデルなどのような潜在変数モデルに対して適用されている[9]．

■参考文献

1)　T.M. Cover and J.A. Thomas：“Elements of Information Theory,” John Wiley & Sons, 2012.
2)　P. Grúnwald：“The Minimum Description Length Principle,” The MIT Press, 2007.
3)　J. Rissanen：“Stochastic Complexity in Statistical Inquiries,” World Scientific, Singapore, 1989.
4)　J. Rissanen：“Optimal Estimation of Parameters,” Cambridge University Press, Cambridge, 2012.
5)　K. Yamanishi：“A learning criterion for stochastic rules,” Machine Learning, vol.9, Issue 2, pp.165-203, Kluwer Academic Publishers, July, 1992.
6)　K. Yamanishi：“A decision theoretic extension of stochastic complexity with its applications to learning,” IEEE Trans. on Information Theory, vol.44, Issue 4, pp.1424-1439, Jul. 1998.
7)　K. Yamanishi and Y. Maruyama：“Dynamic model selection with its applications to novelty detection,” IEEE Trans. on Inform. Theory, vol.5, no.6, pp.2180-2189, 2007.
8)　山西健司：“情報論的学習理論,” 共立出版, 2010.
9)　山西健司：“情報論的学習とデータマイニング,” 数理工学シリーズ 3, 朝倉書店, 2010.

■S3 群－3 編－3 章

3-3 アプローチ

3-3-1 説明による学習

3-3-2 強化学習

（執筆者：木村　元）［2016 年 5 月 受領］

強化学習（Reinforcement Learning）とは，試行錯誤を通じて環境に適応する学習制御の枠組である．教師付き学習（Supervised Learning）と異なり，状態入力に対する正しい行動出力を明示的に示す教師が存在しない代わりに，一連の行動に対して結果としての良し悪しを評価する「報酬（Reward）」というスカラーの評価値が与えられ，これを手がかりに学習を行うが，報酬や状態遷移には不確実性や時間的遅れがある．そのため，一般に行動を実行した直後の報酬をみるだけでは，学習主体はその行動が正しかったかどうかを判断できないという困難を伴う．

(1)　強化学習の基礎理論

強化学習では，環境のダイナミクスをマルコフ決定過程（Markov Decision Process：MDP）によってモデル化し，アルゴリズムを解析するのが一般的である[4),6)]．

(a) マルコフ決定過程（MDP）モデル

環境のとりうる有限な状態集合を $\boldsymbol{S}=\{s_1, s_2, \cdots, s_n\}$，エージェントがとる有限な行動集合を $\boldsymbol{A}=\{a_1, a_2, \cdots, a_l\}$と表す．環境中の状態 $s \in \boldsymbol{S}$において，エージェントが行動 a を実行すると，環境は確率的に状態 $s' \in \boldsymbol{S}$ へ遷移する．その遷移確率を条件付確率 $P(s' \mid s, a)$ により表す．このとき環境からエージェントへ報酬 r が確率的に与えられるが，その期待値を条件付期待値 $R(s, a, s')$ により表す．エージェントにおける状態集合から行動集合への写像関数を政策と呼び πと表す．

ある時間ステップで実行した行動が，その後の報酬獲得にどの程度貢献したのかを評価するため，その後得られる報酬の時系列を考える．報酬の時系列評価は利得（Return）と呼ばれる．エージェントの学習目標は，利得を最大化すること，あるいはその政策を求めることである．強化学習では，以下に示す割引報酬合計による評価を利得とする場合が多い．ある時刻 t の状態，あるいはそのとき実行した行動の利得 V_tを以下で定義する．

$$V_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \cdots = \sum_{k=0}^{\infty} \gamma^k r_{t+k}$$

ただし，r_tは時刻 t の報酬，γ は割引率（$0 \leq \gamma < 1$）である．この V_tの期待値は，1 ステップ a 当たり $(1-\gamma)$ の確率で停止するエージェントが得る報酬合計の期待値と等価である．未来の報酬を割引く理由としては，計算上都合が良いだけでなく，実環境では環境が変化したり故障などで停止する可能性があるため，時系列上のすべての報酬を同じ重みで考慮するのは妥当ではないことなどがある．MDP においてエージェントが定常政策 π をとるとき，利得の期待値は時間に関係なく状態 s だけに依存する，すなわち利得の期待値は状態 s の関数になり，これを状態価値（State Value）関数と呼び $V^{\pi}(s)$と表す．

(b) 最適な状態-行動価値関数

すべての状態 s において $V^{\pi}(s) \geq V^{\pi'}(s)$ のとき，政策 π は π' より優れるという．MDP で

は，ほかのどんな政策よりも優れる，あるいは同等な政策が少なくとも 1 つ存在する．これを最適政策 π^*という．最適政策は複数存在しうるが，すべての最適政策は唯一の状態価値関数を共有する．これは最適な状態価値関数 V^* と呼ばれ，以下に定義される．

$$V^*(s) = \max_{\pi} V^{\pi}(s) \qquad \forall s \in \mathcal{S}$$

また，すべての最適政策は，以下に示す唯一の行動価値（Action Value）関数を共有する．

$$Q^*(s,a) = \max_{\pi} Q^{\pi}(s,a)$$

$Q^*(s,a)$は最適な Q 関数と呼ばれ，状態 s で行動 a を選択後，ずっと最適政策をとり続けるときの利得の期待値を表す．$Q^*(s,a)$が与えられた場合，状態 s における最大の Q 値は$V^*(s)$に等しく，また，この Q 値を持つ行動 a が最適な行動である．この$Q^*(s,a)$は以下に示す Bellman の最適方程式を満たす．

$$Q^*(s,a) = \sum_{s'} P\,(s'|s,a)\left(R(s,a,s') + \gamma \max_{a'} Q^*(s',a')\right) \qquad \forall s \in \mathcal{S},\ \ a \in \mathcal{A}$$

この非線形連立方程式を解くことで最適な Q 関数及び最適政策を得る．一般に解法としてダイナミックプログラミング（DP）が用いられる．

割引率 γ を 1 へ近づけると，ある割引率 γ^* 以上の割引率における最適政策はすべて同じになり，このときの最適政策は平均報酬に関しても最適政策となっている．

(2)　強化学習アルゴリズム

MDP 環境下での強化学習では，状態遷移確率 $P(s'|s,a)$ や報酬 $R(s,a,s')$ についての知識をあらかじめ持たないため，環境とのやり取りを通じてこれらの情報を得る．前節での説明のとおり，$Q^*(s,a)$ が得られれば，最適な政策を得るのは簡単である．Q-learning [1)] は $Q^*(s,a)$ の推定値 $\hat{Q}^*(s,a)$ を得るための代表的な強化学習アルゴリズムであり，確率的 DP とも呼ばれる．**図 3･1** にその概要を示す．

1. エージェントは環境の状態 s_t を観測する．
2. エージェントは任意の行動選択方法（探査戦略）に従って行動 a_t を実行する．
3. 環境から報酬 r_t を受け取る．
4. 状態遷移後の状態 s_{t+1} を観測する．
5. 状態 s，行動 a に対する特徴ベクトル $x = (x_1, x_2, \cdots, x_n)$を何らかの方法で生成する．ただし，区別が必要な状態−行動ペア間では互いに線形独立に設定し，ノルムを 1 にしておく．
6. 特徴ベクトルの要素数と同数の重み変数 w_i を用いて Q 値を計算．

   $$\hat{Q}(s,a) = \sum_{i=1}^{n} w_i x_i$$

7. 以下の更新式により重み変数 w_i を更新する．ただし，α は学習率，γ は割引率．

   $$\delta_t = r_t + \gamma \max_{a} \hat{Q}(s_{t+1},a) - \hat{Q}(s_t,a_t), \qquad w_i \leftarrow w_i + \alpha x_i \delta_t$$

8. 時間ステップ t を $t+1$ へ進めて手順 1 へ戻る．

図 3・1　Q-learning アルゴリズム

（a）Q-learning の収束定理

エージェントの行動選択においてすべての行動を十分な回数選択し，かつ学習率αが$\sum_{t=0}^{\infty} \alpha(t) \to \infty$ かつ $\sum_{t=0}^{\infty} \alpha(t)^2 < \infty$ を満たす時間 t の関数（例えば $\alpha(t) \propto 1/n$ など）となっているとき，Q-learning アルゴリズムで得る Q 値は最適な Q 値に概収束する[1]．ただし，環境はエルゴート性を有する離散有限 MDP であることを仮定する．その他，解析については文献 2）を参照して欲しい．

（b）連続・多次元空間への対応

図 3･1 の特徴ベクトルを生成するメカニズムを工夫することで連続・多次元空間への対応が可能となる．これは汎化（Generalization）と呼ばれ，様々な基底関数を用いる方法が提案されている[3),7)]．

（c）行動選択方法（探査戦略）

上記の収束定理は，すべての行動を十分な回数選択しさえすれば行動選択方法（探査戦略）には依存せずに成り立つ．よって，行動選択はランダムでもよい．しかし，強化学習ではまだ Q 値が収束していない学習の途中においてもなるべく多くの報酬を得るような行動選択を求められる．序々に挙動を改善していく行動選択方法として，

- ϵ-greedy 選択：確率 ϵ でランダム，それ以外は最大 Q 値を持つ行動を選択．
- ボルツマン選択：$\exp\left(\hat{Q}(s,a)/T\right)$ に比例した割合で行動選択．ただし，T は時間とともにゼロに近付ける．
- 楽天的初期値（Optimistic Initial Value：OIV）：Q 関数の初期値を大きめの値に設定しておき，実行していない状態-行動を選択しやすくする．

などの方法が提案されている[3)]．

また，多次元の行動空間において効率良くボルツマン選択を行う方法として，ギブス（Gibbs）サンプリングを行う方法[7)]が提案されている．

■参考文献

1) C.J.C.H. Watkins and P. Dayan：“Technical Note: Q-Learning,” Machine Learning, vol.8, pp.279-292, 1992.
2) D.P. Bertsekas and J.N. Tsitsiklis：“Neuro-Dynamic Programming,” Athena Scientific, 1996.
3) R.S. Sutton and A. Barto：“Reinforcement Learning: An Introduction,” A Bradford Book, The MIT Press, 1998.
4) 木村　元，宮崎和光，小林重信：“強化学習システムの設計指針,” 計測と制御, vol.38, no.10, pp.618-623, 1999.
5) 木村　元，小林重信：“ロボットの強化学習における状態--行動空間の汎化,” 日本ロボット学会誌, vol.22, no.2, pp.161-164, 2004.
6) 吉本潤一郎，銅谷賢治，石井　信：“強化学習の基礎理論と応用,” 計測と制御, vol.44, no.5, pp.313-318, 2005.
7) 木村　元：“ランダムタイリングと Gibbs-sampling を用いた多次元状態—行動空間における強化学習,” 計測自動制御学会論文集, vol.42, no.12, pp.1336-1343, 2006.

3-3-3 能動学習

3-3-4 機械学習と知識発見

（執筆者：井上克巳）［2016 年 11 月 受領］

機械学習は計算機によって学習能力を実現する技術・研究分野であるが，データから特徴を

抽出し分類する認識やクラスタリング以外にも知識の発見に利用することができる．こうした知識発見の研究では，人間が普段行っている学習・認識能力を実現するというよりは，これまでに知られていない科学法則のような未知の知識の発見について議論されることが多い．このため，知識発見には正解という概念が存在しておらず，人間の身体能力や典型的な思考を模倣することもできず，発見された科学的知識は何らかの形で検証されなければならない．

科学的発見については，古来から科学者が意識して用いたり，無意識に実践してきた発見のための方法論が存在している．このような方法論は科学哲学の分野で論じられてきており，仮説の形成が科学理論の根幹を成している．このため，仮説を用いた思考法をモデル化し計算機上で実現することで仮説発見を自動化していくことが知識発見には望まれる．更に，こうした仮説発見の技術は，自然科学における科学的発見のみならず，社会科学や人文科学にも適用可能であり，ビジネスや日常生活の多くの場面でも活用することができる．

(1)　発見の科学

科学的思考法の探求は，Francis Bacon（1561～1626）や John Stuart Mill（1806～1873）による経験事実から一般規則を導き出す帰納法の体系化に始まった．帰納法により得られる法則は実験結果と合っているかどうかで評価されるが，新しい理論を提示するわけではない．そこで，Charles Sanders Peirce（1839～1914）は，新しい仮説を提示するためにアブダクション（Abduction, 発想）による思考法の重要性を説いた [1),2)]．また Karl Popper（1902～1994）は帰納主義を批判し，仮説は反証可能（Falsifiable）でなければならないとする立場 [3)] を説いた．

科学的発見のプロセスとは，観測，仮説形成，予測，実験，観測，…，のサイクルを形成し，この繰り返しにより仮説が洗練されていく過程である [4)]．このサイクルにおいて，予想外の観測は科学的探究の出発点であり，観測の説明を可能にするために仮説が導入される．この仮説は新たな予測を導き，それは実験によって確かめられる．その際に新しい観測によって，前の仮説は採択または修正される．これをパースによる推論の 3 分法と重ね合わせると，推測による仮説生成はアブダクションであり，仮説からの帰結計算は演繹であり，実験による結果の検証は帰納であるとみなすことができる．例として，天王星の軌道が計算と異なることから（観測），海王星の存在を仮定し（アブダクションによる仮説 [2)]），その天体のあるべき位置を計算し（演繹による予測 [3)]），実際に観測によって確証する（実験～帰納），という流れに沿って天文学の発展がなされている．こうして従来の学説が更新されると，別の観測により更なる仮説が提示されサイクルが形成されていく．

(2)　発見の自動化

科学的探究を自動化する企ては，そもそもコンピュータの出現時からあり，最初は科学計算のために計算機が使われた．人工知能（AI）では，1960 年代後半に AI プログラム Dendral が世界初のエキスパートシステムとして有機化合物の特定を行った [5)]．この流れを引き継ぎ，Lenat はヒューリスティックス（経験則）を組み込んだ AM や後継の EURISKO プログラム [6)] において新概念の発見を試みた．AM は Automated Mathematician の略で数学を対象としていたが，Eurisko はギリシャ語で発見するという意味で他分野に対象を拡げた．これらのプログラムでは，ヒューリスティックをあらかじめ必要なだけ用意しておく必要があり，獲得できる規則の数は限定的であった．1990 年代には，多くの観測事例から一般法則を導くデータマイニングやサンプルを分類する統計的学習を中心に機械学習の研究開発が進んだ．2000 年代に入り，King らは「ロボット科学者」（Robot Scientist）によって，仮説生成，実験計画作成，実験遂行

に至る知識発見サイクルを生化学分野で部分的に具現化し[7]，物理学ではコーネル大学のグループが物理現象の観測から物理法則を導き出す実験を行った[8]．

こうした機械学習を用いた知識発見の研究では，パターン認識のような学習と比べて困難な点がいくつかある．まず，仮説を知識として明示する必要があるため，ニューラルネットワークのように学習器の精度を高めるだけでなく，そこから有意義な知識を人間が理解できる形式で抽出しなければならない．このため，学習器をブラックボックスとして扱うのではなく，提示された仮説に対する意味付けや根拠が必要となる．次に，知識体系を一から構築するということはなく，既存の知識を背景知識として用いなければならない．背景知識を用いた帰納を実現し，仮説が宣言的に表現される枠組みとしては，帰納論理プログラミング（ILP）[9] が研究されてきたが，計算量が高くスケーラビリティが問題となる．ただし ILP では，新述語の発明（Predicate Invention）を実装したシステムも提案されており，新概念発見への応用が期待されている．知識発展サイクルに含まれる帰納とアブダクションを組み合わせ，新概念も含めて仮説を生成できる枠組みとしては，メタ推論の利用も有望視されている[10]．今後，こうした記号計算がスケールアップされれば，知識発見の研究が一段と進むと予想される．

情報技術の発達により膨大な観測データが得られるようになった今日，従来人間の創造的活動により行っていた仮説発見の作業が，計算機抜きには難しくなっている．更に，自動化された発見プロセスが知識の量を飛躍的に増大させ，発見のスピードが加速されることも予測されている．科学的発見は計算機技術や AI の究極の応用であると言ってもよいだろう．

■参考文献

1) C.S. Peirce：“The Collected Papers,” C. Hartshorne and P. Weiss (eds.), Harvard University Press, Cambridge, MA, 1935.
2) 米盛裕二：“アブダクション―仮説と発見の論理,” 勁草書房, 2007.
3) K.R. Popper：“The Logic of Scientific Discovery,” Hutchinson, London, 1959.（邦訳: 大内義一，森　博（訳）：“科学的発見の論理,” 恒星社厚生閣, 1971.
4) 井上克巳：“人工知能による科学的発見,” 電子情報通信学会誌, vol.98, no.1, pp.35-39, 2015.
5) D. Waltz and B. Buchanan：“Automating science,” Science, vol.324, pp43-44, 2009.
6) D.B. Lenat and J.S. Brown：“Why AM and EURISKO appear to work,” Artificial Intelligence, vol.23, pp.269-294, 1984.
7) R. King, K. Whelan, F. Jones, P. Reiser, C. Bryant, S. Muggleton, D. Kell, and S. Oliver：“Functional genomic hypothesis generation and experimentation by a robot scientist,” Nature, vol.427, no.6971, pp.247-252, 2004.
8) M. Schmidt and H. Lipson：“Distilling free-form natural laws from experimental data,” Science, vol.324, pp.81-85, 2009.
9) S. Muggleton, L. De Raedt, D. Poole, I. Bratko, P. Flach, K. Inoue, and A. Srinivasan：“ILP turns 20: biography and future challenges,” Machine Learning, vol.86, no.1, pp.3-23, 2012.
10) K. Inoue：“Meta-Level Abduction,” IFCoLog Journal of Logics and their Applications, vol.3, no.1, pp.7-35, 2016.

■S3 群－3 編－3 章

3-4 対象，表現手法

3-4-1 識別関数の学習

（執筆者：小野田崇）［2016 年 7 月 受領］

ここでは，識別関数の学習について述べる．機械学習は大きく「教師あり学習」と「教師なし学習」の 2 つに分類される．様々な見方はあるものの，ここでは，識別関数の学習は教師あり学習に属するものとする．識別関数の学習が属する教師あり学習は，ある規則に従う「入力」と「出力」の例（教師）から，機械が「入力」と「出力」との関係 $f : \mathcal{X} \to \mathcal{Y}$ を学習する以下の状況を意味する．

1. 「入力」と「出力」との例：学習データ集合 $D = \{(\boldsymbol{x}_i, y_i), i = 1, \cdots, n\}$ を観察
2. 学習データ集合 D から $y \simeq \hat{y} = f(\boldsymbol{x})$ なる機械 f を学習

ここで，$\boldsymbol{x}$ は説明変数，独立変数，入力，特徴量などと呼ばれる．y は応答変数，従属変数，出力，目的変数などと呼ばれる．また，教師あり学習は，y が離散値の場合に判別問題，y が連続値の場合に回帰問題と呼ばれる．

機械が学習する際にデータの生成過程にどの程度確率を組み込むかによって，教師あり学習を以下のように分類することができる．

生成モデル：ナイーブベイズなどのように，あるクラス y が確率 $p(y)$で選択され，そのクラスから特徴量 $\boldsymbol{x}$ が確率$p(\boldsymbol{x}|y)$で生成されたと考えるアプローチ

識別モデル：ロジスティック回帰，Conditional Random Field（CRF）などのように，事後確率 $p(y|\boldsymbol{x})$を学習データ集合 D から直接推定するアプローチ

識別関数：線形判別分析，サポートベクターマシン（SVM），ニューラルネットワークなどのように，学習データ集合 D に基づき，その特徴量 $\boldsymbol{x}$ から直接クラスラベル y 推定する関数（機械）f を求めるアプローチ

以下では，識別関数の学習における代表格である線形判別分析と SVM について紹介する．

(1) 線形判別分析

d 次元の多次元正規分布は以下の式で表現される．

$$N(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{2\pi^d}\sqrt{|\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{x} - \boldsymbol{\mu})\right)$$

この正規分布の指数部分は，データ点 $\boldsymbol{x}$ と平均$\boldsymbol{\mu}$間の距離を表しており，これはマハラノビス距離と呼ばれる．マハラノビス距離はユークリッド距離に共分散行列$\boldsymbol{\Sigma}$の逆行列をかけているので，正規分布の広がりを考慮した距離になっている．i 番目のクラス C_i のクラス条件付き確率を以下のように仮定する．

$$p(\boldsymbol{x}|C_i) = \frac{1}{\sqrt{2\pi^d}\sqrt{|\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_i)^\top \boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_i)\right)$$

クラスの事前確率を $p(C_i)$とすれば，事後確率は以下のように表現できる．

$$p(C_i|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|C_i)p(C_i)}{p(\boldsymbol{x})} \propto \frac{p(C_i)}{\sqrt{2\pi^d}\sqrt{|\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_i)^\top \boldsymbol{\Sigma}_i{}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i)\right)$$

上式の対数をとると以下を得る．

$$\ln p(C_i) - \frac{d}{2}\ln(2\pi) - \frac{1}{2}\ln|\boldsymbol{\Sigma}_i| - \frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_i)^\top \boldsymbol{\Sigma}_i{}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i)$$

以上より，i 番目のクラスの事後確率から求まる評価値は次式となり，識別クラスとしてこの値が最も小さなクラスを選択すれば，誤り最小基準のベイズの識別規則を得ることができる．

$$g_i(\boldsymbol{x}) = (\boldsymbol{x}-\boldsymbol{\mu}_i)^\top \boldsymbol{\Sigma}_i{}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i) + \ln|\boldsymbol{\Sigma}_i| 2\ln p(C_i)$$

クラス間の識別境界では 2 クラスの事後確率が等しくなるので，クラス i と j の識別境界は以下の 2 次曲面となり，その値の正負でクラスの識別ができる．

$$g_i(\boldsymbol{x}) - g_j(\boldsymbol{x}) = \boldsymbol{x}^\top(\boldsymbol{\Sigma}_i{}^{-1} - \boldsymbol{\Sigma}_j{}^{-1})\boldsymbol{x} + 2(\boldsymbol{\mu}_j)^\top \boldsymbol{\Sigma}_j{}^{-1} - \boldsymbol{\mu}_i)^\top \boldsymbol{\Sigma}_i{}^{-1})\boldsymbol{x} + \mathrm{const}$$

上式は 2 次識別関数と呼ばれる．更に，2 クラスの共分散が等しいと仮定する場合， $\boldsymbol{\Sigma}_i{}^{-1} - \boldsymbol{\Sigma}_j{}^{-1} = \boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1} = 0$ なので，識別境界は以下の線形識別関数となる．

$$g_i(\boldsymbol{x}) - g_j(\boldsymbol{x}) = 2(\boldsymbol{\mu}_j - \boldsymbol{\mu}_i)^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{x} + \mathrm{const}$$

この線形識別関数の学習は，学習データ集合 D から$(\boldsymbol{\mu}_j - \boldsymbol{\mu}_i)^\top \boldsymbol{\Sigma}^{-1}$（平均と共分散行列）を推定することである．

(2)　SVM

学習データ集合 D が与えられ，$f(\boldsymbol{x}_i) = y_i, i = 1, \cdots, n$を満たす識別関数$f(\boldsymbol{x}) = \mathrm{sgn}\big((\mathbf{w}^\mathrm{T}\boldsymbol{x}) + b\big)$を推定する問題を考える．分類可能な 2 次元上の 2 クラス判別問題を示したものが**図 4･1** である．識別関数を一意に定めるため，図 4･1 には，$y_i \cdot \big((\mathbf{w}^\mathrm{T}\boldsymbol{x}) + b\big) \geq 1, i = 1, \cdots, n$の制約を導入している．SVM では，この 2 クラス間の最小距離 $2/\|\mathbf{w}\|$ を最大化するように識別関数のパラメータ $\mathbf{w}$ と b を決定する [1]．

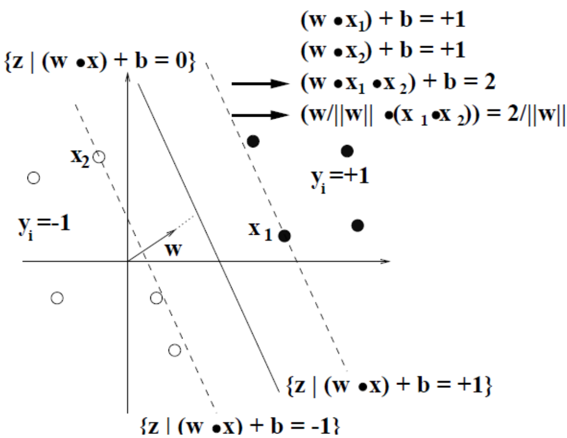


図 4・1　2 クラス分類問題

SVM では，制約条件 $y_i \cdot \left((\mathbf{w}^{\mathrm{T}} \boldsymbol{x}) + b \right) \geq 1,\ i = 1, \cdots, n$ のもと，以下の目的関数の最小化を行い，2 クラス間の最小距離を最大化する．

$$\tau(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|^2 \qquad (4・1)$$

SVM では，この最適化問題をラグランジェ乗数$\alpha_i \geq 0$とラグランジアン $L(\mathbf{w}, b, \alpha) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{n} \alpha_i \left(y_i \left((\mathbf{w}\boldsymbol{x}_i) + b \right) - 1 \right)$を導入して，双対問題を考える．このラグランジアンをα_iについて最大化し，$\mathbf{w}$ と b について最小化する．パラメータ $\mathbf{w}$ と b についてラグランジアン L の導関数は，鞍点において $\frac{\partial}{\partial b} L(\mathbf{w}, b, \alpha) = 0,\quad \frac{\partial}{\partial \mathbf{w}} L(\mathbf{w}, b, \alpha) = 0$ となる．このことから，$\sum_{i=1}^{n} \alpha_i y_i = 0,\ \mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i x_i$である．また，Karush-Kuhn-Tucker 条件により，鞍点においてラグランジェ乗数α_iは，$\alpha_i \left[y_i \left((\mathbf{w}^{\mathrm{T}} \boldsymbol{x}_i) + b \right) - 1 \right] = 0,\ i = 1, \cdots, n$ の制約条件に対して非ゼロでなくてはならない．$\alpha_i > 0$を有するパターン $\boldsymbol{x}_i$をサポートベクトルと呼ぶ．図 4･1 中，$\boldsymbol{x}_1$ 及び $\boldsymbol{x}_2$ がサポートベクトルである．

ラグランジアンの導入により，制約条件$y_i \left((\mathbf{w}^{\mathrm{T}} \boldsymbol{x}_i) + b \right) \geq 1, i = 1, \cdots, n$ のもとでの$\tau(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|^2$の最小化問題は，以下の双対問題等価となる．

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j (\boldsymbol{x}_i^{\top} \boldsymbol{x}_j) \\ \text{subject to} \quad & \alpha_i \geq 0, \quad i = 1, \cdots, n, \\ & \sum_{i=1}^{n} \alpha_i y_i = 0. \end{aligned} \qquad (4・2)$$

識別関数は$f(\boldsymbol{x}) = \mathrm{sgn}\left(\sum_{i=1}^{n} \alpha_i y_i (\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{x}) + b \right)$のように表現される．以上より，識別関数$f(\boldsymbol{x}) = \mathrm{sgn}\left((\mathbf{w}_i^{\mathrm{T}} \boldsymbol{x}) + b \right)$を得ることができる．これが基本となる線形 SVM である．

■参考文献

1)　小野田崇：“サポートベクターマシン,” オーム社, 2007.

3-4-2 形式言語の学習

（執筆者：吉仲　亮）［2018 年 8 月 受領］

通常の定義では，形式言語は有限記号集合上の有限列を要素とする可算集合である．形式言語は形式文法と呼ばれる有限記述を用いて表現されることが一般的であるため，形式言語の帰納学習は文法推論とも呼ばれる．数学的には，関数のグラフは集合であり，集合は特性関数を用いて表現可能であるが，一般には形式言語の学習と関数の学習とは別個に考えられる．正例からの言語学習など，一方で自然な学習の枠組みが他方では自然な定義を持たないからである．したがって，言語として符号化することが意味論的に適切な概念の獲得が形式言語学習研究の射程にあると言える．特に，自然言語は研究史以来の重要な動機である．自然言語の複雑さに比して，幼児に与えられる母語に関する情報量は貧困に思われるが，幼児は驚くべき効率で母

語を習得することから，Chomsky は，自然言語は極めて特殊な構造を持ち，人間には自然言語学習のための特殊な機構が生得的に備わっていると主張した．このような理論を背景に，母語獲得の数理モデルの探究という視点から，形式言語学習の体系的研究が開始された [2]．また，近年は生物情報学の発展に伴い DNA や RNA の塩基配列などの学習に関する研究も盛んになっている．その他，自動翻訳や音声認識などのパターン認識などへの形式言語学習理論の応用の試みもある．

形式言語の学習の数学的な研究のためには，学習の対象，学習者に与えられる情報，学習のアルゴリズム，また何をもって学習したと認めるのかの基準を定義しなければならない．学習の代表的な枠組みとしては，極限同定（Identification In The Limit）[2]，質問（Query）による学習 [1]，PAC（Probably Approximately Correct）学習 [5] が挙げられる．

極限同定では，学習対象言語に由来する情報が無限列として学習者に与えられ，学習者は無限の文法列を出力するが，出力は正しく学習対象を生成する文法に収束しなければならない．入力情報は，学習対象言語の要素（正例）のみであるか，もしくはすべての文字列について対象言語に属するか否かの所属性が与えられることが一般的である．質問による学習においては，学習者は教師（オラクル）に質問し回答を得，それに基づいて正しい文法を推測する．様々な質問と回答のプロトコルが提案されているが，特に，任意の文字列が学習対象言語に含まれるかを問う所属性質問と，学習者の推測が正しいか否かを問う等価性質問を用いる組み合わせは，MAT（Minimally Adequate Teacher）学習と呼ばれている．これらのモデルは厳密で誤りのない学習結果を要請するのに対して PAC 学習は確率的に精度の良い学習のモデル化を可能にした．学習者への入力は未知の確率分布に従う有限個の文字列及びそれぞれの文字列の対象言語に関する所属性であり，その確率分布において学習対象言語と推測言語の誤差が ε を超える確率が δ 以下に抑えられるような「多分大体正しい」推測を行う．これらの基本的な枠組みに加え，変種や組合せによる学習モデルも提案されており，例えば導出木を入力とする学習モデル，学習者に質問を許す PAC 学習などがある．また，これらの異なるモデル間の強弱関係についても研究がある．各モデルの詳細については本章の該当項目を参照されたい．

また，統計学習の文脈では，言語を既知として，隠れマルコフ決定過程（Hidden Markov Decision Process）や確率文脈自由文法によって表現される，言語の文字列の確率分布を求めることが学習の課題となる．詳しくは 3 章 3-1-2 項「統計的学習」を参照されたい．

これまで多種多様な形式言語のクラスが学習理論の研究対象とされてきた．それらは，形式言語理論においてその数理的性質が既によく研究された言語クラスであることも多いのはもちろんであるが（正規言語，文脈自由言語など），むしろ，学習理論上望ましいとされる性質を持つように，形式文法やオートマトンに対する自然な制約を考察することが一般によく行われている（リバーシブル言語（Reversible Language）など）．その場合もよく知られた制約の組合せである場合から，新しい制約の提案と言える場合まで様々である．更に，通常は学習対象となる形式言語のクラスは文法のクラスによって定義されるが，サブスティテュータブル文脈自由言語（Substitutable Context-free Language）のように，文法的特徴づけを持たず，統語的性質による定義のみを持つクラスの学習に関する肯定的な結果もある．また，パターン言語（Pattern Language）のように，学習理論の文脈から体系的な研究が始まった言語クラスでも，その定義の単純さや数理的性質の重要性から，学習理論の文脈から離れて純粋に形式言語理論的性質が議論されるようになったものもある．基本形式系（Elementary Formal System）は，本来は帰納

関数論の道具であったが，形式言語を定義する表現系とみなした場合，Chomsky 階層に対応する自然な制約を持つこと，生成/認識系の二重性を有することなどの特長があり，学習理論上有用な枠組みとして研究されている．この形式は，言語の正例からの極限同定に関する一般的で強力な肯定的結果をもたらしたほか，論理プログラム学習の研究において取り上げられるなどの側面も持つ形式である．

学習理論研究の道具として，形式言語理論で得られている性質がしばしば利用される．例えば, 正規言語の部分クラスの学習においてはMyhill-Nerodeの定理に基づいた標準形(Canonical)の決定性有限オートマトンが出力されることも多い．また，線形文法の導出言語（Szilard 言語とも呼ばれる）が正規言語になることを利用して，線形言語の学習を正規言語の学習に帰着することで，より広いクラスを学習可能にする方法論も提案されている．

本文中に引用したものに加えて，下記の参考文献が参考になると思われる．文献 3）と 4）は最新の研究動向もカバーした良質のサーベイである．文献 6）は形式言語学習の日本語の良質な教科書である．文献 7）はその最終章で基本形式系の学習に関する研究結果をよくまとめている．

■参考文献

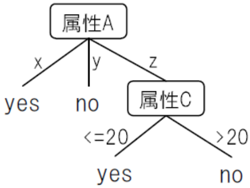
1) D. Angluin：“Queries and concept learning,” Machine Learning, vol.2, no.4, pp.319–342, 1988.
2) E. M. Gold：“Language identification in the limit,” Inform. Control, vol.10, no.5, pp.447–474, 1967.
3) C. de la Higuera：“A bibliographical study of grammatical inference,” Pattern Recognition, vol.38, pp.1332-1348, 2005.
3) S. Jain, D. Osherson, J.S. Royer, and A. Sharma：“Systems That Learn - An introduction to Learning Theory (second edition),” MIT Press, 1999.
4) S. Lange, T. Zeugmann, S. Zilles：“Learning indexed families of recursive languages from positive data: a survey, Theoret,” Comput. Sci. vol.387, pp.194-232, 2008.
5) L.G. Valiant：“A theory of the learnable,” Commun. ACM, vol.27, no.11, pp.1134-1142, 1984.
6) 榊原康文, 小林　聡, 横森　貴：“計算論的学習,” 培風館, 2001.
7) 西野哲朗, 石坂裕毅(著), 有川節夫(監)：“形式言語の理論,” 丸善, 1999.

3-4-3 決定木の学習

（執筆者：尾崎知伸）［2016 年 7 月 受領］

決定木 1)~3) とは，**図 4･2**(左)に示すような 1 枚の表（属性-値表）形式で表されるデータを対象とした分類器であり，葉ノードに予測クラス（目的属性），中間ノード（節）に分割テスト（説明属性）を配置した木構造を持つ．図 4･2(中央)に，決定木の例を示す．この木は，2 つの中間ノード A と C を持ち，また 4 つの葉ノードには，それぞれ予測クラス yes または no が割り当てられている．決定木は，クラス未知の事例に対し，根から順に葉ノードに至るまで分割テストを繰り返すことで，そのクラスを予測する．例えば，図 4･2 の決定木を使って，事例 < 属性 A = z, 属性 B = b, 属性 C =25 > のクラスを予測することを考える．まず，根ノードである属性 A の値を確認することで“z”の枝をたどり，次いで中間ノードである属性 C の値を確認することで“> 20”の枝をたどる．これにより葉ノードへと到達するので，そのラベルを用いクラスを“no”と予測する．

属性 A	属性 B	属性 C	クラス
x	a	5	yes
x	a	10	yes
x	b	20	yes
y	a	15	no
y	b	20	no
y	b	30	no
z	a	15	yes
z	a	25	no
z	b	17	yes
z	b	27	no



if 属性 **A = x then** クラス **= yes**
if 属性 **A = y then** クラス **= no**
if 属性 **A = z and**
属性 **C <= 20 then** クラス **= yes**
if 属性 **A = z and**
属性 **C > 20 then** クラス **= yes**

図 4・2　学習データ（左）と決定木（中央），ルール（右）の例

良い決定木の条件として，構造の簡潔さとクラス未知の事例に対する高い予測能力が挙げられる．しかし一般に，所与の学習データ集合から決定木を構築する際，どの位置にどの属性を配置すれば良い決定木が得られるかは自明ではなく，また考えられる組合せ数の多さから，全数探索によって最適な木を発見することは必ずしも現実的ではない．したがって，多くの場合，木の良さの基準である簡潔性と分類・予測精度を考慮した発見的な探索（ヒューリスティック探索）に基づく手法が採用される．**図 4・3** に，広く用いられているトップダウン探索に基づく決定木学習アルゴリズム（Top-Down Induction of Decision Trees：TDIDT）の概要を示す．このアルゴリズムは，学習データ集合を分割したうえで同じ操作（決定木の構築）を繰り返す分割統治法に基づいており，またその時点で最適な分割テスト（属性）を選択する貪欲探索を採用している．

アルゴリズム TDIDT（入力：学習データ集合 S，属性の集合 Att，出力：ノード v）

1: **if** 停止条件を満たす **then** S 中の最頻クラスをラベルとする葉ノード v を返す．
2: 最良の分割テストとして，属性値 $\{a_1, \cdots, a_n\}$ を持つ属性 $A \in Att$ を選択する．
3: 分割属性として属性 A を持つ中間ノード v を準備する．
4: **for each** a_i in $\{a_1, \cdots, a_n\}$
5: $\quad S_i := \{s \in S \mid$ データ s において属性 A の値は a_i である $\}$
6: $\quad v_i :=$ TDIDT($S_i, Att \setminus \{A\}$)
7: $\quad v_i$ を v の子ノードとする．また v–v_i 間の辺ラベルを a_n とする．
8: **end**
9: v を返す

図 4・3　決定木学習アルゴリズム

TDIDT の停止条件（1 行目）としては，「S 中の全データが同一クラスに属する」や，「分割するための属性が残されていない（$Att = \emptyset$）」などが利用される．一方，分割テストの選択（2 行目）では，分割後の各ノードにおいてデータのクラス分布が偏ること，すなわち多様性や不純度が小さくなる属性を選択する．そのような基準として，これまでに情報利得や情報利得比，ジニ係数などが提案されている．例えば，情報利得は，データ分割前のデータ集合に対するエントロピーと，分割後の各データ集合のエントロピーの加重平均との差であり，形式的には，

k 個のクラスから成る学習データ S に対する属性 A の情報利得は$\mathrm{Gain}(S,A) = I(S) - \sum_{i=1}^{n} |S_i|/|S| \cdot I(S_i)$ と定義される．ここで，S 中のデータのクラスが j である確率を $p_j(S)$ と表記し，$I(S) = -\sum_{i=1}^{k} p_j(S) \log p_j(S)$ である．分割テストの選択では，この値を最大にする属性 $A = \mathrm{argmax}_{A \in Att}\ \mathrm{Gain}(S,A)$ が最良の属性と判断される．

決定木を学習する際，単純に分割テストの選択を繰り返すだけでは，学習データ集合中の誤りや外れ値などにも過剰に適合した決定木が生成されてしまう．この過学習の問題を回避し，クラス未知のデータに対する予測精度を向上させるために，一般に，枝刈り技法が適用される．枝刈りは，その適用のタイミングによって，事前枝刈りと事後枝刈りに分けられる．事前枝刈りは，決定木構築中に適用される枝刈り技法であり，分割に対する統計的重要性や誤り減少性などに基づき過学習を引き起こすと予測された場合に，それ以上の分割を終了する．一方，事後枝刈りは後処理として適用される枝刈りであり，決定木構築後に過学習が起きていると考えられる部分木をまとめて一つの葉に置き換える．これら枝刈り手法を適用することで，より簡潔かつ高い予測性能を持つ木が構築されることが期待できる．

決定木は，図 4･2(右)に示すように，複数の分類ルール（if then ルール）をまとめたものとみなすことができる．また，予測に利用される分割テスト（属性）が明示されるので，結果に対する可読性や理解容易性に優れている．加えて，記号属性と数値属性が混在しているデータや欠損値を持つデータを対象とすることができる柔軟性や，大規模データに対しても高速に学習・予測が可能であるという特徴を持つ．

また，決定木は，分割テストの作り方を工夫することにより，容易に属性−値表以外のデータへと応用することができる．例えば，文字列データの場合，ある部分文字列を含むか否かを分割テストとして採用すればよい．これまでに，文字列データに加え，時系列データやグラフなどの構造データ，述語論理式など，より複雑なデータを対象とした決定木やその効率的な構築手法が提案されている．

■参考文献

1) J.R. Quinlan：“Induction of decision trees,” Machine Learning, vol.1, no.1, pp.81-106, 1986.
2) J.R. Quinlan：“C4.5: Programs for Machine Learning,” Morgan Kaufmann Publishers Inc., 1993.
3) L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone：“Classification and Regression Trees,” Chapman & Hall, 1984.

3-4-4 帰納論理プログラミング

（執筆者：平田耕一）［2018 年 8 月 受領］

帰納論理プログラミング（Inductive Logic Programming，以後 ILP と略す）とは，一般に「論理プログラムを対象とした帰納（学習）」のことを指し，現在では，仮説の表現として決定木やブール関数よりも複雑な関係表現を伴うデータマイニングに広く適用されている [1),8)]．なお，ILP という言葉は 1990 年代初頭に Muggleton が導入したものである [4),5)]．

まず，ILP を説明するために必要な記法を導入する．アトム（素論理式，原子論理式）とその否定をリテラル（Literal）という．このとき，アトムを正リテラル（Positive Literal），アトムの否定を負リテラル（Negative Literal）という．リテラルを選言（∨）で結んだもの（もしくはそれを集合とみなしたもの）を節（Clause）という．特に，高々一つの正リテラルしか含まない節をホーン節（Horn Clause）という．節の集合（節集合）を論理プログラム（Logic Program）

という．変数を含まないリテラル，節，プログラムを基礎（Ground）という．

節 C, D に対して，$C\theta \subseteq D$ となる代入 θ が存在するとき，C は D を包摂する（C subsumes D）といい，$C \succeq D$ と表す．また，C を真にするすべての解釈が D を真とするとき，C は D を伴意する（C entails D，もしくは C は D を含意する（C implies D））といい，$C \models D$ と表す．$C \succeq D$ ならば $C \models D$ であるが，逆は一般に成り立たない．更に，節 C，D と論理プログラム B に対して，$B \models \forall(C\theta \square D)$ となる代入 θ が存在するとき，C は D を B に関して相対包摂する（C subsumes D relative to B）という [7,10]．$C \succeq D$ は $C \succeq_{\theta} D$ のことである．節，及び，ホーン節の伴意関係や相対包摂関係は一般に決定不能であり，包摂関係は決定可能であるが一般に NP 完全である．

以上の準備のもとで，ILP を実現するシステムである ILP システムについて概説する．ILP システムの論理的設定（Logical Setting）は，以下のように記述することができる．B を背景知識と呼ばれる論理プログラム，$E = E^{+} \cup E^{-}$ を例の集合（E^{+} は正例の集合，E^{-} は負例の集合）とし，$B \not\models E^{+}$（事前不十分性），かつ，$B \cup E^{-}$ が無矛盾である（事前無矛盾性）とする．このとき，ILP システムの目的は，仮説空間 $\mathcal{H}$ の中から，$B \cup H \models E^{+}$（事後十分性），かつ，$B \cup H \cup E^{-}$ が無矛盾（事後無矛盾性）となる仮説 H を求めることである．また，ILP システムは，常に事後十分性を満足する仮説を生成するとき健全である（Sound）といい，すべての事後十分性を満足する仮説を生成するとき完全である（Complete）という．

ILP システムの仮説空間 $\mathcal{H}$ としては，節全体 $\mathcal{C}$ やホーン節全体 $\mathcal{HC}$ などが用いられる．更に，仮説空間 $\mathcal{H}$ から仮説を求めるために，ILP システムでは，仮説空間に仮説間の擬順序としての一般化関係 $\geq$ を導入する．一般化関係 $\geq$ としては，伴意関係 $\models$，包摂関係 $\succeq$，相対包摂関係 $\succeq_{B}$ などが用いられる．今後，仮説空間 $\mathcal{H}$ を一般化関係 $\geq$ を合わせて $\langle \mathcal{H}, \geq \rangle$ と記述する．

仮説空間 $\langle \mathcal{H}, \geq \rangle$ と仮説 $H_1, H_2 \in \mathcal{H}$ に対して $H_1 \geq H_2$ となるとき，H_1 は H_2 より一般的である（More General，もしくはより強い（Stronger））といい，H_2 は H_1 より具体的である（More Specific，もしくはより弱い（Weaker））という．特に，H_1 を H_2 の一般化（Generaliz(s)ation）という．また，仮説の集合 $S \subseteq \mathcal{H}$ の $\geq$ に関する一般化のうち最小なものを S の最小一般化（Least Generaliz(s)ation）という．

$\langle \mathcal{C}, \succeq \rangle$（または $\langle \mathcal{HC}, \succeq \rangle$）において，$S \subseteq \mathcal{C}\,\mathcal{H}$（または $S \subseteq \mathcal{HC}$）の最小一般化は（変数の名前を除いて）一意に決まり，しかもそれを求める多項式時間アルゴリズムが存在する [9]．一方，$\langle \mathcal{C}, \models \rangle$，$\langle \mathcal{HC}, \models \rangle$，$\langle \mathcal{C}, \succeq_{B} \rangle$，$\langle \mathcal{HC}, \succeq_{B} \rangle$ において，$S \subseteq \mathcal{C}$（または $S \subseteq \mathcal{HC}$）の最小一般化は一般には存在しない．ただし，$S \subseteq \mathcal{C}$ が関数記号を含まない節を含む場合には $\langle \mathcal{C}, \models \rangle$ で，B が基礎リテラルだけからなる場合には $\langle \mathcal{HC}, \succeq_{B} \rangle$ で，B が基礎正リテラルからなる場合には $\langle \mathcal{HC}, \succeq_{B} \rangle$ でそれぞれ S の最小一般化が存在し，それらはすべて $\langle \mathcal{C}, \succeq \rangle$ における最小一般化アルゴリズムを用いて求めることができる [7,10]．

ILP システムでは，一般化関係に基づく精密化演算子（Refinement Operator）[7,13] を用いることで仮説空間を探索する．仮説空間 $\langle \mathcal{H}, \geq \rangle$ と $\mathcal{H}$ の二項関係 π に対して，$H \pi H'$ ならば $H \geq H'$ となるとき，π を $\langle \mathcal{H}, \geq \rangle$ の精密化演算子という．精密化演算子 π に対して，$\rho(H) = \{H' \in \mathcal{H} \mid H \pi H'\}$ となる ρ を $\langle \mathcal{H}, \geq \rangle$ の下向き精密化演算子（Downward Refinement Operator）といい，$\delta(H) = \{H' \in \mathcal{H} \mid H' \pi H\}$ となる δ を $\langle \mathcal{H}, \geq \rangle$ の上向き精密化演算子（Upward Refinement Operator）という．具体的に下向き精密化演算子 ρ を設定することで，$\rho(H)$ として仮説 H より具体的な複数の仮説を得ることができ，具体的に上向き精密化演算子 δ を設定することで，$\delta(H)$ として仮説 H より一般的な複数の仮説を得ることができる．下向き精密化演算

子を用いた探索をトップダウン（Top Down）探索，上向き精密化演算子を用いた探索をボトムアップ（Bottom Up）探索という．特に，ボトムアップ探索では，上向き精密化演算子として包摂関係や相対包摂関係に関する最小一般化を用いることが一般的である．

仮説空間の探索の基準となる仮説の選択や仮説空間の探索法である精密化の設定により，これまでに様々な ILP システムが構築されてきた．ここでは，有名な ILP システムである Muggleton の Progol [2),6)] と Quinlan の FOIL [11)] を紹介する．

Progol [2),6)] は，伴意の逆演算である逆伴意法（Inverse Entailment）を用いたトップダウン探索システムである．この逆伴意法は，一つの正例 $e \in E^+$を説明する無矛盾な仮説 H のなかで，$B \cup H \models e$ を変形した $B \cup \{\neg e\} \models \neg H$ を満たす H を，B と $\neg e$ から帰結される基礎アトムの連言（$\wedge$）の否定である最弱仮説（Most Specific Hypothesis）として求める．そして，最弱仮説に下向き精密化演算子を適用することで仮説空間をトップダウンに探索して最適な仮説を選択し，その仮説によって説明できる正例を削除する．この操作を正例がなくなるまで繰り返すことで節を学習する．ただし，Progol の逆伴意法は健全であるが完全ではなく，特に，H が B と e から Progol の逆伴意法で得られるとき，そのときに限り，$H \succeq_B e$ となる [14)]．また，Progol の学習対象は一般の節ではなく，ホーン節である．一方，現在では，節を対象とした完全な逆伴意法として，Residue 手続き [15)] や CF 帰納法 [3)] が知られている．

FOIL [11)] は，情報量に基づくリテラルの選択を用いたトップダウン探索システムである．FOIL の背景知識と例の集合は基礎アトムの集合であり，学習対象は関数記号を含まない述語 P の定義節 $P(x_1, \cdots, x_k) \leftarrow L_1, \cdots, L_n$（ただし，$L_i$はリテラル）である．ここで，$\leftarrow$ の右側を本体と呼ぶ．FOIL では，変数を 1 つずつ増やしながらリテラルを定義節の本体に追加するという下向き精密化演算子を用いて，仮説をトップダウンに探索する．例の集合 $E = E^+ \cup E^-$に対して，E 中の一つの例が正例であるかどうかを知るために必要な情報量は $I(E) = -\log_2(|E^+|/|E|)$ である．リテラル L，及び，L を追加する前の例の集合 E_i と追加した後の例の集合 E_{i+1} に対して，E_i^+のうち L を真にするものを $E_i^+(L)$ とし，L の利得（Gain）を $|E_i^+(L)|\big(I(E_i) - I(E_{i+1})\big)$とする．このとき，追加候補のリテラルのうち利得が最大となるようなリテラルを定義節の本体に追加する．この操作を仮説によって説明できない正例がなくなるまで繰り返すことで定義節を学習する．なお，Quinlan は，FOIL を拡張し，関数記号を許した定義節を学習対象とした FFOIL を構築している [1),12)]．

以上，ILP について ILP システムの観点から概説した．ILP の研究は，ILP システムの構築だけでなく，一般化関係の定式化とそれに基づく組合せ問題の解析，精密化演算子の定式化と解析，逆伴意法の演繹的解法，ILP のための新たな道具立てとその解析，論理プログラムの計算学習理論による学習可能性，ILP のデータマイニングへの応用など，非常に幅広い．このような ILP に関する最新の研究成果は，毎年開催されている International Conference on Inductive Logic Programming をはじめとする機械学習・計算学習理論に関する国際会議で発表されている．

■参考文献

1) S. Džeroski, N. Lavrač (eds.)：“Relational data mining,” Springer, 2001.
2) 古川康一，尾崎知伸，植野　研：“帰納論理プログラミング,” 共立出版, 2001.
3) K. Inoue：“Induction as consequence finding,” Machine Learning, vol.55, pp.109-135, 2004.
4) S. Muggleton：“Inductive logic programming,” Proc. 1st ALT, pp.42-62, 1990.

5) S. Muggleton (ed.)：“Inductive logic programming,” Academic Press, 1992.
6) S. Muggleton：“Inverse entailment and Progol,” New Generation Computing, vol.13, pp.245-286, 1995.
7) S.-H. Nienhuys-Cheng, R. de Wolf：“Foundations of inductive logic programming,” LNAI vol.1228, Springer, 1997.
8) 元田　浩, 津本周作, 山口高平, 沼尾正行：“データマイニングの基礎,” オーム社, 2006.
9) G.D. Plotkin：“A note on inductive generalisation,” Machine Intelligence, vol.5, pp.153-163, 1970.
10) G.D. Plotkin：“A further note on inductive generalisation,” Machine Intelligence, vol.6, pp.101-124, 1972.
11) J.R. Quinlan：“Learning logical definitions from relations,” Machine Learning, vol.5, pp.238-266, 1990.
12) J.R. Quinlan：“Learning first-order definitions of functions,” Journal of Artificial Intelligence Research, vol.5, pp.139-161, 1996.
13) E.Y. Shapiro：“Inductive inference of theories from facts,” Technical Report 192, Yale University, 1981. Reprinted in: J.-L. Lassez, G.D. Plotkin (eds.)：“Computaitonal logic,” pp.199-234, The MIT Press, 1992.（有川節夫（訳）：“知識の帰納的推論,” 共立出版, 1986.
14) A. Yamamoto：“Which hypotheses can be found with inverse entailment?,” Proc. 7th ILP, pp.296-308, 1997.
15) A. Yamamoto：“Hypothesis finding based on upward refinement of residue hypotheses,” Theoretical Computer Science, vol.298, pp.5-19, 2003.

■S3 群－3 編－3 章

3-5 評　価

3-5-1 極限同定

（執筆者：赤間陽二）［2018 年 8 月 受領］

例題から背後にある命題，法則，文法を学習する過程は，一般には，無限に続く過程であり，それに着目して学習現象を計算可能性の理論（Computability Theory）[19),21)] を用いて形式的に研究することは，ヒラリー・パットナム（Hilary Putnam），ソロモノフ（R. J. Solomonoff）やゴールド（E. Mark Gold）[13)] らにより 1960 年代から始められ，現在，極限同定（Identification in The Limit），極限での学習（Learning in The Limit）や帰納推論（Inductive Inference）と言われている．

極限同定の典型例として，未知だがあらかじめ決まっている自然数係数多項式 u により生成される $u(0), u(1), \cdots$ の最初の何個かから次の自然数を予測するゲームを考える．予測が成功し続けるようになるための戦略としては，次の 2 つを考える．

1. 自然数係数多項式の係数列 $\boldsymbol{a}^1, \boldsymbol{a}^2, \cdots$ を枚挙し，例 $u(t)$ を受けとるごとに$\sum_j \boldsymbol{a}^i_j s^{j-1} = u(s)$ $(0 \le s \le t)$ となるかを調べ，異なれば次の候補 $\boldsymbol{a}^{i+1}$ を試す，という仮説の枚挙と検査による同定法がある．
2. 各時点でそれまでのデータに無矛盾な最も「単純」な仮説を計算するという同定法もある．それは，次数 $d = 0$ から始めて，各時点 t で $\big(s, u(s)\big)$ $(0 \le s \le t)$ を通る多項式を計算して，その多項式 $\hat{u}$ の $t+1$ での値が $u(t+1)$ と違うならば $\big(s, u(s)\big)$ $(0 \le s \le t+1)$ を通る次の次数の多項式を計算することを繰り返す，というものである．

前者の同定法では無限個のすべての整数係数多項式を走査しようとするが，後者は有限個であり効率が良い．このように，可能な仮説の多さと仮説の変更回数は関係がない．

どういう同定問題には正解を出せるようになる（極限同定可能）かを正確に議論するための論点として以下がある．

(1) 同定の対象全体．それは一般に関数，言語，述語である．上の例では同定の対象は勝手な自然数係数多項式 u である．

(2) 仮説全体．同定の各対象は適当な仮説で表現されているものとする．上の例では u の係数列 $\boldsymbol{a}$ である．

(3) 同定する対象の例示法．上の例では u のグラフを $(0, u(0)), (1, u(1)), \cdots$ と提示しているが，どんな順序で提示しても上述の 2 つの同定法で同定できる．

(4) 同定法．

(5) 同定の達成基準．

(6) 同定の効率．

効率の古典的基準としては

- (6-1) 同定を達成するまでに必要な例の個数．
- (6-2) 同定を達成するまでに仮説を変更する回数，すなわち試行錯誤の回数（Mind-change Complexity）．

以下 $\mathbb{N}$ から $\mathbb{N}$ への帰納的関数（Total Recursive Function）全体を R とおき，ここではまず R に属する関数の極限同定可能性について述べる．以下 $i \in \mathbb{N}$ が符号化しているプログラムが

計算する ℕ から ℕ への部分帰納的関数（Partial Recursive Function）をφ_iで表す．i は時系列$\left(\varphi_i(t)\right)_t$の「法則」であり，また，$\varphi_i$の入出力関係という「振る舞い」の「説明」とみなせる．

冒頭の例の予測問題に対応して，まずチューリング機械 P が $u \in R$ を$\mathcal{NV}$で同定することを，

i. ℕ のどんな有限列 $(x_1, \cdots, x_n)$ を P に入力しても M は出力し，

ii. 時系列の最初の適当な t_0 個の例を入力すると，P はそれ以降の例を P は次々と予測するようになる，すなわち $\exists t_0 \forall t \geq t_0 . P\left(u(0), \cdots, u(t)\right) = u(t+1)$ が成立する，

とする．M が $U \subseteq R$ を$\mathcal{NV}$ 同定するとは，U に属するどんな関数 u も M が$\mathcal{NV}$ 同定することである．適当な P で$\mathcal{NV}$ 同定できる関数の集合 U 全体を$\mathcal{NV}$ と表す [6),7)].

冒頭の例の最初の解法「枚挙による同定法」であるが，帰納的関数の集合で「枚挙による同定法」で同定できるもの全体は，

$$\mathcal{NUM} = \left\{U \subseteq R\ ;\ \exists a \in R. U \subseteq \left\{\varphi_{a(i)}\ ;\ i \in \mathbb{N}\right\}\right\}$$

であるが，帰納的枚挙 $a \in R$ があれば予測できるので$\mathcal{NUM} \subseteq \mathcal{NV}$ が分かる．冒頭の例の 2 番目の解法「極限でのプログラムの同定」に対応して，$R\text{-}\mathcal{TOTAL} \subset 2^R$ という同定可能性がある [26)]．$U \in R\text{-}\mathcal{TOTAL}$ の定義は適当なチューリング機械 M が存在し，すべての入力 x に対して $\varphi_{M(x)} \in R$，かつ

$$\forall u \in U \exists i\ \left(\varphi_i = u\ \&\ \lim_t M(u(0), \ldots, u(t)) = j\right) \qquad (5\cdot1\cdot1)$$

が成立としてある．各 t について $M(u(0), \cdots, u(t))$ は u のプログラムの仮説であり，集合 $\{t \in \mathbb{N}\ ;, M(u(0), \cdots, u(t)) \neq M(u(0), \cdots, u(t+1))\}$ の要素数が，仮説変更（試行錯誤）の回数である．初期値 $u(0), u(1), \cdots, u(t_0)$ を十分与えてから P で u を予測し続ける，というプログラムi_{t_0}は十分 t_0 が大きいと u を計算するプログラムである．そのことを利用すると，$\mathcal{NV} \subseteq R\text{-}\mathcal{TOTAL}$ が導ける．

実をいうと$\mathcal{NV}, \mathcal{NUM}, R\text{-}\mathcal{TOTAL}$ はいずれも$\{U \subseteq R\ ;\ U$ は計算量クラス$\}$に一致する [6)]．ここで計算量クラスとは勝手な $U \subseteq R$ で，ただし，適当な $F \in R$ が存在し U のどの関数も，ほとんどすべての入力 n に対して時間が $F(n)$ 以内で計算できるものである．したがって，$R \notin \mathcal{NV}$ である．

本稿では極限同定を次の 4 個の視点で考える．

1. 予測．例えば$\mathcal{NV}$ より強い同定可能性の概念として$\mathcal{NV}'$ [6),7)]，$\mathcal{NV}''$ [20)] がある．予測を通じて同定モデルを考えると，オンライン予測による同定モデル [17)] につながり，後者は PAC 学習の関連で重要である．
2. 同定対象の帰納的枚挙.「枚挙による同定法」で用いられる．正解をすべて汲まなく列挙する帰納的枚挙があれば，それを用いた「枚挙による同定法」は，同定を達成するまでに必要な例の個数が「最適」な同定法である [13)].
3. プログラムの同定．次の同定可能性の概念が関係している：$R\text{-}\mathcal{TOTAL}$，無矛盾な同定可能性たち$\mathcal{CONS}$ [5)]，$R\text{-}\mathcal{CONS}$ [15)]，$T\text{-}\mathcal{CONS}$ [24)]，$T\text{-}\mathcal{CONS}^{arb}$ [7)]，及び確実な同定可能性$\mathcal{REX}$ [7)]，説明による同定可能性$\mathcal{EX}$ [12),13)] など．なお，同定の効率は，仮説変更の回数などで計ることが多い．

4. 関数の抽象計算複雑度 [8]. 「現象を同程度うまく説明する仮説が複数あればより単純なものを選ぶべきである」という指針（科学哲学においてオッカムの剃刀（Occam's Razor）と呼ばれる）があるが，極限同定では，仮説であるところのプログラムの複雑度はそのプログラムの抽象計算複雑度が用いられる．例を説明するプログラムの複雑度の一様な限度があらかじめ分かっていれば，正解を含むかもしれないプログラムたちをその限度の範囲内で枚挙し検査すれば，同定ができる．

最後の 3 つの観点から様々な同定可能性が特徴づけられ [2),25)]，また，様々な学習現象の解明が試みられている．例えば，試行錯誤の回数が限られても，例たちからそれが従う法則の大まかなところは推定できるし，そのようにして大まかに推定できる法則は比較的多い．例全体 $u\in R$ が従う法則の大まかなところ，ただし，大まかさの限度は $a\in\mathbb{N}_*:=\{0<1<\cdots<n<\cdots<*\}$ とする，の説明の一つ i を提出し続けられる場合に，同定が達成できるとしよう．そのような同定達成基準を定式化すると，φ_i のグラフと u のそれとの対称差 [*1] $\#(\varphi_i\triangle u)$ の要素数が高々 a である i を提出し続けられる場合に同定が達成できる，ということになる．

この同定基準で仮説変更の回数の上界が $b\in\mathbb{N}_*$ である，プログラムの同定可能性 $\mathcal{EX}_b^a$ は次を満たす $U\subseteq R$ 全体と定義される：適当なチューリング機械 M が存在し，すべての $u\in U$ に対して

$$\#\{t\in\mathbb{N}\,;\ M(u(0),\ldots,u(t))\neq M(u(0),\ldots,u(t+1))\}\leq b;\ \text{かつ}$$
$$\exists j\left(\#(\varphi_j\triangle u)\leq a\ \&\ \lim_t M(u(0),\ldots,u(t))=j\right)$$

が成立することである. なお, 説明によるプログラムの同定可能性 $\mathcal{EX}$ は $\mathcal{EX}_*^0$ のことである．全順序集合 $\mathbb{N}_*$ の自分自身との自然な直積 $(\mathbb{N}_*)^2$ において，(a,b) に関する自然な順序と，$\mathcal{EX}_b^a$ の包含関係による (a,b) の順序が同値になり，$\mathcal{EX}_b^a$ が $a=*$ と $b=*$ で不連続になる [10)]．また，どの $\mathcal{EX}_b^a\,(a,b\in\mathbb{N}_*)$ も集合和で閉じてなくて，実際 $m\,(>1)$ 人で同定すると少なくとも m 倍精密に同定できる [23)]．

一方，同定法が，各時点で提出する仮説に，それまでに提示されたすべての例たちが，正確に反映されている場合，その同定法を無矛盾と言う．枚挙による同定法は明らかに無矛盾な同定法であり，実際，$\mathcal{NUM}\subsetneq\mathcal{CONS}$ である．無矛盾な同定法による同定可能性にも前述の変形 $\mathcal{CONS}$，$R\text{-}\mathcal{CONS}$，$T\text{-}\mathcal{CONS}$，$T\text{-}\mathcal{CONS}^{arb}$ があるが，相違 a で無矛盾に同定するという概念はない．というのは，無矛盾な同定法が出力する仮説の極限が計算する関数は同定目標と一致するからである．無矛盾性の条件は自然にみえるが，説明による学習可能性 $\mathcal{EX}$ の真部分クラスになっている．文献 2) では，それら無矛盾性の各条件を緩和し，仮説が例に $\delta\in\mathbb{N}$ ステップ以内で追従しているという条件，すなわち，最後の δ 個を除くすべての例を正確に仮説が反映している，という条件を導入し，それを δ 遅れの無矛盾性条件と呼んでいる．更に，無矛盾性の条件の別の緩和として一貫性の条件（再び δ 遅れも考える）を導入している．それは，各時点 t での仮説に t 番目（$t-\delta$ 番目）の例が正しく反映していると条件である．δ 遅れのある各々の無矛盾同定可能性たちを，帰納的枚挙により特徴づけ，また，抽象計算複雑度を用いて特徴づけている．また，これら同定可能性たちが $\mathcal{REX}$ の中で遅れ $\delta\in\mathbb{N}$ に関して真の階層になり，そして，δ 遅れのある一貫同定可能性が δ 遅れのある対応する無矛盾同定可能性と同じ

---

[1] A と B の対称差とは $A\triangle B:=(A\setminus B)\cup(B\setminus A)$ である．

強さであることを証明している．

近年はスキル習得の U 字学習曲線の必然性を極限同定の理論で説明する研究が帰納的枚挙の観点から行われている [9]．

なお，関数の極限におけるプログラムの同定に対応して，言語の極限における文法などの同定が盛んに研究されてきた．言語の完全提示（すなわち，その言語に属しているか否かのラベルを付けてすべての語を列挙したもの）により，どんな添字つき帰納的言語クラス$\{L_i\,;\,i\in\mathbb{N}\}$（すなわち L_i たちの特徴関数たちが一様に帰納的である）も同定可能であるが，言語の正提示（すなわち，その言語の列挙）では，正規言語全体すら同定可能ではない．実を言うと言語クラスが有限言語をすべて含み無限言語を一つでも含む場合，正提示では同定できない [13]．しかし，アングルイン（D. Angluin）は，添字付き帰納的言語クラスが正提示から同定可能である使いやすい必要十分条件と十分条件を与え，有用な添字付き帰納的言語クラスの一つであるパターン言語が正提示から同定可能であることを示し [3]，後の研究の大きな出発点となった．

彼女は，添字つき帰納的言語クラス$\mathcal{C}=\{L_i\,;\,i\in\mathbb{N}\}$の各言語 L_i に対して$\mathcal{C}$に関する有限証拠集合 $T_i\subseteq L_i$ という考えを導入した．それは雑に言うと有限次元ベクトル空間の有限基底に対応する．$\mathcal{C}$が正提示で同定可能であることと，$\mathcal{C}$の有限証拠集合族が帰納的に枚挙可能であることが同値であることを示した．また，文献 18）は$\mathcal{C}$の同定可能性の十分条件の一つとして有限の弾力性を導入した．それは，雑に言うと$\mathcal{C}$の中の言語たちの無限列に包含関係で真に大きくなっているものがないことである．一般に集合のクラス$\mathcal{C}_i$ がすべて有限の弾力性を持つ場合，$\mathcal{C}_1$と$\mathcal{C}_2$の要素ごとの集合和$\mathcal{C}_1\,\tilde{\cup}\,\mathcal{C}_2:=\{L_1\cup L_2\,;\,L_1\in\mathcal{C}_1,L_2\in\mathcal{C}_2\}$は Ramsey の定理により有限の弾力性を持つ [18]．また，文献 22）は，任意不定個の要素ごとの集合和（$\tilde{\cup}$）が，有限の弾力性の十分条件を調べ，ヒッグマン（Higmann）の補題と合わせることにより，各種のパターン言語のクラスの同定可能性を証明した．これは多言語環境での言語習得との関連が示唆される．また，ネーター（Noether）性に関連する計算可能な数学対象の族は，有限の弾力性を持つため，学習可能である [29]．

最後に数理論理学との関連を述べる．(1) における M が帰納的関数ならば，適当な帰納的汎関数 G が存在してすべての $u\in U$ に対して $\lim_t M\big(u(0),\cdots,u(t)\big)=\lim_t G(t,u)$ が成立する [12]．このことはシェーンフィールド（Shoenfield）の極限補題 [19),21)] を想起させる．すべての $A\subseteq\mathbb{N}$ に対して次の 3 つは同値である．

a. 適当な $g\in R$ が存在して$c_A=lim_t g(t,\bullet)$.

b. A の特徴関数 $c_A:\mathbb{N}\to\{0,1\}$は停止集合（空集合の飛躍）$\emptyset'$を神託として計算可能である．

c. A は Δ^0_2 述語である．すなわち，適当な計算可能述語 S, R が存在し，述語 A, $\forall x\exists y.R(x,y,\bullet\ )$ 及び $\exists x\forall y.S(x,y,\bullet\ )$ は，すべて同値である．

したがって，目標の述語が基本的な述語たちの「極限」を意図している場合（すなわち，(a)），その述語は無限への見通しを表す述語 $\emptyset'$から基本的な法則だけで推論できる（すなわち，(b)）が，その述語は結局，基本的述語を適当に限量化して得られる述語である（すなわち，(c)）．これに伴い，構成的（≈ 計算可能）算術であるハイティング算術（Heyting Arithmetic）に，Δ^0_2 論理式に関する排中律，$\big(\forall x\exists y.P(x,y,z\ )\leftrightarrow\exists x\forall y.Q(x,y,z)\big)\to\exists x\forall y.Q(x,y,z)\vee\exists x\forall y.\neg P(x,y,z)$ を加えると，いわば極限で同定可能（≈ 近似計算可能）な算術対象に関する算術体系が得られ，このようなことを反復すると，計算可能性の理論における算術階層

（Arithmetical Hierarchy）[19),21)] に対応して，ハイティング算術とペアノ算術（Peano Arithmetic）の間の算術の形式体系の階層ができる [1)]．逆に，証明論からの極限同定の理論への貢献としては，仮説変更のオーダータイプである順序数を評価するために，その同定可能性が証明できる形式体系の証明論的順序数との関係を用いる試みがあげられる [11)]．

本稿は極限同定の一部分しか紹介していないが，極限同定に関する成書としては文献 14)，帰納的関数と添字付き帰納的言語の極限同定に関する包括的なサーベイ論文としては文献 27) 及び 16)，日本語による入門書としては文献 28) がある．極限同定のサーベイ文献 4) は PAC 学習の観点が散見され興味深い．

■参考文献

1) Y. Akama, S. Berardi, S. Hayashi, and U. Kohlenbach：“An arithmetical hierarchy of the law of excluded middle and related principles,” H. Ganzinger (ed.), Proc. of the 19th Annual IEEE Symposium on Logic in Computer Science (LICS 2004), pp.192-201, IEEE Computer Society Press, 2004.
2) Y. Akama and T. Zeugmann：“Consistent and coherent learning with δ-delay,” Inf. Comput., vol.206, no.11, pp.1362-1374, 2008.
3) D. Angluin：“Inductive inference of formal languages from positive data,” Inform. Control, vol.45, no.2, pp.117-135, May 1980.
4) D. Angluin and C.H. Smith：“A survey of inductive inference: Theory and methods,” ACM Comput. Surv., vol.15, No.3, pp.237-269, Sep. 1983.
5) J.M. Barzdin：“Inductive inference of automata, functions and programs,” Amer. Math. Soc. Transl., pp.107-122, 1977.
6) J.M. Barzdin and R.V. Freivald：“Onthepredictionofgeneralrecursive functions,” Soviet Math. Doklady, vol.13, pp.1224-1228, 1972.
7) L. Blum and M. Blum：“Toward a mathematical theory of inductive inference,” Inform. Control, vol.28, no.2, pp.125-155, Jun. 1975.
8) M. Blum：“A machine-independent theory of the complexity of recursive functions,” Journal of the ACM, vol.14, no.2, pp.322-336, 1967.
9) L. Carlucci, S. Jain, E. Kinber, and F. Stephan：“Variations on u-shaped learning,” Inf. Comput., vol.204, no.8, pp.1264-1294, 2006.
10) J. Case and C. Smith：“Comparison of identification criteria for machine inductive inference,” Theoret. Comput. Sci., vol.25, no.2, pp.193-220, 1983.
11) M. de Brecht and A. Yamamoto：“Mind change complexity of inferring unbounded unions of pattern languages from positive data,” In Algorithmic Learning Theory, 17th International Conference, ALT 2006, Barcelona, Spain, Oct. 2006, Proceedings, vol.4264 of Lecture Notes in Artificial Intelligence, pp.124-138, Springer, Oct. 2006.
12) E.M. Gold：“Limiting recursion,” J. Symbolic Logic, vol.30, pp.28-48, 1965.
13) E.M. Gold：“Language identification in the limit,” Inform. Control, vol.10, no.5, pp.447-474, 1967.
14) S. Jain, D. Osherson, J.S. Royer, and A. Sharma：“Systems that Learn: An Introduction to Learning Theory, second edition,” MIT Press, Cambridge, Massachusetts, 1999.
15) K.P. Jantke and H,-R. Beick：“Combining postulates of naturalness in inductive inference,” Elektronische Informationsverarbeitung und Kybernetik, vol.17, no.8/9, pp.465-484, 1981.
16) S. Lange, T. Zeugmann, and S. Zilles：“Learning indexed families of recursive languages from positive data: A survey,” Theoretical Computer Science, vol.397, no.1-3, pp.194-232, 2008.
17) N. Littlestone：“Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm,” Machine Learning, vol.2, no.4, pp.285-318, 1988.
18) T. Motoki, T. Shinohara, and K. Wright：“The correct definition of finite elasticity: corrigendum to Identification of unions,” Proceedings of the Fourth Annual Workshop on Computational Learning Theory, p.375, San Mateo, CA, 1991. Morgan Kaufmann.
19) P.G. Odifreddi：“Classical Recursion Theory,” North Holland, Amsterdam, 1989.

20) K. Podnieks：“Comparing various concepts of function prediction, Part 1,” Theory of Algorithms and Programs, Latvian State University, Riga, vol.210, pp.68-81, 1974.
21) H. Jr. Rogers：“Theory of Recursive Functions and Effective Computability,” McGraw-Hill, 1967. Reprinted, MIT Press, 1987.
22) T. Shinohara and H. Arimura : “Inductive inference of unbounded unions of pattern languages from positive data,” Theoret. Comput. Sci., vol.241, no.1-2, pp.191-209, 2000, Special issue for ALT '96.
23) C. H. Smith：“The power of pluralism for automatic program synthesis,” J. ACM, vol.29, no.4, pp.1144-1165, 1982.
24) R. Wiehagen and W. Liepe：“Charakteristische Eigenschaften von erkennbaren Klassen rekursiver Funktionen,” Elektronische Informationsverarbeitung und Kybernetik, vol.12, no.8/9, pp.421-438, 1976.
25) R. Wiehagen：“Characterization problems in the theory of inductive inference,” In Automata, Languages and Programming, Fifth Colloquium, Udine, Italy, July 17-21, 1978, vol.62 of Lecture Notes in Compute Science, pp.494-508. Springer-Verlag, 1978.
26) R. Wiehagen : “Zur Theorie der Algorithmischen Erkennung,” Dissertation B, Humboldt-Universit ¨ at zu Berlin, 1978.
27) T. Zeugmann and S. Zilles：“Learning recursive functions: A survey,” Theoretical Computer Science, vol.397, no.1-3, pp.4-56, 2008.
28) 榊原康文, 小林　聡, 横森　貴 : “計算論的学習,” 培風館, 2001.
29) 徳永浩雄, 山本章博, 小林正典 : “人工知能における計算論的学習理論と Noether 環,” 数学, vol.59, no.3, pp.307-318, 2007.

3-5-2 ブースティングと PAC 学習

3-5-3 EXACT 学習

（執筆者：中村篤祥）［2016 年 1 月 受領］

EXACT 学習とは，PAC 学習に対比して使われる言葉である．未知の概念（関数）を何らかの情報から学習する問題において，PAC 学習では指定された精度の近似概念（関数）を求めるのが目標であるのに対し，EXACT 学習では等しい概念（関数）を求めることが要求される．その意味では極限同定もこの範疇に入りそうではあるが，一般に，質問による学習（Learning Via Queries）のみを指す．

概念学習では，何らかの条件を満たす定義域 X の部分集合のことを概念と呼び，概念の集合（X の部分集合族）を概念クラスという．概念学習とは，ある概念クラスに属している未知の概念 c の学習を意味するが，EXACT 学習の枠組みでは c の同定を目標とする．

例えば，n 以下の自然数の集合 $X = \{1, 2, \cdots, n\}$ において，k 以上数の集合 $c_k = \{x \in X : x \geqq k\}$ を概念と考え，$k = 1, 2, \cdots, n$ のすべての c_k から成る X の部分集合族を概念クラス C とする．C に属する概念 c を，「x は c に含まれているか？」という質問に対してオラクル（教師）により正しい解答（Yes/No）が与えられるという設定において，できるだけ少ない質問数で同定する問題を考えてみよう．

最悪ケースで評価した場合，この問題対する最適戦略は，それまでに得られた事例に無矛盾な仮説 $h \in C$ による（x が c に属すか属さないかの）予測が半分に割れる点 x を選ぶというもので，$R_{\mathrm{up}}(\log_2 n)$ 回の質問で同定可能である．ただし，R_{up} は切り上げを行う関数とする．このように未知の概念 c に対して「x は c に含まれているか？」と尋ねる質問は所属性質問（Membership Query）と呼ばれ，最もよく研究されている質問形態である．質問形態としてもっと直接的に，ある仮説 $h \in C$ に対して「$c = h$ か？」という質問を使う枠組みも考えられる．オラクルの答え

が Yes/No のみであれば，1 回の質問で 1 概念ずつしか可能性が減らないので上の例では同定に最悪 $n-1$ 回の質問が必要である．

そこでもう少し情報が得られるように，No の場合は反例が返ってくるという設定にすると，所属性質問のときと同様，$R_{\mathrm{up}}(\log_2 n)$ 回の質問で同定可能である．最適戦略において質問に使う仮説 h は，それまでに得られた反例に無矛盾な仮説すべてを使って多数決で予測する関数と等しい概念 c_k を選べばよい．このように，未知の概念 c に対し仮説 h を使って行われる「$c=h$ か？」という質問は，等価性質問（Equivalence Query）と呼ばれ，他の学習モデル（極限同定，PAC 学習，オンライン学習）との関連が強い質問形態である．また，オラクルにより反例が返ってくると仮定するのが普通であり，返ってこない場合は制限された等価性質問（Restricted Equivalence Query）といわれる．これまでに様々な質問形態（部分性質問（Subset Query），包含性質問（Superset Query），非交差性質問（Disjointness Query），全体性質問（Exhaustiveness Query）など [2)]）が提案されてはいるものの，質問による学習ではこの 2 つの質問を使った研究が大半を占める．

質問による学習において草分け的な存在である Angluin は，所属性質問と等価性質問に答えられる教師を極小最適教師（Minimally Adequate Teacher）と呼んだ．上の例では所属性質問のみでも等価性質問のみでも効率的な学習が可能であったが，一般意はどちらの質問も重要であり，実際，片方の質問のみでは効率的な学習が不可能な概念クラスが存在する．決定性有限オートマトン（DFA）により受理される正則言語（Regular Language）のクラス及び単調 DNF（Disjunctive Normal Form）で表現される $\{0, 1\}^n$ 上の概念クラスはその代表格と言える．どちらのクラスも両方の質問を使った多項式時間アルゴリズム [1),2)] が存在するが，片方のみの質問を使った多項式時間アルゴリズムが存在しない [2),3)]．

所属性質問と等価性質問に答えられるオラクルの存在を仮定することは自然であろうか？所属性質問は，概念 c のメンバーシップ関数 $\mathbf{1}_c$ の関数値（$x \in c$ ならば $\mathbf{1}_c(x)=1$，そうでないなら $\mathbf{1}_c(x)=0$）を尋ねる質問とみなせるが，最適実験計画などでは入力 x に対する出力 $f(x)$ が得られるという枠組みで関数 f を推定する問題を扱っており自然な仮定といえる．それに対し，等価性質問に対して正しく答えられるオラクルの存在は無理な仮定であるが，ある信頼度で正しく答えられるオラクルであれば，（定義域 X 上の確率分布に従った）ランダムサンプリングによる等価性の統計的検定により代用できるため，それほど不自然ではない．PAC 学習は定義域 X 上の確率分布に従って得られた点 x に対する事例（$x, f(x)$）をくれるオラクルの存在を仮定するので，等価性質問を使って多項式時間 EXACT 学習可能であれば PAC 学習可能であると言える [2)]．

このように等価性質問による EXACT 学習は，PAC 学習と関連があるが，その他の学習モデルとも関連する．極限同定モデルでは，定義域 X のすべての要素が一度は必ず現れる事例の無限列 $(x_1, \mathbf{1}_c(x_1)), (x_2, \mathbf{1}_c(x_2)), \cdots$ に対し学習アルゴリズムは仮説の無限列 $h_1, h_2, \cdots$ を出力する．学習アルゴリズムが概念 c を極限において同定するとは，アルゴリズムから出力される仮説はある時点 m からすべて c と等しくなる．つまり，ある数 m が存在し，$h_m = h_{m+1} = \cdots = c$ が成り立つことをいう．時点 i に対し $h_i \neq h_{i+1}$ となることをマインドチェンジ（Mind Change）と呼ぶが，高々 k 回の等価性質問により概念 c を EXACT 学習するアルゴリズム A を用いて，高々 k 回のマインドチェンジにより c を極限同定するアルゴリズムを構成することができる．構成は簡単で，A の等価性質問に使う仮説 h を，極限同定のアルゴリズムが出力する仮説の列とし

て使うだけでよい．ただし，事例が仮説 h に無矛盾な間は同じ仮説を使い，矛盾した場合，つまり $h(x_i)\neq\mathbf{1}_c(x_i)$ となった場合は，x_i を等価性質問の反例として A に返し，次の等価性質問に使う仮説 h' を以降の出力仮説として使うということを繰り返す．一般性を失うことなく，A の最後の等価性質問に対する答えは必ず yes と仮定してもよいことに注意されたい．

また，等価性質問による EXACT 学習は，オンライン学習モデルとも関係ある．オンライン学習モデルでは，定義域 X の点 x がオラクルから与えられ，それに対する予測値 y' を学習アルゴリズムが返すと，オラクルから正しい答え y が与えられるというサイクルを繰り返すモデルである．この枠組みは極限同定と少し似ているが，目的が損失関数により与えられる損失の累積を最小にすることであるという点が大きく異なる．損失関数は実際の値との違いから生ずる予測値による損失を測るものであるが，最も単純なものは予測が間違っていたとき $(y'\neq y)$ に 1 でその他の場合は 0 という関数であり，このとき累積損失は予測が間違った回数となる．

上述の極限同定のアルゴリズムの構成と同様な方法で，高々 k 回の等価性質問により概念 c を EXACT 学習するアルゴリズム A を用いて，高々 k 回予測を間違えるオンライン学習アルゴリズムを構成することができる[4]．ただし，オンライン学習の場合は，等価性質問に使う仮説 h を直接オラクルに渡すのではなく，与えられた点 x に対する仮説を使った予測値 $h(x)$ を渡すところが異なる．

所属性質問は，EXACT 学習のみならず PAC 学習の学習能力向上のためにも用いられる．実際，決定性有限オートマトンや単調 DNF などは，所属性質問を使わない場合に PAC 学習困難であることを示す証拠が示されているが，所属性質問を用いた PAC 学習アルゴリズムは存在する．これは，ある確率分布に従ったランダムサンプルしかもらえない受動学習に対し，それに加えて欲しい点のサンプルを積極的に取得する能動学習が効果的であることを示す結果と言える．

EXACT 学習に関して，更に詳しく知りたい方は文献 5) の解説記事を参照されたい．

■参考文献

1) D. Angluin :“Learning Regular Sets from Queries and Counter-Examples,” Information and Computation, vol.75, pp.87-106, 1987.
2) D. Angluin :“Queries and Concept Learning,” Machine Learning, vol.2, pp.319-342, 1988.
3) D. Angluin :“Negative Results for Equivalence Queries,” Machine Learning, vol.5, pp.121-150, 1990.
4) N. Littlestone :“Learning Quickly When Irrelevant Attributes Abound: A New Linear-Threshold Algorithm,” Machine Learning, vol.2, pp.285-318, 1987.
5) 榊原康文，西野哲朗 :“EXACT 学習―質問からの概念学習―,” 情報処理, vol.32, no.3, pp.246-256, 1991.

3-5-4 オンライン学習