Vol.37 No.10　　情報処理　　Oct. 1996

解説

ゲノム情報

6. ゲノムデータからの知識発見†

宮野　悟††

1. はじめに

「発見」をすることは科学的研究における最も重要な活動である．その発見は新しい彗星の発見であったり，ケプラーの法則のように観測データから導き出された物理法則であったりする．人間の知的活動として位置づけられてきたこうした知識発見は，最近注目されているデータマイニングのように，人間に代わって計算機上のシステムが行うことが強く期待されている．

ゲノム研究の進展により核酸配列やその翻訳であるアミノ酸配列データが大量に生産されてくるなかで，ゲノムデータからの知識発見とその利用を計算機によって支援することが重要な課題となっている．そして，それは，熟練した人間による職人芸的な発見的な方式ではなく，産業革命により家内製手工業から工場生産方式に転換したように，広い範囲に網を打つような自動的な知識発見方式による mass discovery を目指さなければならないであろう．

本稿では，こうした状況に対応するために著者らのグループが取り組んできた確率的近似学習（Probably Approximately Correct Learning, PAC 学習）[10]とよばれる学習方式に基づいた知識発見の考え方とゲノムデータからの知識発見を行うためのシステムについて解説する．

2. PAC 学習による知識発見

PAC 学習[10]は，ある概念を学習することが本質的にどれほどの数のサンプルや時間を必要とするかということをテーマとして，これまで学習可能性などについて多くの理論的研究が展開されてきた．ここでは，PAC 学習のパラダイムによる正負の文字列データからの知識発見の考え方について述べる．

文字列データはアルファベット Σ を用いて表現されているとしよう．このとき有限文字列全体からなる集合 Σ^* は，生起する可能性のあるデータ全体と考えられ，この集合を**学習領域**とよんでいる．一般に，学習領域 Σ^* の部分集合を**概念**とよび，概念の族を**概念クラス**とよんでいる．概念 c に対して，文字列 x は，c に属するとき概念 c の**正の例**とよばれ，c に属さないとき**負の例**とよばれている．概念クラス C に属する任意の未知の概念 c をいくつかの正の例と負の例から同定するプロセスを**例からの概念学習**とよんでいる．

知識発見の観点から概念学習を用いるときの問題を考えよう．DNA 配列データからの知識発見を目的とするときには，アルファベットを $\Sigma = \{a,c,g,t\}$ のように定義するばかりでなく，学習の効率化や知識の鮮明化のために，他のアルファベットに変換することも考えられている[1),2)]．図-1 のような有限個の正負の例がデータとして与えられたとき，アルファベット Σ 上の「言葉」で表現された「知識」を発見することが要求されるわけであるが，通常，そのデータに対して，どのような「言葉」を用い，どのように知識を表現すればよいかは知識発見の前提としてわかっているわけではない．これは，学習アルゴリズムを用いて知識発見に取り組むときに直面する最も困難な問題の1つである．そして，この問題を解決することは，その概念が属していると思われる概念クラスとその表現法を定義することに対応する．したがってゲノムデータを扱う場合，ゲノムについての知識を仮定することになり，それに応じたデータの抽象化が重要な鍵となる．

概念クラス C の各概念を表現するためのアル

† Knowledge Discovery from Genome Data by Satoru MIYANO (The University of Tokyo, The Institute of Medical Science, Human Genome Center).

†† 東京大学 医科学研究所 ヒトゲノム解析センター

ファベットをΛとする．概念クラスCの**表現**とは，Cの各概念cにその概念を表す名前の集合$R(c) \neq 0$を対応させる関数$R: C \rightarrow 2^{\Lambda^*}$として与えられる．ただし，2つの異なる概念は異なる名前で表現されていなければならない．この名前が概念についての知識を表現することになる．以後，概念クラスCとその表現Rの組(C,R)を概念クラスとよぶことにする．

次の(a)，(b)の条件を満たすランダムアルゴリズムAを概念クラス(C,R)に対する学習アルゴリズムということにする[5]．

(a) Aは，**文字列長パラメータ**とよばれる自然数$n \geq 1$，**誤差パラメータ**とよばれる実数ϵ，および**信頼度パラメータ**とよばれる実数δ $(0 < \epsilon, \delta < 1)$を入力とする．

(b) AにはEXAMPLE callの状態があり，この状態でAは例(x,a) $(x \in \Sigma^*, a \in \{0,1\})$を受け取る．$c$を概念クラス$C$の任意の概念とする．EXAMPLE callの状態において，長さがn以下のどのような例$(x,c(x))$がAに返されてもAのすべての計算は停止し，Cに属する概念hの名前$\nu(h) \in R(h)$を出力する．

さらに，学習アルゴリズムAが次の(c)，(d)の条件を満たすとき，概念クラス(C, R)に対する**多項式時間PAC学習アルゴリズム**であるという．また，このような多項式時間PAC学習アルゴリズムが存在するとき，(C,R)は**多項式時間PAC学習可能**[5]であるという．

(c) Aの計算時間は$1/\epsilon$，$1/\delta$，nについての多項式$p(1/\epsilon, 1/\delta, n)$で押えられる．

(d) Pを$\Sigma^{\leq n} = \{x \in \Sigma^* \mid |x| \leq n\}$上の任意の確率分布とする．EXAMPLE callの状態で，概念cの例$(x,c(x))$が確率$P(x)$でAに返されるとするとき，Aは$1-\delta$以上の確率で$\Sigma_{c(x) \neq h(x)} P(x) < \epsilon$（すなわち誤差が$\epsilon$未満）となる概念$h$の名前$\nu(h)$を出力する．

この定義の(d)は，次のように解釈できる：「学習アルゴリズムは，未知の確率分布に従って生じる例に対応できなければならないであろう．そのとき，その未知の確率分布のもとで，とても低い確率でしか現れないような例もあるだろうから，そのような例も含めてすべての例を正しく判別できるような仮説を出すことを学習アルゴリズムに要求することは不合理であろう．したがって，仮説hが概念cとϵ以下しかくい違わなければよい仮説とみなそう．また，未知の確率分布によって例が学習アルゴリズムに与えられるが，その例が，きわめて特殊な場合もあろう．このような列が与えられたときにもよい仮説を出力することを要求することは不合理であるので，出力される仮説がϵ以上の誤差をもつことを許そう．しかし，そのような誤差の大きい仮説が出力される確率は高々δとしよう．」

正の例：

```
CACAGGAGGCCAGCGAGCAGGTCTGTTCCAAGGGCCTTCGAGCCAGTCTG
GGAGGCGGAGGACCTGCAGGGTGAGCCCCACCGCCCCTCCGTGCCCCCGC
ACGTCCTTCCCCAGGAGCCGGTGAGAAGCGCAGTCGGGGGCACGGGGATG
GCTGGTCACATTCCTGGCAGGTATGGGGCGGGGCTTGCTCGGTTTTCCCC
CAGGTCACCCAGGAACTGACGTGAGTGTCCCCATCCCGGCCCTTGACCCT
GGACAACAAAACCTTCAGCGGTAAGAGAGGGCCAAGCTCAGAGACCACAG
```

負の例：

```
AAGCTGGAGGCATCACGCTACCTGACTTCAAACTATACTACAAGGCTACA
CTGATTGGTCCAGCTTAGTCCATGTCCCTACCCTGAACAGGGGCATGGGG
TGAATTCTCTCCACATTATTATTATTATTTTTTGAGACAGTCTTGCTCTG
GAAAAAGGAAATATCTTCCGTTCAAAACTGGAAATAAGCTTTCTGAGAAA
TGGTGGAAGCCCTCTACCTGGTGTGCGGGGAGCGAGGTTTCTTCTACGCA
CAATATCTTCTTCTCCCCAGTGAGCATCGCTACAGCCTTTGCAATGCTCT
```

図-1 正の例はエクソン・イントロンのスプライシング部位を含む配列，負の例はイントロンの部分配列．

概念クラスとその表現を定義し，その多項式時間PAC学習アルゴリズムを作ることができれば，データとして与えられている正負の文字列データを未知の確率分布に従って生じている例とみなし，その例をEXAMPLE callの際の例として学習アルゴリズムに与えることにより仮説を得ることができる．そして，その仮説を表現している仮説の名前として，そのデータについての知識を獲得することができる．通常，PAC学習アルゴリズムは，与えられた例に矛盾しない仮説をみつけるアルゴリズムとして実現されている．

以上が，PAC学習アルゴリズムによる正負の文字列データからの知識発見の方法である．現実的には多項式時間という制約のほか，サンプルとして用いる例の個数や仮説表現の本質的大きさなどの障害があり，それを克服する必要がある．また，計算機実験によって，どうしてもよい仮説と知識が得られない場合は概念クラス自身を見直す

ことになる．概念クラスの反駁についての理論も展開されている[4]．

3. パターン言語と文字列上の論理プログラム

文字列データから仮説を作るためには，何らかのデータの見方が必要となる．有限アルファベットΣと変数集合 $X=\{x_1,x_2,\cdots\}$の文字列$\pi\in(\Sigma\cup X)^+$をΣ上の**パターン**という．パターン $\pi=\alpha_0 y_1\cdots y_n\alpha_n(y_i\in X,\ \alpha_j\in\Sigma^*)$に現れている変数に$\Sigma^+$の文字列を代入することで$\Sigma^+$の部分集合 $L(\pi)$ を定義できる．これを**パターン言語**とよんでいる．このパターンを通して文字列データを見ることにより，知識の抽出を行うことができる．

さらにこのパターンを論理プログラムにおける項のように扱い，文字列上の論理プログラムを考えることで，より高次の知識表現が可能となる．

ここでは **elementary formal system**（EFS と略す）という文字列上の論理プログラムの定義を与え，その PAC 学習可能性と知識発見への応用について述べる．

論理プログラムと同様に，p を述語記号とするとき，$p(\pi_1,\cdots,\pi_n)$ の形の式を**アトム**という．ただし $\pi_1,\cdots,\pi_n$ はパターンである．A および $A_1,\cdots,A_t$ をアトムとするとき，$A\leftarrow A_1,\cdots,A_t$ の形の式を**確定節**という．A を確定節の**頭部**，$A_1,\cdots A_t$ を**本体**という．EFS は，このような確定節の有限集合 S として定義される論理プログラムである．

確定節 C が EFS S から**証明可能**であるという概念を論理プログラムと同様に定義することができる．引数 1 の述語 p と EFS S に対し，$L(S,p)=\{w\in\Sigma^+\mid p(w)\leftarrow$は S から証明可能$\}$と定義する．パターン π に対して，$S=\{p(\pi)\leftarrow\}$とするとき，$L(\pi)=L(S,p)$である．EFS の記述能力や意味論についての基礎的な結果も知られている[3]．

確定節

$$q(\pi_1,\cdots,\pi_n)\leftarrow$$
$$q_1(\pi_1^1,\cdots,\pi_{r_1}^1),\cdots,q_t(\pi_1^t,\cdots,\pi_{r_t}^t)$$

を考えよう．本体に現れているパターン π_j^i が，頭部のいずれかのパターン π_i の部分文字列になっているとき**継承的**(hereditary)であるという．たとえば，確定節 $p(a x_1 b c)\leftarrow q(a x_1), r(x_1 b)$ は継承的であるが，$q(b x_1)\leftarrow q(a x_1)$ は継承的ではない．

文字列	クラス
FDCLE	P
GKFHP	N
SGDE	P
EDFIYR	P
MDIWAQ	N
ERLK	N

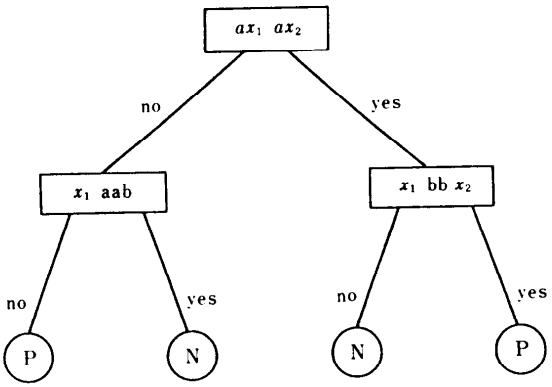
(a)

文字列	クラス
abbab	P
ababa	N
aabb	P
bbaaab	P
abaaab	N
bbab	N

(b)

Σ	ACDEFGHIKLMNPQRSTVWY
Γ	abbbaababaababbaaaaa

インデックス化



正規パターン上の決定木

図-2 (a)の正負の例はアルファベットのインデックス化により(b)に変換され正規パターン上の決定木によって正確に分類される．

自然数 $m, k, t, r\geq 0$ に対し，以下の条件を満たす m 個以下の継承的な確定節からなる EFS によって定義される言語のクラスを H-EFS(m,k,t,r) と表す．

(a) 頭部の変数の出現回数は k 以下である．

(b) 本体のアトムの個数は t 以下である．

(c) 述語記号の引数は r 以下である．

例 1 次の EFS S の確定節は継承的であり，$L(S,p)=\{a^n b^n c^n\mid n\geq 1\}\in$ H-EFS$(3,3,1,3)$である．

$$S=\left\{\begin{array}{l} p(x_1x_2x_3)\leftarrow q(x_1,x_2,x_3)\\ q(ax_1,bx_2,cx_3)\leftarrow q(x_1,x_2,x_3)\\ q(a,b,c)\leftarrow \end{array}\right\}$$

正負の文字列の集合 P と N に対して，$P\subseteq L(S,p)$ と $N\cap L(S,p)=0$ を満たす EFS(S,p) は，どのような事実と規則が正の例を導き出し，また負の例を排除しているのかの論理構造を与えるため，データについての知識を抽出するために有用である．EFS の多項式時間 PAC 学習可能性については次のことがわかっている[1]．

定理 1

(1) H-EFS(m,k,t,r) は多項式時間 PAC 学習可能である．

(2) FU-H-EFS$(m,k,t,r)=\{c_1\cup\cdots\cup c_n \mid c_i\in$ H-EFS$(m,k,t,r)\}$ は多項式時間 PAC 学習可能である．

定理 1 の PAC 学習アルゴリズムは，m, $k\geq 10$ のときには現実的ではない時間を要するものである．これは実行時間は多項式的とはいえ，定数 m と k に関しては指数的であることによる．したがって，現実的な学習のためには，変数の出現数 k と確定節の個数 m のどちらも小さく制限する必要がある．

現実的な対処の方法として，パターンにおける変数の出現を 10 以下におさえ，そうしたパターンの有限集合として定義される EFS に制限した概念クラスが考えられている[1]．また定理 1(2) の学習アルゴリズムとして，最小集合被覆の近似アルゴリズムを応用すると，膜タンパク質の膜貫通領域のアミノ酸列データからの知識発見に有効であることが確認されている[1]．

4. 正規パターン上の決定木とインデックス化

パターン π において，同じ変数が重複して現れていないものを**正規パターン**(regular pattern)という．すなわち，$x_1,\cdots,x_n$ を互いに異なる変数として，$\pi=\alpha_0x_1\alpha_1\cdots x_n\alpha_n(\alpha_i\in\Sigma^*)$ と表現できるパターンである．ただし本稿では変数 x_i には Σ^+ の代わりに Σ^* の文字列を代入するものとしている．正規パターンはアミノ酸配列や DNA 配列のモチーフを一般化した概念である．**正規パターン上の決定木**は，各ノードにこうした正規パターンを判定規則に用いた決定木であり(**図-2**)，与えられた文字列がどのクラスに属するかを決める手続きを記述したものである．その各ノードは，クラスの名前(この場合 N または P)もしくは正規パターンをノードのラベルとしてもっている．正規パターンをラベルとしてもつノードでは，与えられた記号列がその正規パターンとマッチするかがテストされ，その結果(yes または no)により右または左に分岐していく．このテストと分岐は，根で始まり，葉にたどり着くまで行われる．そして到達した葉のラベル(N または P)がその列の属するクラス名を答えることになる．正規パターン上の決定木 T によってクラス P と判定される文字列の集合を $L(T)$ と書く．

膜タンパク質やシグナル配列のアミノ酸配列データからの知識発見において，アミノ酸をその親水度により 3 つのカテゴリーに分類し，20 種のアミノ酸をそのカテゴリーにより，記号 *, +, -に変換すると，知識発見が容易になることが知られている[1),2),7)]．アミノ酸配列からなる正負の文字列データをこのように 3 つの文字からなる文字列に変換すると，一般には，正の例と負の例にオーバラップが生じる可能性がある．しかし，膜貫通領域データについて，文献 1)，2)では，変換後も正負のデータにオーバラップが起こらないことが観察されている．この観察にもとづいて，**アルファベットのインデックス化**の概念が次のように定義されている[7)]．

P と N を互いに共通部分をもたないアルファベット Σ 上の文字列の集合とする．Γ を $|\Sigma|>|\Gamma|$ を満たすアルファベットとする．P と N に関する Γ による Σ の**インデックス化**(indexing) ψ とは，変換 $\psi:\Sigma\rightarrow\Gamma$ で $\psi(P)\cap\psi(N)=0$ を満たすものである．ここで $\psi:\Sigma^*\rightarrow\Gamma^*$ は $\psi(a_1\cdots a_n)=\psi(a_1)\cdots\psi(a_n)$ $(a_1,\cdots,a_n\in\Sigma)$ で定義されるものである．

すなわち，アルファベットのインデックス化とは，入力文字列の正負の情報を欠落させることなく，あらかじめ設定された，より少ない数の文字に変換する対応づけである．こうしたアミノ酸の分類は分子生物学において重要な意味をもっている．

知識発見の対象となっている文字列データが，アルファベット Σ で記述されているとき，より文

字数の少ないアルファベットΓへの変換ψ：Σ→Γ とΓ上の正規パターンを内部ノードのラベルとする決定木Tの組(T,ψ)を概念の表現とする概念クラスを考える．組(T,ψ)によって表現される概念は$L(T,\psi)=\{w\in\Sigma^{+}\mid \Psi(w)\in L(T)\}$である．

こうした概念表現を用いるとき，どのような概念クラスが多項式時間PAC学習可能となるのであろうか．一般にアルファベットのインデックス化をみつけることは計算量的に困難であることがわかっている[8]．一方，正規パターン上の決定木のPAC学習可能性については次の定理が成り立つ[2]．

定理2

正規パターンにk個以下の変数しか現れておらず深さがd以下の正規パターン上の決定木によって定義される言語のクラスをDTRP(d,k)とする．このとき，DTRP(d,k)は任意のd, $k\geq 1$に対して多項式時間PAC学習可能である．

決定木において，定数dやkによる制約を外すと多項式時間PAC学習が困難になることがわかっている．この定理の証明に用いられている学習アルゴリズムは，仮説を枚挙するアルゴリズムであるため，実用には不適切である．また，決定木の深さdや変数の出現回数の上限kは，与えられる文字列データから決まるものではなく未知のものであるため，実際の知識発見においては，上の定理がそのまま有効なわけではない．

こうした問題に現実的対処するために，アルファベットのインデックス化には局所探索法を用い，決定木の学習にはQuinlan[6]の情報量の計算による決定木作成法が用いられている[7]．この方式を実働化したシステムBONSAI[7]は，正負の文字列集合の組(POS, NEG)が入力として与えられると，仮説として正規パターン上の決定木Tとアルファベットのインデックス化ψの組(T,ψ)を出力する．ただしインデックス化に使うアルファベットはあらかじめ決められたものを用いている．仮説(T,ψ)の精度は，正しく認識されている正の例の比率pと負の例の比率nを用いて$\sqrt{p\times n}$で計られている．PIRデータベースを使った膜貫通領域の同定の実験で，このシステムはKyteとDoolittleの親水度にほぼ対応したインデックス化と非常にコンパクトな決定木を発見することに成功している．

知識発見の対象となる文字列データがノイズを含んでいたり多様な種類の配列から構成されると考えられるとき，これらの文字列データをいくつかのクラスに分類すると同時にこれらの各クラスのデータを高い精度で説明する仮説を発見することを目的に開発されたシステムにBONSAI Gardenがある[9]．このシステムは，前章で触れた正負の文字列データからの知識発見システムBONSAI[7]とそのコーディネータであるプログラム*Gardener*とからなり，「サイズの大きな仮説を作るシステムはデータを失う」という単純な原理に基づいて文字列データの分類と知識発見を行っている．

5. おわりに

知識発見はゲノム研究の支援の重要な位置をしめ，精度のよい遺伝子発見システムや機能予測システムの開発は，ゲノム研究の進捗に大きく影響するものである．こうしたゲノム情報の展開は，情報科学の新たな分野をつくり出しつつあり，アメリカにおいてはこうした研究のできる人材の需要が増していることが報告されている．

参考文献

1) Arikawa, S., Kuhara, S., Miyano, S., Shinohara, A. and Shinohara, T. : A Learning Algorithm for Elementary Formal Systems and its Experiments on Identification of Transmembrane Domains, Proc. 25th Hawaii International Conference on System Sciences, pp. 675-684 (1992).

2) Arikawa, S., Miyano, S., Mukouchi, Y., Shinohara, A. and Shinohara, T. : A Machine Discovery from Amino Acid Sequences by Decision Trees Over Regular Patterns, New Generation Computing, Vol. 11, pp. 361-375 (1993).

3) Arikawa, S., Shinohara, T. and Yamamoto, A. : Learning Elementary Formal Systems, Theoretical Computer Science, Vol. 95, pp. 97-113 (1992).

4) Matsumoto, S. and Shinohara, A. : Refutably Probably Approximately Correct Learning, Proc. 5th International Workshop on Algorithmic Learning Theory, Springer-Verlag, pp. 469-483 (1994).

5) Natarajan, B.K.: Machine Learning, 217 p., Morgan Kaufmann, San Mateo, California (1991).

6) Quinlan, J.R.: Induction on Decision Trees, Machine

Learning, Vol. 1, pp. 81-106 (1986).
7) Shimozono, S., Shinohara, A., Shinohara, T., Miyano, S., Kuhara, S. and Arikawa, S. : Knowledge Acquisition from Amino Acid Sequences by Machine Learning System BONSAI, Trans. Information Processing Society of Japan, Vol. 35, pp. 2009-2018 (1994).
8) Shimozono, S. and Miyano, S. : Complexity of Alphabet Indexing, IEICE Transactions on Information and Systems, Vol. E78-D, No. 1, pp. 13-18 (1995).
9) Shoudai, T., Lappe, M., Miyano, S., Shinohara, A., Okazaki, T., Arikawa, S., Uchida, T., Shimozono, S., Shinohara, T. and Kuhara, S. : BONSAI Garden: Parallel Knowledge Discovery System for Amino Acid Sequences, Proc. 3rd International Conference on Intelligent Systems for Molecular Biology, AAA Press, pp. 359-366 (1995).
10) Valiant, L. : A Theory of the Learnable, Commun. ACM, Vol. 27, pp. 1134-1142 (1984).

(平成8年7月17日受付)

宮野 悟 (正会員)

1954年生．1977年九州大学理学部数学科卒業．1979年同大学院理学研究科数学専攻修士課程修了，1979年同専攻博士課程中退．同年より同大学理学部附属基礎情報学研究施設助手．1981年同大学理学部数学科助手．1987年同大学理学部附属基礎情報学研究施設助教授，1993年同研究施設教授．1996年東京大学医科学研究所ヒトゲノム解析センター教授，現在に至る．理学博士．ゲノム情報，発見科学，計算量理論に興味をもつ．人工知能学会会員．