クラスを規定するパターンの構造解析とその応用に関する研究

工藤峰一

目　次

第1章　序章

1．1．　本論文の目的

近年の計算機の著しい発達につれ、人間と比較して計算面での計算機の優位性は疑うべくもないのに対し、推論、判断などの面やパターン認識では未だ人間の方が精度よく、効率よく行う事ができる。これは計算の量の問題ではなく、質の問題であり、現在の音声認識あるいは文字認識における認識精度や認識時間にも一つの限界として現れている。この点に関して、基礎的なパターンあるいはクラスに関する扱いを再検討してみる事は有効な試みと思われる。

特に、パターン認識に関する研究は認識機構全体ではなく、部分の各処理毎に性能向上や効率化を議論しているものが殆どで、加えて、問題毎に経験的な知識を利用しているのが多数見られる。しかし、主目的である認識精度の向上に限っても、訓練サンプル集合や特徴集合、識別規則の型、各々が互に作用して機構全体の精度に影響しているのは明白であるから、各処理の関係を有機的に考察する必要がある。

本論文はこれらの点を考慮して、本質的にクラスはそのクラスに属するパターン間の共通性と他クラスのパターンとの相違性の二点において規定されるものと考え、逆にその考えに基いて従来の識別方法を見直した場合に幾つかの特性が明らかになる事を示す。また、明らかになった特性を踏まえて幾つかの改善が計れる事を述べる。さらに、応用面での成果を論ずる。

1．2．　本論文の構成

本論文の構成は以下の通りである。また、各章の関係を章末に示す。

第１章では、本論文の背景ならびに目的、構成について示す。

第２章では、一般的なパターン認識過程の全体像を示し、各処理毎の問題点を挙げる。また、各処理を有機的に結び付けて考える必要性を示し、それに対して本論文がどの様に対処するかについて概略を述べる。

第３章では、クラスを規定するパターンの性質を表現する一つの型として、文字列表現したパターンがある文法規則から生成されると考える構文的パターン認識の立場に立って、有限のサンプルから文法規則を正規文法に限って推論する問題を議論する。また、従来の代表的な手法を共通のアルゴリズム上で再構成する事により手法間の相違点を明確にする。加えて、明らかになった従来手法の性質を考慮して、欠点を補い、拡張となる手法を提案する。

第４章では、第３章で提案した方法を実際の遺伝子における信号配列の解析に応用する。遺伝子配列中にある特定の位置（スプライスサイト）を定める信号がその位置の周辺配列にあるとして、周辺配列の多数サンプルから規則の抽出を試みる。

第５章では、特徴が二値の場合に、一つのクラスの訓練サンプル集合中に排他性及び極大性を満足する部分集合（部分クラス）を発見する効率的なアルゴリズムについて述べる。

第６章では、質量混在の特徴を有するパターンが特徴空間の点として表現される場合に、第５章の部分クラスを発見する方法を論ずる。特徴空間において、部分クラスを各特徴軸に平行または垂直な超区間として確保するもので、最終的に識別規則は部分クラスの重要度を考慮して構成される。また、部分クラス全体がクラスの構造を十分反映すると考え、クラスの構造を定量的に表現する幾つかの指標を示し、それらの指標と識別率との関係を考察する。また、それらの指標を用いて、特徴数に線形なオーダーの計算量で済む特徴抽出のアルゴリズムを提案する。最後に、識別機構全体としての性能向上の目的にクラス構造の指標を用いる事を提案し、現在の環境（特徴集合、訓練サンプル集合）の十分性を議論する。

第７章では、本論文の総括として、各章毎にこれまでの問題点とそれに対する本論文の成果を要約する。また、全体を通しての問題点ならびに今後の課題を述べる。

第２章　パターンの性質ならびにパターン識別

２．１．　序論

本章では、パターン認識問題の定式化を行い、認識手法の概略を特にサンプル数と特徴数との関係に主眼をおいて述べる。また、一般的なパターン認識過程の枠組みを考えた時、どの様な問題点が残されているかを検討し、それらに対する対処法を論ずる。

２．２．　パターンの性質とクラス

パターンは常にクラスとの関係において考察される。パターンが通常物理的性質を有する実体であるのに対し、クラスはパターンと明らかに次元を異にするむしろ”概念”的な存在である。また、クラスの解釈として、パターンに先んじて存在すると考える場合と、ある共通性を有するパターン集合の概念化として生成されると考える場合の二通りの解釈が可能である。前者の解釈が通常のパターン認識問題で、この場合パターンはすでに存在するクラスから発生したのであるから、未知パターンに正当なクラスを割当てる問題を考える事ができる。後者の解釈は通常クラスタリングと呼ばれる問題である。本研究においては全体を通して、前者の立場を取る。パターン認識の大きな分類として、統計的決定理論的パターン認識および構文的パターン認識が挙げられる。前者は特徴空間上でパターンを一つの点あるいはベクトルとして捉え、大まかに言えば空間を分割する事によって識別規則を構成する。大部分の問題はこの形で取り扱う事ができ、また自然である。それに対し、後者はパターンが形状や構造を表現する場合に有効で、通常パターンは文字列として表現され、識別規則は文字列の生成規則として構成される。

特徴空間のパターン識別に関して、さらにクラスの統計的な性質の知識程度に応じて、

１）分布が既知の場合

２）分布型が既知の場合

３）分布が未知の場合

が考えられる。１）の場合は実際問題としてまずあり得ない。２）はパラメトリックな分布型を通常仮定し、有限のサンプル集合からそれらのパラメータを推定する。この場合は理論的に扱いやすく、特に正規分布の場合などは詳しく研究されており、その場合の識別手法はパラメトリックな手法と呼ばれる。しかし、実際の大部分のパターン認識問題では分布は未知であり、ノンパラメトリックな手法が有効となる。ノンパラメトリックな手法の代表的なものには：

線形、非線形、区分的線形の各識別関数

ポテンシャル関数法

ＮＮ（Nearest-Neighbor）法

などがある。

クラスをパターンの集合と考えた場合、本論文の根底になる考えは以下の通りである。「クラスというのは本来ある種の”まとまり”をなすもので、それは同一クラスのパターン間の高い共通性として現れるであろう。また、一つのクラスは”排他的”、つまり、他のクラスと区別される筈であり、一つのクラスのパターンは他のクラスのパターンと区別できる様な特徴を有するであろう」。この様な考えは、ラベル付けされていないサンプルを幾つかのグループに分けるクラスタリングにおいては、級内分散、級間分散といった形で自然に用いられている。しかし、ラベル付けされているサンプルが提供されるパターン認識ではこれらはあまり考慮されていない。実際、パターンクラスの分布型を仮定してパラメータを推定する場合、一つのクラスのサンプルパターン間の共通性はパラメータに平均的に表現されてはいるけれども、これでは一般の分布の場合には共通性の表現が不十分である。また、クラスの排他性に関しては、同じくパラメトリックな方法では、クラス毎に分布（パラメータ）が推定され、最終的な識別結果の構成の段階で他のクラス分布との比較において表現されるに過ぎず、積極的にクラス間の差異を考慮しているとは言い難い。従って、典型的なパラメトリック手法では共通性や

排他性を積極的に利用していないと思われる。また、ノンパラメトリックの代表的な手法であるＮＮ法を考察すると、未知パターンは最も近いサンプルパターンの属するクラスに割り当てられるから、この場合には同じクラスのサンプルパターン間の共通性はもはや考慮されず、各々のサンプルパターンに関して排他性のみが重要視される。その意味で、パターン間の共通性を平均的に用いるパラメトリックな方法と、共通性は全然考慮せずに排他性のみを重視するＮＮ法は両極端にあると言える。

これらの方法を現実の問題に適用するには前者は分布が仮定の型と大きく異なる時、後者はサンプルが十分にクラスの分布を反映しない時などにそれぞれの過度な面が識別に影響し過ぎる事があり、それらの中間的な性質を有する区分的線形識別関数などが現実には多用されている。区分的線形識別関数には複数のクラス核がクラスタリングの援用によって決定され、それぞれのクラス核を独立なクラスと考えて規則が構成される。これは確に問題毎のクラス分布に追従できるだけの柔軟性を持ち、個々のサンプルに影響され過ぎない平均的な扱いもしている。また、ＮＮ法も未知パターンに近いｋ個までのサンプルパターンを考慮する事で、同一クラスにおけるサンプル間の共通性をいくらか取り入れるように拡張されている。その意味では残された問題はクラス核の決定方法ならびに最適なｋの決定方法を確立する事である。しかし、これらの一般的解決は難しい問題である。さらに、重要な点はこれらの方法のいずれもが現在の環境（サンプル集合、特徴集合）において、それらを肯定して行われている事である。しかしながら、この状況では、不適当なサンプルが混入していたり、サンプルが十分クラスを反映していない場合、あるいは識別に何の情報も持たない特徴が混在していた場合、それらを見極める事はできない。最後の場合は通常、特徴選出の問題として識別と分離して考察されるが、その方法論自体もまだ十分確立されてはおらず、それにもまして、識別と特徴選出はその相互関係を十分考慮して行われるべきであると思われる。

本論文ではこれらの点を考慮して、パターン認識の諸問題を同一クラスのパターン間の共通性およびクラス間の差異の扱いを中心に捉え直す。

具体的には、複数クラスからの訓練サンプル集合が与えられた時、一つのクラスの訓練サンプル集合中に「他のクラスのサンプルを区別するような共通の特徴を有し、しかも包含関係において極大となるような部分集合」（”部分クラス”）を見出す事を考える。この考えは既にStoffelの研究［１］に端を発する。しかし、その研究はどちらかと言えばスイッチング理論の枠組みで、表現の効率化の目的で導入され、クラスとしての積極的な意味付けはなされてはいない。また、アルゴリズムが非実際的であった事もその後の発展を妨げたように思われる。部分クラスは定義から明らかに排他性を重視しつつ、最大限に共通性も表現するように構成されているので、逆に部分クラスからクラスの構造を調べる事も考えられる。その意味で、現在の環境（サンプル集合、特徴集合）に左右される従来の大部分の方法と異なり、逆に環境の評価を行う事もできる。例えば、特徴選出の問題を考える時、部分クラスが大きくなればなる程、つまりその部分クラスが数多くの訓練サンプルを含めば含む程、その部分クラスを規定する特徴の数が減る事は一般的に予想される。なぜなら、一般に個体数の増加に応じて、すべての個体に共通な特徴は減ずるからである。その場合、排他的共通性の低い特徴は用いられなくなり、結果として特徴選出が自動的に行われる。また、不適当なサンプルは同じクラスに属する別のサンプルとの排他的共通性が低いであろうから、その様なサンプルを除去するなどの対策を講ずることができる。

２．３．　パターン認識過程

２．３．１．　各処理の働き

まず、標準的なパターン認識過程を図２．１に示す。この図において、下段は実際の”認識過程”、つまり未知パターンに対するクラスの割当を実行する過程で、多段の処理を受けて実行される。それに対し上段はその各処理内容を規定するための”学習過程”で、通常、有限の訓練サ

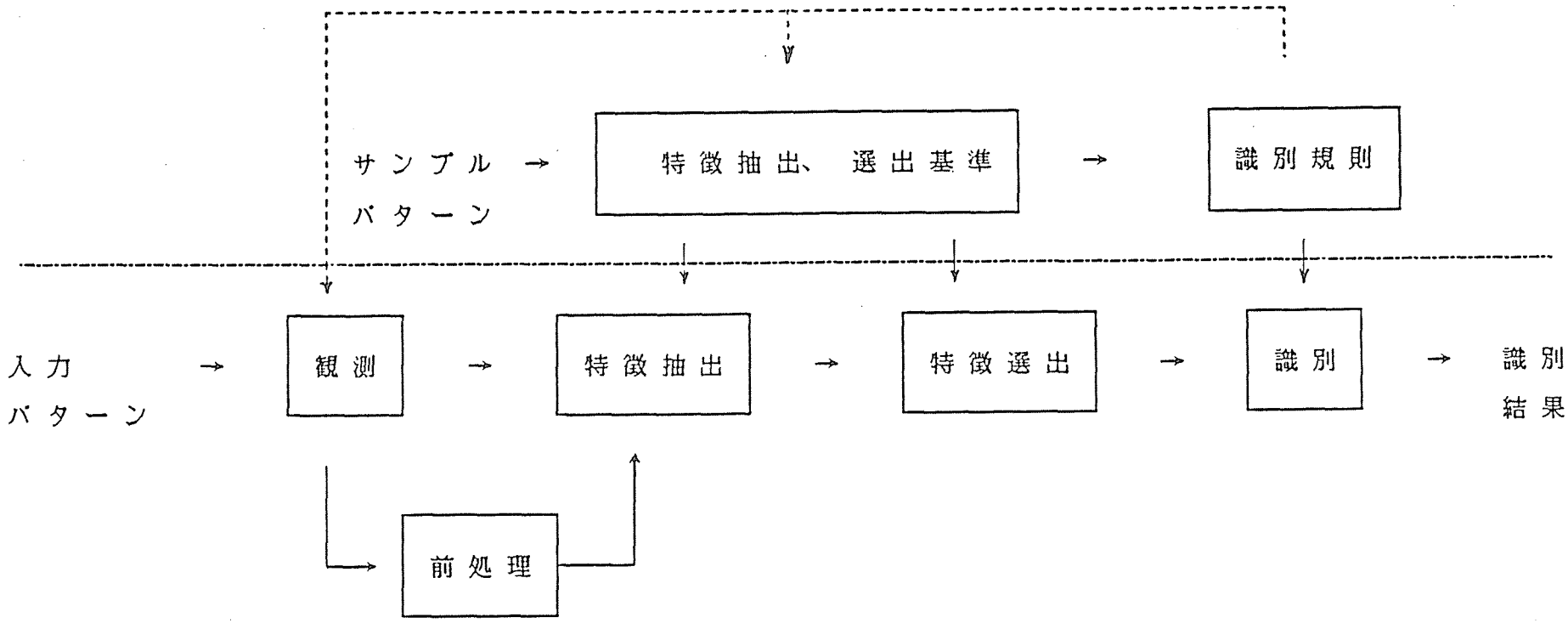

図２．１　標準的パターン認識過程

ンプル集合から最もふさわしいと思われる処理内容を推論する。従って、一つのパターン認識系を考えた時は、上段の学習過程が本質的にそのパターン認識系の良否を決める。

次に、各処理の働きを認識過程に沿って考えると、まず、入力されたパターンはその物理的性質が色々な測定機器により観測され、数値化される。その意味では機器の選定が”第一次特徴選出”に相当する。測定機器の出力はそのまま次の特徴抽出処理に受け渡されるか、雑音の除去、観測系の歪みの補償、セグメンテーション、などの前処理の後受け渡される。特徴抽出と次の特徴選出の処理の間には明白な違いはない。しかし、狭義には次の様にその二つを区別する事ができる。「特徴抽出とは測定された特徴（前処理を受けていてもよい）を識別に有効と思われる特徴に”変換”する事であり、特徴選出は実際に識別に有効な少数の特徴を”選ぶ”事である」。その意味での明確な区別はさらに「特徴抽出によっては測定コストは節約されないが、特徴選出により測定コストが節約される可能性がある」となる。特徴抽出の例としては、フーリエ変換、Ｋ－Ｌ変換、あるいは各種の非線形変換が挙げられる。この処理の目的は測定値の特徴を考慮中の問題に最もふさわしい特徴に変換する事と理解される。従って、問題毎の特殊性を吸収できる処理である。また、特徴抽出においてＫ－Ｌ展開の固有値の大きいものだけを残す事なども考えられており、その意味で”第二次特徴選出”を行っていると言える。しかし、この段階での特徴選出には積極的な意味付けがなされていない。次に、特徴選出の処理を考える。ここでは測定値の特徴がそのまま入力となる場合と特徴抽出の変換後の特徴が入力となる場合があるものの、いずれにせよその特徴集合から識別に有効な少数の特徴が選び出される”第三次特徴選出”を行っている。最後に、識別処理にパターンが渡され、クラスの割当がその特徴と予め定められている規則に応じて行われる。

学習過程は各処理の具体的な内容を有限のサンプル集合から定める過程である。基本的に、特徴抽出ならびに特徴選出の内容設定と識別規則の設定を行う。それらの方法ならびに問題点を次に各処理毎に考える。

2．3．2．　各処理における問題点

ここでは、本論文全体を通しての問題意識ならびに基本姿勢を明示すると共にそれらに対する各章の持つ役割を示す。具体的には、パターン認識過程の各処理毎にその問題点を明らかにし、それらに対する本論文の扱いを述べ、対応する章を示す。

まず、観測に関して考察する。観測は先に述べた様に第一次特徴選出の役割を担っていると考えられる。その意味では、識別に重要な特徴は少なくとも観測しなければならない。結果として識別に有効と思われる特徴すべてを測定する事が望ましい。しかし、必ずしも測定機器の精度が十分であるとは限らず、機器の選定においても、意識せずに知覚している特徴の様なものもあるかも知れない。何よりも、人間が一つのパターンを一つのクラス（概念）に結び付ける時、そこに文脈の強い支持がある事は常に議論されているし、同じ様な意味で現下のサンプルの生じた状況の想定を常に意識していると考えられる。従って、本質的なパターンの認識にはパターンの測定可能な物理的性質ばかりでなく、文脈や心理的状況なども考慮しなくてはならない。少なくとも観測に関しては十分に検討された上で機器の選定をしなければならない。最終的に、やはりできるだけの特徴を測定する他はないように思われる。しかし、その場合でも現在の観測が十分であるかどうかを吟味する事は有意義であると考えられる。本研究では、「現在の特徴集合が十分であるか」の問いに答えるために、パターン認識機構上で、識別規則から観測へのフィードバック（図2．1の点線）を考える。具体的には、識別規則ともなる部分クラスを構成し、その後、それから導かれるクラスの構造を基準に識別性能を評価して不十分であれば、すべての特徴選出に関わった処理に対して再考を促し、結果として観測の処理も再検討される。

もう一つの考慮すべき問題として訓練サンプルの妥当性がある。「すべてのサンプルがそれぞれの属するクラスの代表として十分であるか」とか「誤ってラベル付けされてはいないか」といった問題である。本研

究は、これに対しても同じ様に識別規則からのフィードバックを考え、個々のサンプルがどの程度クラスを代表しているかを評価する一つの手段を与える。また、パターン認識の重要な問題として、サンプル数に比べて余りに多くの特徴を識別規則の構成に用いた時には、識別規則が特徴を最適に用いる事ができず、その一部の特徴を用いた場合の識別性能の方が高い場合がある。これはサンプルの有限性から生じているので、サンプル数を増やす事により改善が計られるが、実際には医療診断などどうしてもサンプル数を制限される場合が多い。従って、先ほどの「測定する特徴はできるだけ多い方が望ましい」という立場とサンプルの有限性は互に識別性能の向上の目的に関して相反するものと考えられる。故に、サンプル数と特徴数はその両方を同時に考察しなければならない。

特徴抽出に関しては本論文は直接それを扱わない。これは、特徴抽出が問題毎の特殊性を吸収する処理であるという判断から、処理の問題点も一般的に扱う事は難しいと考えるからである。逆に問題毎の特殊性が特徴抽出で吸収されるとすれば、その後の処理である特徴選出は一般の問題に対して理論的に考察できる。特徴選出の目的は特徴集合全体の中から識別性能を高く維持する少数の特徴の組を発見する事である。次の二点が特徴選出の有効性を支持する。１）測定コストや識別コスト（識別効率）の節約、２）識別性能の向上。１）のコストの節約は従来から様々な研究がなされているが、２）はサンプルの有限性との関係から比較的新しく論じられていて、未解決の部分が多い。特徴選出の基本的な問題点は、１）特徴数が多い時にその実行が不可能になる程の計算量を要する方法が多く、その場合、より計算量の少ないアルゴリズムを用いて準最適解で満足しなければならない、２）識別性能をサンプルから見積る必要があり、サンプル数が少ない場合に選んだ特徴の最適性が疑わしい、などが挙げられる。

最後の処理である識別規則の構成はパターン認識問題の中核をなす部分であり、これまでに膨大な数の研究がなされており、その各々がそれぞれの問題点を有している。従って、ここではそれには触れず、識別機構全体における識別規則の役割に注目する。この時、サンプル数、特徴

数、識別規則の型などの相互作用の結果が識別性能に現れ、それぞれを個別に考えているだけでは識別機構全体としての最適化は計れないと考えられる。本研究はこの点を踏まえて前述のフィードバックに基き、何回か各処理に関する調節を全体の識別性能を調べながら行う事を目的とする。この実現は処理毎に問題毎の制約を受けるため、総合的な発展はオペレーターの判断に基いて行われると考えられる。

これらの問題点に対する対処法ならびに扱う章はそれぞれ以下の通りである。まず第３章で、パターンを文字列として表現した場合のパターン識別を考察する。この時、同じクラスのサンプル間の共通性を基準として、クラスを規定する性質の抽出を行う方法を述べる。また、これは同時にパターン識別規則としても有用である事を示す。第４章で、その手法を実際に遺伝子塩基配列の特徴的な配列のクラスに適用した例を述べる。第５章では、パターンを二値の特徴で表現した場合に部分クラスを効率よく求めるアルゴリズムを述べる。そして、第６章で部分クラスを一般の特徴の場合に拡張し、超区間として部分クラスを求める方法を述べる。また、一つの試みとして、クラスの構造を部分クラス全体の大きさから定量化する。また、実際にその定量化による値を認識過程におけるすべてのフィードバックの外的基準として、各処理へのフィードバック方法および適用法を述べる。具体的には、特徴選出ならびにサンプル選出のアルゴリズムを述べる。

2．4．　結論

本章では、パターンの性質ならびにクラスの構造を、特にサンプルの排他性と共通性の面から捉え、それらの視点により、従来の認識方法を見つめなおした。また、認識機構全体の問題点を論じ、本論文のそれらに対する姿勢を示した。

第３章　正規文法推論問題

３．１．　序論

対象となるパターンが形状や構造を表す時は、ベクトルとしての表現よりも、文字列としての表現の方が自然にその性質を表現できる。文字列の個々の文字はプリミティブ（局所的な構造）を表し、全体的な構造がプリミティブ間の関係、特に、一次元または多次元のつながりとして表現される。構文的パターン認識はある種の文字列の集合が、決められた”文法”（規則）から生成されたと考えるもので、その場合一つの文法が一つのクラスを定める。また、訓練サンプルの文字列から適切な文法を”推論”する事も重要な問題となる。この問題を”文法推論”（Grammatical Inference）と呼ぶ。

文法には幾つかの型があり、Chomsky［２］の階層が枠組を与える。その階層は制約のゆるい順に、句構造文法（０型文法）、文脈依存文法（１型文法）、文脈自由文法（２型文法）、正規文法（３型文法）から成る。この中で正規文法は表現力が最も制限されるものの、二つの文法が等価（生成する文字列集合が同じ）であるかどうかが決定可能であるなど実用上有効な性質を多数有する。また、正規文法により生成される文字列の集合（”言語”）は有限オートマトンと呼ばれる仮想機械により、より明確な形で図表現される。

現在、正規文法の推論手法は多数考案されている。しかし、それぞれの持つ特質が明確でない。これはそれらを共通に評価する視点が欠如しているのが原因と思われる。本章では手法の特定化が有限オートマトンの状態間の同値関係にある事、また、その関係は状態の前の部分列と後の部分列の類似性に基く事に着目して、なるべく多くの同値関係を表現できる基本的アルゴリズムを提案する。そのアルゴリズム上で従来法の幾つかが再構成され、用いる同値関係により手法毎の特質が明確になる。また、パラメータを持つ自然な同値関係を考え、典型的と思われる三手法を提案する。これらは、その同値関係の基準から推論される文法の性質が明確で、その中から推論目的に応じて手法を選択する事が容易である。加えて、実際に二つの実用的な目的を設定し、目的別に従来手法ならびに提案手法を分類する事も考える。

３．２．　基本的アルゴリズム

３．２．１．　基本的定義

ここでは、今後の議論に必要な定義を行う。

[定義３．１]

(1)　Σを終端記号の有限集合とし、”アルファベット”と呼ぶ。一つの”文”は　$x=a_1a_2\cdots a_n \in \Sigma^*$,　$a_i \in \Sigma$と表され、その長さを$|x|(=n)$と書く。ここで、Σ^*はアルファベットΣ上のすべての可能な文の集合を表す。特に、長さ０の文を”空列”と呼び、λで表す。

(2)　５個組　$A=(\Sigma, Q, \delta, q_0, F)$を非決定性有限オートマトンと呼ぶ。

ここで、　Σ：アルファベット，

Q：状態の有限集合，

q_0：初期状態　$(\in Q)$，

F：最終状態の有限集合　$(\in 2^Q)$，

δ：推移関数　$(Q\times\Sigma \to 2^Q)$.

オートマトンの言語は次のように定義される。

$$L(A) = \{x \mid \hat{\delta}(q_0, x) \cap F \neq \phi\},$$

ここで、　$\hat{\delta} : Q\times \Sigma^* \to 2^Q$, かつ

(a)　$\forall q \in Q,\ \hat{\delta}(q, \lambda) = \{q\}$,

(b)　$\forall q \in Q,\ \forall a \in \Sigma,\ x \in \Sigma^*$,

$$\hat{\delta}(q, xa) = \bigcup_{r \in \hat{\delta}(q,x)} \delta(r, a).$$

非決定性有限オートマトンはその推移関数が $\delta : Q \times \Sigma \rightarrow Q$ に制限された時、決定性有限オートマトンと呼ばれる。

(3) オートマトン $A=(\Sigma, Q, \delta, q_0, F)$から導かれるある種のオートマトンは導出オートマトンと呼ばれ、その集合は $D(A)$と書かれる。導出オートマトンは任意の状態の分割に基いて生成される。すなわち、Q上の任意の同値関係により各状態を同値類に対応するブロックに分ける。その時、導出オートマトン A_D は次の様に定義される：

$A_D = (\Sigma_D, Q_D, \delta_D, q_{0D}, F_D) \in D(A)$,

ここで、

(a) $\Sigma_D = \Sigma$,

(b) Q_D の各状態は Q上に定義された分割の一つのブロックに対応する、

(c) 状態 q_{0D} は 状態 q_0を含むブロックに対応する、

(d) F_Dの各状態は Fの要素を一つでも含むブロックに対応する、

(e) 推移 $\delta_D(q_D, a) = q'_D$ は推移 $\delta(q, a) = q'$に対応する。ここで、q_D (q'_D) は q (q')を含むブロックに対応する。

次の事はすぐにわかる。

$$L(A) \subseteq L(A_D).$$

(4) 文の集合 S^+ と S^- が $S^+ \subseteq L(A)$ ならびに $S^- \subseteq L^c(A)$を満す時、それぞれは正サンプル集合、負サンプル集合と呼ばれる。ここで、肩字 cは集合の補集合を意味する。また、$S=(S^+, S^-)$は単にサンプル集合と呼ばれる。

アルファベット Γ 上のサンプル集合 S^+ は次の条件を満足する時、非決定性有限オートマトン $A=(\Sigma, Q, \delta, q_0, F)$に対して”完備”である[7]と言われる。

(a) $S^+ \subseteq L(A)$,

(b) $\Gamma = \Sigma$,

(c) 各推移 δ は少なくとも一度 S^+の一つの文の生成に用いられている。

(5) オートマトン Aは $L(A) = S^+$ を満足する時"標準"であると言われる。最も基本的な標準オートマトンは"極大標準オートマトン" $A_M(S^+)$ であり、以下の様に定義される：

正サンプル集合を $S^+=\{x_1, x_2, \cdots, x_N\}$, $x_i=a_{i1}\ a_{i2}\ \cdots a_{im}$, $m=|x_i|$ とする。この時、

$$A_M(S^+) = (\Sigma_M, Q_M, \delta_M, q_{0M}, F_M),$$

ここで、

(a) Σ_M は S^+の異なる終端記号の集合、

(b) Q_M は次の $(1+\Sigma|x_i|)$個の状態から成る：

$$q_0, q_{11}, q_{12}, \cdots, q_{1|x_1|},$$

$$q_{21}, q_{22}, \cdots, q_{2|x_2|}, \cdots, q_{n|x_N|},$$

(c) $F_M = \{q_{i|x_i|} \mid 1 \leqq i \leqq N\}$,

(d) $\delta_M(q_{0M}, a_{i1}) = q_{i1}$,

$\delta_M(q_{ij}, a_{i(j+1)}) = q_{i(j+1)}$, $1 \leqq j \leqq |x_i| - 1$, $1 \leqq i \leqq N$。

(6) サンプル集合を $S=(S^+, S^-)$とする時，S の一つの"解"とは次の条件を満たすオートマトン Aである：

(a) $S^+ \subseteq L(A)$, かつ

(b) $S^- \subseteq L^c(A)$。

本章の残りでは専ら $S^-=\phi$ の場合を考えるので、その場合、条件(b)は無視できる。

解と導出オートマトンの間には次の二つの重要な性質が知られている［３、７］。

1. A_D を極大標準オートマトン $A_M(S^+)$の任意の導出オートマトンとする。この時、$S^+ \subseteq L(A_D)$.

2. $A_M(S^+)$を極大標準オートマトンとする。オートマトン Aに対してS^+が完備な時，$A \in D(A_M)$.

これらの性質は「A_Mのすべての導出オートマトンはすべてS^+ に対する解であり、 S^+が”真の解”、すなわちサンプルがそこから取られたオートマトン、 に関して完備であれば、 真の解は導出オートマトンの集合$D(A_M)$の中に見出される」事を言っている。そこで、サンプルの性質として自然に次の事を要求する。

（仮定）正サンプル S^+は真の解に対して完備である。

３．２．２．アルゴリズム（Kudo and Shimbo［５］）

以上の議論により、 正規文法推論問題は極大標準オートマトン A_Mから導出オートマトンを見出す事、 つまり、A_Mの状態集合上に一つの同値関係を定める事に換言される。 従って、従来の各手法も本章で提案される方法も、 その同値関係の性質を解析する事で各手法の特質が明確になる。

まず、提案手法に必要な定義を行う。

［定義３．２］

[1] 推移文 x^* ならびに推移記号の集合 V^*

正サンプル集合を$S^+=\{x_1, x_2, \cdots, x_N\}$、$x_i=a_{i1}a_{i2}\cdots a_{in_i}$ $(1 \leqq i \leqq N)$とする。一つの正サンプルに対応して文$x^*_i=S_i a_{i1} Z_{i1} a_{i2} \cdots Z_{i(n_i-1)} a_{in_i} T_i$ $(1 \leqq i \leqq N)$は"推移文"と呼ばれる。また、記号S_i, T_i, Z_{ij}は"推移記号"と呼ばれる。推移記号の集合は

$$V^*=\{ S_i, T_i, Z_{ij} \mid 1 \leqq j \leqq n_i-1,\ 1 \leqq i \leqq N\}$$

と書かれる。

[2] 推移標準オートマトン A^*:

$A^* = (\Sigma, Q, \delta, q_0, F)$,

ここで、

(1) ΣはS^+における異なる終端記号の集合,

(2) $Q = V^* \cup \{q_0\}$,

(3) $F = \{T_i \mid 1 \leqq i \leqq N\}$,

(4) a. $\delta(q_0, \lambda) = S_i$, $1 \leqq i \leqq N$,

b. 各$x^*_i=S_i a_{i1} Z_{i1} a_{i2} \cdots Z_{i(n_i-1)} a_{in_i} T_i$に対して、

$$\delta(S_i, a_{i1}) = Z_{i1},$$
$$\delta(Z_{i1}, a_{i2}) = Z_{i2},$$
$$\vdots$$
$$\delta(Z_{i(n_i-1)}, a_{in_i}) = T_i, \quad (1 \leqq i \leqq N).$$

推移標準オートマトン A^*は極大標準オートマトンと初期状態からの推移に関してのみ異なる。また、$D(A_M) \subseteq D(A^*)$なので、真の解 Pに対して完備なサンプル集合から A^*を構成した場合、$P \in D(A^*)$が言える。

各推移記号ごとに次の"定長先行語"と"定長後続語"がそれぞれ定義される。

[3] 定長先行語

一つの推移文 $x^*_i=S_i a_{i1} Z_{i1} a_{i2} \cdots Z_{i(n_i-1)} a_{in_i} T_i$ $(1 \leqq i \leqq N)$を考え

た時、各推移記号 S_i, T_i, Z_{ij} に対する定長先行語は次の様に定義される：

$$a.\ pre(S_i)_k = \lambda,$$

$$b.\ pre(T_i)_k = \begin{cases} a_{i(n_i-k+1)}a_{i(n_i-k+2)}\cdots a_{in_i} & (n_i-k+1 \geqq 1) \\ x_i & (\text{それ以外}), \end{cases}$$

$$c.\ pre(Z_{ij})_k = \begin{cases} a_{i(j-k+1)}a_{i(j-k+2)}\cdots a_{ij} & (j-k+1 \geqq 1) \\ a_{i1}a_{i2}\cdots a_{ij} & (\text{それ以外}). \end{cases}$$

[4] 定長後続語

一つの推移文 $x^*_i = S_i a_{i1} Z_{i1} a_{i2} \cdots Z_{i(n_i-1)} a_{in_i} T_i$ $(1 \leqq i \leqq N)$を考えた時、各推移記号 S_i, T_i, Z_{ij} に対する定長後続語は次の様に定義される：

$$a.\ suc(T_i)_k = \lambda,$$

$$b.\ suc(S_i)_k = \begin{cases} a_{i1}a_{i2}\cdots a_{ik} & (k \leqq n_i) \\ x_i & (\text{それ以外}), \end{cases}$$

$$c.\ suc(Z_{ij})_k = \begin{cases} a_{i(j+1)}a_{i(j+2)}\cdots a_{i(j+k)} & (j+k \leqq n_i) \\ a_{i(j+1)}a_{i(j+2)}\cdots a_{in_i} & (\text{それ以外}). \end{cases}$$

特に、次の記法を採用する：

$$pre(A) = pre(A)_\infty,$$

$$suc(A) = suc(A)_\infty。$$

また、簡単の為、誤解がない時に限り”定長”を省いて、これらを単に”先行語”、”後続語”と呼ぶ事にする。

次に、先行語ならびに後続語の集合を以下に定義する。

[5] XをV^*の一つの部分集合とする。この時、

(1) 先行語の集合

$$P(X)_k = \{ pre(A)_k \mid A \in X \},$$

(2) 後続語の集合

$$S(X)_k = \{ suc(A)_k \mid A \in X \},$$

(3) k-tailの集合

$$S^k(X) = \{ suc(A) \mid |suc(A)| \leqq k, A \in X \}.$$

これらの定義の下で、推移記号間に同値関係を定める。

[6] V^*上の同値関係 R

(1) Rp_k : $A\ Rp_k\ B \longleftrightarrow pre(A)_k = pre(B)_k$.

(2) Rs_k : $A\ Rs_k\ B \longleftrightarrow suc(A)_k = suc(B)_k$.

更に、論理記号 ∧（かつ）ならびに ∨（または）を導入して、

(3) $Rp_m \wedge Rs_n$: $A\ (Rp_m \wedge Rs_n)\ B \longleftrightarrow A\ Rp_m\ B$ かつ $A\ Rs_n\ B$.

(4) $Rp_m \vee Rs_n$: $A\ (Rp_m \vee Rs_n)\ B \longleftrightarrow A\ Rp_m\ B$ または $A\ Rs_n\ B$.

(5) $\overline{Rp_m \vee Rs_n}$:

この関係は関係 $Rp_m \vee Rs_n$の推移的閉包によって定義され、次の配列が存在する時、推移記号 Aは推移記号 B に関係 $\overline{Rp_m \vee Rs_n}$ によって関係づけられる：

$C_0(Rp_m \vee Rs_n)C_1(Rp_m \vee Rs_n) \cdots (Rp_m \vee Rs_n)C_t$,

ここで、$t \geqq 1$で $C_0=A$ かつ $C_t=B$ とする。

(6) $\bigvee_{m+n=k} Rp_m \wedge Rs_n$: $A\ (\bigvee_{m+n=k} Rp_m \wedge Rs_n)\ B$

$$\longleftrightarrow A\ (Rp_0 \wedge Rs_k)\ B,\ A\ (Rp_1 \wedge Rs_{k-1})\ B, \cdots,$$

A $(Rp_k \wedge Rs_0)$ B のどれかが成立。

(7) $\overline{\bigvee_{m+n=k} Rp_m \wedge Rs_n}$: 関係 $\bigvee_{m+n=k} Rp_m \wedge Rs_n$ の推移的閉包.

ここで、関係 (4)と(6)は推移的閉包の操作によって始めて同値関係(5)、(7)にそれぞれなっている事に注意する必要がある。

[7] 2^{V*}上の同値関係 Q

X と Y を V^* の互いに素な部分集合とする。この時、

(1) Qp_k : $X\ Qp_k\ Y \leftrightarrow P(X)_k = P(Y)_k$,

(2) Qs_k : $X\ Qs_k\ Y \leftrightarrow S(X)_k = S(Y)_k$,

(3) Qs^k : $X\ Qs^k\ Y \leftrightarrow S^k(X) = S^k(Y)$,

(4) $\bar{Q}\bar{p}_k$: 以下で定義される関係 $\hat{Q}p_k$ の推移的閉包:

$X\ \hat{Q}p_k\ Y \leftrightarrow P(X)_k \cap P(Y)_k \neq \phi$,

(5) $\bar{Q}\bar{s}_k$: 以下で定義される関係 $\hat{Q}s_k$ の推移的閉包:

$X\ \hat{Q}s_k\ Y \leftrightarrow S(X)_k \cap S(Y)_k \neq \phi$.

推移標準オートマトンの各状態は推移記号のそれぞれに対応し、先行語、後続語は各推移記号の前後の配列を示しているから、自然に状態間の類似性が先行語、後続語の比較によって測られ、それを反映する同値関係 R ならびに Q により、状態間に同値関係が設定される。その具体的なアルゴリズムを次に示す。

[アルゴリズム３．１]

Step1 ：推移標準オートマトン A^*の生成。

S^+から推移文$\{x^*\}$ならびに推移記号の集合 V^*，推移標準オートマトン A^*を生成。

Step2 ：V^* の同値関係 Rによる分割。

同値関係 Rを用いて，V^* を同値類に分割する。

Step3 ：2^{V^*} の同値関係 Qによる合併。

Step 2で生成された同値類のうち同値関係により関係付くものを一つのまとめる。

Step4 ：$\{S_i\}$の単一化。

$\{S_i\}$ $(1 \leqq i \leqq N)$を一つの状態 Sにまとめる。

Step5 ：$\{T_i\}$の単一化。

$\{T_i\}$ $(1 \leqq i \leqq N)$を一つの状態 Tにまとめる。

Step6 ：導出オートマトン A_Dの構成。

Step 6までに行われた状態の分割に沿って導出オートマトン A_Dを A^*から構成。

3．2．3．　提案手法［5］

アルゴリズム3．1上で自然に定義される三手法を提案する。それらは、アルゴリズムで使用するステップならびに同値関係R、Qを明示する事により定義される：

1．∧（m，n）法

$1 \rightarrow 2 \rightarrow 4 \rightarrow 6 \quad (R = R\,p_m \wedge R\,s_n)$

2．∨（m，n）法

$1 \rightarrow 2 \rightarrow 6 \quad (R = \overline{R\,p_m \vee R\,s_n})$

3．＋（k）法

$1 \rightarrow 2 \rightarrow 6 \quad (R = \overline{\bigvee_{m+n=k} R\,p_m \wedge R\,s_n})$

具体例を次に示す。

（例3．1）

正サンプルをS^+＝｛d a b a e，c a b a f，e b a d，e c e d，e b e d｝とする。この時の推移標準オートマトンを図3．1に示す。推移記号の集合は｛S_1～S_5，T_1～T_5，Z_1～Z_{17}｝であり、推移標準オートマトンの各状態に対応している。推移記号間の同値関係に関しては、それぞれ∧（1、1）法、∨（3、3）法、＋（3）法の場合を表3．1に示す。特に、＋（3）法を用いた場合の導出オートマトンを図3．2に示す。

3．3．　評価

3．3．1．　従来法との比較

従来提案されてきた主な手法は以下の通りである：

1）k-tail法［1］

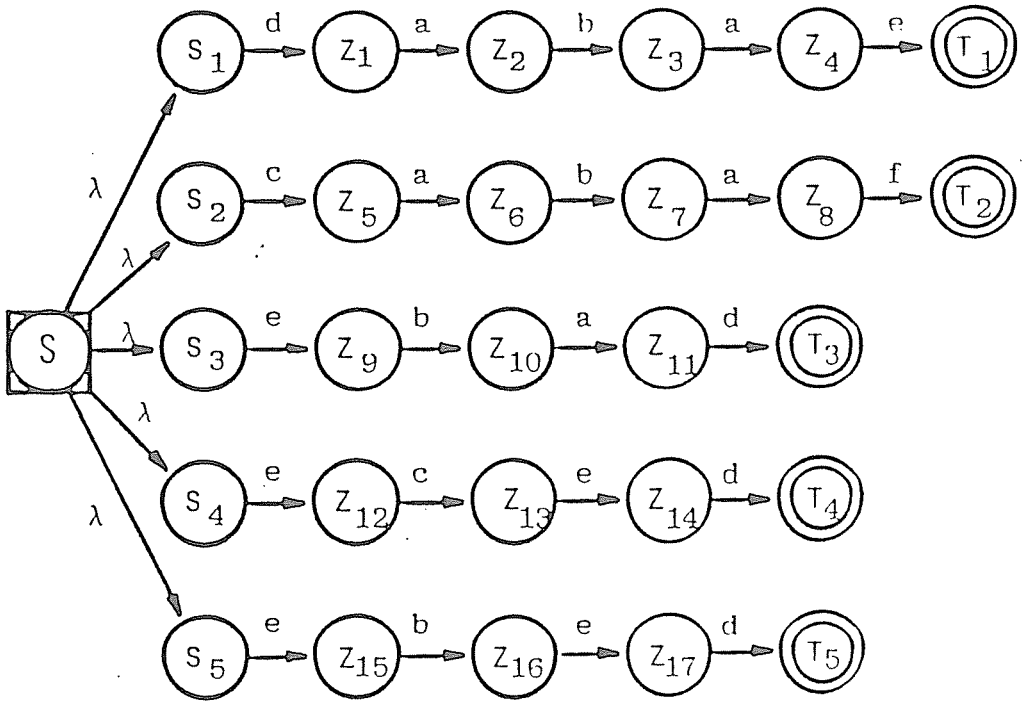

図３．１　例３．１の推移標準オートマトン

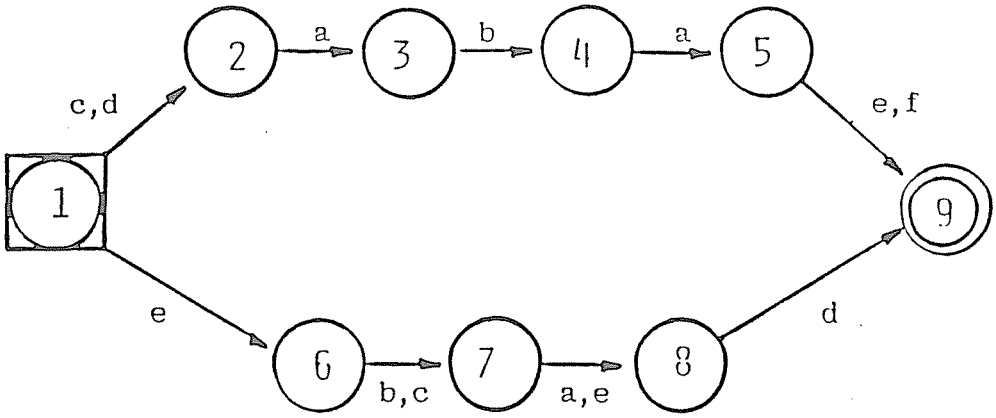

図３．２　例３．１の導出オートマトン　（＋（３）法）

表3．1　例3．1の推移記号の分割

手　法	推移記号の分割
∧ (1,1)	$\{S_1\},\{S_2\},\{S_3,S_4,S_5\},\{Z_1\},\{Z_2,Z_6\},\{Z_3,Z_7\},\{Z_{10}\},\{Z_4\},$ $\{Z_5\},\{Z_8\},\{Z_9,Z_{15}\},\{Z_{11}\},\{Z_{12}\},\{Z_{13}\},\{Z_{14}\},\{Z_{17}\},\{Z_{16}\},$ $\{T_1\},\{T_2\},\{T_3,T_4,T_5\}$
∨ (3,3)	$\{S_1,S_2,S_3,S_4,S_5\},\{Z_1,Z_5\},\{Z_2\},\{Z_6\},\{Z_3\},\{Z_7\},\{Z_4,Z_8\}$ $\{Z_9,Z_{12},Z_{15}\},\{Z_{10},Z_{13},Z_{16}\},\{Z_{11},Z_{14},Z_{17}\},$ $\{T_1,T_2,T_3,T_4,T_5\}$
+(3)	$\{S_1,S_2,S_3,S_4,S_5\},\{Z_1,Z_5\},\{Z_2,Z_6\},\{Z_3,Z_7\},\{Z_4,Z_8\}$ $\{Z_9,Z_{12},Z_{15}\},\{Z_{10},Z_{13},Z_{16}\},\{Z_{11},Z_{14},Z_{17}\},$ $\{T_1,T_2,T_3,T_4,T_5\}$

2）UV^{k}Π法［６］

3）tail-clustering法［８］

4）successor法［９］

5）predecessor and successor法［１０］

始めに、従来法と提案手法の推論性質を考察する。従来法の幾つかはアルゴリズム３．１上で再構成される：

1）k-tail法

$$1 \rightarrow 2 \rightarrow 3 \rightarrow 6 \quad (R = R_p,\ Q = Q_{s^k})$$

2）successor法

$$1 \rightarrow 2 \rightarrow [3] \rightarrow 6 \quad (R = R_{p_1},\ Q = Q_{s_1})$$

ここで、［３］は最簡形を導くために必要とされるステップで、生成する言語には影響しない。また、厳密には異なるが、着眼している先行語、後続語の長さが等しく、大部分の問題において同じオートマトンを推定する手法を tail-clustering 法に準ずるものとして考えると、

3）tail-clustering法［８］に準ずる手法

$$1 \rightarrow 2 \rightarrow 6 \qquad (R = \overline{R_p \vee R_s})$$

従って、等価な言語を生成するという意味においては、successor法は∧（１、０）法に、tail-clustering法に準ずる手法は∨（∞、∞）と見なす事ができるので、∧（m，n）法はsuccessor法の、∨（m，n）法はtail-clustering法のそれぞれ自然な拡張と考えられる。

特に先行語、後続語の長さに関して手法を比べて見ると次の事がわかる。

a）successor法は先行語の長さが１、後続語の長さが０と非常に短い。これは同じ文字の繰返しの表現としてはよい性質で、簡単な（状態数の少ない）オートマトンを構成する目的には向いているものの、文字列全体のつながりを表現するには不十分である。また、そのままではアルゴリズム３．１上で再構成できないが、predecessor and successor法はこの欠点を前後の文字の制約を加えて改善したものである。しかし、この手法もさらに１文字程度考慮する程度であるので、本質的に同じ問題点を抱える。

b）k-tail法は先行語を最大長、後続語を長さｋ用いる。特に、手続きのはじめ

に先行語のみを考慮して分割が行われる為、その後の手続きで行われる長さkの後続語比較は前段階の影響を強く受けすぎてそれほど推論に反映しない。その為、生成するオートマトンはかなり標準オートマトンに近くなる。

c）tail-clustering法は先行語も後続語も最大長を考慮する。これはsuccessor法と対照的で、同一文字の繰返しの表現には不向きである一方、文字列全体のつながりを忠実に反映する性質を有する。また、推移的閉包の操作から考えても k-tail法よりは広い言語を受理するオートマトンを生成する。

この様に、従来の手法を先行語ならびに後続語の長さの観点から見た場合、提案手法の持つパラメータの有効性が明確になる。つまり、文字列全体のつながりを反映するほどに長く、局所的なつながりを効率的に表現するほど短く先行語、後続語を考慮するのが望ましい。しかし、実際の問題ではむしろそのどちらかをある程度犠牲にしても片方を強調したい事が多いので、これに関しては後節で目的を明確にして議論する。

今まで、手法の性質を先行語、後続語の長さの観点から捉えてきたので、次に幾つかの実験を通してそれらを検証する。従来法のなかで、successor法はpredecessor and successor法に拡張され、また、UV^{k}W法も生成する言語がpredecessor and successor法にほぼ等しく、計算時間はUV^{k}W法の方がかかり過ぎるという理由から、これら二つの手法の代表として、predecessor and successor法を選ぶ。従って、三従来法と三提案手法を比較する。

（例３．２）

動的システムの出力波形を一定時間間隔で測定し、波形の傾きに応じて文字列表現したものを２種類扱う［１０］。アルファベットは$\Sigma = \{a, b, c, d, e\}$である。二種類の正サンプル集合を次に示す：

クラス１：$S^+ = \{d^{16}c^{15}, d^{17}c^{14}, d^{18}c^{13}, d^{10}c^{21}, d^{20}c^{11}\}$,

クラス２：$S^+ = \{c^6b^2a^6e^7d^5c^5, cba^7e^{10}d^8c^4, a^9be^{21},$
$c^5b^2a^7e^8d^5c^4\}$,

但し、肩字は同一文字の繰返し数を表す。

この例は同じ文字の繰返しが多く、その部分を効率よく推論する事が望まれる。

その意味で、推論手法中効率的に推論した手法はpredecessor and successor法とΛ（1，1）法のみである。その結果を図3．3（a）～図3．4（b）に示す。双方ともほぼ同じ言語を受理するけれども、次の点で明確な差異がある。

○ Λ（m，n）法により生成されたオートマトンは同一文字の繰返しが最低（m＋n）回なければ文字列を受理しないのに対し、predecessor and successor法のオートマトンは最低1回の繰返しで受理する。

この事は、Λ（m，n）法の方が同じ文字の繰返しに制限を加える事ができるという点で好ましい様に思われる。また、Λ（1、0）法はpredecessor and successor法と同一のオートマトンを生成する。

クラス2の結果に関しては、

○ predecessor and successor法はサンプル間の相違性を別種なサンプル群と見なし、本質的に二つのオートマトンを内部に含む様なオートマトンを生成している。これに対し、Λ（m，n）法はその相違性をつながりが部分的に省略されたとして表現している。

これは本質的にすべてのサンプルが一つの型の波形から生成されたとすれば、後者の扱いが自然であろう。

（例3．3）

正サンプル集合をS^{+}＝｛aabba，bb，abbaa，aaabb｝とする。一つの評価規準は部分列”bb”の表現である。六手法の推論結果を図3．5（a）～（f）に示す。部分列の的確な表現としてはtail-clustering法、∨（3，3）法、＋（3）法が望ましい結果を与えている。その中でさらに両端の”a”を効率よく推論しているのは後の二手法である。

（例3．4）

これまではサンプルを先に与え推論を行ってきた。今度はモデルを設定し、それを推論する事を考える。モデルを図3．6（a）に示す。正サンプルはモデルから乱数により得た13個で以下に示す：

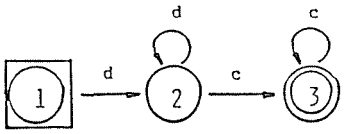

図３．３（ａ） 例３．２のクラス１のオートマトン
(predecessor and successor 法)

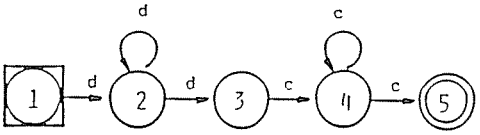

図３．３（ｂ） 例３．２のクラス１のオートマトン
(∧(1,1) 法)

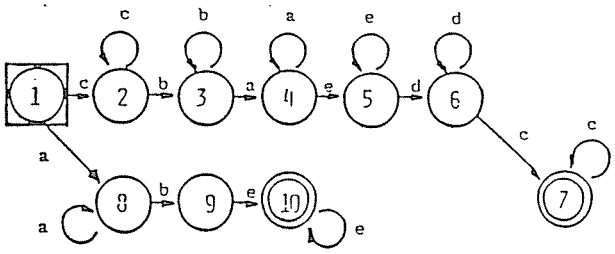

図３．４（ａ） 例３．２のクラス２のオートマトン
(predecessor and successor 法)

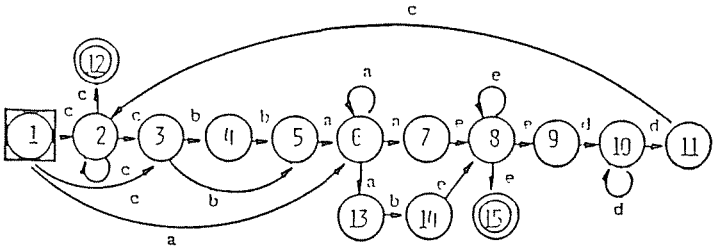

図３．４（ｂ） 例３．２のクラス２のオートマトン
(∧(1,1) 法)

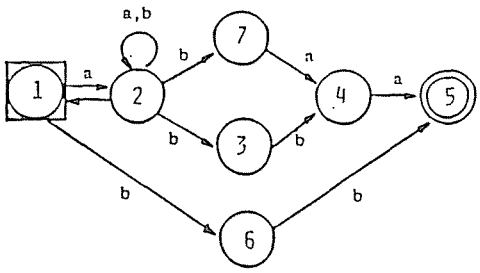

図３．５（ａ）　例３．３のオートマトン
(k-tail法)

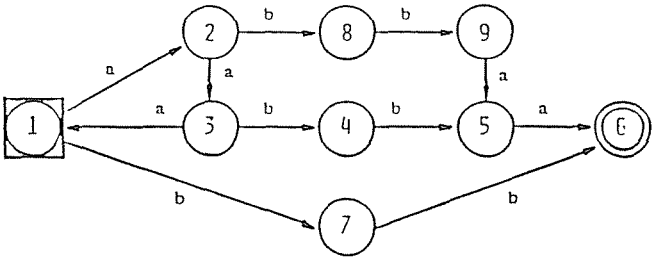

図３．５（ｂ）　例３．３のオートマトン
(tail-clustering法)

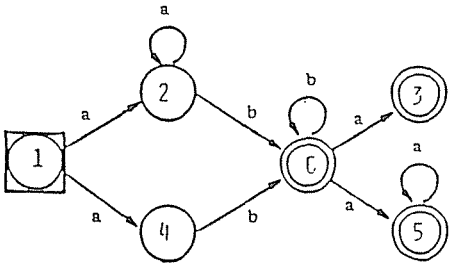

図３．５（ｃ）　例３．３のオートマトン
(predecessor and successor法)

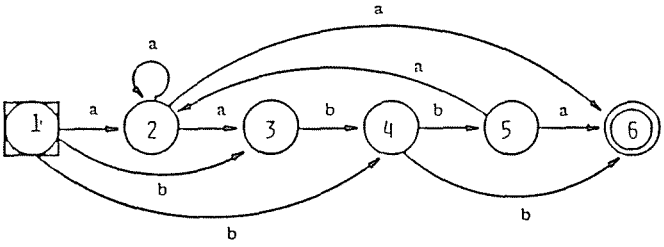

図３．５（ｄ）　例３．３のオートマトン
（∧（１，１）法）

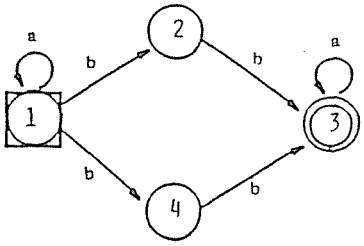

図３．５（ｅ）　例３．３のオートマトン
（∨（３，３）法）

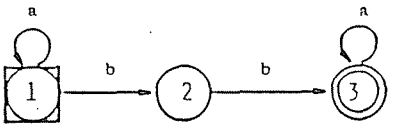

図３．５（ｆ）　例３．３のオートマトン
（＋（３）法）

S^+ ={ 1. cbdcadba　　2. bcdacdc　　3. adbdba

4. cbdacdab　　5. bcdcadab　　6. cbdacdba

7. bcdbdc　　8. adacdba　　9. bcdacdba

10. adacdc　　11. adcadab　　12. adbdab

13. bcdcadba }

モデルを再現できたのはtail-clustering法、∨（4、4）法、＋（4）法の三手法であった。サンプル数との関係を見る為に、上の順序で提示した時に幾つのサンプル数でモデルを推論するかを調べたところ、tail-clusteringと∨（4、4）法が１３個、＋（４）法が８個であった。サンプル数８の時のそれぞれの手法を図３．６（ａ）～（ｃ）に示す。

これにより、∨（m，n）法ならびに＋（k）法が少数サンプルでも有効な推論を行う事がわかる。しかし、この事は反面”過度の推論”をする危険をも意味する。対処法の一つはパラメータの調節であろう。

これらの実験例からも先に考察した先行語ならびに後続語の長さに関して次の従来法の二つの欠点が明白になっている：１）用いる長さが短かすぎる場合、文字列全体のつながりがうまく表現できない、２）用いる長さが長すぎる場合、反対に効率のよい表現ができない。また、サンプル間の共通性が部分列として文字列の内部にある場合を想定した場合、先行語ならびに後続語として最大長を取ることは、その部分列の前後の文字列も考慮する事を意味し、特徴的な部分列を抽出することは難くなる。実際に、その傾向は例にも現れている。

３．３．２．　逐次学習

推論手法を推論結果だけでなく、実用性の面でも考察を行う事は重要である。

はじめに、逐次学習（sequential learning ）を考慮する。これは新しいサンプルを付加して推論する時に、はじめから推論し直さなくてもオートマトンを構成できるかどうかを問題とする。これが可能ならば、推論結果のオートマトンを評価しながらサンプルを加えていき、満足できる結果が求まるまでその操作を繰

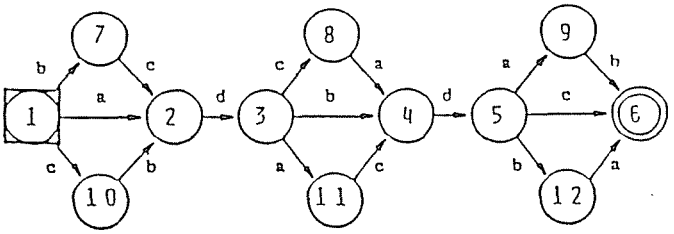

図３．６（ａ）　例３．４のオートマトン
（モデル、＋（４）法）

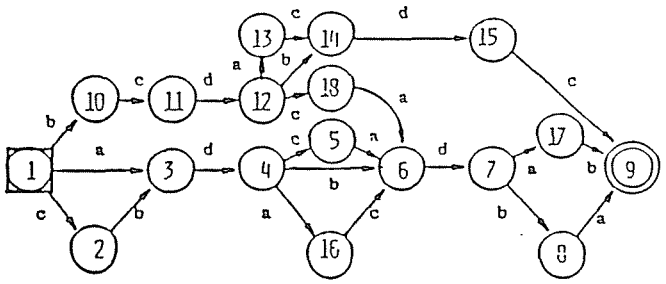

図３．６（ｂ）　例３．４のオートマトン
(tail-clustering法)

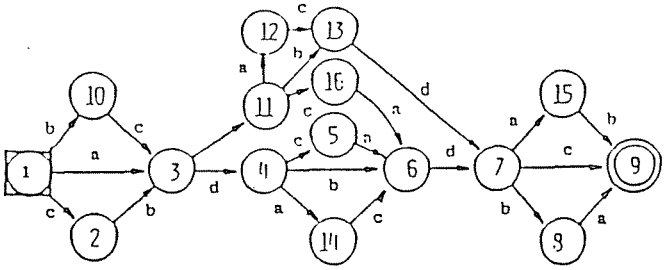

図３．６（ｃ）　例３．４のオートマトン
（∨（４，４）法）

返すなどが効率よく行える。

アルゴリズム３．１において、ｓｔｅｐ３の集合的操作を行わなければ、このアルゴリズム上のどの手法も以下の様な手順で逐次学習を行える。

まず、導出オートマトンに必要なものを考察すると、

１）推移標準オートマトン

２）同値関係Ｒによる分割

の二つである。残りのステップはすぐに行える。そこで、それぞれの逐次学習を考察する。S^+に新しいサンプルｘが付加された場合、１）の推移標準オートマトンは推移文x^*を付加するだけでよい。次に、同値関係Ｒによる分割に関しては、次の三つの操作のいずれかを、推移文x^*中の各推移記号Ｚと現在迄の分割における各ブロックの各要素との比較結果に応じて行う：

ａ）既存のブロックの一つにＺを入れる

ｂ）新しいブロックをつくり、そこにＺを入れる

ｃ）既存の複数のブロックを一つにまとめ、そこにＺを入れる。

どの操作をするかは同値関係Ｒによって異なる。もし、Ｖ（ｍ，ｎ）法ならば、長さｍの先行語と長さｎの後続語をＺと現在のすべての状態（推移記号全体）に関して調べ、Ｚと先行語あるいは後続語が一致する状態がすべて一つのブロックに含まれているならば、そのブロックにａ）の操作を行う。先行語で関係の成立するブロックと後続語で関係の成立するブロックが異なる時はｃ）の操作を行なう。関係がどのブロックとも成立しない場合ｂ）の操作を行う。他の手法に関しても同様な操作を同値関係Ｒに応じて行う事ができる。

３．３．３．　確率オートマトン

次に、確率オートマトン（stochastic automaton）の構成を考える。これは受理する文字列に確率を付与するもので、複数のクラスのオートマトンが同一文字列を受理する場合、受理確率の高いクラスをその文字列に割り当てるなどの判定が行える。また、閾値を定めて、受理文字列を制限する事も可能であり、主に、構文的なアプローチの本質的な欠陥である受容域の狭さを補償する目的に用いられる。

提案手法のどれに関しても、successor法［９］により示されたのと同様な方法により確率オートマトンの構成は可能である。手続きの詳細は文献［９］に譲る。successor法の確率オートマトンとの違いは、前者が決定性オートマトンを生成するのに対し、後者は非決定性オートマトンを生成する点である。決定性の場合は受理される文字列は唯一のパス（path）を通って受理されるが、非決定性の場合は同一文字列が異なるパスを通って受理される可能性がある。従って、すべてのパスの受理確率の最大値をもってその文字列の受理確率とする必要がある。

３．４．　推論目的と手法の分類

第３．３節で議論した様に、手法の性質は主に先行語、後続語の長さに見る事ができる。手法の両極端にsuccessor法とtail-clustering法があり、提案手法は長さをパラメータとして持ち、それらの中間的な性質を有する。パラメータを調節して最適な推定を行う事を考えた場合、客観的な評価規準が必要となる。これには通常、負サンプル、状態数などを用いる事が可能である。しかし、パラメータに無関係にＡ（ｍ，ｎ）法とＶ（ｍ，ｎ）法の間には大きな違いがある。例えば、サンプル数を増やした時、Ａ（ｍ，ｎ）法は状態数が減少する事はないが、Ｖ（ｍ，ｎ）法ではあり得る。これは推移的閉包の操作により、新なサンプルを介して従来の複数の状態が一つにまとまる事があり得るからである。また、推論面での特徴として、Ａ（ｍ，ｎ）法は比較的小さいｍ，ｎで文字の繰返しを効率よく表現する反面、ｍ、ｎを大きくすると関係が殆ど成立しなくなり、受理する言語が小さくなる。対照的にＶ（ｍ，ｎ）法はあまり小さいｍ，ｎにおいては関係が成立し過ぎるが、比較的大きなｍ，ｎで文字列の類似性を最大限に活用する。実用的な推論目的もこれらの性質に対応する。従来の推論目的は

１）効率的な推論を目的とするもの、つまり、サンプルの構造を反映しつつ、できるだけ状態数の少ないオートマトンを求める事を目的とするもの

２）忠実な推論を目的とするもの、つまり、サンプルを生成した文法規則が存在するとして、その忠実な再現を求める事を目的とするもの

の二つに大別する事ができる。この二つの目的は互に相容れない部分を持つ。例

えば、すべてのサンプルが３個以上の”ａ”から始まる文字列である場合、１）の目的には”ａ”のループを一つ作ればよい。しかし、２）の目的には”ａ”を３個以上つなげた部分が必要とされるであろう。より実際的には、すべてのサンプルが３個以上の”ａ”から始まるが、個数の変動が著しいならば、１）で十分であろう。しかし、変動が少なかったり、他のクラスのサンプルも”ａ”の繰返しで始まるけれども、その個数が３個未満であれば、むしろ２）の目的が重要であろう。また、偶数個の”ａ”で始まる文字列が推論すべき言語であれば、１）の目的は的はずれになる。簡単な選択法としては、サンプルを効率よく処理する為の文法推論なら１）、サンプルが生成された規則が存在すると思われ、その推論を目的とするなら２）を考えればよい。

これらの事実と前節の実験例を考え合わせると、推論における手法の性質を考慮して推論目的別に分類するのが有用と思われる。上述の二つの目的に沿って、従来法と提案手法を分類した結果は以下の様になる：

[効率的な推論に適する手法]

predecessor and successor法、∧（ｍ，ｎ）法

[忠実な推論に適する手法]

tail-clustering法、∨（ｍ，ｎ）法、＋（ｋ）法

３．５．　結論

本章では正規文法推論問題を扱った。その際、従来手法の推論に関する特徴を明らかにする為、同一アルゴリズム上で従来手法の幾つかを再構成した。その結果、手法の推論特徴が標準オートマトンの状態間に定義される同値関係として捉えられ、特に先行語あるいは後続語の長さとその扱い方の差異として明確になった。先行語ならびに後続語から考慮される手法の特徴と幾つかの実験例から、主な従来法中、predecessor and successor法がサンプルの局所的な類似性の表現に優れ、効率のよい文法を推論する反面、大局的な表現が不十分である事、tail-clustering法がそれと全く対照的な性質を有する事が判明した。また、それらの方法がそれぞれに極端な場合であり、それに対応した欠点も明らかなので、

新しく、predecessor and successor法の型の拡張として∧（m, n）法を、tail-clustering法の型の拡張として∨（m, n）法、＋（k）法を提案した。また、これらの手法により、特にサンプル文字列間の特徴がサンプル文字列内部の特定部分列として存在する場合、その特徴を自然に抽出できるようになった。今後の課題として、パラメータm, n, kの最適な設定の問題が残される。現在では負サンプル、状態数などをその指標としている。これに関しては、より明確な設定法を考察する必要がある。

第４章　遺伝子の特徴的配列の解析

４．１．　序論

近年、遺伝子工学の発達に伴い急速に遺伝子配列データが集められている。しかし、その膨大な資産を解析する手段は十分とは言えない。遺伝情報を担っているのは遺伝子であり、遺伝子の２本鎖のうち１本を取り出すと４種の塩基（Ａ：アデニン、Ｃ：シトシン、Ｇ：グアニン、Ｔ：チミン）を文字としてもつ文字列と見なす事ができる。この枠組みにおいては種々の情報を文字列から読み取る事が問題となり、情報論的アプローチも有効な手段になり得る。

パターン認識の面からも遺伝情報を扱う事には以下の様な利点がある。構文的パターン認識の有効性は主にある種の構造解析にあり、通常文字列表現されたパターンの生成規則を解析する。その際、個々の文字はプリミティブ、つまり局所的な構造を表している。しかし、実際のプリミティブは各研究者毎に決められる為に人工的で、本来の構造を最適に表現するプリミティブが取られているという保証はない。一方、遺伝情報は４種の塩基がプリミティブとなっており、全く自然な文字列である。その意味で、遺伝子配列は文字列としてのパターン表現が自然な対象で、構文的アプローチを無理なく導入できる。

遺伝子情報の働きを大きく分けると、１）自己のもつ遺伝情報を子孫に伝える事、２）自己のもつ遺伝情報を形や性質に発現する事、３）機構の制御をする事、の三つに分けられる。本論文では特に３）の制御に関する情報を扱う。具体的には、転写やタンパク質合成などの過程においては長い塩基配列のうちの一部分だけが処理対象となる。その場合、処理部分の開始位置、終了位置がどの様な信号で確定されているかが問題となる。現在、その様な位置は複数見つかっている。しかし、具体的に信号の知られているものは殆どない。また、ある配列が位置確定の信号となっているとしても、より進んで、どの様な機構がどの様に位置を確定し、処理を行うかも重要な問題である。

位置を規定する情報が塩基配列そのものに含まれていると考えるのは自然であり、実際、特定の位置における周辺配列を多数集めて解析を行っている研究が多数見られる。しかし、その殆どは位置毎にどれ程の頻度で塩基が出現するかを考慮するもので、それに基く識別はどちらかといえば決定理論的パターン認識を行っていると言える。これは信号が文字列のつながり具合によっている場合には十分にそれを解析できない。実際の生物のスプライスサイト認識機構が前者の様に特定の位置に特定の塩基があるかどうかで位置を確認しているか、後者の様に塩基のつながりを調べて確認しているかは不明である。本章では、後者の立場に立って、構文的パターン解析を用いてある種の開始位置周辺配列の解析を試みる。

4．2．　mRNAスプライシング

DNA上の遺伝情報はタンパク質の合成を指令している。そして、DNAからタンパク質への遺伝情報はRNAによって伝達される。まず、遺伝情報がmRNAへ転写され、mRNAが各種のプロセッシングを受けて成熟mRNAになる。成熟mRNAの情報がtRNAの助けを借りてタンパク質へと翻訳される。このプロセッシングの一つがスプライシング（つなぎ合わせ）と呼ばれるものである。mRNAは実際にタンパク質をコードしている部分（エクソン）ばかりでなく、最終的に取り除かれる部分（イントロン）も含んでおり、スプライシングとは介在しているイントロンを取り除き、隣合ったエクソンをつなぎ合わせる処理をいう（図4．1）。ここで、エクソンーイントロン境界の事をドナーサイト（5'ースプライスサイト）、イントロンーエクソン境界の事をアクセプターサイト（3'ースプライスサイト）と呼ぶ。

実際にスプライシングを起こしている箇所は成熟mRNAと元のDNAを比べる実験により判定できる。これまで多数のドナーサイトあるいはアクセプターサイトの周辺配列が調べられており、高等真核生物ではかなりの類似性が認められている。また、下等真核生物のスプライス信

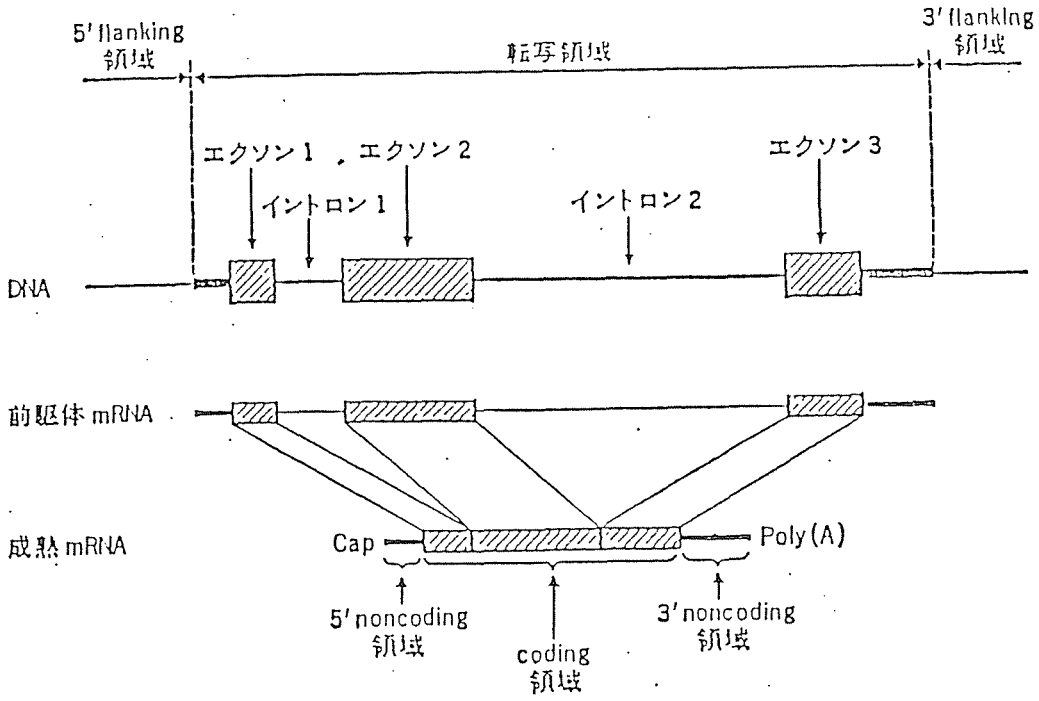

図4．1　β－グロビン遺伝子の構造の模式図

表4．1　ドナーサイト周辺配列のサンプル例

Gene		Sample strings[b]
Human α-globin	IVS1	GAGAG/GTGAGGCTCC
	IVS2	TCAAG/GTGAGCGGCG
Rabbit β-globin	IVS1	GGCAG/GTTGGTATCC
	IVS2	TCAGG/GTGAGTTTGG
Mouse β-globin	IVS1	GGCAG/GTTGGTATCC
	IVS2	TCAGG/GTGAGTCTGA
Rat cytochrome C	IVS1	GAAAG/GTAAGTGTGG
Goat α_1-globin	IVS1	GAGAG/GTGAGCACCG
	IVS2	TTAAG/GTGAGCTCGC
Mouse α-amylase	IVS1	ACATG/GTATGTAATC
	IVS2	AAATG/GTGAGATTTC
	IVS3	TGCAG/GTGTGCAGGT

[a]Genes taken from Gen Bank Library (Genbank, 1987).
[b]A stroke (/) indicates the boundary between exon and intron.

号は高等真核生物のそれと幾らか異なる事が報告されている［８、１３］。高等真核生物におけるイントロンはほぼ１００％ＧＴで始まりＡＧで終わる（ＧＴ－ＡＧ規則）。しかし、この規則だけでは実際の生物のスプライスサイトを確定できず、周辺配列の相同性が Mountら［９，１０］によって百数十の配列により調べられ、コンセンサス配列が提出されている。ドナーサイトに関しては、

（C/A）ＡＧ／ＧＴ（A/G）ＡＧＴである。ここで、”／”はドナーサイトを示す。しかし、これは位置毎に塩基の発生頻度を測定して、特に頻度の高い塩基を並べたに過ぎず、現実のドナーサイト周辺配列では大多数が９塩基の幾つかが異なるものになっている。従って、コンセンサス配列はどの程度の合致を要求するのかといった曖昧さを残し、未知のドナーサイトの検出に用いる事は殆ど不可能である。これに対する改善が重み行列法［１４］、パーセプトロンアルゴリズム［１１］、数量化２類による解析［４］などにより試みられているが、まだ十分にドナーサイトを確定するとは言えない。また、配列パターンとしての直接の発展が Iida and Sasaki［２］によって行なわれ、ドナーサイトに関して４パターン、ＡＧ／ＧＴＡ、／ＧＴＡＡＧＴ、ＲＧ／ＧＴＧＡＧ、ＡＧ／ＧＴＸＸＧＴが提出されている。ここで、ＲはＡまたはＧ、Ｘはどの塩基でもよい事を示す。しかし、これらのパターンも未だ十分にドナーサイトを確定するには至っていない。

本章では、これらの統計的手法がパターンの横のつながりを十分反映できない事を踏まえて、ドナーサイト周辺の配列からその様な特徴を抽出する事を考え、ドナーサイト認識装置として有限オートマトンを構成する。

４．３．　正規文法推論手法の応用（Kudo et al.，［５］）

４．３．１．　適用方法

ドナーサイト周辺配列を文字列パターンと見なし、正規文法を推論する事を試みる。この際、手法としては他の手法に比べてパターンに共通な特徴が文字列の中に部分列として存在する際に、その特徴を的確に捉えられるV（m, n）法（第２章参照）を用いる。なぜなら、どの範囲が信号配列として用いられているかがはっきりしない為、信号を表現すると予想される範囲を大きめに取る必要があるからである。本解析においては、ドナーサイトを中心にエクソン側５塩基、イントロン側１０塩基の計１５塩基配列をパターンとして切り出した（表４．１）。コンセンサス配列の範囲の外側では相同性が殆どない事が確認されており、生物がスプライシングに関して、コンセンサス配列の内部の変異には弱くイントロン内部の欠損には強い事が塩基の変異調査から報告されている［３、１５、１６］。これらの事実から判断して、コンセンサス配列に対応する部分を完全に含んでいるこの切り出し範囲は適当と思われる。次に、解析に用いた条件をまとめる。

（解析条件）
パターン長：１５（エクソン側５塩基、イントロン側１０塩基）
サンプル　：１５６個の哺乳類の遺伝子のドナーサイト周辺配列
　　（データベース Genbank(1987)［１］より採取）
手法　　：V（m, n）法を用いて確率オートマトンを構成

一つの遺伝子は複数のイントロンを含み、結果としてドナーサイトも複数存在する。一例として Human-βの遺伝子に関して、その全体配列を図４．２に示す。

４．３．２．　結果

推論により構成された有限オートマトンの性能を調べる為に、次の２タイプの全体配列を検査した。
１）サンプルとして用いた配列を含む１３遺伝子（３２イントロン）の

```
          1         2         3         4         5         6         7         8         9         10
     ....+....0 ....+....0 ....+....0 ....+....0 ....+....0 ....+....0 ....+....0 ....+....0 ....+....0 ....+....0
   0 CCCTGTGGAG CCACACCCTA GGGTTGGCCA ATCTACTCCC AGGAGCAGGG AGGGCAGGAG CCAGGGCTGG GCATAAAAGT CAGGGCAGAG CCATCTATTG

 100 CTTACATTTG CTTCTGACAC AACTGTGTTC ACTAGCAACC TCAAACAGAC ACCATGGTGC ACCTGACTCC TGAGGAGAAG TCTGCCGTTA CTGCCCTGTG

 200 GGGCAAGGTG AACGTGGATG AAGTTGGTGG TGAGGCCCTG GGCAGGTTGG TATCAAGGTT ACAAGACAGG TTTAAGGAGA CCAATAGAAA CTGGGCATGT
                                      ******************************************************************************

 300 GGAGACAGAG AAGACTCTTG GGTTTCTGAT AGGCACTGAC TCTCTCTGCC TATTGGTCTA TTTTCCCACC CTTAGGCTGC TGGTGGTCTA CCCTTGGACC
     ********** ********** ********** ********** ********** ********** ********** ******

 400 CAGAGGTTCT TTGAGTCCTT TGGGGATCTG TCCACTCCTG ATGCTGTTAT GGGCAACCCT AAGGTGAAGG CTCATGGCAA GAAAGTGCTC GGTGCCTTTA

 500 GTGATGGCCT GGCTCACCTG GACAACCTCA AGGGCACCTT TGCCACACTG AGTGAGCTGC ACTGTGACAA GCTGCACGTG GATCCTGAGA ACTTCAGGGT
                                                                                                          **

 600 GAGTCTATGG GACCCTTGAT GTTTTCTTTC CCCTTCTTTT CTATGGTTAA GTTCATGTCA TAGGAAGGGG AGAAGTAACA GGGTACAGTT TAGAATGGGA
     ********** ********** ********** ********** ********** ********** ********** ********** ********** **********

 700 AACAGACGAA TGATTGCATC AGTGTGGAAG TCTCAGGATC GTTTTAGTTT CTTTTATTTG CTGTTCATAA CAATTGTTTT CTTTTGTTTA ATTCTTGCTT
     ********** ********** ********** ********** ********** ********** ********** ********** ********** **********

 800 TCTTTTTTTT TCTTCTCCGC AATTTTTACT ATTATACTTA ATGCCTTAAC ATTGTGTATA ACAAAAGGAA ATATCTCTGA GATACATTAA GTAACTTAAA
     ********** ********** ********** ********** ********** ********** ********** ********** ********** **********

 900 AAAAAACTTT ACACAGTCTG CCTAGTACAT TACTATTTGG AATATATGTG TGCTTATTTG CATATTCATA ATCTCCCTAC TTTATTTTCT TTTATTTTTA
     ********** ********** ********** ********** ********** ********** ********** ********** ********** **********

1000 ATTGATACAT AATCATTATA CATATTTATG GGTTAAAGTG TAATGTTTTA ATATGTGTAC ACATATTGAC CAAATCAGGG TAATTTTGCA TTTGTAATTT
     ********** ********** ********** ********** ********** ********** ********** ********** ********** **********

1100 TAAAAAATGC TTTCTTCTTT TAATATACTT TTTTGTTTAT CTTATTTCTA ATACTTTCCC TAATCTCTTT CTTTCAGGGC AATAATGATA CAATGTATCA
     ********** ********** ********** ********** ********** ********** ********** ********** ********** **********

1200 TGCCTCTTTG CACCATTCTA AAGAATAACA GTGATAATTT CTGGGTTAAG GCAATAGCAA TATTTCTGCA TATAAATATT TCTGCATATA AATTGTAACT
     ********** ********** ********** ********** ********** ********** ********** ********** ********** **********

1300 GATGTAAGAG GTTTCATATT GCTAATAGCA GCTACAATCC AGCTACCATT CTGCTTTTAT TTTATGGTTG GGATAAGGCT GGATTATTCT GAGTCCAAGC
     ********** ********** ********** ********** ********** ********** ********** ********** ********** **********

1400 TAGGCCCTTT TGCTAATCAT GTTCATACCT CTTATCTTCC TCCCACAGCT CCTGGGCAAC GTGCTGGTCT GTGTGCTGGC CCATCACTTT GGCAAAGAAT
     ********** ********** ********** ********** *********

1500 TCACCCCACC AGTGCAGGCT GCCTATCAGA AAGTGGTGGC TGGTGTGGCT AATGCCCTGG CCCACAAGTA TCACTAAGCT CGCTTTCTTG CTGTCCAATT

1600 TCTATTAAAG GTTCCTTTGT TCCCTAAGTC CAACTACTAA ACTGGGGGAT ATTATGAAGG GCCTTGAGCA TCTGGATTCT GCCTAATAAA AAACATTTAT

1700 TTTCATTGCA ATGATGTATT TAAATTATTT CTGAATATTT TACTAAAAAG GAATGTGGGA GGTCAGTGCA TTTAAAACAT AAAGAAATGA TGAGCTGTTC

1800 AAACCTTGGG AAAATACACT ATATCTTAAA CTCCATGAAA GAAGGTGAGG CTGCAACCAG CTAATGCACA TTGGCAACAG CCCCTGATGC CTATGCCTTA

1900 TTCATCCCTC AGAAAAGGAT TCTTGTAGAG GCTTGATTTG CAGGTTAAAG TTTTGCTATG CTGTATTTAC ATTACTTATT GTTTAGCTGT CCTCATGAAT

2000 GTCTTTTCAC TACCCATTTG CTTATCCTGC ATCTCTCTCA GCCTTGACT
```

図４．２　Human β－グロビン遺伝子の塩基配列

（”＊”はイントロン部を示す）

全体配列、

2）サンプルとして用いた配列を含んでいない7遺伝子（20イントロン）の全体配列。

実験は手法の持つパラメータを幾つか変えて正規文法推論（有限オートマトンの構成）を行った。また、構成したのは確率オートマトンなので、受理パターンは確率値を有する。そこで、真正なドナーサイトとしての判断は受理確率がある閾値以上である事をその条件とした。また、閾値は全訓練パターンの受理値における最小値を用いた。結果を表4．2に示す。また、特にV（9、9）法により構成された有限オートマトンを図4．3に示す。ここで、実際に構成されたオートマトンは非決定性であるが、見やすさの為に決定性に変換した図を載せている。また、オートマトンの性能の評価として、次式で表される検出率、識別率を用いた。

$$\text{検出率} = \frac{\text{正しく検出されたドナーサイトの個数}}{\text{存在するドナーサイトの個数}} \qquad (6.1)$$

$$\text{識別率} = \frac{\text{正しく検出されたドナーサイトの個数}}{\text{存在ドナーサイトの個数} + \text{誤発見したエクソン部の個数}} \qquad (6.2)$$

この識別率で、誤発見したイントロン部の個数を考慮しなかったのは次の説による。「ドナーサイト識別機構は5'端から3'端へと順に配列を走査すると思われる（図4．4）。その場合、一旦ドナーサイトを識別した後その機構は不活性化し、逆にアクセプターサイト認識機構が活性化し、アクセプターサイトを捜す。従って、アクセプターサイトを発見するまでの間はドナーサイト候補は見つけられても無視される」。走査の方向性の仮定を支持するいくつかの報告［6、7、11］がなされている。

解析の結果、V（8、9）法、V（9、9）法によって、サンプルを含んだ遺伝子の検査において、識別率94%－97%、未知の遺伝子に

表４．２ V（m，n）法による識別結果

手法	閾値 ($\times 10^{-10}$)	訓練サンプルを含んだ遺伝子				未知の遺伝子				
		接合部位	エクソン	イントロン	識別率（%）	接合部位	エクソン	イントロン	識別率（%）	検出率（%）
V（8，8）	61	32	61	59	34	14	38	60	24	70
V（8，9）	337	32	2	12	94	11	0	3	55	55
V（9，9）	381	32	1	6	97	10	0	4	50	50
V（10，10）	1640	32	0	3	100	7	0	0	35	35

識別率は式（６．１），検出率は式（６．２）を参照

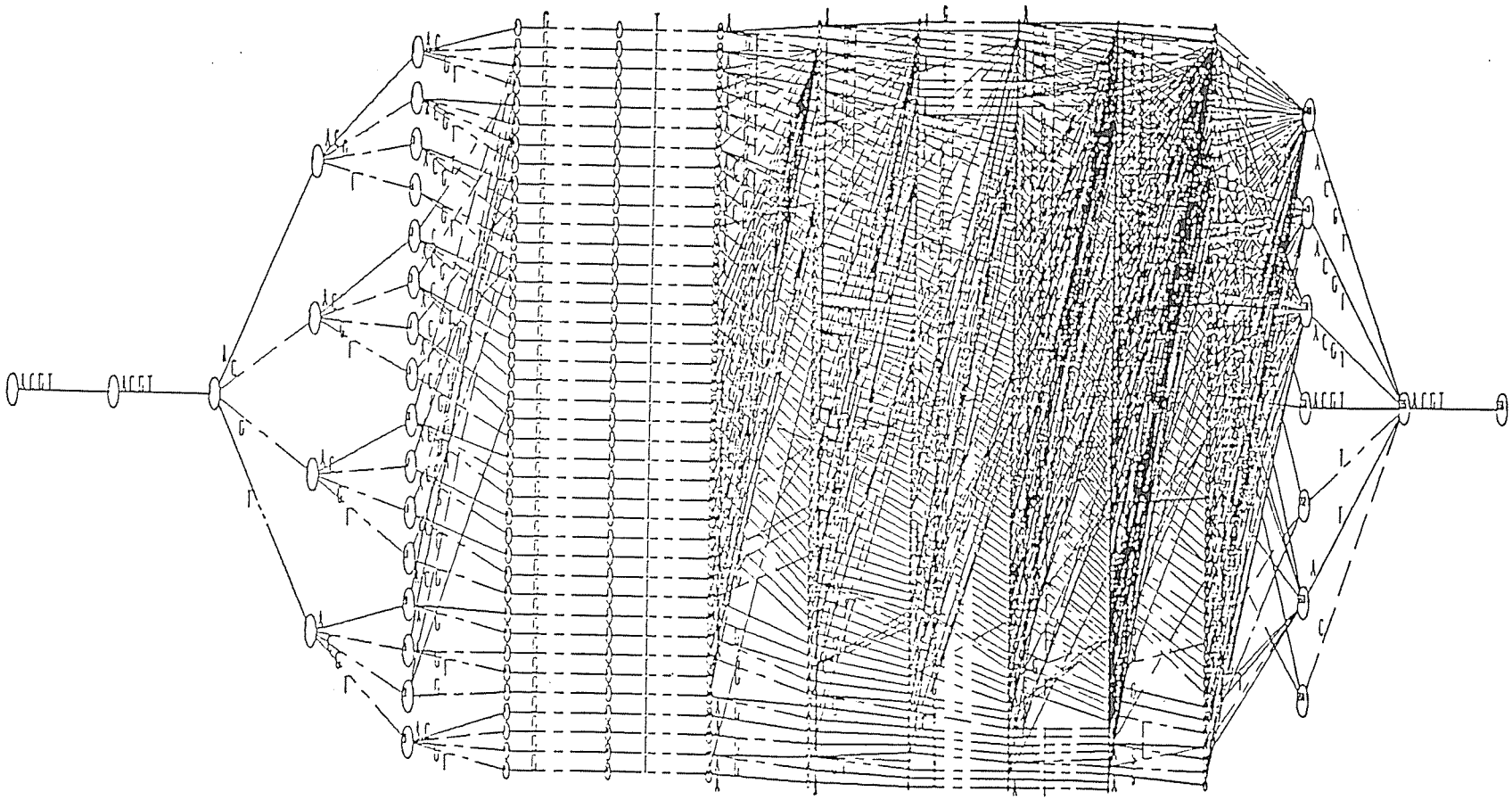

図4．3　∨（9，9）法により構成されたオートマトン

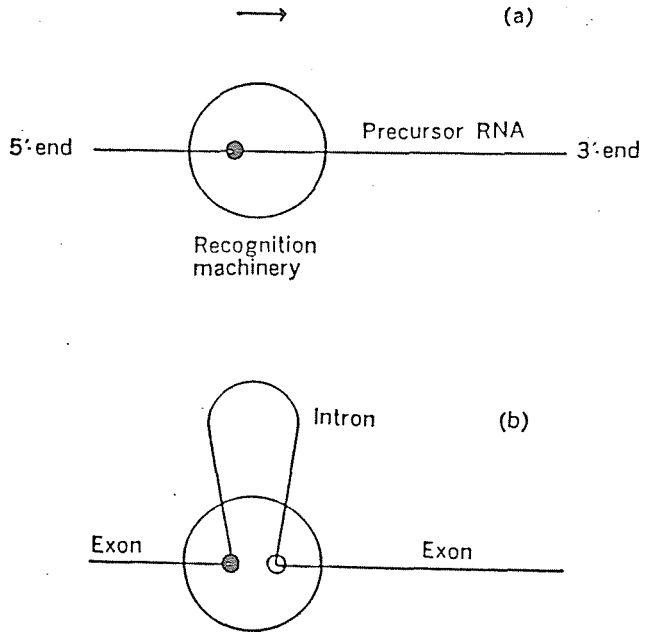

図４．４　スプライスサイト認識機構のモデル
（Iida and Sasaki[2]より引用）

関しては、識別率５０％－５５％、検出率５０％－５５％を達成した。これらの値は手法毎に用いた遺伝子および評価方法などが異なり、直接の比較はできないが、従来のものと比べて同程度かそれ以上の識別性能を有すると言える。

この解析結果は特殊な文字列がドナーサイトを規定する信号になっている可能性が高い事を示している。従来の解析が特定の位置に特定の文字がある事をもって信号である可能性を求めたのに比べると、識別結果からは未だどちらの形が実際の信号であるかは結論づけられない。また、これらの解析はいずれも信号がドナーサイト周辺配列にあるとの仮定で行われているが、生物の機構はより複雑に様々な要因に基いて位置を確定しているかもしれない。加えて、同一の配列が異なった仕方でスプライスされる事例も確認されており、その場合ドナーサイトの位置が変化する為、位置確定信号はそれほど決定的なものでなくなる。

4．3．3．　Mutation の実験

正常なスプライス信号配列が存在していたのに、遺伝病などの突然変

異によりある塩基の置換や欠落が起こり、信号配列が改変され、認識されなくなる場合がある。変異の例として、Human β-globin の第1ドナーサイトを取り上げる。正常な周辺配列はＣＡＧ／ＧＴＴＧＧＴであり、生物認識機構も前節で構成されたオートマトンもこの配列を正しく認識する。この配列には３種の変異体（ＣＡＧＡＴＴＧＧＴ、ＣＡＧＧＴＴＧＣＴ、ＣＡＧＧＴＴＧＧＣ）（下線部は変異箇所を示す）が知られている。いずれの場合も生物認識機構は変異体を認識せず、正常なｍＲＮＡを生成しない。反対に、正常な遺伝子では信号配列でなかった場所に変異が起こり、信号を発生した例もある。この場合、生物認識機構もこの信号を誤認して間違ったｍＲＮＡを生成する。

この様な例は多数報告されており、それらに対して前節のオートマトンと生物認識機構とを比較して働きの差異を調べた。変異体を４０例解析した。Ｖ（８、９）法、Ｖ（９、９）法の結果を表４．３に示す。正答率はそれぞれ８９％、９２％であった。

特定の塩基の置換、欠落によって生じる影響は位置毎の塩基の重要性を測った従来の方法では、重要な位置に関しては影響が大きいが、それ以外のところでの影響は小さい。それに比べて、特徴的な塩基配列に基く本章の解析においては、一つ塩基の置換、欠落も影響は大きい。今回の実験による精度の良さは生物の認識機構とオートマトンとの整合性が高い事を示している。今後、他の手法との比較実験が望まれる。

４．４．　結論

遺伝子の特徴的な配列（スプライス信号配列）の解析に構文的なアプローチを適用した。これは従来の手法に比べ、文字の横のつながりが特徴である時に有効であり、今回の実験結果から生物がその様な形で信号を保持している可能性が示された。また、Mutationの実験からも推定された信号規則がかなり実際の生物の挙動に合致する事が確かめられた。

実際問題としてスプライスサイトの未知な遺伝子配列が多数あり、それらの位置確定に関して生物に関する研究分野からの需要は高く、その

表４．３　Mutationの実験結果（正誤表）

	遺伝子名	配列	有効（無効）	V（8、9）法 受理	V（8、9）法 正誤	V（9、9）法 受理	V（9、9）法 正誤
		↓					
1	Human β-globin	TCAGG GTGAGTCTAT	○	○	正	○	正
2	IVS 2	A	×	×	正	×	正
3	Human β^E-globin	TGGTG GTGAGGCCCT	×	×	正	×	正
4		A	○	○	正	○	正
5		A	○	×	誤	×	誤
6		T	○	○	正	○	正
7	無血糖　rat	CAGTG GTAGGTTTCC	○	○	正	○	正
8		CGAGCT	×	×	正	×	正
9	Ad 2 E1A	AGAGG GTGAGGAGTT	○	○	正	○	正
10		G	×	×	正	×	正
11		AT	×	×	正	×	正
12	Ad 2 pm 1114	CTACA GTAAGTGAAA	○	○	正	○	正
13	and dl1500	T	×	×	正	×	正
14		G CT AA T ATGG	×	×	正	×	正
	Human β-globin thalassemia						
15	(745)	TCCAG GTACCATTCT	○	×	誤	×	誤
16		C	×	×	正	×	正
17	(705)	CTGAG GTAAGAGGTT	○	○	正	○	正
18		T	×	×	正	×	正
19	(654)	TTAAG GTAATAGCAA	○	×	誤	×	誤
20		C	×	×	正	×	正
21	Human β-globin	GGCAG GTTGGTATCA	○	○	正	○	正
22	thala. IVS-1	A	×	×	正	×	正
23		C	×	×	正	×	正
24		C	×	×	正	×	正
25	Chicken ovalbumin	GAAAG GTAAGCAACT	○	○	正	○	正
26	polymorphism	G	○	○	正	○	正
27	Human α-thala.	GAGAG GTGAGGCTCC	○	○	正	○	正
28		CTCCCTC	×	×	正	×	正
29	Rabbit β-globin	TCAGG GTGAGTTTGG	○	○	正	○	正
30		G	○	○	正	○	正
31		A	○	○	正	○	正
32		A	○	○	正	○	正
33		A	×	×	正	×	正
34		A	○	○	正	○	正
35		G	○	○	正	×	誤
	Rabbit β-globin						
36	(mutation)	C CT	○	○	正	○	正
37	Human $^G\gamma$-globin	TCAAG GTGAGTCCAG	○	○	正	○	正
38	(deletion)	AGCT	○	○	正	○	正
正解率					92%		89%

＊　V（8、9）法のしきい値　$337*10^{-18}$
　　V（9、9）法のしきい値　$381*10^{-18}$

意味でも今回構成したオートマトンは有効である。

現在構成したオートマトンの認識精度は十分なものとは言えない。これに関しては、一つの原因としてデータ面での制約が考えられる。現在の解析データはデータベースから解析可能な殆どの遺伝子配列を用いているものの、十分な量とは言えない。今後さらにデータが増える事により、より正確な識別を行うオートマトンの構成が見込まれる。

第５章　二値的特徴を持つパターンの解析

５．１．　序論

パターンを文字列とする時、同じクラスのパターン間の共通性の表現には大きく二通りの表現が考えられる。一つは第３章で扱った”横のつながり”としての表現である。これは特にパターン長が可変の場合有効である。それに対し、本章ではパターン長一定の場合のもう一つの表現、”縦の共通性”の表現を扱う。

例として、同じクラスから取った四つのサンプルパターンを考える：

１：ＢＣＢＣＣＡ

２：ＡＣＢＢＡＣ

３：ＢＣＢＣＡＡ

４：ＢＢＤＣＤＡ

この時、共通性の表現として

１）＊ＣＢ＊＊＊　　（パターン１、２、３に共通）

２）Ｂ＊＊Ｃ＊Ａ　　（パターン１、３、４に共通）

が考えられる。ここで”＊”は任意の文字と合致するとする。また、各表現に対しその表現に合致するパターン集合を考えた場合、集合の包含関係において極大となる表現のみを抽出した。従って、表現　ＢＣＢＣ＊Ａ　などはパターン１、３と合致するけれど、これは表現１）に合致するパターン集合に含まれるので抽出しない。この”極大性”の要請は冗長度を減ずる為にも、また、積極的になるべく大きな共通性を求める目的にも自然であると思われる。

この様な共通性をパターン認識の問題に持ち込む時にはさらに次の点が考慮される。いくら共通であってもその共通性がほかのクラスと区別的でない場合は意味がない。例えば、他のクラスのパターンが

５：ＢＣＤＣＢＡ

のみの場合、２）の表現ではパターン５にも合致してしまう。従って、要請として、

○　排他的な表現で極大にサンプルパターンと合致するもの

が望まれる。この形の表現を　Stoffel［２］は”prime event”と呼び、それを

求める基本的なアルゴリズムを提案した。上の例で発見される prime eventは表現１）とパターン４自身の表現

３）ＢＢＤＣＤＡ

である。

本研究ではより積極的に表現の”効率化”を考察する。これは、同じクラスに属するパターン間の共通性に対する表現としてはすべての共通な文字を用いる必要はないという主張である。実際、上の例における１）の表現はパターン５を考慮すると　＊＊Ｂ＊＊＊　で十分である。これは、多数のサンプルならば消える様な共通性も、少数サンプルの場合は残る可能性が高いので、「できるだけ少数の共通特徴（共通文字）で排他性を維持」する事を考え、それによりサンプル数の少なさの補償を行う事を考慮するからである。それを目的とした場合、上の例において得られる表現は

４）＊＊Ｂ＊＊＊

５）＊Ｂ＊＊＊＊　または　＊＊＊＊Ｄ＊

となる。この試みは”過度”の推論を行う可能性が高い。しかし、多クラスの問題ではクラス毎にこの様な推論を行うと、複数のクラスが推論結果において共通部分を有する事になり、むしろその部分に対する処理が問題になる。また、完全に排他的、つまり他クラスの一つのサンプルにさえも合致を許さないと言うのは、他クラスの訓練サンプル数がさほど大きくない場合でさえ十分きつい制約と思われ、その面での抑制がかなり働くと考えられる。

この様にして得られた表現は、その表現に合致するすべての文字列の集合を考えると、構文的なアプローチの場合と同様にある種のクラスを定義する。しかし、このクラスは元来一つのクラスのサンプルから得られている。とすれば、”部分クラス”と考えるのが妥当であろう。すべての訓練サンプルが一つの表現に合致するならば、部分クラスはそのまま一つのクラスになり、理想的な状態と言える。しかし、一般には一つのクラスは複数の部分クラスから構成されると思われる。

Stoffelの”prime event 理論”はサンプル数に関して指数的な計算量を必要とするので、その原型のアルゴリズムは実用的ではなかった。また、近似的に線形のアルゴリズムも提案しているが、１）発見しない prime eventがある、２）発見した prime eventは必ずしも極大性を維持しないなど、本質的な達成度が不十

分である。本章では Stoffelの”prime event”を訓練サンプル集合の”排他的な極大部分集合”として捉え、さらに表現の効率化の結果に”部分クラス”という名を与えて、それらを求める効率的アルゴリズムを提案する。また、Stoffelは特徴を離散値としているのに対し、本研究ではすべての特徴を二値として考察する。この事が本質的な制約でない事は明らかであろう。

部分クラスを考える根拠はさらに一般の場合に求められる。人間のパターン認識の過程を想定した場合、与えられたパターンのすべての特徴を見てクラスを割り当てるとは思われない。現下のパターンが有する各特徴の中から最もそのクラスに属する根拠を示す特徴あるいは特徴の組を発見し、その特徴だけで不十分な時はさらに別な特徴も併せて判断するかも知れない。あるいは、最初の特徴や特徴の組をもはや見ずに、別な複数の特徴の組で判断する可能性もある。これらが部分クラスを導入する根拠である。一つには、「一つのクラスを規定する特徴は少数で、特徴集合全体の一部である」という予想である。その場合、多クラスの問題ではクラス毎には少数の特徴でクラスを規定できても、全体として多数の特徴を要する事があり得る。また、大きな部分クラス、すなわち一つのクラスの大部分のパターンを含む部分クラスは特に少数の特徴により規定されると予想される。なぜなら、一般に個体数が多くなればなるほど、すべての個体に共通な特徴は減ずるから。従って、より多くのパターンを含む部分クラスは自然に特徴選出を行うと思われる。識別に用いる特徴は少ない方が効率が上がるから、これはまた識別の効率化にもつながる。もう一つの側面として、「一つのクラスのあるパターンは複数の異なる根拠からそのクラスに属すると判断し得る」という事も考えられる。つまり、ある特徴の組によりパターンに対するクラスの割当が行われる場合、別の特徴の組によっても同じクラスへの割当が可能である事が一般に想定される。勿論、一通りの特徴の組でしか正当な割当が行われないパターンが存在する事も考えられる。この場合、クラスを割当る根拠の違いがそのまま部分クラスの存在の主張である。複数の割当方法があるならばその数だけ部分クラスが存在するだろう。結果として、一般的な状況で想定される複数の部分クラスは”重なり”を持つと予想され、また、その様な形で構成されなければ、部分クラスとしては不十分であると思われる。

5．2．　Stoffelの研究［２］

ここでは、Stoffelの理論を要約する。

prime event理論の枠組みはスイッチ理論として見なす事ができる。一つの特徴は有限の値をもつので、特徴数が有限の場合、特徴のすべての組合せが可能なパターンを表現する事になり、その総個数も有限となる。従って、それらすべてを入力とし、クラスラベルを出力と考えると”完全に特定化された入出力関係”が成立する。その場合、効率的な表現の目的に論理関数が問題となる。その一つの実行方法がprime event理論で示されていると考える事ができる。簡単な例を考える。２クラスで各パターンが４変数で表現され、各変数が｛０，１，２｝のどれかの値を取るとする。従って、すべてのパターンの総数は８１（＝3^4）である。すべての８１パターンのうち、クラスω_1のパターンを｛１：（２，０、０）、２：（２，１，０）、３：（２，２，０）、４：（２，０，１）｝、それ以外のすべてのパターンはクラスω_2に属すると仮定する。prime eventの生成の準備として、まず、eventの”合成”の操作を導入する。その操作はパターン（＝初期 event）１、２から、

e_{12}＝（２，０、０）＋（２，１，０）＝（２，－，０）

また、パターン１、４から

e_{14}＝（２，０、０）＋（２，０，１）＝（２，０，－）

の様に行われ、新しい eventを生ずる。ここで、記号”－”は異なる記号間の合成の結果生ずるワイルドカードで、どんな記号とも合致するとする。手順は、合成の操作を既存のすべての eventにおける一対間の組合せに対して行い、合成した eventが他クラスのサンプルと合致するか既に生成されていたかであれば合成した eventを出力せず、そうでなければ新しい eventとして出力する。一通り既存のすべての eventにおける一対間の可能な組合せを合成および検査を行うと新しい eventの集合ができるので、この操作を新しい eventがつくられなくなるまで続け、最終的にprime eventを得る。上の例では、クラスω_1のパターン各々を初期 eventとして、すべての一対間の組合せの結果｛（２，－、０）、（２，０，１）｝が得られる。さらに、これらを合成すると（２，－、－）となる。しかし、これは他のクラスのサンプルと合致するので生成されず、結局、｛（２，－、０）

、（２，０，１）｝が prime eventとなる。

この例では”完全な入出力関係”が与えられている。しかし、”不完全”な場合も同様の操作を適用できる。これが通常の訓練サンプル集合からの prime eventの生成である。その結果得られる prime eventは次の二つの性質を持つ：

１）prime eventは排他的である。つまり、他クラスのサンプルのどれとも合致しない、

２）prime eventは極大である。つまり、訓練サンプル集合において prime eventに合致する部分集合は別の prime eventに合致する部分集合に含まれない。

prime event理論の計算量はサンプル数に関して指数的なオーダーである事がStoffel自身によって示されている。これはその手続きが毎回組合せをとる事を考えてもすぐに了解される。実際、５０個のサンプルに対して一対間の組合せで合成を行い、半分が異なるeventを生成するならば、その個数は約６１２となり、これに対する一対間の組合せは１８６９６６にもなる。従って、さほど大きくないサンプル数に対してもすぐに実行不可能になってしまう。この問題に対して、Stoffelは合成手順を変形して、サンプル数に関して近似的に線形のオーダーの計算量で済む方法を提案しているが、その方法は１）サンプルの順序に結果が依存する、２）生成されないprime eventがある、３）極大性を満たさない prime eventを生成する、など重要な点で目的の達成度が不十分になる。

５．３．　効率的なアルゴリズム

Stoffelの基本アルゴリズムは効率の悪さから非現実的であった。ここでは、これに対し、Kudo and Shimbo［１］による効率的なアルゴリズムを示す。また、クラスの構造にある仮定を置くと、そのアルゴリズムが十分実行可能な計算量で済む事も示す。最後に、より大きな訓練サンプル集合を扱う目的に前処理を提案する。

５．３．１．　基本的定義

はじめに、今後の議論に必要な準備および定義を行う。本章の目的はクラス一つ一つに対し、それらの部分クラスを発見する事であるから、議論を一つのクラスのみに着目して進める。

[定義５．１] 訓練サンプル集合

訓練サンプル集合は $S=S^{+} \cup S^{-}$ で表され、S^{+}は部分クラスを発見しようとするクラスから取られたサンプル集合 $S^{+}=\{x_i\}$, $1 \leq i \leq N_p$ 、S^{-} はそれ以外のクラスからのサンプル集合 $S^{-}=\{x_i\}$, $N_p+1 \leq i \leq N$ を表すとする。

[定義５．２] 特徴集合

特徴はすべて二値の特徴（０または１を値として持つ）とし、特徴の集合を $F=\{f_i\}$, $1 \leq i \leq d$ で表す。従って、各サンプルは $x_i=(x_{i1}, x_{i2}, \cdots, x_{id})$ と書かれる。

これらを基に、サンプルの型を表５．１に示す。

次に、特徴がすべて二値である事から、複数の１である特徴を表す目的に論理式を用いる事が有効である。

[定義５．３] 論理式

論理式は連言標準形 $\alpha=f_{\alpha_1} \wedge f_{\alpha_2} \wedge \cdots \wedge f_{\alpha_n}$ で表されるとする。一つのパターンは１である特徴を結び付けて表現され、論理式 α とパターン x_i に関して、$x_{i\alpha_1}=x_{i\alpha_2}= \cdots =x_{i\alpha_n}=1$ の時、パターン x_i は論理式 α を”真”にすると言う。

この定義を用いて、３種類の集合を定義する。

[定義５．４] 論理式の覆うサンプル

論理式 α とサンプル集合 $S=S^{+} \cup S^{-}$ に関して、

$C^{+}(\alpha)=\{x_i \mid x_i$ は α を真とする, $x_i \in S^{+}\}$

$C^{-}(\alpha)=\{x_i \mid x_i$ は α を真とする, $x_i \in S^{-}\}$

$C(\alpha)=C^{+}(\alpha) \cup C^{-}(\alpha)$

表5．1　訓練サンプルの型

	データ	特徴			
		f_1	f_2	・・・	f_d
正サンプル	x_1	x_{11}	x_{12}	・・・	x_{1d}
	x_2	x_{21}	x_{22}	・・・	x_{2d}
	：		：		
	x_{N_p}	x_{N_p1}	x_{N_p2}	・・・	x_{N_pd}
負サンプル	x_{N_p+1}	$x_{(N_p+1)1}$	$x_{(N_p+1)2}$		$x_{(N_p+1)d}$
	：		：		
	x_N	x_{N1}	x_{N2}	・・・	x_{Nd}

［定義５．５］論理式間の順序

二つの論理式 $\alpha=f_{\alpha_1}\wedge f_{\alpha_2}\wedge\cdots\wedge f_{\alpha_n}$、$\beta=f_{\beta_1}\wedge f_{\beta_2}\wedge\cdots\wedge f_{\beta_m}$ の間に次の半順序を定義する。

$$\alpha\leq\beta\leftrightarrow\{f_{\alpha_1},f_{\alpha_2},\cdots,f_{\alpha_n}\}\subset\{f_{\beta_1},f_{\beta_2},\cdots,f_{\beta_m}\}$$

また、S^+の任意の部分集合Ｇ（｜Ｇ｜＝ｐ）に対して部分クラスを定義する準備として五つの定義をする。

［定義５．６］サンプルの部分集合に対応する論理式

訓練サンプルS^+の任意の部分集合 $G=\{x_{i_1},x_{i_2},\cdots,x_{i_p}\}$ に対して、一つの論理式が対応する。

$$a(G)=a(x_{i_1},x_{i_2},\cdots,x_{i_p})=\bigwedge_{x_{i_1k}=x_{i_2k}=\cdots=x_{i_pk}=1} f_k$$

この論理式 $a(G)$ が部分集合Ｇのすべての要素に対して真となる論理式全体の極大要素（半順序≦に関して）になっている事は容易に分かる。つまり、論理式 $a(G)$ は部分集合Ｇを規定する。

次に、部分集合Ｇの”排他性”を表す目的に、

［定義５．７］部分集合の排他性

$$[G]=\begin{cases}1\ (\ C^-(a(G))=\phi\ \text{の時}\ )\\ 0\ (\ \text{それ以外}\ )\end{cases}$$

［定義５．８］部分集合の検査

部分集合Ｇの大きさｎ（≦ｐ）の全ての部分集合｛Ｈ｝に対してその排他性を調べ、その結果を二種類の族とする。

$$\pi^+(n,G)=\{H\mid[H]=1,\ |H|=n,\ H\subset G\}$$

$$\pi^-(n,G)=\{H\mid[H]=0,\ |H|=n,\ H\subset G\}$$

［定義５．９］ｎ項完全部分集合、充満部分集合

もし $\Pi^{-}(n,G)=\phi$ ならば、部分集合Ｇは”ｎ項完全”であるという。特に、ｎ＝ｐの時は、”充満”であるという。また、S^{+}の充満部分集合全体の族の中で集合の包含関係に関して極大なものを”極大充満部分集合”という。

これらの準備の下に、部分クラスを定義する。

［定義５．１０］部分クラス

一つ部分クラスは一つのS^{+}の極大充満部分集合Ｇに対応し、Ｇのすべての要素に対し真となる様な論理式αでしかも排他性を保持するもの（S^{-}のどのサンプルも論理式αを真としない）を考えた時、論理式αを真とする様なすべてのパターンの集合を”部分クラス”と呼ぶ。この場合、パターン集合は特徴空間全体からとられる事に注意する必要がある。また、その様な論理式として自然に $a(G)$ を考える事ができるが、ここでは、今後部分クラスを拡張する事を考慮して論理式を特定化しない。

この定義から、結局目的の”部分クラスの発見”の問題は”極大充満部分集合の数え上げ”の問題に帰着する。しかし、この数え上げは集合S^{+}の要素のすべての組合せを調べる事を要求し、$|S^{+}|$がある程度以上大きくなると実行不可能になる。そこで、数え上げの効率化を図る為に次の定理を用意する。

［定理５．１］

訓練サンプル集合S^{+}の部分集合Ｇ（$|G|=p$）が任意のｎ（$\leq p$）に対してｎ項完全である事は、Ｇが充満である為の必要条件である。

（証明）

集合Ｇの任意の大きさｎ（$\leq p$）の部分集合Ｈに対して、$a(H) \geq a(G)$ は明らかである。従って、$C^{-}(a(H)) \subset C^{-}(a(G))$ 。Ｇが充満ならば $C^{-}(a(G)) = \phi$ 。結果的に $C^{-}(a(H)) = \phi$ 。

この定理より、極大充満部分集合を数え上げる事に先んじて、小さなnに対して極大n項完全部分集合を数え上げ、nを上げて行く事でさらに調べるべき対象となる部分集合を制限し、目的実現の効率化が計れる。

5．3．2．極大n項完全部分集合の数え上げ

ここでは与えられた集合中において、すべての極大n項完全部分集合を数え上げる方法を検討する。

検査すべき集合をG、$|G|=p$とする。この時、はじめにn（$\leqq p$）個の要素からなるGの部分集合（以後、この部分集合をサイズnの部分集合と呼ぶ）の排他性をすべて検査する。つまり、一つのサイズnの部分集合をHとする時、$[H]=1$かどうかを調べておく。この結果を定義5．8で定義した二つの集合族$\Pi^{+}(n, G)$、$\Pi^{-}(n, G)$に納める。ここで、$\Pi^{+}(n, G)$は排他的な部分集合、$\Pi^{-}(n, G)$は排他的でない部分集合からそれぞれ成っている。そこで、これらの族から極大n項完全部分集合を数え上げる事を考える。この時、両方の族が等価な情報である事に気づけばすぐに、そのどちらかで十分である事がわかる。つまり、二つの族別々にそれぞれを用いた数え上げの方法が考えられる。

まず、$\Pi^{+}(n, G)$を用いる方法の概略を述べる。$|\Pi^{+}|=k$とする。一つの要素$\{x_{i_1}, x_{i_2}, \circ\circ\circ, x_{i_n}\} \in \Pi^{+}$を取る。既にこの集合はn項完全部分集合であるので、この集合を拡張して極大にする事を考える。例えば新しく$x (\neq x_{i_j})$をこの集合に付け加える事を考えた時は、n個の部分集合

$\{x, x_{i_2}, \circ\circ\circ, x_{i_n}\}$、$\{x_{i_1}, x, \circ\circ\circ, x_{i_n}\}$、$\circ\circ\circ$、

$\{x_{i_1}, x_{i_2}, \circ\circ\circ, x\}$がすべて$\Pi^{+}$の要素である事を確認すればよい。もし、そうであれば、新しい集合$\{x_{i_1}, x_{i_2}, \circ\circ\circ, x_{i_n}, x\}$はn項完全部分集合である。逆に、一つでも$\Pi^{+}$の要素でないものがあれば、xを付け加える

事は出来ない。xが付け加えられたとして、さらに要素yを加えてもn項完全部分集合であれば、その操作を続ければよいし、もしどんなyに関してもそうならなければ、この集合が一つの極大n項完全部分集合である。注意すべき事は、yの検査の際に、集合$\{x_{i_1}, x_{i_2}, \cdots, x_{i_n}, x\}$の各要素とyを取り替えるのではなく、$\{x_{i_1}, x_{i_2}, \cdots, x_{i_n}\}$の各要素とyを取り替え、x，yとこの集合における任意の2つの要素とを取り替え、それらすべてがΠ^+の要素である事を調べる必要があり、計$({}_nC_1+{}_nC_2)$回の検査が必要になる事である。すべての極大n項完全部分集合の数え上げはこの様な操作を取りこぼしのないように行う事により達成される。当然、効率化の面で種々の改良が考えられるが、それにしてもこの手続きの計算量は一般の状況では膨大なものになる。

次に、Π^-を用いた方法に関して考察する。実際、大多数の問題においてこちらの方法の方が有効となる事を後述する。まず、その方法の基礎となる定理を示す。

［定理5．2］

Gの部分集合Iに関して、

Iがn項完全部分集合　←→　任意の部分集合　$H \in \Pi^-(n, G)$ 対して、$\bar{I} \cap H \neq \phi$。

ただし、$\bar{I} = G - I$とする。

（証明）

（→の証明）

Iがn項完全部分集合である時に、$\bar{I} \cap H \neq \phi$、$H \in \Pi^-(n, G)$を示せばよい。今、$H = \{x_{h_1}, x_{h_2}, \cdots, x_{h_n}\} \in \Pi^-(n, G)$とする。この時、少なくとも一つの$x_{h_k}$は$\bar{I}$に属する。なぜなら、すべての$x_{h_j}$がIに属するとすると［H］＝0かつH⊂Iとなり、Iがn項完全部分集合だという事に反する。よって、示された。

（⇐の証明）

まず、Gの部分集合で、すべてのH∈Π⁻（n，G）と共通部分を有する集合$\bar{I}=G-I$を考える。そして、任意の$J=\{x_{j_1}, x_{j_2}, \circ\circ\circ, x_{j_n}\}\subset I$を考える。この時、［J］＝0、つまり、集合Hが排他的でないとする。すると仮定より少なくとも一つ$x_{j_k}\in\bar{I}$が存在する。これは$x_{j_k}\in I$に反する。従って、Iのサイズnの部分集合はすべて排他的である事が結論づけられる。

定理5．2は結局、「集合Gの中の任意の部分集合Iのn項完全性を調べるには、Gの排他的でないサイズnの部分集合すべてと$\bar{I}$が共通な要素を持つかどうかを調べればよい」事を示している。また、極大性を考えるに際しては、Gのn項完全部分集合の族Ωならびにn項完全部分集合の補集合の族$\bar{\Omega}$を考えた時、「IがΩにおいて包含関係に関して極大要素なのは、$\bar{I}$が$\bar{\Omega}$の極小要素なのに等しい」事は容易に了解できる。従って、この性質と定理5．2を併せて考えると、Gの極大n項完全部分集合の数え上げ問題は「Gの排他的でないサイズnの部分集合すべてと共通な要素を持つ部分集合の族$\bar{\Omega}$における極小要素を見出す問題」に換言できる。従って、この様な集合を実際に数え上げればよい。その具体的アルゴリズムはここに示さないが、Π^+の時と同様に、丁寧にできるだけ小さな集合を数え上げればよい。その場合、多少の効率化を考えてもやはり一般の場合にはΠ^+の時と同様に膨大な計算量を必要とする。そこで、次に本章の問題の特殊性を考察する。

これまで、Π^+とΠ^-を用いた極大n項完全部分集合の数え上げの概略を述べた。そこで、方法に関して二種類が見つけられた以上、効率的な選択法は次の基準である。「もし、$|\Pi^+|\geqq|\Pi^-|$ならば、Π^-を用いた方法、そうでなければΠ^+を用いた方法を採用すればよい」。しかし、おのおのの場合に述べた様にその計算量は一般の場合実現不可能になる可能性が高い。この点に関しては、今問題にしている集合はパターン認識におけるクラスのサンプル集合であるという問題の

特殊性を考慮する事が有効である。問題の特殊性は次の考察を促す。「クラスと言うのは本来ある種の”まとまり”を持ったものである。従って、ｎ項関係、つまり一つのクラスにおける任意のｎ個のサンプルが共通の排他的特徴を持つという予想は、クラスというのが共通性の高い要素の集まりである事を考えると、大部分肯定されると予想される。特に、小さいｎの値（例えばｎ＝２）の場合にはより顕著であろう」。この予想は｜Π⁺｜が大きく｜Π⁻｜が小さい事を意味する。従って、より効率的な極大ｎ項完全部分集合の数え上げの方法はΠ⁻を用いた方法であると思われる。最悪の場合、ｍ＝（ｎの｜Π⁻｜乗）個の集合を調べ、その後極小なものだけ見つけ出せばよいので、計算量はなんら効率化を考えなければＯ（m^2）である。ここで、Ｏ（ｋ）はアルゴリズムがｋの定数倍のステップ数で計算できる事を示す。実際は、これより効率のよいアルゴリズムはすぐに考えられる。しかし、むしろ本質的なｎと｜Π⁻｜の値に関する低い上界の見積りが重要である。これには現在おおまかではあるけれども、次の二点が考えられる。１）ｎの値はクラスの複雑さを示す。実際、ｎ項完全であるが、充満でない部分集合を考えた時、少なくとも（ｎ＋１）個の部分クラスが存在しなくてはならなくなるのはすぐにわかる。従って、クラスがそれほど複雑な構造を取らない場合、ｎの値は小さく見積られる。２）｜Π⁻｜の値は先に述べた様にクラスのサンプルの非共通性を示している。従って、それほど大きくないと予想される。よって、大ざっぱな言い方をすれば、「クラスが本質的にある程度のまとまりを有し、それほど複雑な構造を取らなければ、この極大ｎ項完全部分集合の数え上げは十分実行可能である」事が予想される。しかし、これらはあくまでも予想であるので、アルゴリズムの現実の実行可能性をより高める目的に、前処理によりサンプル数を見かけ上減少させる方法を考察している。詳細は後述する。

５．３．３．　アルゴリズム

以上の準備の下で、極大充満部分集合を数え上げるアルゴリズムを示す。基本的な構想は以下の通りである。定理５．１は訓練サンプル集合Ｓ⁺の部分集合Ｇ（｜Ｇ｜＝ｐ）に関して、Ｇが充満である為にはＧが任意のｎ（≦ｐ）に対してｎ項完全でなければならない事を言っている。従って、小さいｎから始めて極大ｎ

項完全部分集合｛H｝を見つけ出す。具体的な方法は第５．３．２節で述べた。そして、各極大ｎ項完全部分集合Ｈが充満かどうかを、つまり［Ｈ］＝１かどうかを調べる。もし、Ｈが充満なら当然それが一つの極大充満部分集合である。この操作を徐々にｎを上げながらすべての極大充満部分集合が見つかるまで続ける。この構想を実現したものが次のアルゴリズムである。

[アルゴリズム５．１]（極大充満部分集合の数え上げアルゴリズム）

```
begin
        n ← 1;
        G ← S⁺;
        ENUMSAT(n, G);
end
```

手続き ENUMSAT(n, G)

```
begin
1. if n=|G| then すべてのx∈Gに対してG−{x}を出力;
   else
      begin
2.       Gのすべてのサイズnの部分集合の排他性を検査して、結果を
           Π⁺(n, G)とΠ⁻(n, G)に設定;
3.       Π⁺(n, G)とΠ⁻(n, G)のどちらかを用いて極大n項完全部
           分集合を数え上げる;
4.       数え上げられた極大n項完全部分集合を{ H_1^n, H_2^n, ···, H_t^n }
           とする;
5.       for i←1 until t do
6.          if H_i^n が充満 then H_i^n を出力;
            else
               begin
7.                ENUMSAT(n+1, H_i^n );
                  end
      end
end
```

アルゴリズム５．１の出力集合が必ずしも極大充満部分集合とはならない事に注意を要する。なぜなら、一旦小さいnにおいて生成された二つの極大n項完全

部分集合H''_1、H''_2が大きなｎにおいて、同じ、あるいは片方がもう一方を含む様な極大充満部分集合G_1、G_2をそれぞれ生成する場合が有り得るから。従って、出力集合の検査により、本当に極大なものだけを選ぶ事が後処理として必要である。

次に、アルゴリズム５．１の計算量を見積る事を考える。はじめに手続きＥＮＵＭＳＡＴ（ｎ、Ｇ）を調べる。まず、変数を｜Ｇ｜＝ｐ、ｎ（≦ｐ）、それと特徴数ｄとする。また、排他性の検査を行う他クラスのサンプル数をｍとする。ステップ毎に考えると、ステップ１はＯ（ｐ）、ステップ２はＯ（$p''d(n+m)$）、ステップ４は無視でき、ステップ５～７はＯ（ｔｄ（ｐ＋ｍ））となる。ここで、ｔは発見された極大ｎ項完全部分集合の数を表す。ステップ３は第５．３．２節で述べた様にクラスが”よいまとまり”を持つならば少い計算量になる事も考察されている。従って、そのステップの計算量はとりあえずこの手続きの計算量としては考慮しないとする。さらに、ｔ、ｄ、ｍならびに係数としてのｎ（通常ｍ以下）も定数と考えて、結局、この手続きの計算量をステップ２のＯ（p''）と見積る。これは、さほど大きくないｐ，ｎの値に関してすぐに実行不可能になる。そこで、もう少しこれらの値に関して考察してみる。まず、ｎの値であるが、これは第５．３．２節で述べた様にクラスの複雑さの程度を表し、やはり先と同様に「クラスはそれほどの複雑な構造を持たない」という仮定を置くと、さほど大きくないと考えられる。次に、ｐの値に関しては、実はｎの増加に伴って減少する事がわかる。実際、手続きＥＮＵＭＳＡＴを検討してみると、ｎの増加にともなって調べられる集合は、

$$|S^+| \geqq |H_t^1| \geqq |H_t^2| \geqq \cdots$$

の様になる。これは「極大ｎ項完全部分集合は少なくとも極大（ｎ－１）項完全部分集合である」と言う事実からきている。その結果、大きなｎに対してはｐは小さくなるので、手続きの計算量Ｏ（p''）はさほど大きくはならないと考えられる。最終的にアルゴリズム５．１の計算量は各段階のｎとその時のｐを対にして、p''の最大のオーダーと考えられる。ｎはそれほど大きくはならないとしても、ｐに関しては｜S^+｜以下である事だけが保証されるに過ぎない。これに対しては、S^+が大きい場合にサンプルサイズを縮小する前処理を後述する。

次に、アルゴリズムの動作例を示す。

表5．2　例5：2の訓練サンプル

データ		特徴			
		f_1	f_2	f_3	f_4
正サンプル	x_1	1	1	1	0
	x_2	1	1	0	1
	x_3	1	0	1	1
	x_4	0	1	1	1
	x_5	1	0	0	0
	x_6	1	1	0	0
負サンプル	x_7	0	0	0	0

（例5．1）

サンプル集合を表5．2に示す。アルゴリズム5．1の働きを特に手続きENUMSATの働きを中心にして示す。

0）手続きENUMSAT（n、S^+）（n＝1）を呼ぶ；

1）ENUMSAT（1、S^+）；

$\Pi^- = \phi$；

極大1項完全部分集合：S^+ →充満でない →ENUMSAT（2、S^+）；

2）ENUMSAT（2、S^+）；

$\Pi^- = \{x_4, x_5\}$ → 定理5．2より $\bar{H}_1^2 = \{x_4\}$，$\bar{H}_2^2 = \{x_5\}$；

極大2項完全部分集合：$H_1^2 = \{x_1, x_2, x_3, x_5, x_6\}$ →充満→ 出力；

$H_2^2 = \{x_1, x_2, x_3, x_4, x_6\}$ →充満でない

→ENUMSAT（3、H_2^2）；

3）ＥＮＵＭＳＡＴ（３、H_2^2 ）；

$\Pi^- = \{ x_3, x_4, x_6 \}$ → 定理５．２より

$\bar{H}_1^3 = \{ x_3 \}$, $\bar{H}_2^3 = \{ x_4 \}$, $\bar{H}_3^3 = \{ x_6 \}$ ；

極大３項完全部分集合：

$H_1^3 = \{ x_1, x_2, x_4, x_6 \}$ → 充満 → 出力；

$H_2^3 = \{ x_1, x_2, x_3, x_6 \}$ → 充満 → 出力；

$H_3^3 = \{ x_1, x_2, x_3, x_4 \}$ → 充満でない

→ＥＮＵＭＳＡＴ（４、H_3^3 ）；

４）ＥＮＵＭＳＡＴ（４、H_3^3 ）；

$| H_3^3 | = n = 4$ →

$H_1^4 = \{ x_2, x_3, x_4 \}$ → 出力；

$H_2^4 = \{ x_1, x_3, x_4 \}$ → 出力；

$H_3^4 = \{ x_1, x_2, x_4 \}$ → 出力；

$H_4^4 = \{ x_1, x_2, x_3 \}$ → 出力；

５）出力部分集合を検査、極大充満部分集合のみを選出；

結果

$H_1^2 = \{ x_1, x_2, x_3, x_5, x_6 \} \rightarrow a(H_1^2) = f_1$；

$H_1^3 = \{ x_1, x_2, x_4, x_6 \} \rightarrow a(H_1^3) = f_2$；

$H_1^4 = \{ x_2, x_3, x_4 \} \rightarrow a(H_1^4) = f_4$；

$H_2^4 = \{ x_1, x_3, x_4 \} \rightarrow a(H_2^4) = f_3$

５．３．４．　部分クラスの発見

極大充満部分集合を求めるアルゴリズムは述べたので、次に部分クラスの発見について考察する。定義５．１０により、任意の極大充満部分集合Ｇに関して、Ｇのすべての要素に対し真となる論理式で排他性を維持するものを考えた時、その論理式を真とするパターンの集合が部分クラスである。勿論、ａ（Ｇ）が代表的な論理式である。しかし、S^+の大きさが十分でない時、ａ（Ｇ）は冗長な場合が多い。そこでなるべく少ない特徴数でａ（Ｇ）の排他性を維持する事を考える。目的を完全に実行する唯一の方法はａ（Ｇ）に用いられている特徴の組合せを調べる事であり、実際ａ（Ｇ）の特徴数をｋとし、ｋが十分小さければこれは実行

可能であり、各種の効率的な方法がまた援用できる。しかし、ある程度のサンプル数からGが生成されているならば、次の簡単な手続きで十分であろう。

（論理式の拡張手続き）

a（G）に用いられている特徴を左から順に一つずつ消してみる。この時、排他性が維持されなければその特徴は残し、維持されれば消したまま次の特徴を調べる。この手続きを繰り返す。

本論文の残りでは、すべてこの手続きを用い、これにより生成された論理式により部分クラスが定義されるとする。

5．3．5．　アルゴリズムの変形

アルゴリズム5．1は確に効率的ではあるけれど、このままでは適用できる訓練サンプルS⁺の大きさはせいぜい二、三百であると予想される。しかし、実際の問題ではより大きなサイズのサンプルを扱う事が望まれるかも知れない。これに対処する目的でサンプルサイズを縮小する前処理を考える。基本的な構想は、最も近いサンプル同士を一つにして、サンプルサイズを約半分にする事である。その場合の”近さ”の測定にはHamming距離を用いる事が考えられる。実際には次の手順が考えられる：

0）$S^+=\{x_1, x_2, \cdots, x_n\}$ とする。すべてのサンプルに”未処理”のラベルをつける；

1）Sから未処理のサンプルxを一つ取ってきて、xに”処理済”のラベルを付ける。もし、すべてのサンプルが処理済みならば　4）へ；

2）未処理なサンプルでxと最も距離の近いものをyとする。この時、対（x，y）が以下の基準の両方を満たせばx，yを”論理積”で合併し出力、yに”処理済”のラベルを付ける；

a）$d(x,y) \leq \theta \cdot \min_z d(x,z)$

ここで、dはHamming距離を表し、zは処理済でも構わないとし、

θ（≧１）は与えられた閾値とする。

b）[$\langle x,y \rangle$] = 1　　（排他性）

もし、基準のどちらかでも満さない場合は、ｘを出力；

３）　１）へ；

４）　終了；

この手続によって、サンプルサイズは約半分に縮小される。大きなサンプル集合の場合には、必要な回数この処理を繰返す事によって、適当なサンプルサイズに縮小する事が出来る。

この前処理を通した場合、アルゴリズム５．１により発見される部分クラスは、１）サンプルの順序に多少依存する、３）極大性も多少減ずる、などの欠点を持つ。１）に関しては、なるべく近い２パターンを合併するので、大体は順序に無関係であるが、手順の関係で多少順序による影響がある。また、２）に関しては、前処理により合併されたパターンは二度と別々に用いられる事がなく、微妙な包含関係が考慮できなくなる為である。しかし、双方とも部分クラスの発見に多大な影響を及ぼさない事は容易に推測される。この点は、計算の効率化を目的にStoffelの提案した変形アルゴリズムがサンプル順序にかなりの影響を受け、極大性もかなりの損失が生じるのに比べ有利な点である。

５．４．　結論

一つのクラスの中に、１）排他性と２）極大性を満たす部分クラスを発見する方法を議論した。その発想は Stoffelの prime event理論に始まるが、そのアルゴリズムは非現実的だった。本章では幾つかの改良により、大部分の問題に対して現実に実行可能な計算量であるアルゴリズムを提案した。prime event理論との主な相違点は以下の通りである。

１）特徴を有限離散値ではなく二値に限った、

２）prime eventではなく、その拡張である”部分クラス”を求めた、

３）prime event理論の基本的アルゴリズムは非現実的な計算量であったのに対し、本章の方法によりかなりの効率化がなされた。

ここで、１）の限定は本質的な制約ではでなく、むしろ操作の簡便さをもたらす利点がある。また、２）によりStoffelの立場をより積極的にパターン認識の問題に持ち込む事が出来る様になった。３）で計算効率の向上をはかる事で、より大規模な問題に対する適用の可能性を開いた。

問題点は、本章のアルゴリズムにおいても考慮下のクラスがある程度”まとまり”を持つ事を仮定した上で、その計算量が実行可能な範囲内に収まる事が見積られているに過ぎず、本質的に複雑なクラスの解析にはこのアルゴリズムはまだ実用的ではないかもしれない事にある。この点に関して、より効率のよいアルゴリズムの開発が期待される。

第６章　二値的特徴の解析に基くパターン識別ならびに特徴選出

６．１．　序論

５章で提案した部分クラスを求めるアルゴリズムは二値的特徴を有するパターンに対してであった。本章ではこれを一般の質量混在の特徴を有するパターンの場合に拡張する。また、部分クラスに基く識別規則の構成法を述べ、従来の手法と比較する。加えて、求めた部分クラス全体の集合がクラスの構造を反映すると考え、クラス構造を定量化する指標の幾つかを示し、それらと識別率との関係について考察する。最後に、パターン認識機構全体としての性能向上の目的にそれらの指標を用いる事を提案し、具体的に特徴選出およびサンプル選出の方法を述べる。

６．２．　二値的特徴解析の適用

一般に、パターン認識におけるパターンの特徴は質量混在とするのが望ましい。その際、両方の性質を有するパターンの扱い方が問題となる。問題解決に有効なアプローチの一つは、すべての性質を最低レベルの特徴、つまり「ある性質を持つか否か」の二値的特徴（１：持っている、０：持っていない）に落す事である。この事は、有限の質的特徴はもとより連続の量的データに対しても「ある閾値以上か否か」を考える事により実行できる。本節ではこの考えを定式化すると共に、それを用いたパターン識別の具体的なアルゴリズムを与える。

６．２．１．　特徴の変換

まず、すべての特徴を二値的特徴に変換するとして、変換する前の特徴（特徴として識別に用いられるもの）を”測定値特徴”あるいは単に”特徴”、変換後の二値的特徴を”二値的特徴”と呼び区別する。これにより”特徴選出”などの”特徴”は前者を示す事になり通常の呼称と整合する。

次に具体的な変換を考える。その際、二値的特徴に関しては「ある性質を有す

る」も「ある性質を有しない」も等価な情報と考える。つまり、常に二値特徴 f_i は $\overline{f}_i$ を伴って用いられるとする。個々の測定値特徴を次の方法で変換する。

（１）質的特徴に対して

特徴変数は通常有限個の性質のうちのどれかを性質として持つ。例えば、１：「好き」、２：「嫌い」とか、１：「赤」、２：「青」、３：「黄」などである。一般にｎ個の性質からなる場合、変換は各性質に対応してｎ個の二値的特徴とそれらの「否定」であるｎ個の二値的特徴に変換される。上の最初の例では「好き」、「嫌い」、「好きでない」、「嫌いでない」の４つの二値特徴に変換される。それに対応して、一つのデータは例えばｘ＝（０、１、０、１）（「好きでない」が「嫌いでない」）と表現される。また、質的特徴のうちでも順序が入っている場合は量的特徴と同じ扱いとする。

（２）量的特徴に対して

連続量を取りうる特徴変数を考えると、「ある閾値以上」と「その閾値以下」の二つの二値的特徴に変換できる。ここで、「その閾値より小さい」にしなかったのは両方の二値的特徴の共存、即ちその閾値そのものをも許す為である。閾値の取り方としては、幾つかの方法が考えられるが、ここでは特徴軸毎に全訓練サンプルの持つ特徴値の最小値、最大値を両端として等間隔に閾値を設定する方法を採用する。

これらの変換を各測定特徴毎に独立に行う。その変換全体を今後ｇで表す。つまり、

$$\text{（測定値特徴空間）}\ X \xrightarrow{g} \hat{X}\ \text{（二値的特徴空間）}$$

ここで、「変換ｇは特徴軸のスケーリングに不変」である事に気づく事は重要である。一般に特徴軸のスケーリングは多次元のデータ間の距離（例えば、ユークリッド距離）に直接影響を与えるので、識別やクラスタリングの問題に対しては何らかの”正規化”を行うのが普通であり、その妥当性が問題となる。この点に関して、特徴軸のスケーリングに不変という性質は望ましい。また、特徴を”

独立”に変換する事は、サンプルの高次の構造を表現できにくい様に思われる。しかし、次の考察が重要である。「クラスと無関係に、特徴間に従属関係があるのか、それともクラスが複数の特徴の関係において規定されるのか」である。本章においては特に、前者の関係は”特徴抽出”（変換）の問題であるとし、後者に重点をおく。従って、各特徴は統計的に”独立”あると見なす。その場合でも、もしクラスを規定する特徴が高次構造を取る場合は、それは結果として抽出される。なぜなら、クラスの高次構造とは「特徴の連言（特徴１と特徴２と。。。と特徴ｎ）を有するならこのクラス」と言った形、あるいは複数のその形の選言で表現されると思われるから。この事は本論文において、連言が”部分クラス”に、それらの選言が”複数の部分クラスを持つクラス”に対応する。

６．２．２．　領域確保アルゴリズム

特徴が二値的特徴に変換された後は、第５章の方法を用いて極大充満部分集合を数え上げればよい。そのアルゴリズムを示す（図６．１）。

［アルゴリズム６．１］（領域確保アルゴリズム）

（０）訓練サンプルを集合Ｓ＝（S^+、S^-）、測定値特徴集合を$F=\{f_i\}$とする；

（１）測定値特徴集合Ｆをｇにより二値的特徴集合$\hat{F}$に変換する。すべてのサンプルをｇにより測定値特徴空間Ｘから二値的特徴空間$\hat{X}$に写像する；

（２）必要に応じて、第５章の前処理の方法で正サンプルS^+のサイズを縮小する（一回の処理によりサンプルサイズが約半分になるので、適当な大きさになるまで繰り返す）；

（３）アルゴリズム５．１を用いて極大充満部分集合の族Ω（S^+、Ｆ）を発見する；

（４）すべての論理式ａ（Ｓ），$S \in \Omega$に対して余分な測定値特徴を除去する。具体的には一つの測定値特徴から変換された二値的特徴を一まとめとして１ブロックと考え、論理式ａ（Ｓ）の左からブロック毎に１であるも

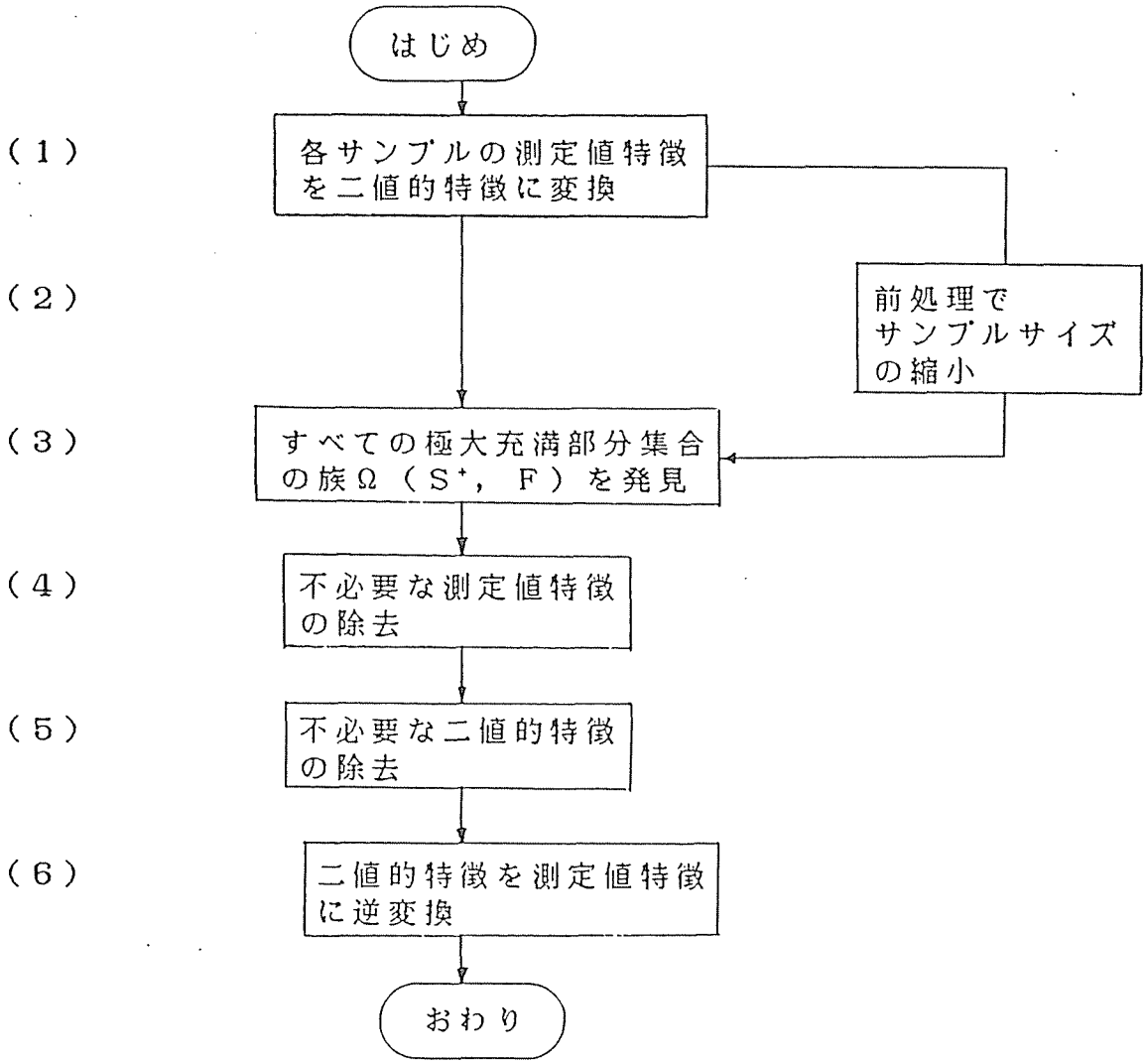

図６．１　領域確保アルゴリズム

のをすべて0にする、この時、排他性が維持されればそのままにして、そうでなければ元に戻して次のブロックへ進む。この操作を繰返す；

（５）個々の二値的特徴に関して（４）と同様の操作を行う；

（６）それぞれの論理式ａ（Ｓ）、Ｓ∈Ωの二値的特徴をg^{-1}により測定特徴空間に逆変換する。結果として部分クラス（領域）Ｒが論理式ａ（Ｓ）に対応して次の様に確保される：

Ｒ（ａ（Ｓ））＝｛ｘ｜ｇ（ｘ）はａ（Ｓ）を真とする、ｘ∈Ｘ｝。

アルゴリズム６．１において、アルゴリズム５．１を除いて注目すべきところは（４）、（５）のステップである。双方のステップにより”冗長な測定値特徴”、”冗長な二値的特徴”の除去が行われ、これにより一つの極大充満部分集合に対応する素朴な論理式ａ（Ｓ）の構成二値特徴が減り、結果的に対応する部分クラスは拡張される。特に、（４）のステップは重要であり、この処理において除去された測定値特徴は実際の特徴の測定コストに節約をもたらす。また、一般的に、「大きな部分クラスになればなるほど共通性が低い」と思われるので、重要な部分クラスに実際用いられる測定値特徴の個数は少なくなると予想される。その意味では、「識別規則の設定そのものが特徴選出を含んでいる」と言う事ができる。そこで、特にアルゴリズムで最終的に得られた論理式に実際用いられた測定値特徴を”極大充満部分集合または部分クラスを規定する測定値特徴”と呼び、後の議論で用いる事にする。

これでクラス毎に領域を確保する事ができたので、次に多クラスの識別方法を述べる。

［識別手続き］

クラスをω_1、ω_2、・・・、ω_kのｋ個、それらの生起確率（事前確率）を$P(\omega_1)$，$P(\omega_2)$，・・・，$P(\omega_k)$とする。生起確率が分からない場合は推定値を用いるものとする。また、各クラス毎に訓練サンプル集合 S_i^+ (i=1,2,・・・,k)があるとする。この場合 S_i^- はすべての S_j^+（ｊ≠ｉ）の和集合とすることができる。更に、各クラス毎にアルゴリズム６．１により、極大部分集合の族 $\Omega_i = \{ S_1^i, S_2^i, \cdots, S_{m_i}^i \}$ (i=1,2,・・・,k) ならびに、それらに対応する部分クラス $\{ R(a(S_1^i)), R(a(S_2^i)), \cdots, R(a(S_{m_i}^i)) \}$

が発見されているとする。

このとき、入力パターンxに対する識別は次のように行われる。

（識別基準）

入力パターンxはxを含むすべての部分クラス $R(a(S_j^i))$ の中で、

$P(\omega_i)\cdot q(S_j^i)$ の値を最大にするクラスiに割り当てる

但し $q(S_j^i) = |S_j^i| / |S_i^+|$ とする

（例6．1）

Fisher［5］の古典的なアイリスデータの識別を試みた。データはアイリスの三種の花から求めた測定値からなる。用いられた測定値特徴はがく片の長さ、がく片の幅、花弁の長さ、花弁の幅の四つであり、サンプルは各クラスから50ずつ取られている。実験は各サンプルの前半25／クラスを訓練集合として領域確保の為に用い、残りの25サンプル／クラスをその領域の識別率を評価する為に検査集合として用いた。各クラスに対する部分クラスを表6．1に、識別規則を表す決定木を図6．2に示す。訓練集合ならびに検査集合に対する識別結果を表6．2に示す。この識別率はこれまでに実験されているどの手法と比較しても劣らないものである。

アルゴリズム6．1の計算量に関して考察する。考察すべき変数は正サンプル数N_p、総サンプル数N、特徴数dである。測定値特徴と二値的特徴間の変換に関しては、二値的特徴の個数は測定値特徴の個数dの定数倍であるので計算量のオーダーは変わらない。従って、アルゴリズム6．1の計算量はアルゴリズム5．1の計算量である。よって、「アルゴリズム6．1の計算量は特徴数に関して線形のオーダーである一方、実際の計算量はクラスの構造に大きく依存する」。しかし、実際的な場面では計算量の低い上界を見積れる。パターンがn次元実数ベクトル空間においてベクトルとして表現されているとする。準備として、各パターンはn次元ベクトル $x_i = (x_{i1}, \cdots, x_{in})$ と表されるとする。また、複数のパターンにより空間に張られる超区間を

表6．1　例6．1の部分クラス

クラス	論理式	がく片長	がく片幅	花弁長	花弁幅	重要度（%）
クラス1	f^1_1				≦0.9	100
クラス2	f^2_1			≦4.93	1.0≦ ≦1.6	96
	f^2_2		2.8≦	≦5.43	1.2≦ ≦1.9	64
	f^2_3	5.05≦ ≦6.25		≦4.93	1.0≦ ≦1.9	56
クラス3	f^3_1		≦3.1		1.6≦	68
	f^3_2			4.93≦		88
	f^3_3	6.25≦			1.6≦	76
	f^3_4				1.9≦	72
	f^3_5	≦6.1	≦2.9		1.5≦	24

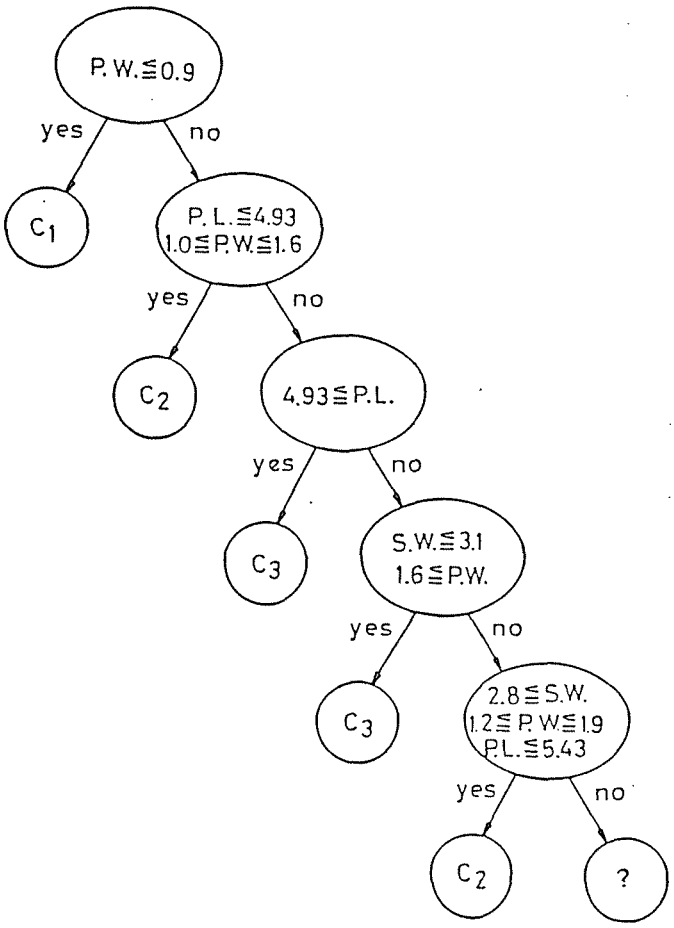

図6．2　例6．1の決定木
（ここでは部分クラスのうちの代表的なもののみを示した。S．W．はがく片の幅、P．L．は花弁の長さ、P．W．は花弁の幅を示す。またC_iはクラスiを示し、”？”は棄却クラスを示す。）

表6．2　例6．1の決定木による識別結果

検査集合

in＼out	C_1	C_2	C_3
C_1	25		
C_2		23	2
C_3			25

設計サンプル集合

in＼out	C_1	C_2	C_3
C_1	25		
C_2		25	
C_3			25

$RECT^n(x_1, x_2, \cdots, x_s) = [\min_i x_{i1}, \max_i x_{i1}] \times [\min_i x_{i2}, \max_i x_{i2}] \times \cdots \times [\min_i x_{in}, \max_i x_{in}]$

で定義するとする。

この時、次の重要な性質が成り立つ。

[性質６．１]

$n \geqq 2$ かつ $s \geqq n$ の条件下で n次元ベクトルの集合 $(x_1, x_2, \cdots, x_s)$ を考えた時、

$y = (y_1, y_2, \cdots, y_n) \in RECT^n(x_1, x_2, \cdots, x_s)$

$\leftrightarrow$ $i_1, i_2, \cdots, i_n$ で $y \in RECT^n(x_{i_1}, x_{i_2}, \cdots, x_{i_n})$ を満たすものが存在する

（証明）

（→の証明）

n次元ベクトル空間のk（≦n）次元部分空間を考え、kに関する帰納法で示す。

１）k＝２の時、次元の対称性を考慮すると点 $y^2 = (y_1, y_2)$ は図６．３の三つの形のいずれかに含まれ、結果二つのパターン x_{i_1}, x_{i_2} により張られる超区間に含まれる。

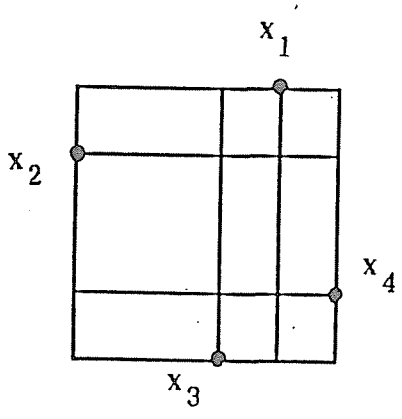

図６．３（a）

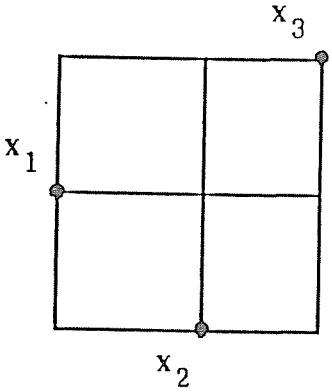

図６．３（b）

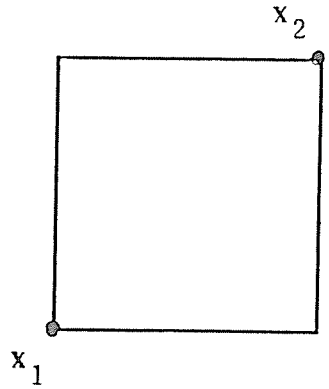

図６．３（c）

2）性質がｋ－１の時に成立するとして、ｋの時に成立することを示す。

$y^{k-1} = (y_1, y_2, \cdots, y_{k-1})$ に対してそれを含むようなパターンの列 $x_{i_1}, x_{i_2}, \cdots, x_{i_{k-1}}$ で $y^{k-1} \in RECT^{k-1}(x_{i_1}, x_{i_2}, \cdots, x_{i_{k-1}})$ は存在するとする。この時、

$y_k \geq \max_j x_{i_j k}$ ならば $x_{i_k k} = \max_j x_{jk}$ とし、

$y_k < \max_j x_{i_j k}$ ならば $x_{i_k k} = \min_j x_{jk}$ とすれば

新しいパターンの列 $x_{i_1}, x_{i_2}, \cdots, x_{i_k}$ によって張られる超区間は y^k を含むであろう。従って、証明された。

（←の証明）

集合 $(x_1, x_2, \cdots, x_s)$ のｎ個の要素からなる部分集合 $(x_{i_1}, x_{i_2}, \cdots, x_{i_n})$ がｙを含むとする（$RECT^n$ において）。従って、

$y \in RECT^n(x_{i_1}, x_{i_2}, \cdots, x_{i_n}) \subset RECT^n(x_1, x_2, \cdots, x_s)$

より証明された。

性質６．１はｎ＝２の場合は既にIchino［１４］に示されているので、その拡張である。

本章のアルゴリズムではｎ次元ベクトルは閾値の集合により二値的特徴に変換されるので、この性質はそのまま成立しない様に思う。しかし、事実上各ベクトルを閾値上の点として考えても問題はないので、結局、この性質が成立する。

さて、性質６．１をアルゴリズム６．１あるいはその基になるアルゴリズム５．１と合わせて考えると次の事がわかる。

「発見されるすべての極大充満部分集合｛Ｓ｝を規定する測定値特徴数の最大値をｐとすると、アルゴリズム５．１において｛Ｓ｝は少なくとも極大ｐ項完全部分集合の数え上げまでに発見される」。一つの極大充満部分集合Ｓに着目し、それがｐ個の測定値特徴のみで排他性を保持しているとする。つまり、Ｓの要素全体が張る超区間はｐ次元部分空間において、その内部に他のクラスのサンプルを含まないとする。この時、性質６．１より、すべてのｐサンプルの組 $(x_{i_1}, x_{i_2}, \cdots, x_{i_p})$ もその張る超区間の内部に他のクラスのサンプルを一つも

含まない。これはＳがｐ項完全部分集合である事を示す。従って、その様なＳは極大ｐ項完全部分集合として見つかるはずである。よって、すべての極大充満部分集合の測定値特徴数の最大値ｐ以下ですべての極大充満部分集合は発見される。一つのクラスを規定する測定値特徴数はそれほど多いとは思われない。従って、この上限はアルゴリズム６．１あるいはその支配的なステップであるアルゴリズム５．１のおいて、クラスがよほど複雑な構造を取らない限り実行に必要な計算量はさほど大きくならない事を示している。具体的には、アルゴリズム５．１で計算量の面で支配的なのは、手続きＥＮＵＭＳＡＴ（ｎ、Ｇ）における、集合Ｇのすべてのサイズｎの部分集合に対する排他性の検査であり、これは｜Ｇ｜のｎ乗のオーダーである。従って、上の性質でｎがｐで押さえられる事は、少なくともその面において計算が実行可能な範疇に入る有効な証拠を与える。よって、残りの極大ｎ項完全部分集合の数え上げの計算量が実行可能な範疇である事がわかれば、アルゴリズム全体を通しての計算量が十分実行可能なものである事が言える。これに関しては第５章でクラスの構造がそれほど複雑でなければその数え上げも実行可能な範疇の計算量で済む事を示したので、最終的に、クラスの構造がある程度単純であるという仮定の下でアルゴリズム６．１は実行可能な計算量である事がわかる。

６．３．　識別規則としての評価

ここでは、部分クラスを識別規則の側面から考え、従来の手法との比較により評価を行う。

パターンの識別手法の分類の一つとして次のものが考えられる。

１）決定規則（決定境界）を定めるもの

２）少数の標準パターンを用意し、それらとの距離を基にするもの

３）クラス毎に特徴空間上に領域を確保するもの［領域確保法］

１）のカテゴリーには統計的分類手法の大部分（線形、非線形識別関数、Bayes識別規則など）が含まれ、２）はＮＮ法（Nearest Neighbor 法）に代表される。厳密には、１）では各クラスの領域が決定境界により定められおり、２）においても異なるクラスの標準パターン間の距離により各クラスの領域は確保されてい

るので３）との区別はないように思われる。しかし、１）ならびに２）により確保される領域は一般に複雑なものであり、それに比べて３）は単純なクラス領域の確保を目的としているのでここでは区別する。

ここでの比較は、本章の手法が３）のカテゴリーに位置付けられるので、特に３）に属する代表的な手法との比較を試みる。実際、識別効率のよさに関しては、これから挙げる様な特徴軸に平行、あるいは垂直な識別面の構成が最もよいと思われる。また、前述の「特徴軸のスケーリングによる影響を受けない」という利点もある。

比較の前に、その構成法の違いから領域確保法をさらに次の二つに細分類する。

１）階層的分割手法、

２）領域拡張手法。

明確な区別として、１）は大分類から細かな分類へと進みつつ領域を細分するのに対し、２）は徐々に領域を拡張しながら含むサンプル数を増加させていくものである。

以下では、それぞれの分類の代表的手法を簡単に説明し、その後比較実験を行う。

６．３．１．　階層的分割手法

この種類に属する手法は、次の手続きで２分木を構成しながら領域を確保する。

０）訓練サンプル集合を$S = S_1 \cup S_2$（S_1、S_2はそれぞれクラスω_1、ω_2からのサンプル）、特徴集合を$F = \{f_i\}$、$1 \leqq i \leqq d$　とする。また、評価関数を$f(i, t_i)$とする。ここで、ｉはｉ番目の特徴を表し、t_iはｉ番目の特徴軸上に設定された閾値を表す。評価関数ｆはすべてのサンプルをｉ番目の特徴軸上の閾値t_iで分けた時の分類の良さを計る関数である。

１）$f(i, t_i) = \max_{j、t_j} f(j, t_j)$　として、特徴軸ｉ上で閾値t_i以下であるサンプルを集合ＳＬ、それより大きいサンプルを集合ＳＲに入れる。

2）ＳＬ、ＳＲのそれぞれについて、ＳＬまたはＳＲの要素のラベル付けを調べ、終了判定を満たすならば　3）へ。それ以外なら、おのおのをＳと考え、1）へ。

3）ＳＬまたはＳＲのサンプルのラベルに応じて、今まで用いてきた（特徴軸、閾値）の組の列

$$(i_1, t_{i_1}), (i_2, t_{i_2}), \cdots, (i_n, t_{i_n})$$

に対応する領域を確保し、ラベルをつける。

この手続きにおいて、終了判定は着目しているサンプル集合におけるすべてのサンプルがS^+またはS^-の片方だけに属すれば、勿論終了となる。この他にもサンプルの比やサンプル総数を考慮している研究もある。ここでは、本質的に手法を特徴づける関数ｆのみに着目する。代表的な方法は以下の通りである。

（1）Friedman(1977)の方法［6］

ｆ：Kolmogorov-Smirnoff 距離

$$f(i, t_i) = |F^1_i(t_i) - F^2_i(t_i)|$$

但し、F^1_i、F^2_iはそれぞれ、特徴ｉにおいてS_1とS_2のサンプルから見積られた累積度数分布とする。

（2）Sethi and Sarvarayudu(1982)の方法［30］

ｆ：平均相互情報量(average mutual information)

$$f(i, t_i) = \sum_{i,j} p(ci,xj)\cdot\log(p(ci|xj)/p(ci))$$

但し、x1，x2は特徴ｉにおける閾値t_i以下か、より大きいかの起こる事象で、c1，c2はクラスω_1、ω_2の起こる事象を表し、p(ci,xj)は事象ci、xjの同時確率、p(ci|xj)はxjの条件下での事象ciの起こる確率、p(ci)は事象ciの起こる確率で、いずれもサンプルから見積られる。

（3）Li and Dubes(1986)の方法［22］

ｆ：順列統計量(permutation statistic)

$$f(i, t_i) = |S(V(i, t_i), C(i)) - 0.5|$$

但し、S（x，y）はx，yを二値ベクトルとした時、yの要素のすべての順列が均一に生ずると考えた時のxとyの合致する1の数を考慮した統計量で、1－1の合致数が多ければ1に近づき、1－0の合致数が多ければ0に近づく、上の式ではどちらかのベクトルにおいて1と0を交換すれば1－0の合致数は1－1のそれになる事を考慮している。また、サンプルを特徴軸iの上で値の増加順にソーティングした後、Vは閾値t_i以下であるサンプルの数だけ0を、残りのサンプルの数だけその後に1を続けた二値ベクトル、Cはソーティング後のサンプルのクラスラベルが1なら0、2なら1とした時の二値ベクトルである。

識別性能だけに着目するといずれの方法もそれほどの差異はない。また、（3）の方法は（2）の終了条件の不安定性を補う事を除いては、識別性能には差異はない事が報告されている［22］。

6．3．2．　領域拡張手法

領域拡張手法の例は、著者の知る限りIchinoのBox-classifier［11、15］のみである。従って、ここではその方法を簡単に述べる。

0）訓練サンプル集合を$S = S^+ \cup S^-$（S^+、S^-はそれぞれ領域を確保したいクラスならびにそれ以外のクラスの訓練サンプル集合）、特徴集合を$F = \{f_i\}$、$1 \leqq i \leqq d$　とする。

1）S^+において最大数の相互近隣を有するサンプルx_1をシード点として、別な点x_2をx_1と併せてその二点を含む最小の超区間を構成し、それが最もよく他クラスのサンプルを排除する様なサンプルx_2を付け加える。次に、さらにもう一点を加えて同じ事を考え、この操作を生成された超区間が排他性を満たさなくなるまで続ける。

2）1）において一つの超区間の構成において用いたS^+のサンプルをすべて除去して、再び1）を行う。こうして、すべてのサンプルを含む超区間の集合を得る。

ここで、x_1の相互近隣とは、点x_2をx_1と併せてその二点を含む最小の超区間を構成し、それが他クラスのサンプルを排除する時、x_2はx_1の相互近隣であるという。

この手順を見ると、本章の方法とかなりの類似が認められる。主な相違点は次の２点である。

（１）Box-Classifierでは、一度超区間の構成に用いたサンプルはその後の超区間の構成には用いない。それに対し、本章の方法は、一つのサンプルは自然に複数の部分クラスに含まれる、

（２）Box-classifierの識別領域はサンプルの値そのものに依存する。これに対し本章の方法は客観的閾値を用いる為サンプルの値には影響されない。

特に、Box-Classifierにおいて、（１）の性質は重要で、あとでも述べる様にサンプルの用い方として不十分になり、二番目以降に得られる超区間の信頼性が低くなる。また、（２）の性質は特に真の識別境界付近にサンプルが得られていない場合、識別領域の近似の精度が落ちる事を意味する。

６．３．３．　比較実験

ここでは、いままで述べてきた手法の幾つかと本章の方法との比較を実験例により行い、その後検討する。

（例６．２）

二クラスω_1、ω_2で各サンプルが二次元ベクトルとする。この時、階層的分割手法の代表として、Sethi and Sarvarayudu［３０］による平均相互情報量に基く識別境界を図６．４（ｃ）に示す。また、Ichino［１１、１５］によるBox-classifierの領域を図６．４（ｂ）に、本章の方法による識別境界を図６．４（ａ）に示す。ここで、Box-classifierによって得られるものはクラス毎の領域であって、識別領域ではない事に注意を要する。実際、Ichino［１５］は、超区間からの一種の距離を識別に用いる事を提案している。

この結果から、平均相互情報量に基く方法は概形としては適度な識別境界を構成しているものの、その細部に不十分な面が見られる。これはこの方法に限らず

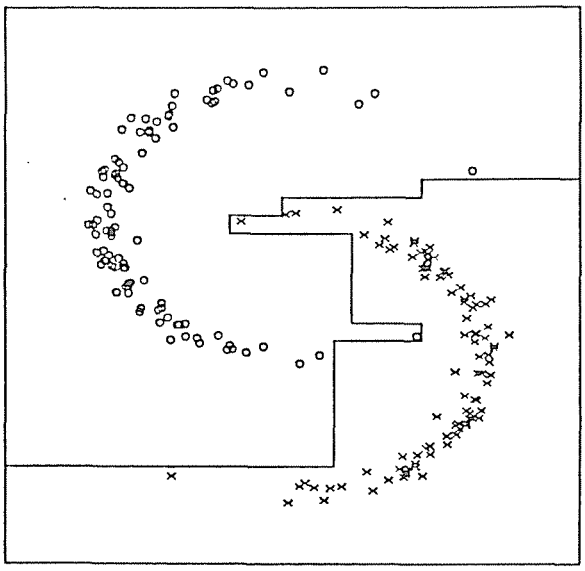

図6．4（a）　例6．2の識別境界
（部分クラスによる）

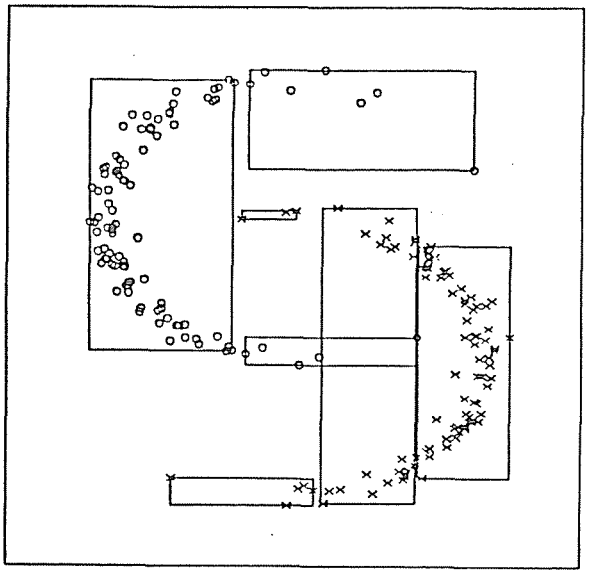

図6．4（b）　例6．2の識別境界
（Box-classifierによる）

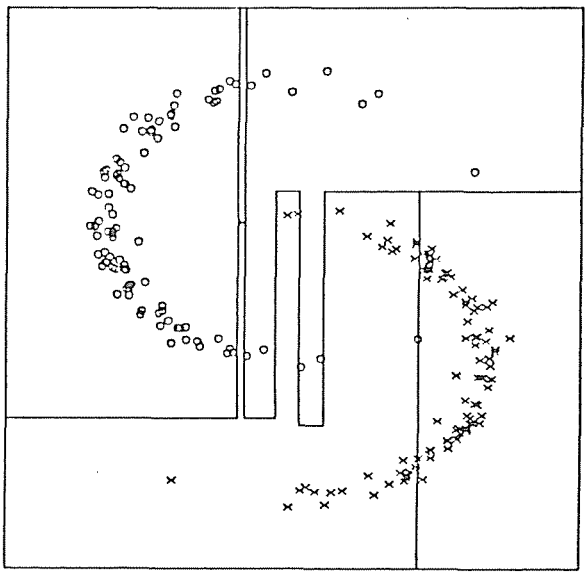

図6．4（c）　例6．2の識別境界
（平均相互情報量に基づく）

階層的分割手法全般に言える事で、分割が進むにつれ境界の設定に用いるサンプル数が少なくなり、それに対応して境界の信頼性が薄れるのが理由である。その進みを制限する終了条件が研究されているが［２２］、そうすると細部の領域の扱いが不十分になるのは、図６．４において、主だった分割面のみを残してみてもわかる。次に、Box-Classifierに関しては、最も大きな（サンプルを多数含む）超区間以外の超区間においてはその領域が妥当とは思われないものが存在している。それらの中には、距離を導入したとしても識別に不適当と思われる領域の存在も確認される。また、同じクラスのBox同士の間に不連続な領域が見られる。これらの現象は一つのサンプルが複数のBoxの構成に用いられる事はなく、階層的分割手法の場合と同様に、手続が段階的で後半に構成される超区間であればあるほど用いるサンプル数が減少する事が原因である。その為、細部における識別面の構成が不十分になる。

総括的に、階層的分割手法も領域確保手法もサンプルの用い方が不十分であると考えられる。それに対し、本手法はすべてのサンプルをどの部分クラスの構成にも用いる為、それによる改善が識別境界の細部に見られる。

６．４．　特徴選出

「特徴選出」の問題はパターン認識における重要な主題の一つである。その重要性は大きく２つの利点からくる。１）コストの節約、２）識別性能の向上がそれである。１）のコストはさらに１－ａ）特徴の測定コスト、１－ｂ）識別の計算コストに分けられる。また、１－ａ）のコストは「特徴抽出」つまり本論文で言うところの”特徴の変換”を行う時はその計算コストも含める事にする。

これまで、上述の２つの面のうち１）のコストに関しては多くの研究がなされている。その理論的考察には”決定木”が有効である［４］。決定木を最も簡単に”２分木”とする。決定木は識別規則そのものをも表し、識別対象ｘが入力された場合、まず”根”においてすべての特徴集合$F = \{ f_i \}$の中から一つの特徴f_1が測定され、その測定値に応じてｘは左か右の”子”に移る、そこでまた新し

い特徴f_2に対して同様の事が行われ、その操作がクラスによりラベル付けのされている”葉”にxが至るまで続けられる。従って、”測定する特徴の集合”が入力パターン毎に異なる。各特徴毎に”測定コスト”を与え、また、誤識別に対しクラス毎（どのクラスをどのクラスとして間違えたか）に”誤りコスト”を考え、その両方のコストの和の期待値を最小にする”最小コスト”問題が研究されている［4、9、28］。また、”識別コスト”（入力パターンの認識にかかる時間）を木の”深さの総和”または”最大深さ”により評価し、これらのコストを最小化する事も研究されている［26］。

それに対して、2）識別性能の向上の側面に関してはまだ研究が十分になされてはいない。有力な一つの視点は”次元数（特徴数）、サンプルサイズ、誤識別率の問題”としての扱いである［2、17］。この視点では、「どの識別規則（識別機構）の型に関して、現在の特徴を用いた時、どれほどのサンプル数が有効な識別規則の構成に必要であるか？反対に、現在のサンプル数に対してはどれほどの次元数が最適であるか？」などが主たる関心事である。楽観的な予想「次元数は多ければ多いほど識別性能の高い規則が構成できる」はごく限られた識別規則（最適なBayes識別規則）だけに見出される理想的なものである事が判っている［2、17］。従って、実際に構成可能な識別規則においては、その殆どが予想の否定「最適な有限の次元数が存在する（ピーク現象）」を考慮する必要がある。これは本質的にサンプル数の不足が原因であるけれども、実際使用できるサンプル数は有限であるからこの問題は避けて通れない。最適次元数の決定に関しては、サンプル数と次元数の比を基に考察する事の有効性が示されている［17］。

本章は2）の性能向上の側面における貢献を目的とする。但し、”識別規則の型”を固定して最適次元数を考察する事は一般性に欠ける為、本章では専ら”識別に必要のない特徴の除去”を考察する。つまり、現実のパターン認識機構の構成には”どの特徴が識別に有効か”を予め判断する事は難しい。従って、一般的に望まれる使用法は”識別に有効と思われる特徴”すべてをできる限り測定し、その数多い特徴から”本当に識別に有効な特徴”のみを選出する事と思われる。識別規則の型を固定しその最適次元数を求めるにしても、こうして選ばれた特徴

集合の中にその最適次元数に対応する特徴が含まれる事は明らかである。従って、本章の目的を明確にすると、少なくとも”識別に貢献しない”むしろ”有害”である特徴の除去を目的とする。これは消極的ではあるが、ピーク現象を考えると確かに識別性能を向上させるであろう。なぜなら、次元数が増加すると識別規則がすべての特徴を生かしきれず、むしろ識別性能を劣化させてしまう（ピーク現象）のが問題で、識別に貢献しない特徴の冗長な使用は当然識別性能を落していると考えられるからである。

6．4．1．　評価関数と探索手続き

特徴選出問題における一般的な目的は「D個の特徴集合から最も識別に有効なd（＜D）個の特徴からなる部分集合を見つける事」である。しかし、この定義にはdの決定の問題が含まれている。他の外的要因（コストの問題など）から制約を受けなければ、この選定は特に前述のピーク現象を意識した場合難しい問題である。実際の特徴選出に関する多くの研究はその点に言及していない。この点に関しては、Ichino and Sklansky［１２］、Foroutan and Sklansky［７］らがそれぞれ「与えられた識別率を維持する最小の特徴集合を見つける」という新しい視点を提供している。本章の解析もこれに準ずるもので、「識別率が著しく減少しない（識別率の差がある閾値以下である）最小の特徴集合の発見」を目的とする。

どの目的の達成にも必要なのは１）選んだ特徴集合による識別率の見積り、２）候補の特徴集合の選出である。本当の識別率を知る事は実際の問題では殆ど不可能であるので、サンプルから見積らなければならない。また、候補となるのはなんの制約もなければ${}_{D}C_{d}$個の部分集合であり、これはさほど大きくないDに対してもすぐに実行不可能な数となる。そこで、１）に対してBayes識別率を十分反映する評価関数、２）に対して特徴の部分集合の探索手続きを述べる事が個々の特徴選出手法を特定化するものである。

評価関数としては、divergence、Bhattacharyya距離など色々なものが提案されており、それぞれとBayes誤識別率との関係が研究されている［１９］。しかし、これらの殆どはサンプルからの分布の見積りを必要とする。従って、やはり少数

のサンプルの場合、推定の信頼性が問題となる。そこで、ある種の”直接法”が有効と考えられている。つまり、使用可能なサンプルの中をできるだけ有効に使って識別率を見積ろうという試みである。代表的な方法は”one-leave-out法”と呼ばれ、一つのサンプルを除いた訓練サンプル集合で識別規則を構成し、その後当初のサンプルを正確に識別できるかどうかを見る方法で、単純にサンプル数だけ試行を繰り返す事ができる。また、サンプルの分け方（識別規則の構成に用いるサンプルとテストに用いられるサンプル）が色々工夫されている。

探索手続きに関しては、最適解を求める唯一の方法は特徴の組合せをすべて調べる事である事がわかっている［１、３４］。従って、種々の制約を用いて探索すべき部分集合の数を減らす事が検討されており、前向き探索［３５］、後向き探索、または少規模のバックトラックを持つ探索など各種の方法が提案されている。しかし、いずれも”準最適解”の探索で満足しなければならない。ここで、前向き探索とは「ｋ＋１個の最適特徴部分集合はｋ個の最適特徴部分集合を含む」と考え、順次ｋを１から上げていく方法で、後向き探索はその逆に、ｋを下げていくものである。探索の効率化において著しい成果を挙げたのは分岐限定法（branch and bound method）の適用［２５］である。しかし、この方法には評価関数の単調性という大きな制約がある。つまり、ある特徴集合の評価値はそのどんな部分集合の評価値より小さくはない事が仮定される。この仮定が満たされれば、分岐限定法は”最適解”の発見を保証する。しかし、この仮定は、実際の識別機構を用いる以上、その識別率に関してはサンプルの有限性から成立しない。

これまでの議論により、標準的な方法は「分岐限定法などの効率の高い探索手続き、直接法による識別率の見積りにより準最適解を見つける」事と思われる。しかし、これでもまだかなりの計算量を要求する。実際、分岐限定法を用いて、Ｄ＝２４、ｄ＝１２の時に６０００の部分集合を検査している例がある。また、直接法もサンプル数に応じてかなりの回数の試行を要求する。特に直接法の１回の試行には１回の識別規則の構成が含まれ、識別規則の構成には一般的にかなりの計算量を必要とするので、この回数は非常に重要である。

６．４．２．　クラスの構造と識別率

識別率は訓練サンプル集合から見積られ、直接法が望ましい。しかし、その場合、見積にかなりな回数の試行を要する。それに対して、本節ではクラスの構造を識別率の代用として用いる事を提案する。

すべての極大充満部分集合ならびにそれに対応する部分クラスの集合は限定された形ではあるけれど、現在のサンプル、現在の特徴集合におけるクラスの構造を完全に規定していると考える事ができる。従って、ここでは極大部分集合の族を基にクラス構造を量的に表現する有効な指標を与える。

[準備]

一つのクラスに対し、そのクラスの訓練サンプル集合$S^+ = \{x_1, x_2, \cdots, x_n\}$ならびに特徴集合$F = \{f_1, f_2, \cdots, f_d\}$を用いて、第6．2節の方法で見つけた極大充満部分集合の族（部分クラスに対応する）を$\Omega(S^+, F) = \{S_1, S_2, \circ\circ\circ, S_m\}$とする。また、$|S_i|$により極大充満部分集合$S_i$の濃度、つまり、極大充満部分集合$S_i$に含まれる$S^+$の要素数を表すとする。但し、任意の訓練サンプルは必ずどれかの極大充満部分集合に含まれるとし、もし、含まれない訓練サンプルが存在する時は、そのサンプルを含む極大充満部分集合として濃度0の空集合がその様なサンプル数だけΩに含まれているものとする。

[定義6．1]

1）$Max(\Omega) = \max_i |S_i| / |S^+|$

2）$Min(\Omega) = \min_i |S_i| / |S^+|$

3）$Ave(\Omega) = \frac{1}{m} \Sigma |S_i| / |S^+|$

これらの量を基に、現在の特徴集合と訓練集合における考慮中のクラスの構造を表す事を考える。

1）クラスらしさ（まとまりの度合）

クラスらしさは一つのクラスが一つの極大充満部分集合よりなる場合、つまりすべての訓練サンプルが同一の排他的な共通特徴を有し、一つの極大充満部分集合に含まれる場合に最良である。反対に、濃度が小さい極大充満部分集合の数が多くなる程クラスらしさは失われると考えられる。従って、一つの測度としてＡｖｅ（Ω）が考えられる。

２）集中度

１）のクラスらしさに近い概念であり、特に密集している部分の大きさを計るものとしてＭａｘ（Ω）を用いる事ができる。

３）離散度

２）と正反対の概念で、同じクラスのサンプル間の共通性が最も低い部分の程度を計る目的にＭｉｎ（Ω）を用いる事ができる。

４）モードの個数

本質的に幾つの部分クラスからなると考えうるかについては、そのモードの個数を計る事が一つの解決法である。その推定法の一つとして次のアルゴリズムが考えられる。

1)適当な閾値θ_A、θ_Bを定める（$0 \leqq \theta_A \leqq 1$）；

2)ｉ＝１；　ｊ＝１；　Ｔ＝ϕ；

3)T^Cと共通部分を最大にする部分集合をＳｉとする；

4)$|Si|/|S^+| > \theta_A$　かつ　$|T \cap Si|/|T^C \cap Si| < \theta_B$

ならば、ｊ＝ｊ＋１　そうでないなら　8)へ；

5)Ｔ＝Ｔ∪Ｓｉ；

6)$T^C = \phi$　ならば　8)へ；

7)ｉ＝ｉ＋１；　3)へ；

8)モードの個数としてｊを出力；　終了。

但し、$T^C = S^+ - T$とする。

閾値θ_A、θ_Bはそれぞれモードとして考えうる最小の部分集合の大きさ、分離の程度を表すもので、$\theta_A = 0.1$、$\theta_B = 1.0$などの値が自然であろう。

次に、これらのクラスの構造を表す指標と識別率との関係を考察する。まず、Ｍａｘ（Ω）に関しては、この値が識別率と関係が深いのは容易にわかる。なぜ

なら、S^+のサンプルだけに関して考え、各部分クラスを単体の識別規則として考えた場合、最大の極大充満部分集合に対応する部分クラスが最もよい識別率を有するからである。また、Ｍａｘ（Ω）に関しては単調性が保証される。なぜなら、次元数ｋにおける特徴集合をF_kとすると、F_kにおける最大極大充満部分集合は当然新しい特徴を一つ付け加えたF_{k+1}においても充満ではある事は確かであるからである。従って、Ｍａｘ（Ω）は理論的なBayes識別率を模倣すると考えられる。次に、Ａｖｅ（Ω）についてである。この値は、総合の識別率を最もよく表すと考えられる。一つの部分クラスが単体で一つの識別規則を構成すると見なす事ができるので、Ａｖｅ（Ω）の値が高いのは全体的にS^+のサンプルに対してどの部分クラスも高い識別能力を持つ事を示している。それに対し、値の低い時は識別能力の低い識別規則を数多く有している事を示す。従って、その値は全体としての識別性能の指標と考えられる。また、クラスのまとまりと言った観点から考察すると、あまり識別に貢献のない特徴の付加に関しては、偶然の共通性をS^+のサンプル中に持ち込み、小さな極大充満部分集合の生成を助長すると考えられる。従って、この指標は”実際に識別に有効な特徴”の付加にだけその値を増加すると考えられるかも知れない。よって、そのピーク現象からもＡｖｅ（Ω）は実際の識別率を模倣すると考えられる。

次に、検証の為に実験例を示す。

（例６．３）

二クラスω_1、ω_2で生起確率は両方0.5とする。特徴は八つで、各パターンは$x=(x_1,x_2,\cdots,x_8)$で表されるとする。また、ω_1は平均（0,0,･･･,0）分散共分散行列が単位行列Ｉの正規分布。ω_2は同様に正規分布で分散共分散行列が単位行列Ｉ、平均(1.036,0.842,0.674,0.524,0.385,0.253,0.126,0.0)である。従って、各特徴のみを用いた場合のBayes識別率はそれぞれ、0.85,0.8,0.75,0.7,0.65,0.6,0.55,0.5となる。実験は最初のｋ個の特徴を用いた場合のω_1の部分クラスを発見する事をサンプル数１００／クラスで独立に５回、それらすべてのサンプルをまとめてサンプル数５００／クラスで１回行った。特徴数ｋと最大部分クラスの大きさＭａｘ（Ω）の関係を図６．５（ａ）に、特徴数ｋと部分クラスの平均の大きさＡｖｅ（Ω）との関係を図６．５（ｂ）に示す。また、それぞれの図にBayes規則を用いた場合のクラスω_1の正識別率を併せて載せている。

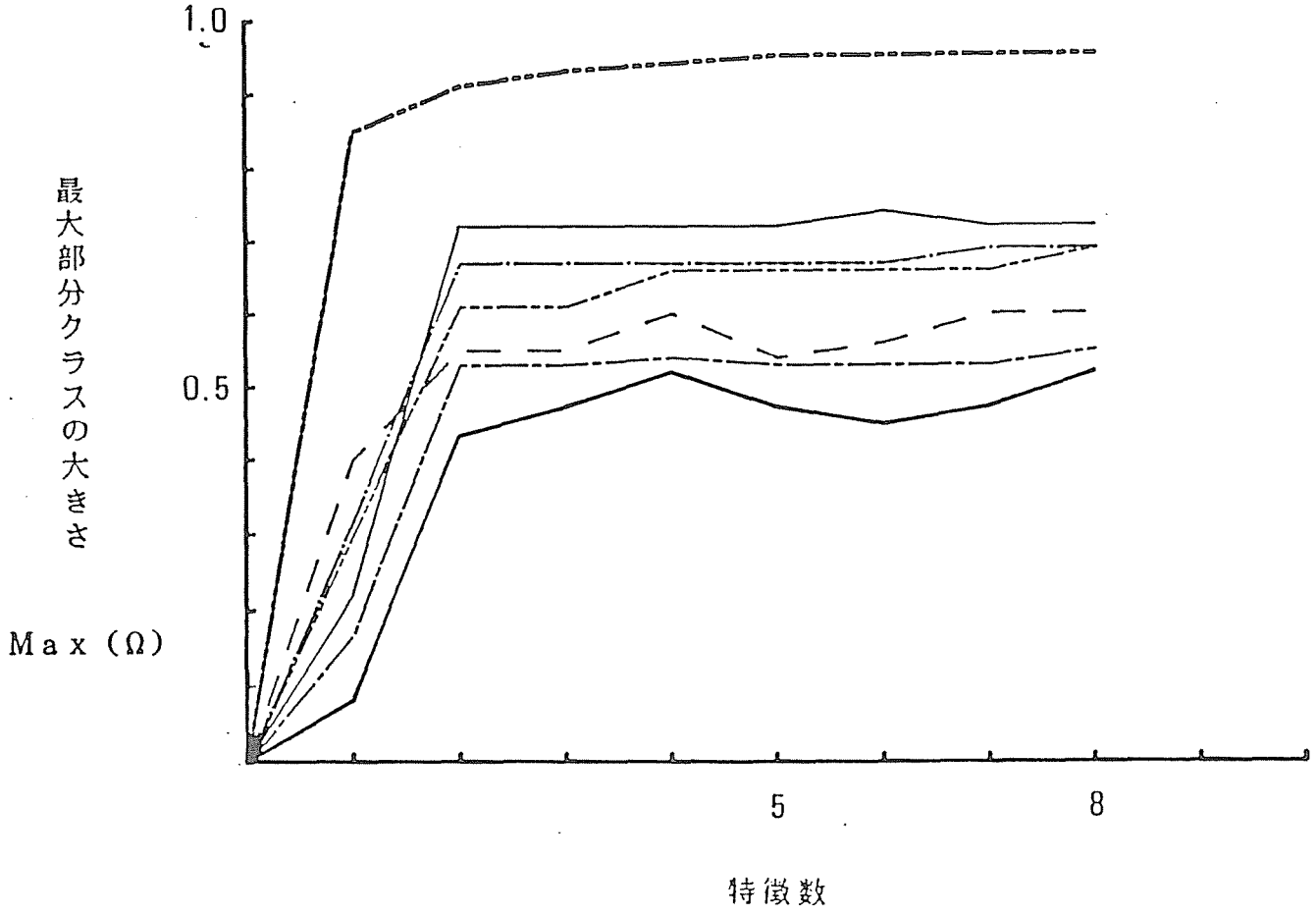

図6．5（a）　例6．3における特徴数と最大部分クラスの大きさとの関係
（最大部分クラスの大きさは訓練集合S^+の大きさ$|S^+|$を1として正規化している。最大の特徴数は8である。細い5種類の線は$|S^+|=100$での5回の独立な結果を示す。太い実線はそれらすべてをサンプルとして$|S^+|=500$の実験結果を表す。また、太い二点鎖線はBayes規則によるこのクラスの正識別率を示す）

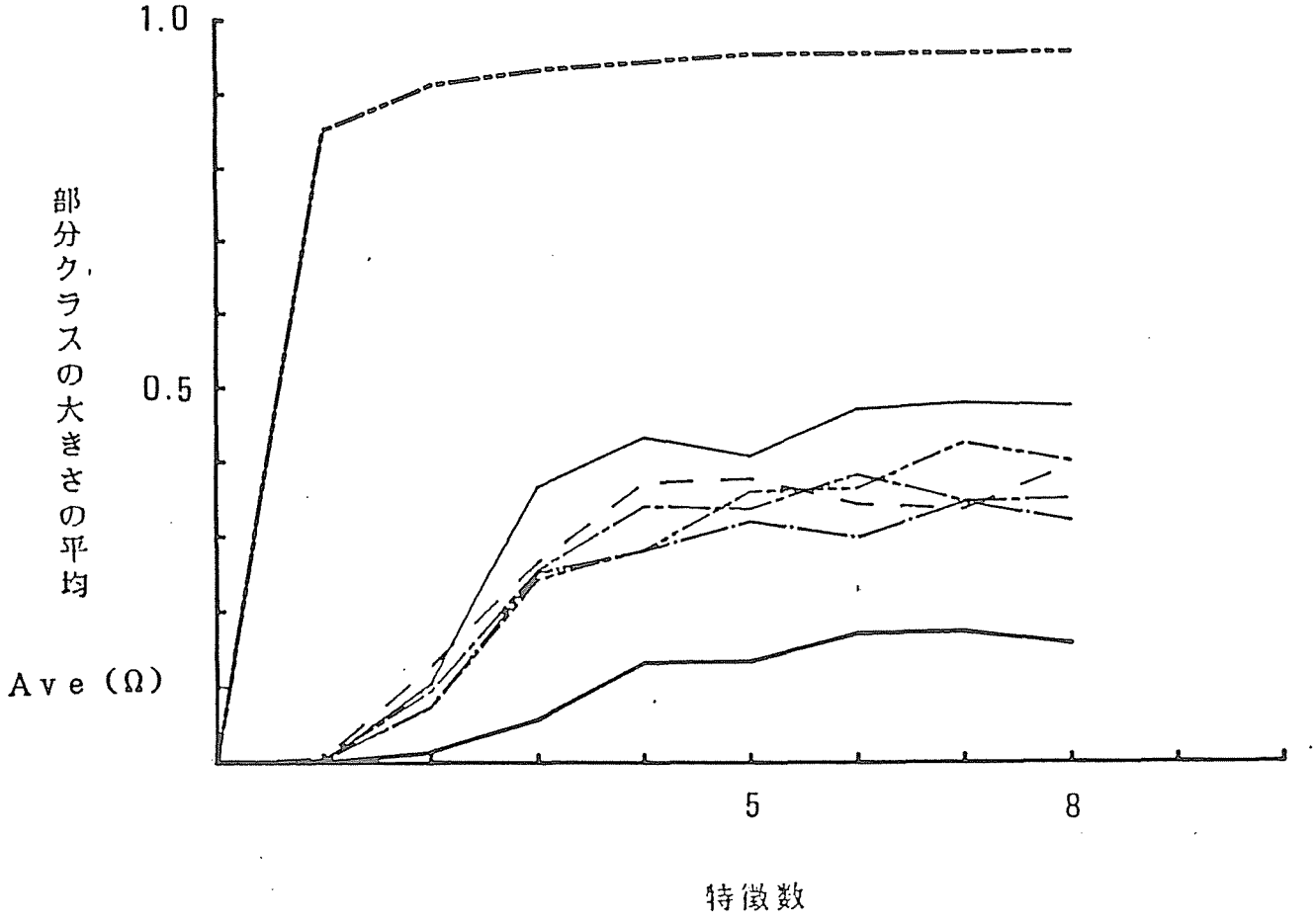

図６．５（ｂ）　例６．３における特徴数と部分クラスの大きさの平均との関係
（部分クラスの大きさの平均は訓練集合Ｓ⁺の大きさＩＳ⁺Ｉを１として正規化している。最大の特徴数は８である。細い５種類の線はＩＳ⁺Ｉ＝１００での５回の独立な結果を示す。太い実線はそれらすべてをサンプルとしてＩＳ⁺Ｉ＝５００の実験結果を表す。また、太い二点鎖線はBayes規則によるこのクラスの正識別率を示す）

図６．５（ａ）から、Ｍａｘ（Ω）の特徴数に関する変化は識別率のそれを満足できる程に模倣しているのがわかる。また、この図においてＭａｘ（Ω）が特徴数に関して単調でない部分がある。これは部分クラスの発見アルゴリズムにおけるサンプル数減少の為の前処理が原因であり、理論的には起こり得ない。一方、Ａｖｅ（Ω）と特徴数との関係を図６．５（ｂ）から考察すると、こちらも識別率の変化を的確に捉えているのがわかる。但し、最大部分クラスの場合と特徴的に異なる部分は特徴数７－８間でグラフの傾きがが負になっているものが５回の独立な実験のうち三つもあり、よりクラスの情報の多い５００サンプルのグラフに関しても見られる事である。これは、８番目の特徴は分類情報をもたず”無意味な”情報であり、前の議論で述べた様にその情報により識別率はそのままで、むしろクラスとしてのまとまりが少し減少した事を表す。

（例６．４）

二クラスω_1、ω_2で生起確率は両方0.5とする。特徴数は十で、各パターンは$x=(x_1,x_2,\cdots,x_{10})$で表されるとする。また、$\omega_1$は平均（0,0,･･･,0）分散共分散行列が単位行列Ｉの正規分布。ω_2の最初の４特徴は球状の厚みのある殻の内部（内径3.5、外径4.0）に一様分布、残りの６特徴は平均（0,0,･･･,0）分散共分散行列が単位行列Ｉの正規分布をしているとする。従って、最初の４特徴以外の特徴に二つのクラスを識別する情報はない。実験は最初のｋ個の特徴を用いた場合（ｋ＝３、４、６、８、１０）のω_1の部分クラスを発見する事をサンプル数１００／クラスで独立に５回、それらすべてのサンプルをまとめてサンプル数５００／クラスで１回行った。特徴数ｋとＭａｘ（Ω）の関係を図６．６（ａ）に、特徴数ｋとＡｖｅ（Ω）との関係を図６．６（ｂ）に示す。

図６．６（ａ）から、どの実験もＭａｘ（Ω）の特徴数に関する変化はｋ＝４で最大になり、その後の付加特徴によるグラフの傾きはほぼ０である。この事は実際の識別率に合致する。この図においてＭａｘ（Ω）が特徴数に関して単調でない部分がある。これは部分クラスの発見アルゴリズムにおけるサンプル数減少の為の前処理が原因である事は前に述べた通りである。また、Ａｖｅ（Ω）

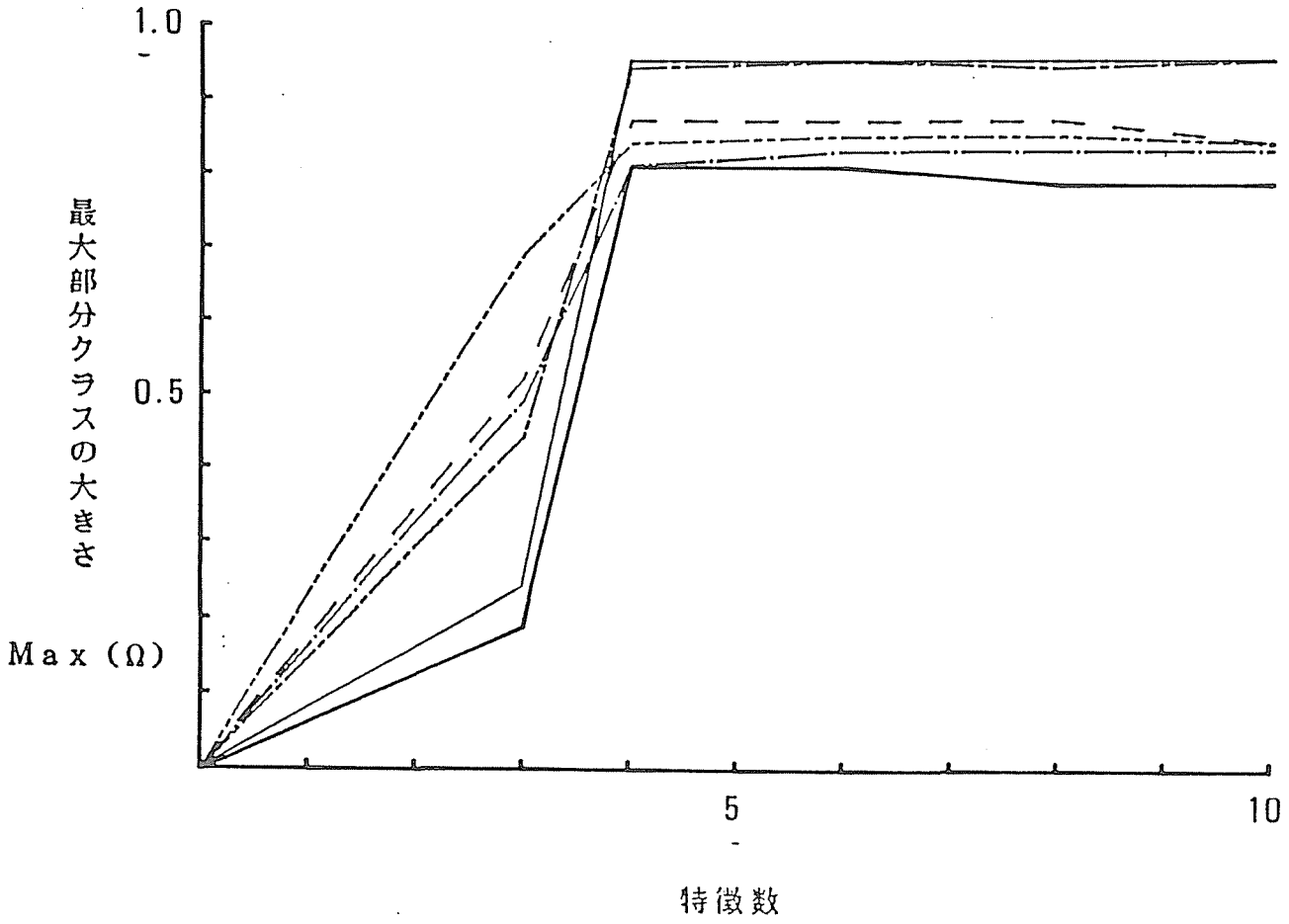

図6．6（a）　例6．4における特徴数と最大部分クラスの大きさとの関係
（最大部分クラスの大きさは訓練集合S^+の大きさ$|S^+|$を1として正規化している。用いた特徴数は3、4、6、8、10である。細い5種類の線は$|S^+|=100$での5回の独立な結果を示す。太い実線はそれらすべてをサンプルとして$|S^+|=500$の実験結果を表す。）

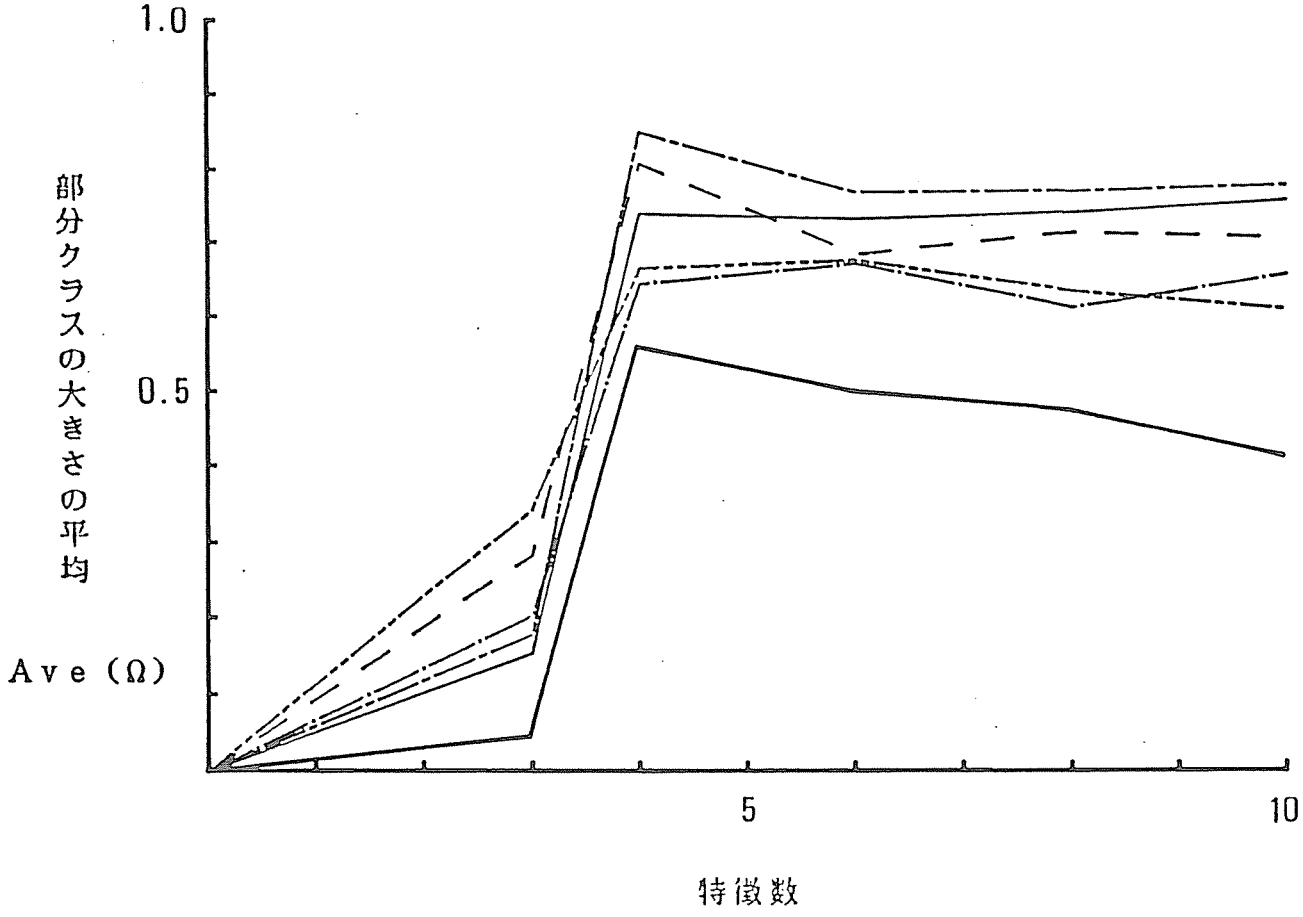

図6．6（b）　例6．4における特徴数と部分クラスの大きさの平均との関係
（部分クラスの大きさの平均は訓練集合S*の大きさ|S*|を1として正規化している。用いた特徴数は3、4、6、8、10である。細い5種類の線は|S*|＝100での5回の独立な結果を示す。太い実線はそれらすべてをサンプルとして|S*|＝500の実験結果を表す。）

と特徴数との関係を図６．６（ｂ）から考察すると、こちらの場合はｋ＝４を最大としそれ以後はむしろ減少しているものが５回中２回、また５００サンプルのグラフにも見られる。多サンプルの方がクラスの構造をより反映すると考えられるので、５００サンプルの場合に着目すると、”無意味”な特徴を付加すればするほどクラスとしてのまとまりが減少している事を示している。

これらの実験と前述の議論から、Ｍａｘ（Ω）、Ａｖｅ（Ω）がかなりよく識別率の代用となる事がわかる。両方の指標のより理論的な解釈が待たれる。ここでは、これらの値が十分識別率の代用となると考え、Ｍａｘ（Ω）またはＡｖｅ（Ω）またはその両方それらを用いた特徴選出アルゴリズムを述べる。

６．４．３．　アルゴリズム

識別率の代用となる評価関数として、１）最大部分クラスの大きさＭａｘ（Ω）、２）部分クラスの大きさの平均Ａｖｅ（Ω）、のどちらかあるいは両方を用いる事にすると、残る問題は探索手続きである。特徴の全体集合から部分集合をとりだして調べる事においては、分岐限定法［２５］などにより効率化が図られているが、やはりまだ多くの部分集合を検査しなければならない。これに対し、本章では特徴の部分集合を検査するのではなく、あくまでもクラスの構造を規準として、部分クラスの検査に基く探索手続を述べる。

準備として、一つのクラスにおける極大充満部分集合の族Ω（S^+、Ｆ）を考え、Ωの部分集合で次の性質を持つものを見つける。

［定義６．２］最小極大充満部分集合族

$$|S^*_i| = \max_{x_i \in S_j \in \Omega} |S_j|, \quad x_i \in S^+$$

として

$$\Psi(S^+, F) = \{S^*_i \mid x_i \in S^+\}$$

このΨが先に述べた識別方法で訓練サンプルS^+のすべての要素を識別した場合

に実際に用いられる部分クラスに対応しているのは定義から容易にわかる。つまり、現在の訓練集合を用いる限り、この最小極大充満部分集合族に対応する部分クラスが最小の識別規則を構成する。その意味で、他のさほど重要ではない部分クラスの影響を受けずに済む。

次に、Ψにより各特徴の重要度を表す重みを定義する。準備として、論理式αにおいて、gにより変換された測定値特徴iの二値特徴すべてを1ブロックと考え、そのブロックに含まれる1をすべて0にした場合の論理式に添字iを付けてα_iとする。この時、

測定値特徴iの重みw_iは

$$\hat{w}_i = \sum_{S \in \Psi} |C^-(a(S)_i)| \circ |S|$$

とした時、その正規化した値

$$w_i = \hat{w}_i / \Sigma \hat{w}_i$$

で与えられる。つまり、各極大充満部分集合に対し着目している特徴が他のクラスのサンプルをどれだけ排除しているかを計っている。

この重みを用いて、アルゴリズムを示す（図6．7）。

［アルゴリズム6．2］（特徴選出アルゴリズム）

（0）S^+を考慮中のクラスの訓練サンプル集合、S^-をそれ以外のクラスの訓練サンプル集合、Fを特徴集合とする；

（1）初期設定。$i \leftarrow 1$；$F_i \leftarrow F$；θ_A、θ_Bを設定；

（2）現在の特徴集合を用いて極大充満部分集合の族$\Omega(S^+, F_i)$を求める；

（3）評価関数fを用いて、Ωから$t_i = f(\Omega)$を求める；

（4）$i = 1$でなく、今回の評価値t_iを前回の評価値t_{i-1}と比較して閾値θ_Aより下回ったら前回の特徴集合を出力して終了する；

（５）最小極大充満部分集合族Ψ（S^+、Ｆ）を求める；

（６）Ψを用いて特徴の重み｛w_j｝を計算する；

（７）閾値θ_Bに対して、$w_j < \theta_B$なる特徴ｊを除去する；

（８）すべての特徴が除去されずに残った場合、現在の特徴集合を出力して終了する；

（９）ｉ←ｉ＋１；

（１０）残った特徴をF_iとして、（２）へ。

ここで、（３）で用いる評価関数ｆの候補として三種類を挙げておく：

１）$f_1(\Omega) = Max(\Omega)$

２）$f_2(\Omega) = Ave(\Omega)$

３）$f_3(\Omega) = (Max(\Omega) + Ave(\Omega))/2$

また、　終了条件は二通りあり、

ａ）特徴集合の評価値が前回の特徴集合のそれより著しく減じた時

ｂ）検査中のすべての特徴の重要度がある閾値以上になった時

のいずれかが生じた時アルゴリズムは終了する。ａ）は識別率規準であり、ｂ）は著しく識別情報の少ない特徴がなくなった事を示すもので、事実上両方の規準は同一の終了条件となると思われる。違いはａ）の判定の出力結果は特徴集合F_{i-1}であるので”後判定”、ｂ）は特徴集合F_iを出力する為”前判定”を行う事であり、ｂ）の判定は一回分の（２）の計算を節約できる。

また、（７）において、θ_Bを固定せず、毎回最低の重み（０でない）を持つ特徴を除去する様にする事も可能である。

（例６．５）

例６．４で用いた二クラスω_1、ω_2、十次元ベクトル空間（最初の四次元のみが識別情報を持つ）を再び用いた。１００サンプル／クラスを訓練集合とし、クラスω_1に対して、アルゴリズム６．２をθ_A=0.1、θ_B=0.05として適用した。測定値特徴をすべて用いた時（$F_1 = F$）のクラス構造の指標Ｍａｘ（Ω）、Ａｖｅ（Ω）はそれぞれ0.95,0.75であった。また、その時の各特徴の重みは

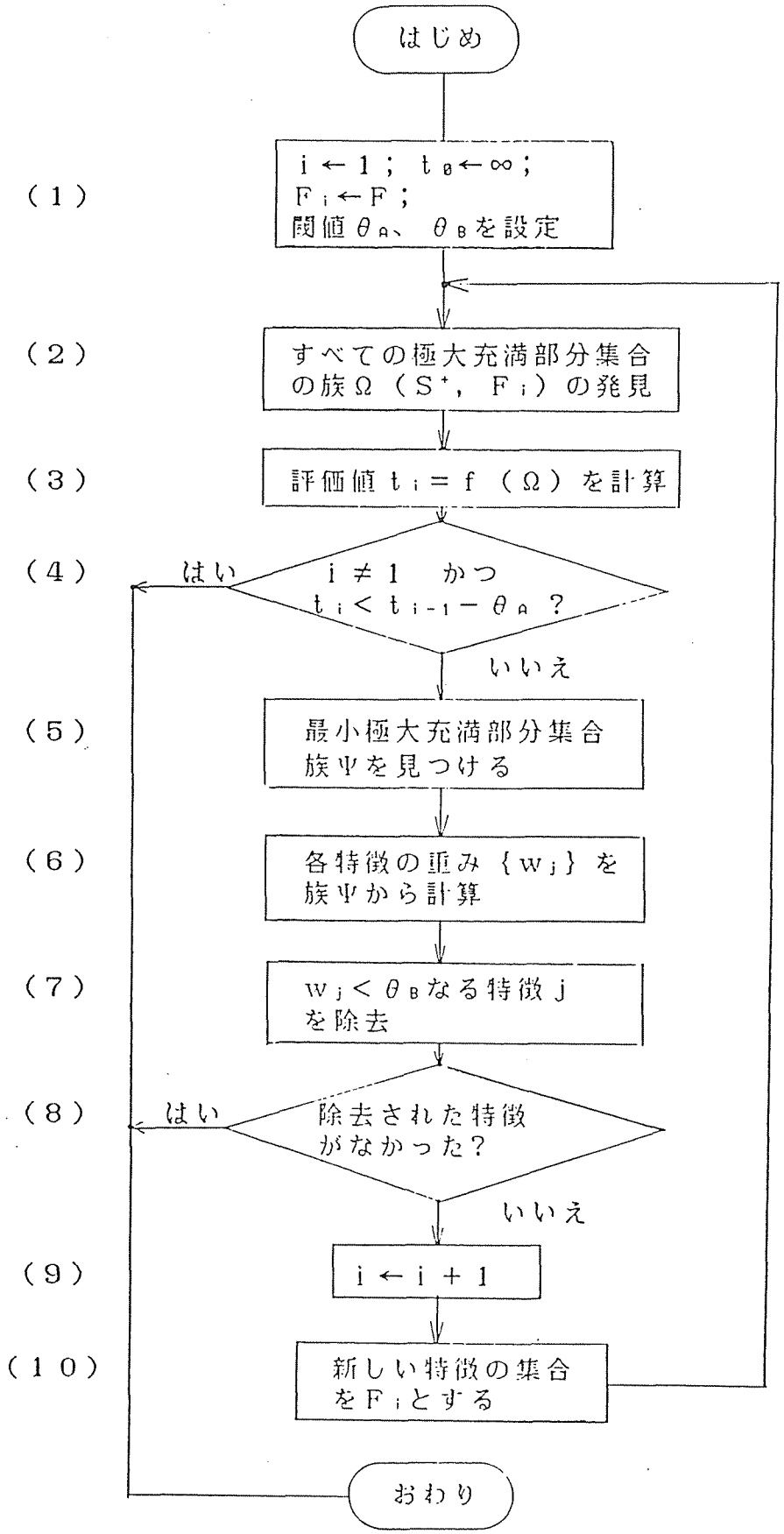

図6．7　特徴選出アルゴリズム

w＝(0.23,0.23,0.26,0.27,0,0,0,0.005,0,0.005)

であった。次に、特徴集合F_1から閾値θ_B=0.05より小さい特徴（特徴５～特徴１０）を除去すると、新しい特徴集合として F_2＝｛特徴１～特徴４｝が選ばれ、再びΩを計算する。その結果、Ｍａｘ（Ω）=0.95、Ａｖｅ（Ω）=0.74となる。ここでＭａｘ（Ω）またはＡｖｅ（Ω）のどちらを用いても前回との評価値の差はθ_Bより小さいので、特徴の重みを計算すると

w＝(0.21,0.25,0.31,0.23)

となり、特徴集合 F_2のすべて特徴の重みが閾値θ_B=0.05以上なので終了条件ｂ）によりアルゴリズムは終了する。特徴集合F_2＝｛特徴１～特徴４｝が選出された。

アルゴリズム６．２による特徴選出に関してその特徴をまとめる。その際に評価関数と探索手続に分ける方が従来の手法との比較が行い易い。まず、評価関数に関しては新たにクラスの構造を識別率の代用とした。その結果、理論的な識別率の代用としてＭａｘ（Ω）が、実際の識別率の代用としてＡｖｅ（Ω）がそれぞれ考えられる事が示された。次に、探索手続ではこれまでの後向き探索と形式を等しくする手続を採用した。しかし、各特徴の重要度はクラスの構造から計られる為、”クラスのまとまりに貢献度の少ない”特徴が除去される。この点については、従来の”重要度が低い”特徴を順次除去するものに比べ、より積極的に特徴間の関係を考慮しているといえる。例えば、識別率によらず、もはや”除去すべき”特徴がない事の判断も可能になった。同時に、除去する特徴を一度に複数選ぶ事も可能である。

次に、性能ならびに実行計算量で現時点においてかなり有効と思われる最近のForoutan and Sklansky［７］の方法との比較をしてみる。Foroutan and Sklanskyの方法は評価関数として区分的線形識別関数による直接法を、探索手続として分岐限定法の変形である”隠的列挙を用いた０－１整数数計画法［１２］”を用いている。それまでの大部分の手法と特徴的に異なるのは、識別関数を探索手続に先んじて構成し、その識別関数の識別率を一つの規準とし、その規準と比べてそれほど識別率を落とさない特徴の部分集合を検査していく事である。これは従来法が個数を規準としていた事と対照的である。本章の規準も Sklanskyの規準に近いけれども、識別率の特徴数に関する単調性に関して決定的な違いがあ

る。つまり、実際の区分的線形識別関数においてはその単調性は保証されない。この時、Sklanskyの方法では最適解は必ずしも発見されない。これに対し、本章の方法は単調性に無関係に、現在の部分集合の識別率が前の大きな部分集合のそれより著しく小さくならなければ探索は続けられる。つまり、現在の識別率の方が上がっても探索は続けられるのでピーク現象に対応できる。その規準は一見Sklanskyの方法にも持ち込めそうであるが、単調性が探索手続の効率化の基になっている為単純にはいかない。本章の方法がこの規準を採用できるのはその探索手続が単調性を利用していないからである。

また、計算量に関しては「特徴数に関して線形のオーダーのステップで実行可能」と言う重要な事実がある。実際、アルゴリズム６．２は特徴数に関して線形のステップのみ存在し、Ωの発見に用いるアルゴリズム６．１も特徴数に線形な計算量である事は前に述べた。この事実は我々に初期の大ざっぱな特徴の測定を許す。つまり、計れるだけの特徴を採用し、特徴空間を作り、その後”識別に有効な特徴”のみを選出する事を可能にする。これは特徴数が計算量において支配的であった従来の方法では実行がかなり困難であった事である。

本章の方法に関しても”最適解”は保証されず、”準最適解”である可能性が高い。しかし、識別率の代用として本章で定義したクラスの構造の指標がどういう仮定の下で、どの程度十分であるかを定量的に示す事ができるならば、本章の方法によって最適な特徴集合を求める事のできる問題を明確にする事ができる。この方向は重要な課題であると思われる。

６．４．４．　特徴の十分性

特徴選出の方法をこれまで検討してきた。しかし、今一度大きな視点で考え直してみると、「果して特徴選出以前の特徴集合はクラスを規定するのに十分であったのだろうか」という問題が生じる。ここでは、この問題に対する答を考察する。

まず、これまでにこの様な議論がそれほど表面化していない最大の理由は「主だった特徴は測定してあり、現在の特徴でどれほどの識別を実行できるかを問題にしている」事と思われる。しかし、事実上、１）特徴の最適な使用ならびにそ

れに伴う識別率の推定が満足できるほどに行えない、２）余り多くの特徴を取り入れると後の特徴選出が困難である、などが原因である事も考えられる。実際、本質的にクラスを規定する特徴が”現在の特徴集合”に含まれていない場合、その集合を用いて識別系を構成しても満足のいくものはできないと思われるので、やはり現在の特徴が十分なものであるかを判断できる指標は有用である。２）の問題点は先に述べた様に、本章の手法においては問題ない。そこで、１）の問題点に焦点を絞って考察する。

十分性の判定の目的には、１）の問題点を克服する為、むしろ識別率と別種な観点を導入する。つまり、これまでにも多く触れた”クラスの構造”を規準とする。これは本章の識別手法も含めて用いる識別規則の型に無関係な規準であるので、特徴集合との関係を純粋に考慮できる。勿論、超区間を用いたクラスであるという制限は残る。判定の根拠として、次の三点に関して注目する。

１）特異的にクラスのまとまりを損なう様なサンプルはないか、

２）一番大きなクラスのまとまりはどれ程か、

３）モードの個数はいくつか。

簡単に、考慮中のクラスに関して１）は下限、２）は上限の測定である。従って、第６．４．２節の記号を用いて、それぞれＭｉｎ（Ω）、Ｍａｘ（Ω）を用いる事ができる。また、３）も同章の方法で測られる。

これらをもとに、例として次の様な判定法が考えられる。

「もし、Ｍａｘ（Ω）が低くてモードの個数も１であれば、クラスのまとまりが不足していると考えられ、特徴数を増やす事が検討される。多モードの時はその数を合わせて判断の根拠とする。また、Ｍｉｎ（Ω）が低すぎる場合（＝０など）の時も、これはサンプルの信憑性が十分高ければ特徴が不十分と考えられる。従って、サンプルの妥当性と併せて検討する。」

６．５．　サンプル選出

特徴選出と双対にサンプルを選出する事が考えられる。その背景には大きく二つの局面がある。

まず、”サンプルの信憑性”の問題である。現在入手しているサンプルは果し

て誤りなくラベル付けされているだろうか。複数のクラスに属するとする事のできる様な、あるいはラベル付けを迷うようなサンプルはなかったか。実際、多くの問題ではクラスが重なりを持つと言われる。この点に関しては、本論文においては、「本来、クラスは重なりを持たないものである。もし、現在の特徴集合上でクラス間に重なりが明らかに存在する時は、むしろ観測が不十分であるとして特徴を増やす必要がある」という立場を取っているので、その意味からはこの局面の問題は最初の純粋なラベル付けのミスのみとなる。

もう一つの局面は”識別規則の構成の為のサンプル選出”である。すなわち、識別規則を構成する際に除いた方が識別性能を向上させるサンプルはないか。これは特にノンパラメトリックな識別規則に関して重要である。ＮＮ(Nearest Neighbor)法を例に取ると、たった一つのサンプルでも境界付近では識別に重要な役割を果すので、そのサンプルの有無は識別性能の面で大きな差異を生ずる。　サンプルを選出する問題は上の議論の成立根拠である「全てのサンプルはすべて、属するクラスの正当なサンプルである」という主張を考えると、必要のないように思う。しかし、結局のところ、実際の問題を考える時に特徴数は限られ、サンプルがすべて正当であるとも言い難い。従って、各サンプルがその属するクラスをどれほど代表しているかを何等かの規準で測り、それに基いて前述の二つの局面への対処を考える事にする。つまり、”現在の特徴集合上での各サンプルのクラスに対する代表性を測る”事を考える。

サンプルのクラスに対する代表性を計るには、第６．５．３節で定義された最小極大充満部分集合族をもう一度考える。つまり、一つのクラスにおいて極大充満部分集合の族Ω（S^+、Ｆ）が求められている場合に、Ωの部分集合Ψで次の性質を持つものを見つける。

$$|S^*_i| = \max_{x_i \in S_j \in \Omega} |S_j|、\quad x_i \in S^+$$

として

$$\Psi(S^+、F) = \{S^*_i \mid x_i \in S^+\}$$

つまり、サンプル$x_i \in S^+$を含む最大の極大充満部分集合がS^*_iである。す

なわち、x_iは（$|S^*_i|-1$）個の同じクラスの他のサンプルと共通の排他的性質を有する。従って、この値をクラスの代表度として採用するのは自然であろう。実際には、正規化した値

$$d(x_i)=|S^*_i|/|S^+|$$

を用いる事にする。もし、x_iを含む様な極大充満部分集合が存在しなければ、それに対応するものとして空集合を考えるので、$d(x_i)=0$となる。この現象は、同じ二値的特徴を有する他クラスのサンプルがあれば、つまり訓練サンプルにおいてクラス間に重なりがあれば、すぐに生ずる。

この代表度関数dを用いると、例えばd＝0なるサンプルは複数のクラスに属すると考えられ、本章の識別方法などの本質的にクラス間の重なりを考慮しない手法に対しては、そのサンプルを除外する方が望ましい結果を与えると思われる。さらに、この条件を緩めて、与えられた閾値θに対してd＜θならサンプルを除去するという様に判定を拡張する事が考えられる。次に、実験例を示す。

（例６．６）

クラスをω_1、ω_2、ω_3の三種類とし、各パターンは二次元ベクトルであるとする。また、ω_1、ω_2、ω_3はすべて、分散共分散行列が単位行列の正規分布で平均がそれぞれ、$(0,\sqrt{12})$、$(2,0)$、$(-2,0)$である。サンプル数は５０／クラスで、乱数により発生させた。本章の方法による識別境界を図６．８（ａ）に示す。次に、$\theta=0.04$として代表度d＜θのサンプルは除去して、再構成した識別領域を図６．８（ｂ）に示す。明かに境界の設定に改善が見られる。

この例はある意味で非実際的な実験と思われる。なぜなら、この例で除外されたサンプルは明らかに”ラベル付けミス”と考えられるからである。しかし、実際の問題にもやはりこの類のミスは特徴の測定の不十分性から起こり得る。もともと訓練サンプルを入手する段階においては、”このサンプルはこのクラスから生成された、あるいは生成する”という、パターンの特徴とは次元を異にする”

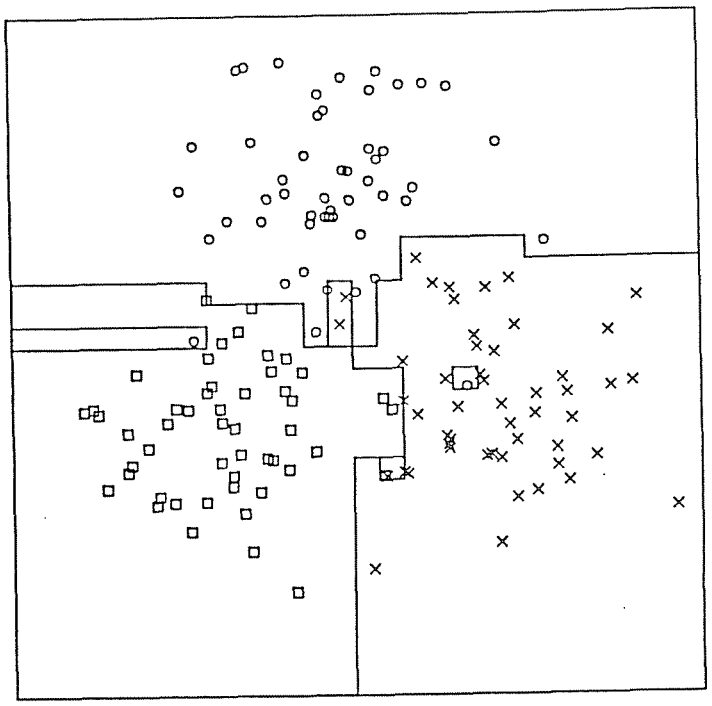

図6．　8　（a）　部分クラスによる識別規則

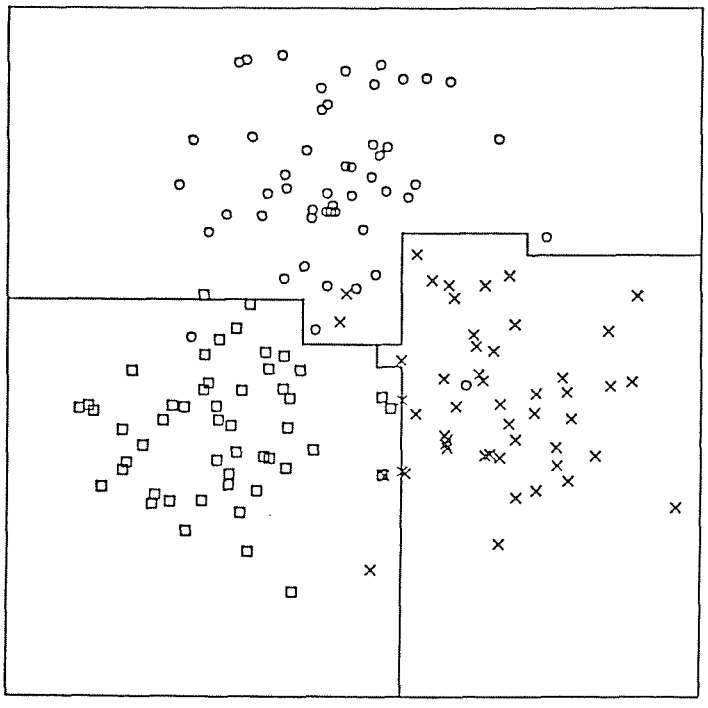

図6．　8　（b）　代表度の低いサンプルを除去した場合の
部分クラスによる識別規則

外的観測”あるいは”意図”に基づいてラベル付けが行われる。つまり、パターンの物理的表現とは別にラベル付けが行われるのが普通である。この点を強調すると、まさに本論文における仮定「特徴の測定をすべて行えばクラスへの割当ては誤りなく行える。つまりパターン間に重なりはない」という主張が、概念の物理的表現の測定可能性の枠内で自然に設定される。しかし、実際の測定特徴集合では物理的表現の捉え方も不十分であるかも知れない、ましてや心理的側面を測る事は今のところ不可能であるから、意図に反した物理的表現を持つサンプルが生成される可能性は高い。例として、「／ａ／と発音しようと意図したが、皆は／ｅ／と聞いた」とか「”７”と書いたつもりが”１”と見られた」などが必ず生じないとは言えない。この状況においては、もはやこの実験例に関して３種類の母分布から自然に生成された正当なサンプルが他クラスのサンプルと見なされる状況となんら変わりがない。

次に、識別性能の向上を目的とするサンプル選出を考察する。まず、クラス間の混在領域を考えてみる。この時、混在領域に属するサンプルは同じクラスの他のサンプルとの共通性は明らかに低い。従って、代表度の低いサンプルを除外する事は混在領域のサンプルを各クラスが引き上げる効果を持つ。この時、もとの混在領域はサンプルの少ない空白領域になる。この状況を想定した場合、各クラスに対して公平な引き上げが行われていれば、識別境界の設定は無理なく行われると思われる。従って、閾値の適度な設定は、混在領域を空白領域に変える役割を担い、本章の方法などの基本的にクラス間に重なりがないと仮定する方法には有効である。

もう一つの重要な視点は特徴の時と同様に「現在のサンプル数で十分なのか」である。これに関しては、もとよりサンプル数は多ければ多いほど識別規則の設定あるいは特徴選出に関して有効であるのは当然である。それ故に、幾らサンプルを集めても十分と言う事はない。そして、得られるサンプルは色々な制約からそれほど取れないのが実状であるのはわかる。しかし、それだからこそ、今少しのサンプルの採取がどれほど有効なのかが判断できれば、それにかかるコストと併せて現状の総合的判断ができるといえる。この面への貢献をクラスの構造の解析から考察してみると、特徴選出に用いた規準が逆の形で有効と思われる。まず、訓練サンプル集合$S = S^{+} \cup S^{-}$に対して、二つの集合をそれぞれｎ等分し、小さ

な集合の対をｎ個作成する。この時、最初からのｋ個の集合対をとる事にする。ｋを横軸に取り、対するクラスの構造（Ｍａｘ（Ω）、Ａｖｅ（Ω）など）を縦軸に取る。この時、ｋの増加に関するクラス構造の変化において、ｋ＝ｎ付近が十分平坦であれば（変化の差がある閾値以下であれば）、現在のサンプルでかなり十分である事が結論できると思われる。なぜなら、サンプル数がクラスの構造を十分正確に表現する程であれば、その後のサンプル数の増加による変化は大きくないと考えられるからである。この点に関しては、さらに検討ならびに議論を必要とし、また実験も不十分なので、ここではその可能性を示すにとどめる。

６．６．　結論

本章では、始めに質量混在の特徴を有するパターンに関して部分クラスを求めるアルゴリズムを述べた。続いて、すべての部分クラスの集合がクラスの構造を規定すると考え、クラス構造を表すいくつかの指標を与えた。それらを基に特徴選出やサンプル選出の方法を提案すると共に、それらの指標をパターン認識機構全体における性能向上の評価規準として用いる事の有効性を示した。従来の研究との比較を中心に、本章により改善された点は以下の通りである：

１）識別規則として高い識別効率（処理速度）を有する矩形領域の設定の一手法を新しく提案した。この手法は従来のものに比べてサンプルの用い方が優れており妥当な識別境界を設定できる事を示した。また、識別規則が自然に特徴選出を内に含み、識別に無関係な多数の特徴に左右され難い性質を持つ事を示した。

２）特徴選出のアルゴリズムに関して、特徴数に関して線形なオーダーの計算量を持つアルゴリズムを提案した。これにより、従来より多くの特徴の中から特徴選出を行う事が可能になった。

３）特徴選出の終了規準として、識別率の変化をとる事を提案した。これは識別率のピーク現象に対応する可能性を持つ。

４）現在のサンプルならびに現在の特徴集合がどれほどの識別率を持つのか、さらに特徴を増やす必要はあるのか、サンプルはこれで十分か、などのパターン認識系全体の現状把握に、新しく”クラスの構造”と言う視点を導入し、それらの問題に対処する一つの試みを示した。これにより、これまで別々に考えられがち

であった処理を有機的に考察する事が可能になった。

残された問題点ならびに今後の課題は以下の通りである：

1）本章のアルゴリズムは主に第5章のアルゴリズムを基礎としており、クラスの持つ構造に依存して計算量が膨大になる事もある。従って、より計算量の少ないアルゴリズムの開発が望まれる。

2）クラスの構造を量的に表す指標をいくつか挙げ、それらと識別率の関係を考察したが、現段階では実験による評価が強く、今後理論的な解釈が待たれる。

3）本章において考察されたクラスは矩形領域を基にしているので、より一般的な形のクラス構造の解析が待たれる。

第７章　結論

パターン認識における識別規則の構成および特徴選出の方法を、同一クラスのサンプルパターン間の共通性および異クラスのサンプルパターン間の相違性に着目して検討した。その結果、これまでとは異なった側面において手法の得失が明らかになり、幾つかの改善を行うことが出来るようになった。本論文において得られた主な成果は以下の通りである。

１）パターンの性質を解析する一つの方法として構文的アプローチを考察し、従来法に比べ推論性質が明確で実用的な三種類の正規文法推論手法を提案した（第３章）。

２）遺伝子配列の解析に１）で提案した構文的アプローチを導入し、生物の制御機構が文字列のつながりを重視している可能性が高い事を示した（第４章）。

３）パターンがベクトル表現されている時に、クラス内に排他性と極大性の要請を満たす部分クラスを見出すことがパターン識別に効果的である事を述べ、部分クラスを数え上げる効率的なアルゴリズムを示した（第５章）。

４）部分クラスから導かれるクラスの構造を外的基準として、パターン認識機構全体の性能向上を計ることを提案し、有効性を示した（第６章）。

５）部分クラスを用いた識別規則が同種のこれまでの方法と比較してサンプルの扱い方に優れ、適切な識別境界を構成することを示した（第６章）。

６）部分クラスを用いて、特徴数に線形なオーダーで済む特徴選出アルゴリズムを提案した（第６章）。

また、現在の問題点ならびに今後の課題として以下のようなものが考えられる。

１）正規文法の推論手法はまだ十分に確立されたとは言えない。特に、例を増やして推論結果をより詳細に調べる必要がある。また、手法の分類に関しても再検討の余地がある（第３章）。

２）遺伝子配列の解析は始まったばかりで、構文的アプローチに関してもかなり改善の余地がある。また、生化学的な実験を考慮して、問題の特殊性を解析手法に柔軟に取り入れる必要を感じる（第４章）。

３）部分クラスを求める現在のアルゴリズムに関しては、問題によって計算量が実用的でなくなる場合があり、一層の効率化が望まれる（第５章）。

4）部分クラスから導かれる指標がクラス構造を十分反映し、識別率と正比例的な関係を持つ事が第6章を通じて議論の根底になっている。この事は部分クラスの性質と幾つかの実験において支持されている。しかし、厳密な解析はしておらず、今後理論面での解析および展開が望まれる（第6章）。

本論文全体を通して、同じクラスに属するパターン間の共通性と異なるクラスに属するパターン間の相違性を積極的に用いて様々な問題を取り扱って来た。この扱いはパターン認識において本論文で論じていない問題においても、従来の方法を見つめ直し、拡張を行う目的に有効であると信ずる。従って、今後もこれらに関する不断の研究を続けることが必要と思われる。

謝　辞

本研究は昭和５８年４月から昭和６３年３月迄の期間、著者が北海道大学大学院工学研究科情報工学専攻博士前期、及び後期課程に在学中、情報処理工学講座において行われたものである。

本論文を作成するにあたり、終始一貫して熱心に御指導頂いた北海道大学大学院工学研究科情報工学専攻　新保　勝　教授に心から感謝致します。また、北海道大学大学院工学研究科情報工学専攻　河口至商　教授、同専攻　宮本衛市　教授、北海道大学工学部情報科学　伊達　惇　教授には本研究全般にわたり貴重な御助言、御指摘を頂き、議論を精密化できたことをここに感謝致します。

北海道大学理学部化学第二学科　飯田陽一　講師には遺伝子配列の解析の手ほどきをして頂き、また、論文作成にも多大な御助力を頂き、ここに深く感謝の意を示します。

また、論文の作成過程において常に熱心な御討論、御助言を頂きました北海道大学大学院工学研究科情報工学専攻情報処理工学講座　宮腰政明　助教授に感謝します。

最後に、論文をまとめるにあたり御助力頂いた同講座　外山　淳　助手、同講座　町田大祐君、亀井洋子さん、北村澄枝さんならびにデータ整理、論文校正などにひとかたならぬ御支援を頂いた石田隆張君に感謝します。また、精神的援助をしてくれた妻　陽子、長男　空、次男　夢人に感謝します。

文献

第２章に関する文献

[1] J. C. Stoffel, A Classifier Design Technique for Discrete Variable Pattern Recognition Problems. IEEE Trans. Comp., C-23(1974), 428-441.

以下、主な文献を挙げる：

[2] 上坂吉則、パターン認識と学習の理論．ＩＣＳライブラリ　５、総合図書、1971.
[3] R. O. Duda and P. E. Hart, Pattern Classification and Scene Analysis. John Wiley & Sons, Inc., 1973.
[4] K. S. Fu, Syntactic Pattern Recognition and Applications. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1982.
[5] K. Fukunaga, Introduction to Statistical Pattern Recognition. Academic Press, 1972.
[6] 中田和男、パターン認識とその応用． コロナ社、1978.

第３章に関する文献

[1] A. W. Biermann and J. A. Feldman, On the Synthesis of Finite-State Machines from Samples of Their Behavior. IEEE Trans. Comp., C-21(1972), 592-597.
[2] N. Chomsky, Three Models for the Description of Language. IEEE Trans. Inform. Theory, IT-2(1956), 113-124.
[3] K. S. Fu, Syntactic Pattern Recognition and Applications. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1982.
[4] K. S. Fu and T. L. Booth, Grammatical Inference: Introduction and Survey-Part I. IEEE Trans. Syst., Man and Cybernet., SMC-5(1975), 95-111.
[5] M. Kudo and M. Shimbo, Efficient Regular Grammatical Inference Techniques by the Use of Partial Similarities and Their Logical Relations. Pattern Recognition, in press.
[6] L. Miclet, Inference of Regular Expressions. Proc. 3rd ICPR, 1976, 100-105.
[7] L. Miclet, Inference de Grammaires Regulieres. These de Doctor-Ingenieur, ENST, Paris, France(1979).
[8] L. Miclet, Regular Inference with a Tail-Clustering Method. IEEE Trans. Syst., Man and Cybernet., SMC-10(1980), 737-743.

[9] M. Richetin and F. Vernadat, Efficient Regular Grammatical Inference for Pattern Recognition. Pattern Recognition, 17(1984), 245-250.
[10] F. Vernadat and M. Richetin, Regular Inference for Syntactic Pattern Recognition: A Case Study. Proc. 7th ICPR, 1984, 1370-1372.

第４章に関する文献

[1] Genbank, Genetic Sequence Data Bank, Release 48.0(1987). BBN Laboratories, U.S.A.
[2] Y. Iida and F. Sasaki, Recognition Patterns for Exon-Intron Junctions in Higher Organizations as Revealed by a Computer Search. J. Biochem., 94(1983), 1731-1738.
[3] Y. Iida, Splice-site Signals of mRNA Precursors as Revealed by Computer Search. Site-specific Mutagenesis and Thalassemia. J. Biochem., 97(1985), 1173-1179.
[4] Y. Iida, DNA Sequences and Multivariate Statistical Analysis. Categorical Discrimination Approach to 5' Splice Site Signals of mRNA Precursors in Higher Eukaryotes' Genes. Computer Applications in the Biosciences, 3(1987), 93-98.
[5] M. Kudo, Y. Iida and M. Shimbo, Syntactic Pattern Analysis of 5'-Splice Site Sequences of mRNA Precursors in Higher Eukaryote Genes. Computer Applications in the Biosciences, 3(1987), 319-324.
[6] T. Kuhne, B. Wieringa, J. Reiser and C. Weissmann, Evidence against a Scanning Model of RNA Splicing. EMBO J. 2(1983), 727-733.
[7] K. M. Lang and R. A. Spritz, RNA Splice Site Selection: Evidence for 5'→3' Scanning Model. Science, 220(1983), 1351-1355.
[8] C. Langford, W. Nellen, J. Niessing and D. Gallwitz, Yeast is Unable to Excise Foreign Intervening Sequenses from Hybrid Gene Transcripts. Proc. Nat. Acad. Sci., U.S.A., 80(1983), 1496-1500.
[9] S. M. Mount, A Catalogue of Splice Junction Sequences. Nucl. Acids Res., 10(1982), 459-472.
[10] S. M. Mount, I. Petterson, M. Hinterbergen, A. Karmas and J. A. Steitz, The U1 Small Nuclear RNA-protein Complex Selectively Binds a 5' Splice Site in Virto. Cell, 33(1983), 509-518.
[11] K. Nakata, M. Kanehisa and C. DeLisi, Prediction of Splice Junctions in mRNA Sequences. Nucl. Acids Res., 13(1985), 5327-5340.
[12] J. C. S. Noble, C. Priver and J. L. Manley, In Vitro Splicing of Simian Virus 40 Early pre-mRNA. Nucl. Acids Res., 14(1986), 1219-1235.

[13] R. Parker and C. Guthrie, A Point Mutation in the Conserved Hexanucleotide at a Yeast 5' Splice Junction Uncouples Recognition. Cleavage, and Ligation. Cell., 41(1985), 107-118.
[14] R. Staden, Computer Methods to Locate Signals in Nucleic Acid Sequences. Nucl. Acids Res., 12(1984), 505-519.
[15] V. L. Van Santen and R. A. Spritz, mRNA Precursor Splicing in Vivo: Sequence Requirement Determined by Deletion Analysis of an Intervening Sequence. Proc. Nat. Acad. Sci., U.S.A., 82(1985), 2885-2889.
[16] B. Wieringa, E. Hofer and C. Weissmann, A Minimal Intron Length But No Specific Internal Sequence is Required for Splicing the Large Rabbit β-globin Intron. Cell, 37(1984), 915-925.

第５章に関する文献

[1] M. Kudo and M. Shimbo, Optimal Subclasses with Dichotomous Variables for Feature Selection and Discrimination, submitted to IEEE Trans. Syst., Man and Cybernet.
[2] J. C. Stoffel, A Classifier Design Technique for Discrete Variable Pattern Recognition Problems. IEEE Trans. Comp., C-23(1974), 428-441.

第６章に関する文献

[1] J. D. Elashoff, R. M. Elashoff and G. E. Goldman, On the Choice of Variables in Classification Problems with Dichotomous Variables. Biometrika, 54(1967), 668-670.
[2] J. M. Van Campenhout, Topics in Measurement Selection. Handbook of Statistics, Vol. 2, North-Holland, 1982, 793-803.
[3] T. M. Cover, The Best Two Independent Measurements Are Not the Two Best. IEEE Trans. Syst., Man and Cybernet., SMC-4(1974), 116-117.
[4] G. R. Dattatreya and L. N. Kanal, Decision Trees in Pattern Recognition. Progress in Pattern Recognition 2, North Holland, 1985, 189-239.
[5] R. A. Fisher, The Use of Multiple Measurements in Taxonomic Problems. Ann. Eugenics, 7(1936), 179.
[6] J. H. Friedman, A Recursive Partitioning Decision Rule for Nonparametric Classification. IEEE Trans. Comp., C-26(1977), 404-408.
[7] I. Foroutan and J. Sklansky, Feature Selection for Automatic Classification of Non-Gaussian Data. IEEE Trans. Syst., Man and Cybernet., SMC-17(1987), 187-197.

[8] K. S. Fu, Recent Devolopments in Pattern Recognition. IEEE Trans. Comp., C-29(1980), 845-854.

[9] P. E. Hart, N. J. Nilsson, and B. Raphael, A Formal Basis for the Heuristic Determination of Minimum Cost Path. IEEE Trans. Syst., Sci. and Cybernet., SSC-4(1968), 100-107.

[10] E. G. Henrichon, Jr. and K. S. Fu, A Nonparametric Partitioning Procedure for Pattern Classification. IEEE Trans. Comp., C-18(1969), 614-624.

[11] 市野　学、 相互近接グラフに基づくパターン認識系の設計。　東京電機大学理工学部紀要, 6(1984), 1-10.

[12] M. Ichino, and J. Sklansky, Optimum Feature Selection By Zero-One Integer Programming. IEEE Trans. Syst., Man and Cybernet., SMC-14(1984), 737-746.

[13] M. Ichino and J. Sklansky, Feature Selection for Classifiers. Proc. 7th ICPR, 1984, 124-127.

[14] M. Ichino, The Relative Neighborhood Graph for Mixed Feature Variables. Pattern Recognition, 18(1985), 161-167.

[15] M. Ichino, A Nonparametric Multiclass Pattern Classifier. IEEE Trans. Syst., Man and Cybernet., SMC-9(1979), 345-352.

[16] M. Ichino, Nonparametric Feature Selection Method Based on Local Interclass Structure. IEEE Trans. Syst., Man and Cybernet., SMC-11(1981), 289-296.

[17] A. K. Jain and B. Chandrasekaran, Dimensionality and Sample Size Considerations in Pattern Recognition Practice. Handbook of Statistics, 2(1982), 835-855.

[18] T. Kaminuma, S. Tomita and S. Watanabe, Covariance Matrix Representation and Object-Predicate Symmetry. Handbook of Statistics, 2(1982), 699-719.

[19] L. Kanal, Patterns in Pattern Recognition: 1968-1974. IEEE Trans. Infomation Theory, IT-20(1974), 697-722.

[20] L. N. Kanal, Interactive Pattern Analysis and Classification Systems: A Survey and Commentary. Proc. IEEE, 60(1972), 1200-1215.

[21] X. Li and R. C. Dubes, The Selection of Significant Dichotomous Features. Proc. 7th ICPR, 1984, 260-263.

[22] X. Li and R. C. Dubes, Tree Classifier Design with a Permutation Statistic. Pattern Recognition, 19(1986), 229-235.

[23] W. S. Meisel and D. A. Michalopoulos, A Partitioning Algorithm with Application in Pattern Classification and the Optimization of Decision Trees. IEEE Trans. Comp., C-22(1973), 93-102.

[24] A. N. Mucciadri and E. E. Gose, A Comparison of Seven Techniques for Choosing Subsets of Pattern Recognition Properties. IEEE Trans. Comp., C-20(1971), 1023-1031.

[25] P. M. Narendra and K. Fukunaga, A Branch and Bound Algorithm for Feature Subset Selection. IEEE Trans. Comp., C-26(1977), 917-922.

[26] 折田三弥彦、小林芳樹、高藤政雄、三島忠明、太田秀夫、パタン認識における決定木の最適化方法とその評価. 情報処理学会論文誌, 28(1987), 12-19.

[27] J. O'Rourke, Computing the Relative Neighborhood Graph in the L_1 and L_∞ Metrics. Pattern Recognition, 15(1982), 189-192.

[28] H. J. Payne and W. S. Meisel, An Algorithm for Construction Optimal Binary Decision Trees. IEEE Trans. Comp., C-26(1977), 905-916.

[29] I. K. Sethi and B. Chatterjee, Efficient Decision Tree Design for Discrete Variable Pattern Recognition Problems. Pattern Recognition, 9(1977), 197-206.

[30] I. K. Sethi and G. P. R. Sarvarayudu, Hierarchical Classifier Design Using Mutual Information. IEEE Trans. Pattern Anal. Mach. Intell., PAMI-4(1982), 441-445.

[31] J. Sklansky and L. Michelotti, Locally Trained Piecewise Linear Classsifiers. IEEE Trans. Pattern Anal. Mach. Intell., PAMI-2(1980), 101-110.

[32] J. C. Stoffel, A Classifier Design Technique for Discrete Variable Pattern Recognition Problems. IEEE Trans. Comp., C-23(1974), 428-441.

[33] G. T. Toussaint, The Relative Neighborhood Graph of a Finite Planar Set. Pattern Recognition, 12(1980), 261-268.

[34] G. T. Toussaint, Note on Optimal Selection of Independent Binary-Valued Features for Pattern Recognition. IEEE Trans. Inform. Theory, IT-17(1971), 618.

[35] A. W. Whitney, A Direct Method of Nonparametric Measurement Selection. IEEE Trans. Comp., 20(1971), 1100-1103.

[36] T. Y. Young, P. S. Liu and R. J. Rondon, Statistical Pattern Classification with Binary Variables. IEEE Trans. Pattern Anal. Mach. Intell., PAMI-3(1981), 155-163.