社団法人 電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE
IT2002-7(2002-5)

計算論的学習と情報圧縮に関する一考察

中澤 真† 松嶋 敏泰†† 平澤 茂一††

† 早稲田大学メディアネットワークセンター
〒 169-8050 東京都新宿区戸塚町 1-104
†† 早稲田大学理工学部経営システム工学科
〒 169-8555 新宿区大久保 3-4-1
E-mail: †nakazawa@mn.waseda.ac.jp

あらまし 情報源符号化における形式文法に基づく符号化法は，サンプルデータからオートマトンを導出する過程と考えることができる．これは情報圧縮の過程を機械学習の枠組みで捉えることができることを意味する．しかし，Chomsky 階層に属する文法あるいは計算機モデルの中で，どのようなものを対象とするのが適切であるかは明らかにされていない．また，従来の文法に基づく符号化はいずれも決定性文脈自由文法を用いているが，この文法が必ずしも最適というわけではない．この点を明らかにするため，計算論的学習理論の枠組みから情報圧縮の過程を捉え，ベイズ学習アルゴリズムを用いて十分シンプルで記述長が短い仮説を出力できることを示す．さらにどのような計算機モデルを考えると現実的計算量で最適な仮説を出力ことが可能になるかを示し，これが事前分布の分布族の複雑さと仮説言語の複雑さの両方によって決まることを示す．

キーワード 形式文法，無歪み圧縮，計算論的学習，ベイズアルゴリズム，タイムコンプレキシティ

A Note on Computational Learning and Information Compression

Makoto NAKAZAWA†, Toshiyasu MATSUSHIMA††, and Shigeichi HIRASAWA††

† Media Network Center, Waseda University
Totsuka-cho 1-104, Shinjuku-ku, Tokyo, 169-8050 Japan
†† School of Science and Engineering, Waseda University
Ohkubo 3-4-1, Shinjuku-ku, Tokyo, 169-8555 Japan
E-mail: †nakazawa@mn.waseda.ac.jp

Abstract Recently, grammar based codes are researched in the area of lossless source coding. But, it is not clear which formal grammars is appropriate in Chomsky hierarchy for information compression. In this paper, we consider grammar based codes as the process of deriving automata based on machine learning and it is a purpose to clarify what influence the class of grammars or languages gives the encoding in the view of machine learning. I will show that Bayes algorithm outputs the shortest hypothesis in the class and some class has feasible complexity in order to ouput optimal hypothesis based on computational learning theory.

Key words formal grammars, lossless compression, computational learning, Bayes algorithm, time complexity

1. はじめに

情報源符号化における無歪み圧縮の手法として，形式文法を用いた圧縮法の研究が行われている．最近の研究としては SEQUITUR [15] [16] や MPM [9] [10] などいくつかの研究があるが，いずれも情報源系列を言語として受理するような決定性文脈自由文法を構成することにより圧縮を行う．形式文法のクラスは Chomsky の階層 [7] とよばれる階層構造を持ち，この階層はそれぞれ対応する計算機モデルが存在する．例えば，決定性文脈自由文法は決定性プッシュダウンオートマトン（PDFA）と等価であり，また最も外の階層に位置する句構造文法はチューリングマシンに相当する．しかし，計算量の視点からどの階層に属する文法を用いるべきかということは明らかではない．そこで，この階層の構造を明確にするために計算論的学習理論の枠組みを用いて考えてみる．

機械学習の立場から情報圧縮過程を眺めると，情報源からのサンプル集合を入力，計算機モデルのクラスを仮説空間とし，入力からこれを説明するシンプルな計算機モデルを導出する枠

組みとして考えることができる．このような情報圧縮の過程が学習の本質であるという捉え方はすでにいくつかの研究で示されている [2] [3] [22] [13]．その中の一つ Occam アルゴリズム [3] はサンプルを説明する仮説の選択基準として，より単純なものを選択するということで，直感的にも情報圧縮との関連が明確である．

しかし，計算論的学習理論の分野において問題の複雑さの議論がなされているのは Occam アルゴリズムではなく，Valiant [20] によって提案された Probably Approximately Correct (PAC) アルゴリズムである．PAC アルゴリズムでは多くのクラスに対する結果が得られているが，このアルゴリズムは無矛盾性のみで仮説の決定を行うため，PAC 学習アルゴリズムは必ずしも Occam アルゴリズムとはならない [22]．これは学習アルゴリズムがただちに圧縮アルゴリズムとなるわけではないことを意味している．

そこで本稿ではベイズアルゴリズムを考える．ベイズアルゴリズムは機械学習の分野において近年多くの研究がなされている．それは最適性の議論を行うことが可能であり，また性能的にも高い能力を持つためである．ベイズアルゴリズムと Occam アルゴリズムとの関係を明らかにし，ベイズアルゴリズムによって十分単純な記述長が短い仮説を出力できることを示す．さらにベイズアルゴリズムを用いた場合に，どのような計算機モデルを考えると現実的計算量で最適な仮説を出力ことが可能になるかを示し，これが事前分布の分布族の複雑さと仮説言語の複雑さの両方によって決まることを示す．

なお，本文ではブール関数を対象とするが，q 元，$(q > 2)$ の事例空間，概念表現空間，あるいはより一般的空間への拡張が可能である．

2. 準　　備

最初に，学習モデルを定義する．アルファベットの集合を Σ_X で表し，このアルファベットの有限系列の集合を X と定義し，これを事例空間という．すなわち事例空間は $X \subseteq \Sigma_X^*$ である(注1)．また，事例空間の部分集合を概念とよび，$c \subseteq X$ で表す．さらに，この概念の集合を概念クラスといい C と表記する．ただし，$c \in C \subseteq 2^X$ である．

計算量の議論をするためには概念をどのように記述するかを明確にする必要がある．アルファベットの集合を Σ_R とし，概念表現空間を $R \subseteq \Sigma_R^*$，概念表現を $r \in R$ と定義する．ϕ を概念表現から概念への写像とする．すなわち $\phi : R \to C$ である．この ϕ によって概念クラスと概念表現空間が対応づけられる．

さて，学習アルゴリズムは概念表現から対応する概念を計算する必要がある．まず，$r \in R, x \in X$ に対し，r が x を含む概念であるかを判別する帰納的関数を $\xi(r, x) : R \times X \to \{0, 1\}$ で表し，次のように定義する：

$$\xi(r, x) = \begin{cases} 0, x \notin c ; \\ 1, x \in c . \end{cases} \quad (1)$$

(注1)：Σ^* は集合 Σ の閉包を表す．

ここで，$c = \phi(r)$ である．

次に，事前確率の分布族を定義する．ベイズ決定理論に基づき，概念表現空間，事例空間の両方に確率分布を定義する．R 上の確率分布族を $\Pi = \{\pi_\theta | \theta \in \Theta\}$，$X$ 上の確率分布族を $\Omega = \{\omega_\gamma | \gamma \in \Gamma\}$ とする．ここで Θ，Γ はそれぞれの分布族のパラメータ空間を，$\pi_\theta, \omega_\gamma$ はそれぞれ θ, γ をパラメータとする分布関数である．学習すべき目標概念 $c^* = \phi(r^*)$ は確率分布 $\pi \in \Pi$ に従ってランダムに選択され，事例 x はある未知の確率分布 $\omega \in \Omega$ に従って，定常独立に生成されると仮定する．ここで，分布族のパラメータ空間も概念のデータ構造を特徴づけていることから，知識表現の一部分であるということに注意する．以後，知識表現を (R, Π, Ω) と表記する．

最後に学習のフローを定義する．学習アルゴリズム L は事前情報として概念表現空間 R とその事前確率分布 P が与えられ，m 個のサンプル S^m を獲得後に仮説を出力する．ここで，サンプル S とは確率分布 G に従って生成された x とそのラベル y の対 (x, y) であり，ラベル y は目標概念 c^* に依存して次にように決定される：

$$y_i = \begin{cases} 0, x_i \notin c^* ; \\ 1, x_i \in c^* . \end{cases} \quad (2)$$

ここで，i 番目のサンプルを (x_i, y_i) で表し，また m 個のサンプルを $S^m = < (x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m) >$ で表記する．ただし，i, m は任意の正整数とする．出力すべき仮説やその評価基準は後述の損失関数に依存する．

3. 学習アルゴリズム

3.1 Occam アルゴリズム

Blumer らは計算論的学習理論の枠組みにおいて Occam アルゴリズムを示した [3]．このアルゴリズムはサンプル S^m に無矛盾で $o(m)$ の複雑さをもつ仮説を出力するアルゴリズムである．厳密に定義すると以下のようになる．

[定義 1]　R の Occam アルゴリズム A は以下の条件を満足する仮説 r を出力する．

（1）　仮説 r はサンプル S^m に無矛盾である(注2)．

（2）　$0 \leq \alpha, 0 \leq \beta \leq 1$ を定数としたとき，仮説 r の長さは高々 $(n \cdot l_{min}(R))^\alpha m^\beta$ である．ただし，$n = \max_{x \in X} |x|$(注3)，$l_{min}(R) = \min_{\{r \in R | r \text{ は } S^m \text{に無矛盾}\}} |r|$ とする．

これはまた PAC 学習アルゴリズム [20] にもなっていることが示されている [22]．

3.2 ベイズアルゴリズム

機械学習の分野ではベイズ決定理論の枠組みを用いた研究が多くなされているが，計算量の問題を解決するために近似的な手法やヒューリスティックな方法を用いている場合が多い．これらの研究の多くは最適解からの乖離については実験的に示されているのみである．そのため，決定理論に基づいた定義が不十分である．まず，統計的決定理論に基づき，ベイズ学習アル

(注2)：無矛盾とは仮説 r がサンプル S^m に対して $y_i = \xi(r, x_i), i = 1, \ldots, m$ であることをいう．

(注3)：$|\cdot|$ は集合の要素数を表す．

ゴリズムを定義する．

D を決定空間[注4]，$l(r, d) : R \times D \to \mathcal{R}$ を損失関数，$\delta : (X \times Y)^m \to D$ を決定関数とする[注5]．また事後確率はベイズの定理より

$$P(r|S^m) = \frac{\pi(r)P(S^m|r)}{\sum_{r \in R} \pi(r)P(S^m|r)} \tag{3}$$

で記述される．

一般に損失関数を $l(\cdot,\cdot)$ で表記し，学習のパラダイムによってこれを決定する．例えば，概念学習のように真の概念を同定することが目的の場合，$c^* = \phi(r^*)$ を真の概念，$r \in R$ をアルゴリズムが出力した概念表現とすると，損失関数を

$$l_1(r^*, r) = \begin{cases} 0, \phi(r) = \phi(r^*) ; \\ 1, \phi(r) \neq \phi(r^*) , \end{cases} \tag{4}$$

と定義する．また，予測を目的としている場合は

$$l_2(r^*, r) = \sum_{x \in X} l_1'(\xi(r, x), \xi(r^*, x))\omega_\gamma(x) \tag{5}$$

$$l_1'(y, y') = \begin{cases} 0, y = y' ; \\ 1, y \neq y' , \end{cases} \tag{6}$$

とする．

事後期待損失 r_{post} [1] は以下のように表される．

$$r_{\mathrm{post}}(\delta, S^m) = \sum_{r \in R} l(r, \delta(S^m))P(r|S^m) \tag{7}$$

また，事後期待損失を最小にする決定関数 $\delta^*(\cdot)$ を

$$\delta^*(S^m) = \arg\min_{d \in D} \sum_{r \in R} l(r, d)P(r|S^m) \tag{8}$$

とする．すなわち，この決定は事後ベイズリスク [1] を最小にする決定である．

ベイズアルゴリズムはこれらの損失関数が定義された下で，事後期待損失を最小にする手続きである．例えば損失関数を l_2 とした場合にベイズアルゴリズムは以下のような決定を行なっていることになる．

$$\delta'(S^m, x_{m+1}) = \begin{cases} 0, \dfrac{\sum_{r \in R_0} P(r|S^m)}{\sum_{r \in R_1} P(r|S^m)} > 1 ; \\ 1, \dfrac{\sum_{r \in R_0} P(r|S^m)}{\sum_{r \in R_1} P(r|S^m)} \leqq 1 . \end{cases} \tag{9}$$

ただし，$R_0 = \{r | r \in R, \xi(r, x_{m+1}) = 0\}$，$R_1 = \{r | r \in R, \xi(r, x_{m+1}) = 1\}$，とする．

PAC アルゴリズムでは無矛盾性を成功基準としているため，記述長を最小にする仮説を選ぶというような問題には適さない．一方 Occam アルゴリムは情報圧縮という意味では直感的に適しているが，一般的な最適性を議論したり，計算量の問題を扱うことについて明らかではない．

ここではベイズ最適性考えて，ベイズアルゴリズムを用いるがこのアルゴリズムが Occam アルゴリズムを真に含んでいることを以下に示す．

[定理 1] ベイズアルゴリズムは Occam アルゴリズムを模倣可能である．

[証明] ベイズアルゴリズムにおける事前確率の分布クラス Π_{Occam} を以下のように定義する．$\forall r, r' \in R$ に対し，$|r| \leqq |r'|$ であるならば，

$$\Pi_{Occam} = \{\pi(r) | \pi(r'|\theta) \leqq \pi(r|\theta)\}. \tag{10}$$

このとき損失関数を先に定義した $l_1(\cdot,\cdot)$ とすると，ベイズアルゴリズムはサンプル S^m に無矛盾な仮説を出力するのは明らかである．さらにこの仮説は無矛盾な仮説の中で事前確率が最大であることから，記述長が最小の仮説となる．よって Occam アルゴリズムの定義から題意が示された． □

この定理により，ベイズアルゴリズムが Occam アルゴリズム同様に記述長最小の仮説を出力できることは示すことができた．そこで次に，このときの計算量に注目してみる．議論を簡単にするために，計算論的学習理論で一般的に用いられるブール関数を概念表現空間のクラスとし，損失関数も真の概念の同定を目的とする $l_1(\cdot,\cdot)$ を用いるものと仮定する．

[定理 2] 事例空間を $X = \{0, 1\}^n$，概念表現空間を主加法標準形によるブール関数のクラス R_{DNF} とする．このとき，知識表現 $(R_{DNF}, \Pi_{Occam}, \Omega)$ と損失関数 $l_1(\cdot,\cdot)$ の下で概念クラス C に対し，ベイズアルゴリズムの計算量は n の多項式オーダーである．

[証明] 次のようなアルゴリズムを考える．アルゴリズムの仮説は恒偽からスタートし，正例を受け取る度にそれを項と持つような仮説に更新する．負例の場合には仮説の更新はしない．このときこの仮説はサンプルに対し必ず無矛盾となり，また Π_{Occam} の定義から事後確率も最大となる．このとき，1 つの例に対する計算量がオーダー n であることから題意が示される． □

4. 学習不可能なクラス

Π_{Occam} のような事前分布族の場合には効率的に記述長の短い仮説を出力することが可能であることを示したが，これは一般的な分布族に対して成り立つわけではない．定理 1 の逆が成り立たないことからもわかるように，ベイズ学習で扱うパラダイムのほうが，Occam アルゴリズムよりも一般的だからである．また，クラスをチューリングマシンとした場合には，Kolmogorov complexity [11] を求めることと同義となり，これが NP 完全問題であることから，仮説言語のクラスについても制約条件が必要であることは明らかである．

一般的にベイズアルゴリズムはその定義から，概念表現空間あるいはその上の事前分布族によって事後確率の更新や平均損失を最小にする仮説を探索する手続きに必要な計算量が指数オーダーになる場合がある．

そこでまず事前分布族を Π_{Occam} のように制限せずに任意のものとした場合について議論する．計算論的学習理論の分野では概念表現としてブール関数のクラスを対象にすることが多いが，その中でも構造が極めて単純な単調連言形について考えてみる．

(注4)：いわゆる概念学習では $D = R$ である．
(注5)：$\mathcal{R}$ は実数空間とする．

[定義 2] n 変数の単調連言形 $f(\cdot,\cdot,\ldots,\cdot)$ は負のリテラルを含まない連言形で次のように表すことができる.

$f(x_1, x_2, \ldots, x_n) = x_{\eta_1} \wedge x_{\eta_2} \wedge \cdots \wedge x_{\eta_k}$,

ただし, $\eta_i \in \{1, 2, \ldots, n\}$, $k(k \leq n)$ は正のリテラルの個数である. この単調連言形の概念表現空間を R_{MC} で表す. □

$r \in R_{MC}$ とし, R_{MC} 上の事前分布の族として Π_{MC} を以下のように定義する.

[定義 3] $\Pi_{MC} = \{\pi(r_i)|\pi(r_i|\theta) = \theta_i\}$. ただし, $\theta = (\theta_1, \theta_2, \ldots, \theta_{||R_{MC}||})$ とする[注6].

これは R_{MC} 上にパラメータの自由度が最大となる事前分布族を仮定していることになる. このとき, 次の定理が成り立つ.

[定理 3] 事例空間を $X = \{0, 1\}^n$ とする. このとき, 知識表現 $(R_{MC}, \Pi_{MC}, \Omega)$ と損失関数 $l_1(\cdot,\cdot)$ の下で概念クラス C に対し, ベイズアルゴリズムの計算量は n の指数オーダーである.

[証明] パラメータの次数が $O(2^n)$ であるため, 事後確率の計算量が n の指数オーダとなる. □

この結果は事前分布族を任意のクラスにしてしまうと単調連言形のような単純な表現クラスであっても現実的な計算量では扱えなくなってしまうことを示している.

5. 学習可能なクラス

ここでは合理的な分布族が仮定できれば, 複雑な概念クラスであっても最適解を求める多項式時間アルゴリズムが存在することを示す.

5.1 k-CNF・k-DNF

概念クラス k-CNF, k-DNF は PAC 学習可能であることは Valiant [20] によって示されている. 最初にこのクラスの定義を以下に示す.

[定義 4] (k-CNF) k-CNF (conjunctive normal form) はリテラルの数が高々 k 個の節によって構成される和積標準形である. この k-CNF の概念表現クラスを R_{k-CNF} で表す[注7].

このクラスに対し, 概念表現に含まれている節を直交基底とし, これをパラメータとする分布族を定義する. 事前知識としてそれぞれの節の確からしさが情報として利用できる場合には上記の定義は自然なものと考えることができる.

[定義 5] リテラルの数が k 個の節に対し, インデックス i を与え, その節を ψ_i で表す. このとき概念表現はこの節の集合と考えることができるため,

$$R_{k-CNF} = \{r|r \subseteq \{\psi_i|i = 1, 2, \ldots, \binom{2n}{k}\}\}. \tag{11}$$

このとき R 上の事前分布族を以下のように定義する.

$$\Pi_{\text{clause}} = \{\pi(r)|\pi(r|\theta) = \prod_{\{i|\psi_i \in r\}} \theta i \prod_{\{j|\psi_j \notin r\}} (1 - \theta j)\}. \tag{12}$$

この分布族の下で k-CNF のクラスが多項式時間アルゴリズムで計算可能であることを以下の定理で示すことができる.

[定理 4] 事例空間を $X = \{0, 1\}^n$ とする. このとき, 知識表現 $(R_{k-CNF}, \Pi_{\text{clause}}, \Omega)$ と損失関数 $l_1(\cdot,\cdot)$ の下で概念クラス C に対し, ベイズアルゴリズムの計算量は n の多項式オーダーである.

[証明] 以下の 2 つの補題から証明される. □

[補題 1] k-CNF の概念クラス C は知識表現 $(R_{k-CNF}, \Pi_{\text{clause}}, \Omega)$ とサンプル S^m が与えられたもとで, 事後確率を計算する手続きが n の多項式時間で上界される.

[証明] 付録 1. を参照 □

[補題 2] k-CNF の概念クラス C は知識表現 $(R_{k-CNF}, \Pi_{rmclause}, \Omega)$ とサンプル S^m による事後確率のもとで平均損失を最小にする仮説を出力するのに必要な計算時間が n の多項式でおさえられる.

[証明] 付録 2. を参照 □

[系 1] 定理 4 と同様な結果が k-DNF(disjunctive normal form) についても成り立つ.

5.2 しきい値関数

しきい値関数のクラスは Pitt ら [19] によって PAC 学習不可能なことが証明されている. しかし, ベイズアルゴリズムの場合, 事前分布族によっては効率的に学習することができる場合がある. 最初にこのクラスの厳密な定義を示す.

[定義 6] (しきい値関数) ある整数 $\tau \geq 0$, ベクトル $v \in \{0, 1\}^n$ に対し, $f(x_1, x_2, \ldots, x_n) = \begin{cases} 0, & v \cdot x < \tau; \\ 1, & v \cdot x \geq \tau, \end{cases}$

となる C をしきい値関数の概念クラスとする.

事前分布の重要なタイプとして階層的事前分布がある [1]. この分布を仮定することで連続値の場合は積分計算を簡略化して計算の効率化が図れる. ここでは概念学習のような離散値の場合に, 以下のような階層的事前分布を定義することで, この複雑なクラスが多項式時間で計算可能になることを示す.

[定義 7] まず概念表現を 2 次に分け, ベクトル部分としきい値を明確にわけて表す.

$$R_{th} = \{r|r = (r_1, r_2), r_1 \subseteq \{x_1, \ldots, x_n\}, r_2 = \{1, \ldots, n\}\}$$

.

この時概念表現についての事前分布族を以下のように表す.

$$\pi_{th}(r|\theta) = \sum_{k=1,\ldots,n} \prod_{\{i|x_i \in r_1\}} \theta_{ik} \prod_{\{j|x_j \notin r_1\}} \theta_{jk} \pi_2(\theta_{\cdot k}|\rho)$$

.

ただし, $\pi_2(\theta_{\cdot k}|\rho) = \rho_k, \rho = (\rho_1, \ldots, \rho_n)$ とする.

ここで, π_2 は階層的に上位 (ハイパー) の事前分布であり, ρ はハイパーパラメータとなっている. つまり, この階層は, 最初にしきい値が確定し, その後必要なリテラルがどのようなものか定まる構造を持っている.

[定理 5] 事例空間を $X = \{0, 1\}^n$ とする. このとき, 知識表現 $(R_{th}, \Pi_{th}, \Omega)$ と損失関数 $l_1(\cdot,\cdot)$ の下で, 概念クラス C に対し, ベイズアルゴリズムの計算量は n の多項式オーダーである.

[証明] 重みが t の事例 x を正例として受け取った場合, $\pi_2(\theta_{\cdot k})$ の確率は $k > t$ において 0 となる. また $k \leq t$ での計

(注6): $||\cdot||$ は集合の要素数を表す.

(注7): 例えば $(x_1 \vee x_2) \wedge (\neg x_2 \vee x_3 \vee \neg x_4)$ は 3-CNF である.

算は n の多項式で可能である．一方，重みが t の事例 x を負例として受け取った場合，$\pi_2(\theta_{*k})$ の値は変化しない．このため階層的事前分布の高次の分布について，事後確率の計算はどのような事例を獲得した場合でも n の多項式オーダで計算可能である．

階層的事前分布の低次の分布についての計算は定理 4 の証明を一部修正することによって証明される． □

6. むすび

本稿はベイズ統計学の立場から最適性と計算量を考慮し，ベイズアルゴリズムによって Occam アルゴリズム同様に記述長最小の仮説を選択できることを示した．

概念表現空間上の事前分布を任意のものとした場合には仮説言語がどんなに単純な構造を持っていても計算量的に仮説を導出するのは困難になる．一方，仮説言語が複雑で従来の PAC 学習アルゴリズムなどで計算不可能であったものも事前分布の構造によっては効率的な推定が可能であることを示した．

今回の結果はブール関数での結果であるが同様の議論をオートマトンのクラスへと拡張することが可能である．文法に基づく符号化について計算量理論的立場から考慮するためにはこのクラスでの結果より，適切な文法を用いることが重要である．

また，計算量的に困難なクラスについては，$\epsilon-$ ベイズ解のように，最適解からの誤差を認めることにより，現実的な計算量の中でどの程度まで最適解に近い仮説を導き出せるかという問題も扱えるようになるであろう．

謝辞 著者の一人中澤は本研究に際して貴重な御助言をいただきました横浜商科大学の浮田善文先生と早稲田大学の石田崇氏に深く感謝いたします．また暖かく支援してくださった法政大学の西島利尚先生に深く感謝いたします．なお，本研究の一部は科学研究補助金（課題番号 12875172）の助成による．

文　献

[1] J.O.Berger, "Statistical decision theory and Bayesian analysis, " Springer-Verlag, New York, 1985.

[2] A.Blumer, A.Ehrenfeucht, D.Haussler and M.Warmuth, "Occam's Razor", Information Proccessiong Letters, 24, pp.377-380, 1987

[3] A.Blumer, A.Ehrenfeucht, D.Haussler and M.Warmuth, "Learnability and the VC-dimension", J. ACM, 36(4), 929-965, 1989.

[4] D.Haussler, M.Kearns, and R.Schapire, "Bounds on the sample complexity of Bayesian learning using information theory and VC dimension, " Proc. of the Fourth Workshop on Computational Learning Theory, pp.61-74, 1991.

[5] D.Haussler and A.Barron, "How well do Bayes methods work for on-line prediction of {±1}values?, " Proc. 1992 NEC Symp. on Computation and Cognition, Chapter 4, 1992.

[6] J.E.Hopcroft and J.D. Ullman, "Introduction to Automata Theory, Languages and Computation I," Addison Wesley College, 2000.

[7] J.E.Hopcroft and J.D. Ullman, "Introduction to Automata Theory, Languages and Computation II," Addison Wesley College, 2000.

[8] J.C.Kieffer and E.Yang, "Sequential Codes, Lossless Compression of Individual Sequences, and Kolmogorov Complexity," IEEE Trans. Inform. Theory, vol.42, No.1, pp.29-39, January 1996.

[9] J.C.Kieffer and E.Yang, "Grammar-Based Codes: A New Class of Universal Lossless Source Codes," IEEE Trans. Inform. Theory, vol.46, No.3, pp.737-754, May 2000.

[10] J.C.Kieffer, E.Yang, G.Nelson, and P.Cosman, "Universal Lossless Compression via Multilevel pattern Matching," IEEE Trans. Inform. Theory, vol.46, No.4, pp.1227-1245, July 2000.

[11] M.Li and P.M.B.Vitányi, "An Introduction to Kolmogorov Complexity and Its Application," Springer-Verlag, 1993.

[12] M.Li and P.M.B.Vitányi, "A New Approach to Formal Language Theory by Kolomogorov Complexity," SIAM J. Comput. vol.24, No.2, pp.398-410, 1995.

[13] T.Matsushima, J.Suzuki, H.Inazumi, and S.Hirasawa, "Inductive inference scheme at a finite stage of process from a view point of source coding, " IEICE Trans. Fundamentals, vol.E73-A, No.5, pp.644-652, 1990.

[14] T.Matsushima, T.Inazumi, and S.Hirasawa, "An inductive inference scheme subject to prediction, " IEEE Trans. Syst., Man, and Cybern., pp.547-554, Vol.23, No.2, 1993.

[15] C.G.Nevil-Manning, I.H.Witten, and D.L.Maulsby, "Compression by Induction Hierarchical Grammars," Proc. IEEE DCC'94, pp.244-253, Snowbird, Utah, USA, March 1994.

[16] C.G.Nevil-Manning and I.H.Witten, "Compression and Explanation Using Hierarchical Grammars," The Computer Journal, vol.40, No.2/3, pp.104-116,1997.

[17] C.G.Nevil-Manning and I.H.Witten, "Identifying Hierarchical Structure in Sequences: A Linear-Time Algorithm," J. of Artificial Intelligence Research, vol.7, pp.67-82, 1997.

[18] B.K.Natarajan, "Machine learning", Morgan Kaufmann Publishers, San Mateo, 1991.

[19] L.Pitt, "Computational limitations on learning from examples, " J. ACM, vol.35, No.4, pp.965-984, 1988.

[20] L.Valiant, "A theory of learnable, " Commun. ACM, vol.27, No.11, pp.1134-1142, 1984.

[21] V.N.Vapnik and A.Ya.Chervonenkis, "On the uniform convergence of relative frequencies of events to their probabilities, " Theory Probab. Appl., vol.16, No.2, pp.264-280, 1971.

[22] 瀧本英二，丸岡章，"PAC 学習モデルにおける学習過程と情報圧縮過程との関係,"信学論 D-1, Vol.J74, No.10, pp.702-711, 1991

付　録

1. 補題 1 の証明

事後確率の計算は概念表現一つ一つに対してベイズの定理を適用する必要はない．なぜなら仮定により事前分布族のパラメータがある制約を持つため，パラメータの各成分について事後確率の更新を行うことが可能となる．パラメータ θ_i は節 ψ_i を含む概念表現の事前確率の総和にほかならないことに注意してほしい．そこで便宜上以下の定義を用いる．

$\Lambda_i = \{r | \psi_i \in r\}$.

学習アルゴリズムが 1 個のサンプル (x, y) を受け取ったもとでの Λ_i の事後確率を $P(\Lambda_i|x, y)$ で表す．

ベイズの定理から，事後確率の計算は以下の式で求めることができる．

$$P(\Lambda_i|x, y) = \frac{\pi(\Lambda_i)P(y|\Lambda_i, x)}{\sum_{\lambda \in \{\Lambda_i, \Lambda_i^C\}} \pi(\lambda)P(y|\lambda, x)} \qquad (A\cdot1)$$

最初に $P(\Lambda_i)$ についての計算量は，$j \leqq (2n)^k$ であることか

らこの関数が n の多項式オーダで計算可能であることは明らかである．

次に $P(y|\Lambda_i, x)$ の計算量を評価するために以下の関数を定義する．

$$A(\psi_i, x) = \begin{cases} 1 : x \to \psi_i ; \\ 0 : otherwise . \end{cases} \quad (A\cdot 2)$$

この評価関数 $A(\psi_i, x)$ が n の多項式オーダで計算可能であることは，ブール関数の性質から導くことができる．そこでこの式を用いて，以下の 4 つの場合分けで考える．

(CASE1 : $A(\psi_i, x) = 0, y = 0$ の場合)

式 (A·1) の分母は x を偽とする概念表現の事前確率の総和を意味している．概念表現 r が x を真とするためには，$A(\psi_i, x) = 0$ となる節 ψ_i を 1 つも持たないことが必要十分条件であり，この条件を満たす概念表現の部分集合は $\bigcap_{t\in\{j|A(\psi_j,\ x)=0\}} \Lambda_t^C$ となる．

よって式 (A·1) の分母は

$$1 - \prod_{t\in\{j|A(\psi_j,\ x)=0\}} (1 - \theta_t) , \quad (A\cdot 3)$$

で書き換えられる．ゆえに，$j \leqq (2n)^k$ であることから，この分母が n の多項式オーダで計算可能であることが示される．次に式 (A·1) の分子であるが，Λ_i に含まれるすべての概念表現は必ず (x, y) を充足する．ゆえに $P(y|\Lambda_i, x) = 1$ であり，事後確率の計算量が n の多項式であることが示された．

(CASE2 : $A(\psi_i, x) = 0, y = 1$ の場合)

式 (A·1) の分子は Λ_i の定義より，$P(1|\Lambda_i, x) = 0$．以上のことから題意は示された．

(CASE3 : $A(\psi_i, x) = 1, y = 1$ の場合)

式 (A·1) の分母は x を真とする概念表現の事前確率の総和である．このことからこの分母が次式で表すことができるのは CASE1 から明らか．

$$\prod_{t\in\{j|x_j=0\}} (1 - P(\Lambda_t)) . \quad (A\cdot 4)$$

次に分子について考える．CASE1，CASE2 では x に対し Λ の中のすべての概念表現が同じ真理値を与えたため，扱いが容易であった．このケースでは異なる真理値を与える概念表現が Λ の中に混在することになる．そこで Λ_i の要素の中で x を真とするものは，領域 $\bigcap_{t\in\{j|A(\psi_j,\ x)=0\}} \Lambda_t^C$ に含まれなくてはならないことに注意する．それゆえ，

$$P(1|\Lambda_i, x) = \frac{P(\Lambda_i)\prod_{t\in\{j|A(\psi_j,\ x)=0\}}(1 - P(\Lambda_j))}{P(\Lambda_i)} \quad (A\cdot 5)$$

が導かれ，上式を式 (A·1) に代入することにより，

$$P(\Lambda_i|x, y) = P(\Lambda_i) . \quad (A\cdot 6)$$

(CASE4 : $x_i = 1, y = 0$ の場合)

CASE1，CASE3 から示すことができる． □

2. 補題 2 の証明

学習アルゴリズム L はサンプル S^m を得た後で，平均損失を最小にする仮説を出力する．これは以下のような式で表すことができる．

$$L(S^m, (R_{k-CNF}, \Pi_{\text{clause}}, \Omega)) = \arg \min_{\hat{r}\in R_{k-CNF}} \sum_{r\in R_{k-CNF}} P(r|S^m) l(r, \hat{r}) , \quad (A\cdot 7)$$

$$l(r, \hat{r}) = \begin{cases} 0, r = \hat{r} ; \\ 1, r \neq \hat{r} . \end{cases} \quad (A\cdot 8)$$

$\{0, 1\}$-損失の場合に平均損失最小の決定と，事後確率最大の決定は等価である．そこで次式を計算することにする．

$$L(C, S^m) = \arg \max_{\hat{r}\in R_{k-CNF}} P(r|S^m) . \quad (A\cdot 9)$$

次に概念クラス C に対し，事後確率最大の概念表現を見つけるために，次の式を定義する．

$$D_i = \begin{cases} \Lambda_i : & P(\Lambda_i|S^m) \leqq P(\Lambda_i^C|S^m) ; \\ \Lambda_i^C : & P(\Lambda_i|S^m) > P(\Lambda_i^C|S^m) . \end{cases} \quad (A\cdot 10)$$

この式は各 $\Lambda_i, i = 1, \ldots, \binom{2n}{k}$ に対し，その集合の中に真の概念が含まれる可能性とその補集合の中に含まれる可能性を事後確率で比較し，確率の高い領域を選択している．

R_{k-CNF} 全体ではこの積集合を考えればよいことになる．すなわち，

$$\hat{r} = \bigcap_{i\in\{1,\ldots,\binom{2n}{k}\}} D_i , \quad (A\cdot 11)$$

は，事後確率を最大にする領域となっており，この中に含まれる仮説の事後確率はどれも等しい[注8]．これは，積集合 H が互いに排反な事象となるため成立しており，ブール関数の構造から示される．

以上のことから事後確率最大の積集合を見つける手続きが n の多項式時間で求めることが可能なことが明らかにされた．また，積集合 $\hat{r}$ と概念表現とを対応させる手続きも n の多項式時間で可能である．

以上により題意が示された． □

(注8)：$\mathcal{P}_i$ の場合は，この積集合の要素数はすべて 1 である．