The 26th Symposium on Information Theory
and Its Applications (SITA2003)
Higashiura, Hyogo, Japan, Dec. 15-18, 2003

区間で一定なパラメータを持つ情報源におけるベイズ符号化法について
Bayes Coding for Sources with Piecewise Constant Parameters

須子 統太* Tota Suko　　松嶋 敏泰* Toshiyasu Matsushima　　平澤 茂一* Shigeichi Hirasawa

Abstract— Bayes code is one of the wellknowm universal codes. Bayes code has Bayes optimality in point of minimization of mean redundancy. Sources with piecewise constant parameters are one of nonstationary sources. These sources have abruptly changing parameters. Conventionally, nonpredictive Bayes coding for these sources is shown. In this paper, we propose a predictive Bayes coding algorithm. The computational complexity of the algorithm increases quadratically. Then, we increase the efficiency of the algorithm.

Keywords— source coding, universal coding, Bayes code, nonstationary source.

1 はじめに

情報源の確率構造について完全な情報が与えられていない場合の符号化法，すなわちユニバーサル情報源符号化法については従来から様々な研究が行われている．情報源の分布のクラスのみ仮定し，そのパラメータについては未知である場合を扱うユニバーサル符号のなかで，ベイズ符号は冗長度をベイズ基準のもとで最小にする符号である．[1]

また近年，非定常な情報源における情報源符号化の研究が行われている．その一つに，情報源が未知のパラメータに従い発生し，そのパラメータがある時点で突然変化する情報源（以下，区分定常情報源と呼ぶ）についての研究がある．[3][4][7][8]

この区分定常情報源に対し，Willems[4] が提案した符号化法は非適応的な場合のベイズ符号の一種であると考えられる．区分定常情報源におけるベイズ符号化法は計算量が系列長の指数オーダとなる問題点があり，[4] では効率的な符号化法を提案している．

本研究では区分定常情報源における適応的なベイズ符号化法について扱う．この時，非適応的な場合と同様，計算量が系列長の指数オーダとなる問題点がある．そこで，本研究では非適応的な符号化において効率的なベイズ符号化法を提案する．

* 〒169-8555 東京都新宿区大久保 3-4-1 早稲田大学理工学部経営システム工学科．Dept. of Industrial & Management Systems Engineering, School of Science and Engineering, Ookubo 3-4-1, Shinjyukuku, Tokyo, 169-8555 Japan. E-mail: suko@matsu.mgmt.waseda.ac.jp

2 区分定常情報源

情報源アルファベットを x_i，長さ N の情報源系列を $x_1^N : x_1, x_2, \cdots, x_N$，また i 番目から j 番目までの部分系列を $x_i^j : x_i, x_{i+1}, \cdots, x_{j-1}, x_j$ とする．区分定常情報源はパラメータの変化パターン m と，その変化パターンのもとでのパラメータの集合 Θ^m によって定まる確率分布で表される．

今，c 回目にパラメータの変化が起きた時点を t_c とする．但し，$t_0 = 1$ である．また，パラメータの変化パターン m は t_c を用いて $m = \{t_1, t_2, \cdots, t_{C(m)}\} \in M$ で表す．さらに，$\Theta^m = \{\theta^m_{t_0}; \theta^m_{t_1}, \cdots, \theta^m_{t_{C(m)}}\}$ とする．但し，θ_{t_c} は定常分布のパラメータとする．このとき，区分定常情報源における系列 x_1^N の発生確率を，

$$P(x_1^N|\Theta^m, m) = \prod_{c=0}^{C(m)} P(x_{t_c}^{t_{c+1}-1}|\theta^m_{t_c}), \tag{1}$$

と定義する．但し，$t_{C(m)+1} = N + 1$ である．

また情報源系列 X_1^N のエントロピーは，

$$\begin{aligned} &H(X_1^N|\Theta^m, m) \\ &= -\textstyle\sum_{x_1^N} P(x_1^N|\Theta^m, m) \log P(x_1^N|\Theta^m, m), \end{aligned} \tag{2}$$

と定義する．

3 ベイズ符号

情報源系列の符号化確率を仮定すれば算術符号を用いることで符号化が可能になる．そのためユニバーサル情報源符号化の問題は系列の符号化確率を決定する問題に帰着される．ベイズ符号ではベイズ基準のもとで冗長度を最小にする符号化確率を決定する．

以降，パラメータの変化パターン m，変化回数 C(m) およびパラメータ Θ^m を未知とし，m，Θ^m の事前分布 $P(m)$，$P(\Theta^m|m)$ は既知であるとする．

区分定常情報源における，非適応的および適応的符号化に対しベイズ符号の符号化確率を以下で示す．

3.1 非適応的ベイズ符号 [1]

系列 x_1^N の符号化確率を $AP(x_1^N)$ とする．今，損失関数を以下で定義する．

$$\begin{aligned} &V(\Theta^m, m, AP, x_1^N) \\ &= log P(x_1^N|\Theta^m, m) - log AP(x_1^N). \end{aligned} \tag{3}$$

この時，冗長度，つまり平均符号長とエントロピーの差をリスク関数として以下で定義する．

$$\begin{aligned} & R(\Theta^m, m, AP) \\ = & \sum_{x_1^N} P(x_1^N|\Theta^m, m) log \frac{P(x_1^N|\Theta^m, m)}{AP(x_1^N)} \\ = & L(X_1^N|\Theta^m, m, AP) - H(X_1^N|\Theta^m, m). \quad (4) \end{aligned}$$

本来であればこの冗長度を最小にしたいのだが，全てのパラメータについて最小化することはできない．そこでベイズ基準では，このリスク関数に対し事前分布で期待値をとったベイズリスクを最小化する．ベイズリスクは以下で定義する．

$$\begin{aligned} & BR(P(\Theta^m|m), P(m), AP) \\ & = \textstyle\sum_m \int_{\Theta^m} R(\Theta^m, m, AP) P(\Theta^m|m) d\Theta^m P(m). \quad (5) \end{aligned}$$

このベイズリスクを最小化する AP，すなわち非適応的ベイズ符号化確率は以下で与えられる．

$$\begin{aligned} & AP^*(x_1^N) \\ & = \textstyle\sum_m \int_{\Theta^m} P(x_1^N|\Theta^m, m) P(\Theta^m|m) d\Theta^m P(m). \quad (6) \end{aligned}$$

3.2 適応的ベイズ符号 [1]

系列 x_1^{n-1} が与えられたもとでの，次の 1 シンボル x_n の符号化確率を $AP(x_n|x_1^{n-1})$ とする．非適応的の場合と同様に適応的ベイズ符号化確率は以下で与えられる．

$$\begin{aligned} & AP^*(x_n|x_1^{n-1}) \\ & = \textstyle\sum_m \int_{\Theta^m} P(x_n|x_1^{n-1}, \Theta^m, m) P(\Theta^m|m, x_1^{n-1}) d\Theta^m \\ & \qquad \times P(m|x_1^{n-1}). \quad (7) \end{aligned}$$

系列 x_1^N に対して非適応的ベイズ符号の符号長と適応的ベイズ符号の符号長は等しくなることが知られている．[1]

$$-\log AP^*(x_1^N) = -\sum_{n=1}^{N} \log AP^*(x_n|x_1^{n-1}). \quad (8)$$

3.3 計算量

今，パラメータの事前分布に自然共役な事前分布を仮定する．この時，(6)，(7) 式における積分計算はデータ数の線形オーダの計算量で計算することができる．他方，m での期待値計算にかかる計算量は $O(|M|)$ となり，考えうるパラメータの変化パターンが多い場合この部分の計算量が符号化確率計算の主要項となる．非適応的な場合 $|M| = 2^{N-1}$，適応的な場合 $|M| = 2^{n-1}$ となり，系列が長くなるにつれて指数的に計算量が多くなることが分る．そのため，系列が長くなればなるほど符号化確率を計算するのが非常に困難になる．

Willems[4] が提案した符号化法は，ある事前分布を仮定したもとでの非適応的なベイズ符号とみなすことができる．[4] では CTW 法を応用することによって，効率的に符号化確率を計算するアルゴリズムを示している．そこで，本研究では適応的な場合について効率的にベイズ符号化確率を計算するアルゴリズムを示す．

4 適応的符号化法の効率化

4.1 m の事前分布

n 時点目での変化パターンの集合を M_n，その要素を $m_n \in M_n$ とする．n 時点目の符号化確率を計算した後，m_n の事後確率は，

$$P(m_n|x_1^n) = \frac{P(m_n|x_1^{n-1}) P^m(x_n|m_n, x_1^{n-1})}{AP^*(x_n|x_1^{n-1})}, \quad (9)$$

によって計算される．但し，

$$\begin{aligned} P^m(x_n|m_n, x_1^{n-1}) = & \int_{\Theta^{m_n}} P(x_n|x_1^{n-1}, \Theta^{m_n}, m_n) \\ & \times P(\Theta^{m_n}|m_n, x_1^{n-1}) d\Theta^{m_n}, \quad (10) \end{aligned}$$

である．

$n+1$ 時点目においてこの変化パターン m_n は，n 時点目までは m_n と同じ変化で，その後 $n+1$ 時点目で変化が起きる m_{n+1}^a と変化が起きない m_{n+1}^b の 2 つの変化パターンに分けられる．そこで各時点でパラメータの変化が起きる確率を π とする．但し，π は時点によらず一定で，既知であるとする．この時，$P(m_{n+1}^a|x_1^n)$ および $P(m_{n+1}^b|x_1^n)$ は m_n の事後確率を用いて以下で計算することができる．

$$\begin{aligned} P(m_{n+1}^a|x_1^n) &= \pi P(m_n|x_1^n). & (11) \\ P(m_{n+1}^b|x_1^n) &= (1-\pi) P(m_n|x_1^n). & (12) \end{aligned}$$

π は変化パターン m の事前分布を表現するパラメータとして捉えることができる．もし，変化パターンについて事前情報が全く無い場合，$\pi = 0.5$ とすることで全ての変化パターンに対して等確率な事前分布をふることになる．

4.2 効率的アルゴリズム

変化パターンが与えられたもとでのある区間におけるパラメータ $\theta_{t_c}^m$ を考える．今，$\theta_{t_c}^m$ の事前分布 $P(\theta_{t_c}^m)$ が，変化パターンおよび区間によらず全て等しいと仮定する．次に，二種類の変化パターン m' と m'' を考える．但し，$m' \neq m''$ とする．m'，m'' がそれぞれ，

$$m' = \{\cdots, i, j, \cdots\}, \quad (13)$$

$$m'' = \{\cdots, i, j, \cdots\}, \quad (14)$$

である．つまり，変化が起きていない同一の区間が存在するとする．さらに，変化パターン m が与えられたもとでの符号化確率を，

$$AP(x_n|x_1^{n-1},m) = \int_{\Theta^m} P(x_n|x_1^n,\Theta^m,m)P(\Theta^m|m)d\Theta^m \quad (15)$$

とする．この時，$i \leq n \leq j-1$ の区間において，

$$AP(x_n|x_1^{n-1},m') = AP(x_n|x_1^{n-1},m''), \quad (16)$$

が成立する．

また，時点 1 から n までの間で，最後にパラメータが変化した時点を τ_n とする．但し，$\tau_n = 1,2,\cdots,n$ である．この時，τ_n の事前分布 $P(\tau_n)$ を以下で定義する．

$$P(\tau_n) = \sum_{\{m:t_{c(m)}=\tau_n\}} P(m). \quad (17)$$

この時，効率的な適応的ベイズ符号化アルゴリズムを以下で示す．

【効率的アルゴリズム】

step-1.

x_n を読み込む．

step-2.

符号化確率を次式で計算する．

$$AP_{algo}(x_n|x^{n-1}) = \sum_{\tau_n=1}^{n} P^{\tau}(x_n|\tau_n,x_1^{n-1})P(\tau_n|x_1^{n-1}). \quad (18)$$

但し，

$$P^{\tau}(x_n|\tau_n,x_1^{n-1}) = \int_{\theta_{\tau_n}} P(x_n|x_1^{n-1},\theta_{\tau_n})P(\theta_{\tau_n}|x_1^{n-1})d\theta_{\tau_n}. \quad (19)$$

step-3.

$\tau_{n+1} = 1,2,\cdots,n+1$ について $P(\tau_{n+1}|x_1^n)$ を次式で計算する．

$\tau_{n+1} = 1,2,\cdots,n$ の時，

$$\begin{aligned} P(\tau_{n+1}|x_1^n) &= (1-\pi)P(\tau_n|x_1^n) \\ &= (1-\pi)\frac{P^{\tau}(x_n|\tau_n,x_1^{n-1})P(\tau_n|x_1^{n-1})}{AP_{algo}(x_n|x_1^{n-1})}. \end{aligned} \quad (20)$$

$\tau_{n+1} = n+1$ の時，

$$P(\tau_{n+1}|x_1^n) = \pi. \quad (21)$$

step-4.

step-1 へ戻る．

例　$P(x_{t_c}^{t_{c+1}-1}|\theta_{t_c}^m)$ が i.i.d の場合 (19) 式は，

$$P^{\tau}(x_n|\tau_n,x_1^{n-1}) = \frac{\nu(x_n|x_{\tau_n}^{n-1}) + \beta(x_t|\tau_n)}{\sum_{a=0}^{l-1}(\nu(a|x_{\tau_n}^{n-1}) + \beta(a|\tau_n))}, \quad (22)$$

によって計算される．但し，l は記号の数とし，$\nu(a|x_{\tau_n}^{n-1})$ は時点 τ_n から $n-1$ の区間において記号 a が出現した回数とする．また，$\beta(a|\tau_n)$ はベータ分布のパラメータとする．

効率的アルゴリズムにより計算された符号化確率について以下の定理が成り立つ．

定理　効率的アルゴリズムにより計算された符号化確率はベイズ基準のもとで最適な符号化確率と等しい．つまり，

$$AP_{algo}(x_n|x_1^{n-1}) = AP^*(x_n|x_1^{n-1}), \quad (23)$$

が成立する．

（証明は付録参照）

適応的ベイズ符号化の計算量は $O(2^{n-1})$ であったのに対し，効率的アルゴリズムを用いることで計算量は $O(n)$ となる．定理からも，効率的アルゴリズムがベイズ最適性を保持したまま効率的に符号化確率を計算していることが分る．

5　まとめ

本研究では区分定常情報源におけるベイズ基準のもとで最適な符号化法を示した．これには系列長が長くなるにつれて，指数的に計算量が増えてしまうという問題点があった．そこで適応的な符号化に対して，ベイズ最適性を保持したまま効率的に符号化確率を計算する方法を示した．

ベイズ符号化確率は全ての変化パターンにおける符号化確率を事後確率で重みづけることによって求められるため，全ての変化パターン数の和計算が必要であった．しかしながら，事前分布の設定によっては，符号化する記号が過去のどの時点までと同じパラメータにしたがって発生しているかを考え，その全てのバリエーションを重みづけることでベイズ符号化確率が計算可能になる．そのため，系列長だけの和計算で符号化確率が計算できる．

本研究ではパラメータが変化する確率 π を既知として扱った．しかし，本来この確率は未知であることの方が多い．そのため，π を未知とした場合の符号化法が必要となる．π を未知とした場合のベイズ符号化確率は，

$$AP^*(x_n|x_1^{n-1})$$
$$= \int_\pi \sum_m \int_{\Theta^m} P(x_n|x_1^{n-1}, \Theta^m, m) P(\Theta^m|m, x_1^{n-1}) d\Theta^m \times P(m|\pi, x_1^{n-1}) P(\pi|x_1^{n-1}) d\pi, \quad (24)$$

となる．この時，π の事前分布 $P(\pi)$ をどのように設定すればよいのか，その際に効率的なアルゴリズムが構成可能であるのか，などについては今後の課題としたい．また，区分定常情報源におけるベイズ符号の冗長度に対する理論的な評価についても今後明らかにするべき課題である．

謝辞

本研究を行うにあたり，数多くの御助言，御支援を賜りました，浮田善文氏，並びに松嶋研究室，平澤研究室の各氏に感謝致します．なお，本研究の一部は日本学術振興会科学研究費基盤 (C) 一般 (No.15560338) の援助による．

文献

[1] T. Matsushima, H. Inazumi and S. Hirasawa, "A Class of Distortionless Codes Designed by Bayes Decision Theory" IEEE Trans. Inf. Theory, vol.37, No.5, page 1288, 1991.

[2] T. Matsushima, and S. Hirasawa, "A bayes coding using context tree." In Proc. Int. Symp. on. Inf. Theory, page 386, 1994.

[3] N. Merhav, "On the minimum description length principle for sources with piecewise constant parameters." *IEEE Trans. Inf. Theory*, vol.39, No.6, page 1962, 1993.

[4] Frans M. J. Willems, "Coding for Binary Independent Piecewise-Identically-Distributed Source." IEEE Trans. Inf. Theory, vol.42, No.6, page 2210, 1996.

[5] Frans M. J. Willems, Y. M. Shtarkov and T. J. Tjalkens, "The Context-Tree Weighting Method: Basic Propreties," *IEEE Trans. Inf. Theory*, vol.41, No.3, page 653, 1995.

[6] Frans M. J. Willems, Y. M. Shtarkov and T. J. Tjalkens, "Context Weighting for General Finite-Context Sources." *IEEE Trans. Inf. Theory*, vol.42, No.5, page 1514, 1996.

[7] G. I. Shamir and N. Merhav, "Low-Complexity Sequential Lossless Coding for Piecewise-Stationary Memoryless Sources." *IEEE Trans. Inf. Theory*, vol.45, No.5, page 1498, 1999.

[8] G. I. Shamir and D. J. Costello, Jr., "Asymptotically Oprimal Low-Complexity Sequential Lossless Coding for Piecewise-Stationary Memoryless Sources-Part I :The Regular Case." *IEEE Trans. Inf. Theory*, vol.46, No.7, page 2444, 2000.

[9] M. Gotoh and S. Hirasawa, "Statistical model selection based on Bayes decision theory and its application to change detection problem." *Int. J. Production Economics* 60-61, page 629, 2000.

付録

付録では定理の証明を行う．

（証明）

まず，(9)(17)(20) 式より，

$$P(\tau_n|x_1^{n-1}) = \sum_{\{m:t_{c(m)}=\tau_n\}} P(m|x_1^{n-1}), \quad (25)$$

が成立する．また，(1) 式の定義より，

$$P^m(x_n|x_1^{n-1}, m) = P^\tau(x_n|x_1^{n-1}, \tau_n). \quad (26)$$

よって，(25)(26) 式より，

$$\begin{aligned} &AP^*(x_n|x_1^{n-1}) \\ &= \sum_m P^m(x_n|x_1^{n-1}, m) P(m|x_1^{n-1}) \\ &= \sum_m P^\tau(x_n|x_1^{n-1}, \tau_n) P(m|x_1^{n-1}) \\ &= \sum_{\tau_n=1}^{n} \{P^\tau(x_n|x_1^{n-1}, \tau_n) \sum_{\{m:t_{c(m)}=\tau_n\}} P(m|x_1^{n-1})\} \\ &= \sum_{\tau_n=1}^{n} P^\tau(x_n|x_1^{n-1}, \tau_n) P(\tau_n|x_1^{n-1}) \\ &= AP_{algo}(x_n|x_1^{n-1}). \quad (27) \end{aligned}$$

また，(11)(12)(17) 式より，(20)(21) 式の計算により $P(m_{n+1}|x_1^n)$ が計算されていることは明らか．

以上より，効率的アルゴリズムによる符号化確率がベイズ符号化確率と等しくなっていることが示せた．

□