

モンテカルロフィルタ (MCF) における ランダムサンプリング

樋口 知之

統計数理研究所

非線形・非ガウス時系列モデルを一般状態空間で表現すると、状態ベクトルの分布の推定法は、逐次的な公式で理論的には与えられている。しかしながらその実際の計算機上での実現は、状態ベクトルの次元が低いときに限られていた。この制約を取り除く目的で提案されたのがモンテカルロフィルタ (MCF) である。この MCF は、最適化の一手法である遺伝的アルゴリズム (GA) の枠組みとほぼ同じ構造をもつ。本稿ではまず、一般状態空間表現の解説を行う。次に、GA の観点からの説明を織りまぜながら MCF を紹介する。最後に、MCF で得られる尤度値の近似を改善するさまざまな方策を、サンプリング法の比較をも含めて、数値的実験結果を踏まえて考察する。

IMPROVEMENTS ON THE ESTIMATION OF THE LOG-LIKELIHOOD OBTAINED THROUGH THE MONTE CARLO FILTER

Tomoyuki Higuchi

The Institute of Statistical Mathematics

Minami-Azabu 4-6-7, Minato-ku, Tokyo 106, Japan

The algorithm with a name of "Monte Carlo Filter" (MCF) has been presented for the non-Gaussian state space model. In this study, we investigate the connections between the Genetic Algorithm (GA) and MCF. One of the objectives of this paper is to cast the GA into the Bayesian framework by its interpretation from a viewpoint of the MCF. Several ideas to improve an original MCF algorithm are shown based on a numerical experience to estimate the log-likelihood more accurately.

1 はじめに

問題対象の複雑さを柔軟に表現できるようなモデルは、一般にきわめて多数の未知変量（パラメータ）を含む形の数値的表現形式をとる。画像ピクセルの一つ一つの値そのものをパラメータとして取り扱う画像復元法^{1),2)}や、計算機トモグラフィに代表される逆問題において、探求対象を細かいグリッドに分割し、各グリッドにおける物理量を未知数として取り扱う方法^{3),4)}などがその典型的な例である。このような数多くのパラメータを持つ大規模モデルにより対象を表現し、対象の固有の知識や経験を数値的に実現する形でパラメータに対し確率分布を想定する統計的モデリングは、普通ベイズアプローチと呼ばれている^{5)~7)}。

ベイズモデルの中でパラメータ間に一次元的な特殊な構造をもつようなもの—例えば時系列—は、一般化された状態空間表現^{8),9)}の枠組みで問題解決を行うのが、計算手続きの見通しが良く様々な点に於て都合が良い^{7)~9)}。実際、状態ベクトルの次元がきわめて低い（例えば4以下）時には、確率分布を数値的に近似表現することで解決されている^{9),10)}。ただこの方法による解法は、状態ベクトルの次元に強い制約を課すため、問題対象の自在な *parametrization* という観点からは満足できなかった。

この制約を取り除く目的で提案されたものが、モンテカルロフィルタ (Monte Carlo Filter、以後 MCF と略す)¹¹⁾である。このアルゴリズムは、自然淘汰と発生のメカニズムを模倣する最適化の一手法の遺伝的アルゴリズム (Genetic Algorithm、以後 GA)^{12)~15)} とほぼ同じ構造をもつことが指摘されている¹⁶⁾。GA と対応づけることで MCF の理解は容易になるうえに、アルゴリズムに内在する本質的な問題点の認識も深まる。

本稿は、まず一般状態空間表現と MCF を紹介し、その後、対数尤度計算の誤差について、数値実験から得られた経験的事実を中心に示す。

2 一般状態空間表現

2.1 定義

一般状態空間表現は、以下の2つの確率差分方程式のセットで与えられる。

$$\begin{aligned} \text{システムモデル} \quad x_n &= g(x_{n-1}, v_n) \\ \text{観測モデル} \quad y_n &= h(x_n, w_n) \end{aligned} \quad (1)$$

ここで、 x_n は、状態ベクトルと呼ばれる、 k 次元ベクトル、また y_n は、時刻 n ($n = 1, \dots, N$) のスカラー観測値とする。 v_n は、確率分布 $q(v|\lambda_s)$ に

従う、1次元 white noise sequences である。また、 w_n は確率分布 $r(w|\lambda_o)$ に従う 1次元 white noise sequences である。 λ_s や λ_o は、 $q(\cdot)$ 及び $r(\cdot)$ を記述するための parameter vector で、ベイズ統計の枠組みでは、超パラメータ⁶⁾に相当する。例えば、 $r(\cdot)$ がガウス分布の場合、 λ_o の成分は、平均値 μ と分散 σ^2 である ($\lambda_o = [\mu, \sigma^2]^T$)。

2.2 状態ベクトルの推定

(1) で表現される時系列モデルでは、条件付き分布が以下のようなマルコフ性を持つ。

$$\begin{aligned} p(x_n|X_{n-1}, Y_{n-1}) &= p(x_n|x_{n-1}) \\ p(y_n|X_n, Y_{n-1}) &= p(y_n|x_n) \end{aligned} \quad (2)$$

ここで Y_n は、最初から時刻 n までの観測値の集合 ($Y_n \equiv \{y_1, \dots, y_n\}$)。また X_n は、最初から時刻 n までの x_n の実現値の集合 ($X_n \equiv \{x_1, \dots, x_n\}$)。ここで、 $v_n = g^{-1}(x_n, x_{n-1})$ 及び $w_n = h^{-1}(y_n, x_n)$ となる逆関数 g^{-1} と h^{-1} の存在を仮定する。そうすると、条件付き分布 $p(x_n|x_{n-1})$ や $p(y_n|x_n)$ は

$$\begin{aligned} p(x_n|x_{n-1}) &= q(v_n) \\ p(y_n|x_n) &= r(w_n) \end{aligned} \quad (3)$$

と簡単に書き直せる。

(2) と (3) の性質を用いると、予測分布 $p(x_n|Y_{n-1})$ やフィルタ分布 $p(x_n|Y_n)$ は、初期分布 $p(x_0|Y_0)$ さえ与えれば、以下の一期先予測とフィルタの手続きを逐次的に繰り返すことで得られる¹⁰⁾。一期先予測

$$p(x_n|Y_{n-1}) = \int p(x_{n-1}|Y_{n-1}) \cdot q(v_n) dx_{n-1} \quad (4)$$

フィルタ

$$\begin{aligned} p(x_n|Y_n) &= \frac{r(w_n) \cdot p(x_n|Y_{n-1})}{p(y_n|Y_{n-1})} \\ &= \frac{r(w_n) \cdot p(x_n|Y_{n-1})}{\int r(w_n) \cdot p(x_n|Y_{n-1}) dx_n} \end{aligned} \quad (5)$$

全てのデータの情報を使った x_n の分布、つまり平滑化分布 $p(x_n|Y_N)$ は、やはり下のような後ろ向きの逐次的な手続きで定められる。

平滑化

$$\begin{aligned} p(x_n|Y_N) &= p(x_n|Y_n) \int \frac{p(x_{n+1}|Y_N) p(x_{n+1}|x_n)}{p(x_{n+1}|Y_n)} dx_{n+1} \end{aligned} \quad (6)$$

2.3 モデルの選択

λ_s と λ_o を一緒にまとめ λ と記す ($\lambda \equiv [\lambda_s^T, \lambda_o^T]^T$)。最適な λ の探索は、次に与える尤度 $p(Y_N|\lambda)$ ^{8)~10)} の最大化、つまり λ の $p(Y_N|\lambda)$ に関する非線形最適化によって行う。

$$p(Y_N|\lambda) = \prod_{n=1}^N p(y_n|Y_{n-1}, \lambda) \quad (7)$$

ここで $p(y_n|Y_{n-1}, \lambda)$ は、(5) で求められたもの ((5) では、 λ を省略している) である。同様に (1) に含まれる未知のパラメータ値も、尤度最大化で決定できる。分布型を固定したままで、(7) を最大にする λ を λ^* と表すことにする。

一般には未知である $q(\cdot)$ や $r(\cdot)$ の分布型の選択は、データの説明に最適なシステム及び観測ノイズの分布型の選択という観点から、やはり尤度の最大化で行う。手続きは、多くの候補のそれぞれに対し、まず分布型を固定した状態で λ^* の探索を行い $p(Y_N|\lambda^*)$ を得る。次に各々の $p(Y_N|\lambda^*)$ を比較し、最大値をとる分布型を選ぶ。これらのさまざまな最適化をまとめて、モデルの選択と呼ぶことにする。

2.4 数値的実現

一般に条件付き分布 $p(x_n|Y_j)$ は非ガウス分布になるので、その数値的表現にはなんらかの近似が必要になる。これには最も単純な、分布を直接数値的に表現する方法¹⁰⁾、つまり、 $p(x_n|Y_j)$ を階段関数、あるいは折れ線等で近似する方法で充分である。必要とされる計算機のメモリー及び計算量は比較的大量となるが、昨今の計算機のハードの充実を考えると、データ数 N が著しく大きくないという条件のもとで、 x_n の次元が低次元 (たとえば 4 以下) であれば、ほとんど問題とならない。分布を数値的に表現すれば、(4), (5), (6) で構成される状態推定の公式は、単純な数値計算¹⁰⁾ となる。

2.5 例

(1) で記述される時系列モデルの中で簡単なモデルの一つは、次のようなシステム・観測モデルの両方が線形である線形・非ガウスモデルである。

$$\begin{aligned} \text{線形システムモデル} \quad x_n &= Fx_{n-1} + Gv_n \\ \text{線形観測モデル} \quad y_n &= Hx_n + w_n \end{aligned} \quad (8)$$

ここで、 F, G, H はそれぞれ、 $k \times k$ 、 $k \times 1$ 、及び $1 \times k$ の定数行列。もっとも単純な例は、次に示すトレンド推定に用いられるモデルである。

$$\begin{aligned} t_n &= t_{n-1} + v_n \\ y_n &= t_n + w_n \end{aligned} \quad (9)$$

このとき、 $x_n = [t_n]$ 、 $F = G = H = 1$ となる。 $q(\cdot)$ に、ガウス分布よりも裾の重い、例えばコーシー分布などを採用すると、全くの恣意的な操作無しにトレンドのジャンプが自動的に検出できる^{9),10)}。ガウス分布ではこのような鋭い変化を捉えることは不可能である。図 1 にその例を示す。示した実線は、 $n = 51$ で高さが 1 上昇するステップ関数に、平均値 0 分散 0.1 のガウスノイズを加えた模擬データである。データ数 N は $N = 100$ 。太線は $q(\cdot)$ にコーシーを、点線はガウス分布を採用した時の、各々の平滑化分布 $p(t_n|Y_N)$ の中央値である。両者とも $r(\cdot)$ はガウス分布にしてある。この時は、尤度の値が大きいコーシー分布の方が、より良い $q(\cdot)$ として選ばれる。

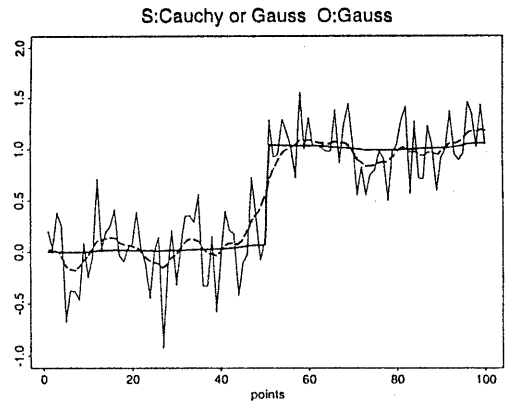


Figure 1: 平滑化の比較 — 点線：ガウス分布
太線：コーシー分布

$r(\cdot)$ に、正の方だけに裾が重い非対称な分布を用いると、大きい正値をとる異常値を自動的に除きながら平滑化できる^{9),17)}。このような単純なモデルであっても、興味深いさまざまな結果が得られる。

$q(\cdot)$ 及び $r(\cdot)$ がともにガウスの場合が線形・ガウスモデル (Dynamic Linear Model と呼ばれる)^{6),18),19)} である。この時は、すべての条件付き分布がガウス分布となるので、(4), (5), (6) で構成される状態推定の公式は、平均と共分散行列を逐次的に計算する公式、いわゆる一般に **Kalman Filter** と総称されるもの (及び Fixed interval smoother)^{20)~22)} になる。

3 モンテカルロフィルタ

先に述べたような数値的実現法が現実的であるのは、状態ベクトルの次元がきわめて低い (たかだか $k \leq 4$) の場合のみである。MCF は k が比較的大きく (20 以下) かつ、システムノイズの次元、 l が、

小さい（例えば $1 \sim 3$ ）ような一般状態空間表現の数値的実現方法として提案された。例えば、経済データの長期的な経済活動変動を捉えるために、季節に関連した変動をデータから取り除く季節調整^{8),9),18)}と呼ぶ時系列解析を状態空間表現の枠組みで行う場合である。この時、最も簡単な標準的季節調整モデル^{8),18)}でも、月毎データの場合 $k=13$ となる。一方、システムノイズの次元は $l=2$ でよい。本章では、MCF と多くのアナロジーのある GA を意識しながら¹⁶⁾、MCF を解説する。

3.1 アルゴリズム

MCF において重要なアイデアは次の二つにまとめられる。

- $p(x_n|Y_j)$ の近似を、その数多くの実現値 ($x_i^{(n)}$ ($i=1, \dots, m$) と表すことにする。 (n) はもちろん時刻を示す) で行う
- すべての時刻 n において、実現値の個数 m を固定する

つまり、 m 個の $x_i^{(n)}$ の集合 $X^{(n)} \equiv \{x_i^{(n)} | i=1, \dots, m\}$ で $p(x_n|Y_j)$ を近似する。従って、 $x_i^{(n)}$ の出現確率 $p(x_n = x_i^{(n)}|Y_j)$ は、 i によらず、すべて $1/m$ である。MCF ではこの $x_i^{(n)}$ を、*particle* (GA では string に対応) と呼ぶ。

時刻 n の一期先予測およびフィルタの手続きで定まる $x_i^{(n)}$ を、 $x_{P,i}^{(n)}$ 及び $x_{F,i}^{(n)}$ とおのおの明示的に記すと、MCF において (4) と (5) の各ステップは、以下に示す手続きによって実現される。

一期先予測:MCF

- $x_{F,i}^{(n-1)}$ に対して以下の操作を行い、 $x_{P,i}^{(n)}$ を得る

$$x_{P,i}^{(n)} = g(x_{F,i}^{(n-1)}, v_i^{(n)}) \quad (10)$$

- この操作を m 回繰り返す

フィルタ:MCF

- $x_{P,i}^{(n)}$ を、

$$\hat{r}_i^{(n)} \equiv \frac{r(w_i^{(n)})}{\sum_{i=1}^m r(w_i^{(n)})} \quad (11)$$

の確率で resampling し、 m 個の $x_{F,i}^{(n)}$ を得る

$v_i^{(n)}$ は $q(\cdot)$ から得られた m 個の乱数の実現値。また、 $w_i^{(n)} = h^{-1}(y_n, x_{P,i}^{(n)})$ である。ここで、 $g(\cdot)$ 、

$h(\cdot)$ 、 $q(\cdot)$ 、及び $r(\cdot)$ は、(1) で現れたものと同じである。

この一連の操作で、(4) と (5) がどのように達成されているのかを以下に説明する。(10) で定まる $X_P^{(n)} \equiv \{x_{P,i}^{(n)} | i=1, \dots, m\}$ は、 $X_F^{(n-1)} \equiv \{x_{F,i}^{(n-1)} | i=1, \dots, m\}$ と $\{v_i^{(n)} | i=1, \dots, m\}$ が⁸⁾、 $p(x_{n-1}|Y_{n-1})$ 及び $q(\cdot)$ からおのおの発生した独立な sequence であるから、(4) で定義される予測分布 $p(x_n|Y_{n-1})$ からの実現値と考えられる。よって、 $p(x_n = x_{P,i}^{(n)}|Y_{n-1}) = 1/m$ である。

(5) より

$$\begin{aligned} p(x_n = x_{P,i}^{(n)}|Y_n) &= \frac{r(w_i^{(n)}) \cdot p(x_n = x_{P,i}^{(n)}|Y_{n-1})}{\sum_{i=1}^m r(w_i^{(n)}) \cdot p(x_n = x_{P,i}^{(n)}|Y_{n-1})} \\ &= \frac{r(w_i^{(n)}) \cdot (1/m)}{\sum_{i=1}^m r(w_i^{(n)}) \cdot (1/m)} \\ &= \frac{r(w_i^{(n)})}{\sum_{i=1}^m r(w_i^{(n)})} \end{aligned} \quad (12)$$

が得られる。 $X_F^{(n)} \equiv \{x_{F,i}^{(n)} | i=1, \dots, m\}$ は、 Y_n を観測したもとの x_n の分布 $p(x_n|Y_n)$ からの実現値と見なせ、 $p(x_n = x_{F,i}^{(n)}|Y_n) = 1/m$ となる。

フィルタで定義される各時点での尤度 $p(y_n|Y_{n-1})$ は、MCF においては (12) より

$$p(y_n|Y_{n-1}) = \frac{1}{m} \sum_{i=1}^m r(w_i^{(n)}) \quad (13)$$

で与えられる。

MCF の枠組みでの smoothing には、*Storing the State Vector* の方法と、*Two-filter Formula* による方法が提案されている¹¹⁾。両者とも一長一短あり^{11),16)}、問題点を解決する方法が模索されている。

3.2 遺伝的アルゴリズムとの関係

フィルタが GA の淘汰の手続きに完全に対応しているのは、(11, 12) の $r(w_i^{(n)})/m$ を GA での *evaluation function* と見なせば自明である。これにより、(13) で定義される各時点の尤度 $p(y_n|Y_{n-1})$ は、第 n 世代の *population fitness* と解釈できる。ただし MCF での *evaluation function* は、 $r(\cdot)$ 自体は不変だが、与えられるデータ y_n のために、 n とともに変化する。フィルタで行っている操作を GA の観点から説明すると、 $x_{P,i}^{(n)}$ を、与えられた環境 y_n への各 *particle* の *fitness* に応じて淘汰させ、淘汰後の集団 $Z_F^{(n)}$ を創り出す手続きとなる。

各時点での尤度の GA 的解釈に従い、(7) で与えられる尤度 $L(Y_N)$ は、次のように書ける。

$$\begin{aligned} p(Y_N) &= \prod_{n=1}^N p(y_n|Y_{n-1}) \\ &= \prod_{n=1}^N (\text{第 } n \text{ 世代の population fitness}) \\ &= \text{一族 (family) の fitness} \quad (14) \end{aligned}$$

よって Y_N を環境の歴史という言葉に言い換えると、最尤法とは、与えられた環境のもとで、family fitness を最大にするようなモデルの選択と解釈できる。逆に GA で ad-hoc に与えるパラメーター Crossover や Mutation の起こる確率を尤度の観点から考察するのも興味深い。

フィルタで淘汰のプロセスを被る $x_{P,i}^{(n)}$ は、 $x_{F,i}^{(n-1)}$ から生み出されていることから、 $x_{F,i}^{(n-1)}$ を parent、 $x_{P,i}^{(n)}$ を子孫と呼ぶことは自然である。従って予測の操作の内容は、 $X_F^{(n-1)}$ から、 $X_P^{(n)}$ への世代交代である。これらの観点から MCF は、GA と総称される population based search アルゴリズムのなかで、Canonical GA と呼ばれるもの¹⁴⁾、あるいは Simple GA¹³⁾ に近い。

Crossover と Mutation の一連の手続きを意味的に捉えれば、parent から次の世代への世代交代であるから、予測は、Crossover と Mutation によってなされる操作に対応している。Crossover と Mutation で加えられる randomness は、世代交代時の情報伝達の際に生じる、情報の揺らぎ (あるいは乱れ) に相当する。MCF では、予測で加えられるシステムノイズがその役割を担っている。表 1 に対応のつくアナロジーをまとめた。

表 1. MCF と GA のアナロジー

MCF	GA
$x_{P,i}^{(n)}$: particle	string
(n): time	generation
$g(\cdot, v_n = 0)$	genetic drift
system noise	crossover mutation
filtering	selection
$p(y_n, x_{P,i}^{(n)} Y_{n-1}, \lambda)$	evaluation function
$\hat{r}_i^{(n)}/m$	fitness
$p(y_n Y_{n-1}, \lambda)$	population fitness
$p(Y_N \lambda)$	family fitness
y_n : observation	environment
Y_N : data	history of environment

3.3 Resampling

3.1 で説明した MCF の計算機上の実行手続きで重要な点は、フィルタ (11) における resampling 手法の選択がある。実際、フィルタに対応する GA の淘汰の仕方は数多く存在し、いくつかの具体的な最適化問題への応用実験をもとにその優劣が議論されている^{13),14),23)}。ただ、“広義の淘汰法の改良”に含まれる、最適化効率を上げる目的のために導入される evaluation function の ad-hoc な変形、たとえば scaling¹³⁾ 等は、MCF では許されない。なぜならば、MCF の evaluation function は、はっきりとした意味をもつ変形できないものだからである。従ってここで述べる淘汰法の改良は、純粹に selection scheme に関連したものだけである。

明確な淘汰法の類別法¹³⁾はないので、ここでは代表的な次の 3 つを MCF に適用してみた。

1. Deterministic sampling
2. Stochastic sampling with replacement
3. Remainder stochastic sampling
 - (a) original
 - (b) Stochastic universal sampling

以上を簡単に MCF の表記法に従い説明する。その前に、fitness $\hat{r}_i^{(n)}$ を $1/m \left((1/m) \cdot \sum_{i=1}^m \hat{r}_i^{(n)} \right) = 1/m$ に注意) で割った値の整数部分を $R_i^{(n)}$ で、また少数部分を $e_i^{(n)}$ で表記しておく。

1. は、まず i 番目の particle を各々 $R_i^{(n)}$ 個ずつ確保する。次に $e_i^{(n)}$ を大きい順に $m' \equiv m - \sum_{i=1}^m R_i^{(n)}$ 個並べ、対応する $x_{P,i}^{(n)}$ を一つずつとっていく。2. は普通ルーレット法 (Roulette wheel selection) と呼ばれ、標準的な GA^{13),15)} で普通採用されているものである。比喩的に説明すると、 i 番目の円弧の長さ (つまり、スロットサイズ) が $\hat{r}_i^{(n)}$ になるようなルーレット盤をつくり¹³⁾、毎回ルーレットを回して particle を m 個選択する。3.(a) は、1. 同様まず i 番目の particle を各々 $R_i^{(n)}$ 個ずつ確保する。次に、今度はスロットサイズが $e_i^{(n)}$ のルーレット盤をつくり、ルーレットを回して m' 個の particle を選ぶ。この手続きよりは、これと等価な働きをする 3.(b)^{14),24)} が便利である。用意するルーレット盤は、2. と同じであるが、場所を指す矢印が一つでなく m 個、円周に等間隔にある。この円盤を 1 回だけまわせばよい。

2. は乱数を m 回、また 3.(a) は m' 回発生させる必要があるが、3.(b) は 1 回だけでよい。1. は、 $e_i^{(n)}$ を大きい順に並べかえる必要があり、 $O(m \log m)$ の計算量を要する。

4 尤度の推定

GA では、最適化というある意味で単一な評価基準で改良法を考察すればよいが、MCF ではさまざまな評価基準が考えられる。ここでは、モデル選択を完全に支配する対数尤度の値に注目する。MCF で得られる尤度の値は近似値であり、良い推定値を得るための改良策を数値実験の結果をもとに提案する。題材とする問題は (9) のトレンド推定で、MCF を適用するデータは、図 1 に示したものである。

4.1 Resampling 法の比較

まず尤度が Kalman filter^{(8), (18)} で正確に計算できる $q(\cdot) r(\cdot)$ が共にガウス分布 (図 1 の点線) の場合の結果を示す。MCF は乱数を用いているので、毎回得られる対数尤度の値は一定ではない。各試行毎に得られる対数尤度の値を l_s^{MCF} ($s = 1, \dots$) と記すことにする。ここでは $m = 1000$ に固定して、5000 回の試行実験を行った。用いた擬似乱数は M 系列にとづくもの⁽²⁵⁾ である。 l_s^{MCF} の平均値を $\langle l_s^{\text{MCF}} \rangle$ 、標準偏差を $\text{s.d. } l_s^{\text{MCF}}$ と、また Kalman filter で計算した対数尤度の値を l^{KAL} と表す。

表 2 に、3 つの resampling 法の比較結果を示す。表に記したバイアスとは、 $\langle l_s^{\text{MCF}} \rangle - l^{\text{KAL}}$ で定義される量である。いずれの方法も、 m が有限である影響でバイアスが存在することがわかる。Deterministic sampling は、 $\hat{r}_i^{(n)}$ の小さい、つまり分布の裾にある particle を消滅させる効果があるので、 $X_F^{(n)}$ の広がりや他の resampling と比較して常に小さくなる。その結果、 $X_P^{(n+1)}$ の広がりも縮小しており、各時点の尤度の値が真値より系統的に小さく、バイアスが大きいと考えられる。ルーレット法の標準偏差が大きいのは、 m 回乱数を発生させて $X_P^{(n)}$ を得るため、フィルタ分布近似の試行毎の揺らぎが大きいことによる。その点、Stochastic Universal sampling は、 $X_F^{(n)}$ を求めるのに 1 回しか乱数を用いていないので標準偏差が小さい。バイアスも小さく、MCF の resampling には最も適している。

表 2. resampling 法の比較: 5000 回実験

Sampling Scheme	バイアス	s.d. l_s^{MCF}
Deterministic	-0.344	0.392
Roulette wheel	-0.127	0.496
stochastic universal	-0.0952	0.455

4.2 その他の改良法

4.2.1 バイアス

尤度推定のバイアスを減らすためには、 $p(x_n|Y_{n-1})$ の表現を精緻に行なえばよい。それにはいくつかの

方法が考えられる。

1. particle 数 m を十分に増やす
2. dummy particle を導入する
その利用法

(a) $\log p(y_n|Y_{n-1})$ の値を求めるだけ

(b) $p(x_n|Y_n)$ の表現に情報を還元する

4.1 節で固定した m を変化させ、各々 100 回 l_s^{MCF} を計算する。図 2 に、その分布を平均値 (\cdot で示す)、下側四分位数、中央値、上側四分位数の形式で示す。 m の増加に従い分布の近似が高まり、その結果バイアスが減少している。図 3 に l_s^{MCF} の分散を各 m に対して示す。分散の m への依存性は、図中の点に最小 2 乗法によって当てはめた直線の傾きより、 $\sim 1/m^{0.86}$ 程度である。

0.25, median, 0.75 quantiles and average (100 trials)

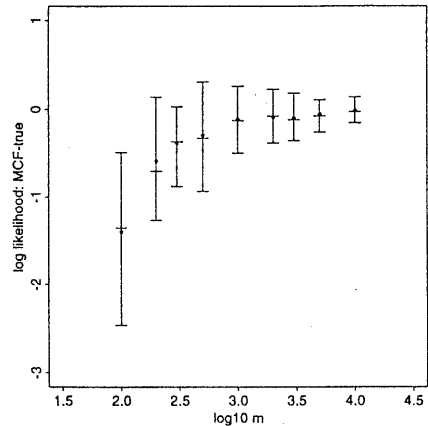


Figure 2: 対数尤度の m への依存性

2. の方法は、一期先予測の手続きで一時的に m を α 倍増やし、 $X_P^{(n)}$ の構成数を αm 個にする方法である。具体的には、(10) で、 $x_{F,i}^{(n-1)}$ を固定したまま $v_i^{(n)}$ を α 個生成し、異なる $x_{P,i}^{(n)}$ を α 個得る。このようにして構成した $X_P^{(n)}$ を、 $X_{P[\alpha]}^{(n)}$ と記すことにする。

2.(a) は、フィルタの手続きで、尤度の計算にのみ $X_{P[\alpha]}^{(n)}$ を使い、resampling には $X_P^{(n)}$ を用いる方法である。2.(b) は、resampling にも $X_{P[\alpha]}^{(n)}$ を使う積極的な手法である。もちろん、resampling する個数は m 個であり、 $X_F^{(n)}$ の構成数は m 個である。この方策は、生命力の弱い種が、その存続のために一個体から多数 (α 個) の子供 (卵) を産むしくみに対応する。

4.2.2 分散

smoothness prior の設定を行った時の一期先予測の本質は、分布の形をばやけさせるといったようなものである。分散を減らすためには確かに上述 1., 2. の方法で可能であるが、非常に多くの計算量を要し、予測の手続きの簡素化という当初の MCF の意図に反する。

分散を減らす簡単なアイデアは、いくつかの I_s^{MCF} の平均値でもって、対数尤度の推定値を定義するやり方である。 N_p 個の I_s^{MCF} の平均値を $I_{s'}^{N_p}$ ($s' = 1, \dots$) と表す。図 4 に、いくつかの N_p に対する $I_{s'}^{N_p}$ の分散を示す。期待どおり、分散は $1/N_p$ となっている（直線の傾きは -1.01 ）。

この方法は、並列プロセス数を N_p とする実効的な並列化が可能なおえ、状態ベクトルの推定に各プロセスの情報を利用でき、現在のところ一番有望である。

4.3 改良法にかんするまとめ

この章に示された計算結果から、尤度推定においてバイアスと分散を減らすための処方箋をまとめてみる。まず、resampling 法は、Stochastic universal resampling を用いるべきである。また、particle 数 m 、dummy 効果 α 、及び平均操作数 N_p を同時に増やすことは不可能なので、応用問題に即した適切な (m, α, N_p) の選択が求められる。たとえば図 1 の問題は、図 2 に示したようにバイアスは $10^3 \leq m$ になるとあまり改善されないので、 m をきわめて大きくとるよりは、計算効率良く分散を減らすために N_p を増大した方が得策である等である。ここでは割愛するが、平滑化をコーシー分布で行った時の尤度推定には、かなり大きい m ($\sim 10^4$ 程度) を

Particle number dependence: Variance (100 trials)

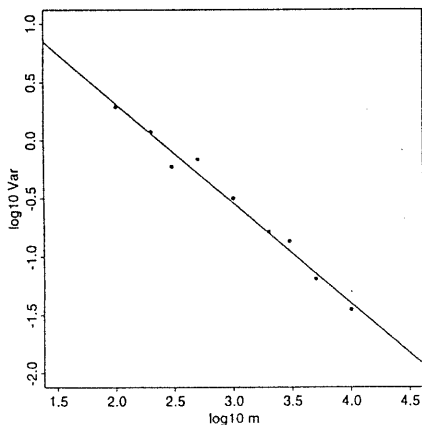


Figure 3: 対数尤度の m への依存性：分散

必要とする。

ここで触れられなかった改良法としては、 $v_i^{(n)}$ の生成に Stochastic universal sampling と同じ考え方を取り入れるものがある。実際、有効な分散の減少が確認されている。また最後になったが、この章で計算した尤度はすべて $\lambda = \lambda^*$ のものである。それ以外の λ の時の尤度の計算結果は、発表時に示す予定である。

謝辞 有益なご討論をして頂いた統計数理研究所・北川教授に深く感謝致します。

参考文献

- 1) Geman, S. and Geman, D. : Stochastic relaxation, gibbs distributions, and the Bayesian restoration of images, *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 6, pp. 721-741 (1984).
- 2) Besag, J. : Towards Bayesian image analysis, *J. Applied Statistics*, Vol. 16, No. 3, pp. 395-407 (1989).
- 3) 田辺國士、田中輝雄：ベイズモデルによる曲線・曲面のあてはめ、月刊 地球、Vol. 5, No. 3, pp. 179-186 (1983).
- 4) Inoue, H., Fukao, Y., Tanabe, K. and Ogata, Y. : Whole mantle P-wave travel time tomography, *Physics of the Earth and Planetary Interiors*, Vol. 59, pp. 294-328 (1990).
- 5) Lindley, D. V. and Smith, A.F.M. : Bayes estimates of the linear model (with discussion), *J. R. Statist. Soc. B*, Vol. 34, pp. 1-41 (1972).
- 6) Akaike, H. : Likelihood and the Bayes procedure (with discussion), in *Bayesian Statistics*, edited by

Np dependence

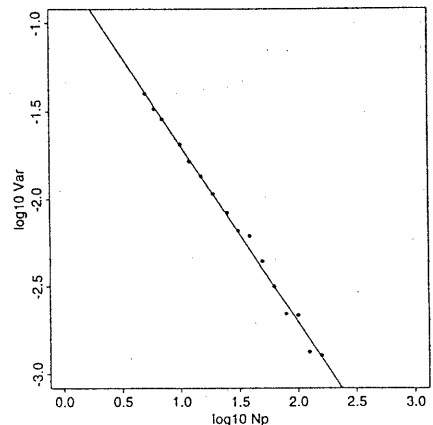


Figure 4: 対数尤度の N_p への依存性

- J. M. Bernardo, M. H. De Groot, D. V. Lindley, and A. F. M. Smith, University Press, Valencia, Spain, pp. 143-166 (1980).
- 7) 伊庭 幸人 : ベイズ統計と統計物理, *Inst. Statist. Math. Res. Memo.*, No. 492, (1993).
- 8) 北川 源四郎 : 時系列解析プログラミング, 岩波書店、東京 (1993).
- 9) 北川 源四郎 : 非ガウス型時系列モデリング、オペレーションズ・リサーチ, Vol. 34, No.10, pp. 541-546 (1989).
- 10) Kitagawa, G. : Non-Gaussian state space modeling of nonstationary time series (with discussion), *J. Am. Stat. Assoc.*, Vol. 79, pp. 1032-1063 (1987).
- 11) Kitagawa, G. : Monte Carlo Filtering and smoothing method for non-Gaussian Nonlinear state space models, *Inst. Statist. Math. Res. Memo.*, No. 462, (1993).
- 12) Holland, J.H. : *Adoption in natural and artificial systems*, Ann Arbor: The University of Michigan Press (1975).
- 13) Goldberg, D.E. : *Genetic Algorithm in Search, Optimization and Machine Learning*, Addison-Wesley Publishing Company, Massachusetts (1989).
- 14) Whitley, D. : A genetic algorithm tutorial, *Statistics and Computing*, Vol. 4, pp. 63-85 (1994).
- 15) デービス・L 編: 遺伝アルゴリズムハンドブック、嘉数侑昇 他共訳、森北出版、東京 (1994).
- 16) 樋口 知之 : 遺伝的アルゴリズムとモンテカルロフィルター, *Inst. Statist. Math. Res. Memo.*, No. 489, (1993).
- 17) 樋口知之 : 大規模ベイズモデルに基づくスピンノイズの除去法、統計数理, Vol. 41, No. 2, pp. 115-130 (1993).
- 18) Gersch, W. and Kitagawa, G. : Smoothness priors in time series, in *Bayesian Analysis of Time Series and Dynamic Models*, edited by J.C. Spall, Marcel Dekker, INC., New York and Basel, pp. 431-476 (1988).
- 19) Harrison, J. and West, M. : *Bayesian forecasting and dynamic model*, Springer, New York, (1989).
- 20) Kalman, R.E. : A new approach to linear filtering and prediction problems, *Trans. ASME, J. Basic Eng.*, Vol. 82D, pp. 35-45 (1960).
- 21) Anderson, B. D. O. and Moore, J.B., *Optimal Filtering*, Prentice-Hall, Englewood Cliffs, N.J., (1979).
- 22) 片山 徹 : 応用カルマンフィルタ, 朝倉書店、東京 (1983).
- 23) Brindle, A. : *Genetic algorithm for function optimization*, Unpublished doctoral dissertation, University of Alberta, Edmonton (1981).
- 24) Baker, J. : Reducing bias and inefficiency in the selection algorithm, In *Proceedings of the Second International Conference on Genetic Algorithms and Their Applications*, ed. J. Grefenstette, Lawrence Erlbaum (1987).
- 25) 伏見 正則、手塚 集 : 多次元分布が一様な疑似乱数列の生成法、応用統計学, Vol. 10, No. 3, pp. 151-163 (1981).