粒子フィルタとプライシングによる時系列区分化の 手法を用いた非線形時系列生成モデルの推定

時永, 祥三 九州大学名誉教授

池田,欽一 北九州市立大学経済学部 : 准教授

https://doi.org/10.15017/1515767

出版情報:經濟學研究. 81 (4), pp. 99-122, 2014-12-26. 九州大学経済学会

バージョン: 権利関係:

粒子フィルタとプライシングによる時系列区分化の 手法を用いた非線形時系列生成モデルの推定

時 永 祥 三 池 田 欽 一

1 まえがき

観測された時系列に基づいて、時系列を生成するパラメータの時間変化を推定する場合に、時系列を区分化をする手法を用いる研究が示されている [1][2]. また、線形システムのモデルにおいても、記述するパラメータの時間的な変化を推定する手法が示されている [3][4][5][6]. しかしながら、非線形システムとして記述されるモデルから生成された時系列 (以下では時系列生成モデルと呼ぶことにする) に対して、これらの方法を直接的に用いることはできないため、別の方法による時系列区分化と生成モデル変化の推定手法を用いる必要がある。これまでの研究で、統計的手法として、観測された時系列データに対する非線形状態方程式により得られる尤度を用いて、システムの状態方程式の変化を抽出する方法が示されているが、ある時点でのシステムの変化だけを議論している限界がある [7][8][9][10]. 非線形時系列生成モデルの時間変化推定には、これまで多くの分野で適用されてきた、粒子フィルタ (Particle Filter:PF) 手法が有効である [9][10][11][12][13][14][15][16]. 本論文では PFと時系列区分化の手法を用いた非線形時系列生成モデルの時間変化の推定について述べる.

本論文ではまず、観測される時系列が非線形状態方程式により生成され、その観測データから PFにより状態を推定する問題を定義し、尤度を用いて、いくつかの候補となる非線形システム記述の、どれが最適かを判断する方法を適用する。この場合、区分化を行うための時間軸は、プライシングと呼ばれる、ある離散系カオス変動により特徴付けられることを仮定として用いている。すなわち非線形システムが時間的に変化する背景にある、決定論にしたがうメカニズムが存在すると仮定する [17][18][19][20][21][22]. これにより、ランダム探索を行う場合と比較して効率が向上することを示す。このような時系列の区分化の方法は、まとまったデータに対してオフラインで行うに適した方法であるが、一方では瞬間的に発生するシステムの変化抽出には適さない。したがって、次に、非線形モデルの時間変化をオンラインで行う1つの方法として、変化後の関数形状を遺伝的プログラミング(Genetic Programming:GP)の手法により推定する方法を提案する [23][24][25][26][27][28][29]. 具体的には、システムの変化は緩やかであり、その関数形状も現在の形状に変形したものであると仮定し、現在の関数に GPによる遺伝的操作をほどこしたものから、適合度が最大となる関数を推定する方法を適用し、システム変化の抽出と、その関数形状の推定を同時に行う方法を提案する。応用

例として人工的なデータを用いて、本論文の手法の有効性を確認するとともに、現実の時系列データに適用して、モデル変化の後の状態方程式推定を議論する.

以下 2. では、について示す.3. ではを議論する. 4. では応用例を示す.

2 非線形時系列生成モデルの時間変化の推定

2.1 時系列牛成モデルの時間変化と問題解明

ここでは、本論文で検証している事項が、問題解明にどのように有効であるかを簡潔に述べておく.

(1) 非線形時系列生成モデルの時間変化解明の意義

これまでの研究において、観測される時系列の特性を記述するパラメータが単純に変化する場合や、時間的に変化する線形システムに基づいて時系列が生成されるケースの分析は行われている[1][2][3][4][5][6]. しかしながら社会生活において、比較的頻繁に見出される事例として存在する、時系列が非線形システムから生成される一般的な場合については、議論されていない。時系列生成モデルの時間変化を推定することは、異常検出のほかに、状況に応じてスケジュールを変更するなどの計画立案にとり重要であり、非線形時系列生成モデルの時間変化の問題を解明することは意義がある。

(2) 時系列生成モデルの時間変化の決定論による明確化

これまでの研究においては、時系列生成モデルの時間変化については特に定義されていないため、変化の発生はランダムであり、その規則性を議論することはなされていない [1][2]. したがって、モデルの時間変化の発生時刻の推定は、いわゆるランダム探索により行われている. しかしながら、モデル変化をイベント発生としてとらえた場合に、その発生の仕組みを同時に推定することは、イベント生起のメカニズムの推定や、将来における発生予測などに有意義である. 本論文では、モデルの時間変化が、背景にあるメカニズムにより発生しているという仮定を用いている [17][18][19][20][21][22]. その結果、発生の特徴の分析や予測に役立つことが期待されると同時に、決定論的な仕組みの記述を行うことにより、計算が容易となる利点がある.

(3) PF による時系列生成モデル変化推定

非線形モデルにより時系列が生成される場合,この関数を推定するだけであれば,簡単な方法は存在する.本論文では,ダイナミックスを導入することにより時系列生成モデル推定を一般化している.また,時系列が生成されるシステムを問題とするケースも少なくない.特に, PF の手法を適用することにより,尤度を定義し,より明確に時系列生成モデルの時間変化を推定することができる.

2.2 非線形時系列生成モデルと時間変化

以下ではまず非線形状態方程式により生成される時系列モデルを記述しておく [9][10][15]. 本論文では時系列生成を記述するモデルは 1 つではなく, 時間的に変化することを仮定している. 時系列を記述する時間は同じ長さ 1 の区間に分割されており, また, いくつかの区間にわたり同じ時系列生成モデルが適用され, j 番目のセグメントを s_j と呼んでおく.1 つのセグメントでは 1 つの非線形時系列生成モデルが適用 (選択) され, 選択されるモデルを区別する記号 (クラス) を z_k , k=1,2,...,K としておく (総数が K 個).

いま1つのセグメントにおける非線形時系列生成モデルは、次のように記述できる(適用される モデルのクラスを zk としておく).

$$x(t+1) = F[x(t)]^{(z_k)} + u(t)^{(z_k)},$$

$$x(t) = [x_1(t), x_2(t), ..., x_m(t)],$$

$$u(t) = [u_1(t), u_2(t), ..., u_m(t)].$$
(1)

$$y(t) = G[x(t)]^{(z_k)} + v(t)^{(z_k)}.$$
(2)

ここで,変数 y(t) は時刻 t における観測データを意味するスカラー (1 次元ベクトル) であり,変数 x(t) は推定されるべき m 次元の状態変数ベクトルである. また $u(t)^{(z_k)}, v(t)$ は m 次元, 1 次元のシ ステムノイズである. $F[.]^{(z_i)}, G(.)^{(z_k)}$ は、変数 x(t) についての関数である. ここで. $G(.)^{(z_k)}$ は既知 の非線形関数であり、関数のセットから選択されると仮定する.

PF による状態推定とモデルの尤度 2.3

いま, $F(.)^{(z_k)}$, $G(.)^{(z_k)}$ が行列を用いた線形表現の形式で与えられ (x(t+1) = Ax(t) + u(t), y(t) =Bx(t) + v(t) の形式, 行列 A, B は時間 t のみの関数), u(t), v(t) が正規乱数に従う場合には, 従来か らの手法である Kalman Filter(KF) を適用して、状態を推定することができる. しかし、状態方程式が 非線形の場合には、KF が適用できないため PF が援用され、有効性が示されている. PF 適用の概 要は、以下のようにまとめられる. 話を簡単にするため、選択された関数のクラス zk の表示を省略 する.

いま、観測データとして時系列 y(t) が与えられた場合に、式 (1)(2) に示すモデルを仮定して、 $x(t) = [x_1(t), x_2(t), ..., x_m(t)]$ を推定する. 記号として, $x_i(t), i = 1, 2, ..., m$ の j 番目の粒子を $x_i^{(j)}(t)$ としておく.

(1) ステップ 1:粒子の初期値の生成

時刻 0 における変数 $x_i^{(j)}(t)$ の初期値 $x_i^{(j)}(0), i=1,2,...,m$ について, あらかじめ定めた確率分布 に従がう乱数により Np セットの粒子の集合として生成する.

(2) ステップ 2:システムノイズの生成

それぞれの粒子について, 変数に対する粒子 $x_i^{(j)}(0)$ と同様に, システムノイズである $u_i^{(j)}(0)$ お よび $v^{(j)}(0)$ について、あらかじめ与えた確率分布に従った乱数によって N_P セットの粒子として生 成する.

(3) ステップ $3:x_{k,t}^{(j)}(t)$ の 1 時刻先予測

 $x_i^{(j)}(t)$ の 1 時刻先予測が、次により得られる.

$$x^{(j)}(t+1) = F[x^{(j)}(t)] + v^{(j)}(t).$$
(3)

ただし上の式はベクトル $x^{(j)}(t+1)$ のそれぞれの要素 $x_i^{(j)}(t+1)$ について計算することを意味する.

(4) ステップ 4:重みの計算

次に、それぞれ粒子 j の重み $w_t^{(j)}$ を計算する.

$$w_t^{(j)} = R_t(y(t+1)|x^{(j)}(t+1)). (4)$$

ここで $R_t(.)$ は $x^{(j)}(t+1)$ が与えられた場合の y(t+1) の条件付分布である. これらの重みを, 集計しておく.

$$W_t = \sum_{j=1}^{N_P} w_t^{(j)}. (5)$$

(5) ステップ 5:粒子のリサンプリング

それぞれの粒子を, 重みに従って, リサンプリングする. すなわち i 番目の状態変数については, 粒子の集合 $X_{t+1}=(x_i^{(1)}(t+1),x_i^{(2)}(t+1),...,x_i^{(N_P)}(t+1))$ から粒子を $w_t^{(j)}/W_t$ の確率に従って復元抽出して, 生成する. この結果として, 変数 $x_i(t)$ の確率分布を推定することができる. 時刻 t における状態の推定値は, 集合 X_t の平均として与えられる.

(6) ステップ 6:時刻 t = 2, 3, ..., T について繰り返し

ステップ 2~ステップ 6 の操作を, 時刻 t=2,3,...,T について繰り返す. この結果として, 時刻 t=2,3,...,T についても状態変数 $x_i(t)$ の確率分布を推定することができる. 最終的に得られるモデルの対数尤度は, 次により与えられる.

$$L_m = \sum_{t=1}^{T} l_t - T \log N_P, l_t = \log W_t.$$
 (6)

これまでの議論で述べたように、非線形時系列生成モデルが変化する時刻がセグメントにより正しく表現されていれば、時系列全体に対して計算される尤度は最大になる。しかしながら、一般にはランダムに与えたセグメント構成は、このような最適な分割ではないので、この分割を前提として推定された尤度は、最大にはならない。このように観測される時系列から、その時刻に最適と思われる関数モデルとこれを継続した場合の区間セグメントを推定する場合には、これらの2つを同時的に進める必要がある。

3 時系列区分化と非線形時系列牛成モデル変化

3.1 時系列生成モデル変化をプライシングで表現する利点

本論文で提案する、PFと時系列区分化による時系列モデル変化の分析について議論を進めるまえに、特徴(意義)について、簡単にまとめておく、まず時系列モデルの変化が周期的である場合には、変化事象の推定そのものが無意味であり、一方では、反対に変化がまったくランダムである場合には、ランダム探索以外に適用方法がない問題がある。本論文で提案するプライシングを用いる時系列区分化の利点と特徴については、(1)プライシングが説明する間歇的現象、(2)計算の容易さ、(3)現実データへの適用可能性の3点にまとめることができる。まず第1点目については、プライシングは待ち行列へのフロー到着のような「事象の持続・連続の蓄積にともなう放出」を表現するものであり、関数やパラメータ変化など、間歇的な現象を表現する1つの方法論として有効であると言える。しかも、原理そのものが一般的には理解されやすく比較的簡単なアルゴリズムとなっている利点がある。例えば、確定的モデルではなく、ランダム変数が含まれるダイナミックスで時間変化を記述する場合には、関数の推定誤差とランダム変数のあいまいさが、相互に判別できない結果

が発生する.また本論文で用いるカオス変動モデルは振動や不規則な現象を表現する手段としては、 確率的なモデルに劣らない.

第2点目の計算の容易さについては、モデル変化の背景にプライシングが存在するとする仮定した確定的なモデルを用いることにより、推定が相対的に容易となり、推定結果の解釈が明確となる利点がある. 比較として、文献 [2] において提案されている方法論を検討すると、区分化の境界をランダム探索により行うため計算時間が長くなること、得られた結果は規則性を含まない偶然の帰結であるとしか言えないなど、の問題がある.

第3番目の現実データへの適用可能性に関しては, 時系列モデルの変化を検出した後で, 分析結果を更に活用できる点がある. 結果を解釈することにより変化が発生する理由の推定, その性質の分析や将来の発生予測なども可能となる.

3.2 プライシングを用いた時系列区分化手法

本論文では、個別の時系列モデルの時間変化を、離散系カオス変動を用いてモデル化する。具体的には、施設などへの客の到着と退去をモデル化する、いわゆるプライシング(pricing)と呼ばれる現象のモデルである。このモデルに関する記述はすでに多く存在するので、ここではプライシングの基本モデルを簡単にまとめておく。なお、ここに示す方法は時系列モデルの時間変化を記述する1つの方法であり、更に実際に発生している現象を外部から指標化する方法であり、限界が存在することを明示しておく。

サービスを処理する窓口をノードと考え, ノードでのサービスの処理能力 (容量) を μ としておく [17][18][19][20][21][22][23][24]. ノードに流入するフロー $\lambda(t)$ は, ノードでの待ち時間や処理コストの予測値 $\pi(t)$ (pricing:プライシングと呼ばれる) に関連して調整される. フロー $\lambda(t)$ はノードに対してすべてが供給されるものではなく, この部分が供給されるモデルとなる. $\lambda(t)$ は, 次の式により与えられる (a>d).

$$\lambda(t) = \begin{cases} 1, & (0 \le \pi(t) \le d), \\ (a - \pi(t))/(a - d), & (d < \pi(t) < a), \\ 0, & (\pi(t) \ge a). \end{cases}$$
 (7)

プライシングの値は、指数平滑和により与えられる $(0 < \omega < 1)$.

$$\pi(t+1) = \omega \pi(t) + (1-\omega)p(t), p(t) = Q_{\mu}(\lambda(t)).$$
 (8)

$$Q_{\mu}(\lambda(t)) = \frac{1}{(1 - \lambda(t)/\mu)\mu}.$$
(9)

なお、式 (5) において λ が μ より大きい場合は、待ち時間がマイナスになり、理論的にはありえないが、シミュレーションにおいては、過剰な入力がある場合に相当する。このケースでは、無限 (計算上では極めて大きな値で代替する) の待ち時間を与えることで、入力を拒否された現象を表現する.

関数 $Q_{\mu}(\lambda(t))$ は、いわゆる待ちシステムにおいて入力フローが到着率 λ のポアソン到着、処理時間の分布が指数分布である場合の平均待ち時間である.

図1には、プライシングとして生成されるカオス時系列の例を示している。図1に示すように、時 系列は間歇的に大きな値をとり、この時点で時系列モデルの変化が発生したとみなす。また分岐図

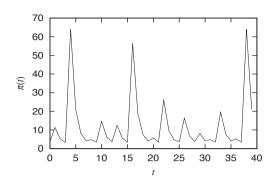


図 1: プライシング時系列の例

は省略するが, パラメータが $0.2 < \mu < 0.9$ となる範囲では変数 $\pi(t)$ はカオス変動にしたがった挙動となる. したがって μ の値を調整することにより, 間歇的に大きな値をとる時刻 (ジャンプの発生時刻) の間隔を調整することができる.

なお図 1 に示すように、プライシング時系列が大きな値をとる間隔は $10\sim 20$ であり、これをそのまま非線形時系列の生成における変化発生の時間間隔に用いると、時系列変化が激しくなり、推定が難しいケースが生じる.そのため本論文では、プライシングにおける時間間隔 Δ_P と、非線形時系列のモデル変化における時間間隔である Δ_T の関係を、 $\Delta_P = 5\Delta_T$ としておく.これにより観測される非線形時系列の変化は、緩やかになる.

なお、プライシングのモデルにおいて、 $\pi(t)$ から $\lambda(t)$ を計算する式 (7) においては、 a,d,ω を固定する (定数とする) ようにしている.その理由としては 2 つあり、(1) 極端な値に設定しない限り類似したカオス変動を示すこと、(2) 計算の煩雑さを避けることである.いまプライシングを生成する目的を考慮すると、 a,d,ω の値は意図的に $a\approx d,\omega=1,\omega=0$ などと設定をすることはないであろう.このような極端な値の設定を避けながら a,d,ω の組み合わせをさまざまに変化させたシミュレーションを繰り返して、プライシング時系列が生成されるパターンを分析した結果、パルス状の変化が間歇的に発生する現象が観測される.

次に、第2番目の a,d,ω を可変とした場合の計算の複雑化の問題については、値を可変として推定対象とした場合には、これを含むすべての計算を暫定的なものにする必要があり、手順は極めて複雑となることに起因している。PF、セグメント構成の推定などの処理が一巡したあとで、再度 a,d,ω の値を最適化する操作が必要となる。を要することになる。なお、われわれは以前の研究(文献 [20])において、これらのパラメータを観測データから推定する方法を示しており、計算の手間さえかければ推定可能である。

3.3 システム構成の概要

以下では、本論文で提案する PF とプライシングによる時系列区分化を用いた観測データからの時系列モデル変化の推定について、システム構成を述べる。システム構成における主要なポイントは、次の3つである。

- (1) プライシングにしたがって時系列生成モデルが変化する
- 一般的には時系列生成モデルの変化は、何らかの規則性や背景となるメカニズムは仮定されていない. しかしながら本論文では、時系列生成モデルの変化の背景にはプライシングというメカニズムは存在することを前提としており、推定可能であるとしている.
 - (2) 推定のデータとしては観測される時系列データだけを用いる
- 一方では、パラメータ変化を規定するプライシングそのものは観測可能ではなく、観測できるデータは時系列データ y(t) だけであり、このデータから最適なセグメント分割とセグメントごとの非線形モデルの最適なクラス z_k を推定する.
 - (3) パラメータ変化時点を与える閾値 P_J
- $\pi(t)$ の値が P_J 以上となる時刻が、パラメータ変化時点を与えると仮定する. 式 (3) に示す $\pi(t)$ の値が大きくなるので、次の時刻 t+1 においては、入力フロー $\lambda(t)$ の値が極めて小さくなるので、変数 $\pi(t)$ は再び小さな値に復帰する.

本論文の手法において推定すべきパラメータは、次のものである。プライシング、すなわちセグメントの最適分割に関しては、容量 μ 、パラメータ変化時点を与える閾値 P_J 、プライシング時系列の初期値 $\pi(0)$ の推定を行う必要がある。これらのパラメータの推定には、よく知られている遺伝的アルゴリズム(Genetic Algorithm:GA)手法を適用する。一方、これらのパラメータの推定と並行して、それぞれのセグメントに割り当てる非線形時系列生成の状態方程式のクラス z_k の推定を行う必要がある。これら 2 つの手順は、同時には進められないので、ある時刻では一方を固定しておいて、他方を最適化する手法を用いる。

(ステップ1) プライシングの初期値(初期個体)の設定

プライシングの特性を推定する場合には、カオス変動を規定するパラメータである待ち行列の容量 μ 、変数の初期値 $\pi(0)$ のほかに、プライシング閾値 P_J を推定対象とする必要がある。まず GA 手法の適用にあたり $\pi(0)$ 、 μ 、 μ 、 μ 0 の値の候補となる個体(初期個体)を複数個ランダムに生成しておく.

(ステップ 2) 個体からのプライシング時系列の生成とセグメントへの分割

それぞれの個体に対応して、プライシング時系列の生成と時間軸のセグメントへの分割を行う. 具体的には、それぞれの個体が与える μ , P_J , π (0) の設定が最適であると仮定して、式 (7)(8)(9) にしたがって 1 つの時系列をプライシング時系列として生成し、時系列が閾値 P_J を超える時点を検出し、時間軸をいくつかのセグメントへと分割する.

(ステップ 3) セグメント s_i へのクラス z_k の割り当て

次に、個体から生成されるこれらのそれぞれのセグメント s_j において、尤度が最大となるようにセグメントに割り当てる時系列生成モデルのクラス z_k を決定する。このような計算によって得られる時系列全体についての尤度を、この個体の適合度とする。

(ステップ 4) 個体への遺伝的操作

それぞれの個体の適合度が得られたあとで、GAにおける遺伝的操作を用いて個体の性能を改善する.

(ステップ 5) 手順の繰り返し

ステップ 2, 3, 4 において個体すなわちプライシングを記述するパラメータが更新されるので, このデータを用いて, 更にステップ 2, 3, 4 における非線形時系列生成のモデルの推定結果を向上させる. この結果, 推定のレベルがあらかじめ設定した基準に達したら, 繰り返しを停止する. そうでな

い場合には、ステップ2に戻る.

なおステップ 2,3の処理については,次のような具体的な手順が含まれている.ここで考察しているプライシング時系列 $\pi(t)$ は,直接的には観測できないデータであるので,時系列の推定誤差を直接的に個体の適合度に用いることはできない.これに代わって,観測データである $\pi(t)$ から得られるデータを用いた推定方法を用いる必要がある.まずプライシング時系列 $\pi(t)$ からセグメント生成を行う.時系列 $\pi(t)$ が閾値 P_J を超える時点で,時系列モデルが変化すると見なす.すなわち,この時点で,これ以前の k 番目のセグメント s_j が終了し,新しい k+1 番目のセグメント s_{j+1} が開始されると見なす.次にセグメントに含まれる観測データへの時系列モデルの当てはめを行うセグメントの開始と終了の間に含まれる観測データ $\pi(t)$ を,ある時系列生成モデルに当てはめ,分布を推定すると同時に,その当てはまり (尤度)を求める.これが,当該の個体の適合度となる.

なお本論文においては、これ以降、プライシングにおけるパラメータついて、次のように設定して、シミュレーションを実施する。この条件を、プライシングによるパラメータ設定とする(パラメータ設定 P と呼ぶことにする)。以下ではこの記号 P により、シミュレーション条件を引用する. μ は $\mu=0.35$ などの特定の値に設定する場合もあるので、このときには説明文において明示する。

パラメータ設定 P:

 $P_J = 10 \sim 30, \mu = 0.3 \sim 0.7$

π(0): 0.5 ~ 15 の間の一様乱数

推定における仮定について

以下では、この節で述べているシステム構成に関して、推定における仮定をまとめておく.

(1) GA における繰り返し収束基準

GA によるパラメータ推定において、繰り返し計算を収束させる判断基準としては、パラメータの更新により推定値が改善されない限界を、改善の相対的な値として設定している。例えば、容量 μ の推定において、1 つ前の推定値を μ^{t-1} として今回の推定値を μ^t とした場合に、この相対変化が設定 $|\mu^t - \mu^{t-1}|/\mu^t < 0.01$ を満足する場合に限って、繰り返し計算を終了する.

(2) 初期値の推定と時系列の再現性

なお、カオス変動系から生成される時系列には初期値依存性が存在するので、プライシングの初期値の推定をどの程度正確に行う必要があるかを検証する必要がある。これを確認するために、シミュレーションによる検証を行った。 具体的には、初期値を $\pi(0)^{(1)}$ として与えたうえで、人為的にこの値に倍数 h をかけて $h=0.5\sim 2$ の範囲で変化させ、それぞれのケース $\pi(0)^{(2)}$, $\pi(0)^{(3)}$ などについてジャンプの発生個所 t_J を求める。次に、これらのジャンプ発生場所が倍数 h によって、どの程度変化するかを検証してみる。その結果、本論文の応用例の場所で述べているように、 $\pi(0)$ の相対誤差が 10%未満程度であれば、時刻 t_J は同じになることがわかる。したがって、このように推定された $\pi(0)$ と容量 μ とを用いて、あらためて時系列を再現し、閾値を超える時刻を求めセグメントを構成すると、その位置について最初に既知として与えた場所に極めて近いことが分かる。

3.4 現実データへの適用における関数推定

非線形システムの関数形状 $F(.)^{(z_k)}$, $G(.)^{(z_k)}$ をあらかじめ仮定して, シミュレーションにより本論文の方法を検証することが可能である. しかし, 一方では, 現実の観測データだけが与えられてい

る場合には、システム記述の関数形状は未知であり、推定する必要がある。本論文では、この関数形状 $F(.)^{(z_k)},G(.)^{(z_k)}$ の推定に GP 手法を用いる。 GP 手法はこれまで、カオス力学系の推定やプライシングの推定などの、いくつかの分野に適用され、その有効性が確認されている [23][24][25][26][27][28][29]. 本論文では、非線形システムを記述する非線形関数の形状の推定を行いながら、同時にその変化を検出するために、本論文の手法を適用する。そのために、 GP 手法を 2 つに分けて適用する。 すなわち、(1) 初期の時刻における関数形状の推定、(2) システムの変化を仮定した場合の関数形状の推定、である。 最初の初期段階での関数形状の推定には、 GP 手法における個体はランダムに生成したものを用いて、最適な関数形状を推定する。 具体的には、プライシングによる時系列区分化を仮定した場合に、当該のセグメントにおける最適な関数 $F(.)^{(z_k)},G(.)^{(z_k)}$ の形状を GP 手法により推定する。この場合、プライシングを記述するパラメータの最適化には、 GA 手法が適用され、このパラメータ推定のもとでの関数形状 $F(.)^{(z_k)},G(.)^{(z_k)}$ の推定には、 GP 個体を仮定した遺伝的操作を用いる。

しかしながら、このような初期段階での関数形状の推定には、計算時間を要するので、このままの方法を、時間が経過したあとでも同様に適用することは適切ではない。このようなことを考慮して、本論文では、非線形システムの時間変化を考慮した手法へと変換する。 具体的には、非線形システムの時間変化は緩やかであり、 $F(.)^{(z_k)}$ 、 $G(.)^{(z_k)}$ については現在の関数形状から、類似した関数に変形されるとする仮定を適用する。この仮定を用いることにより、計算時間が短縮できるケースが活用できる場合がある。 具体的には、GP 手法において関数を記述する個体を、まったくランダムな個体からスタートするのではなく、現在の関数形状から派生したと仮定する方法である。この方法については、ここで述べることは適切ではないので、付録 A.1 にまとめている。なお付録 A.1 では、通常の GP 手法の操作も含めて述べているので、ここでは GP 手法の全体の手順についての説明は省略する。

4 応用例

4.1 シミュレーションの目的

以下では、次に示す2つの事項について、シミュレーションを基礎として分析を行っていく.すなわち、(1)本論文で提案する時系列モデル変化をPFとプライシングによる区分化を用いて推定した場合の正確さの検証、(2)現実に観測される時系列モデル変化の現象における分析の試み、についてである。まず事項(1)については、本論文で提案するいくつかの人工的な時系列モデル変化の事例において推定の精度を調べることにより、本論文の手法が適用できる範囲を明らかにすることを目的としている。本論文で取り扱うモデルは、一般的な時系列モデル変化の推定問題を考察する場合に有効であるので、これをもとにした検証は、現実問題を分析する場合にも役立つであろう。しかしながら一方では、これらのモデルは人工的に与えられたもので、生成されるシミュレーションのデータも人工的であり、現実に適用可能であるかどうかを更に検証する必要がある。このような目的のために、事項(2)において2種類の現実データについて分析を試みる。

4.2 4つの非線形システムにおける時間変化

以下では、本論文で議論する非線形システムから生成される観測時系列を用いた区分化とシステム変化の推定について、あらかじめ形状を仮定した非線形システムに対して本論文の方法論を適用した場合の結果について述べる。以下のように定義される 4 つの非線形システム (Case I から Case IV としておく)を分析の対象とする。最初に、これらのシステムの概要についてまとめておく。なお説明において、クラスを $z_1, z_2, ...$ 、として表記し、それぞれのクラスにおける非線形システムを区別している。

Case I:離散系カオス変動から導出された方程式

離散系カオス変動のシステムとして、ロジスティック写像、エノン写像の3つを用いる (K=3)となる)。しかし、これらの写像そのものを用いると、カオス変動だけが強調されるため、本論文で行うような分析には適切ではない、不自然な時系列になる。したがって、カオスである制限を緩めるため係数を少し修正 (変形) し、しかもランダム変数 u(t)、v(t) の影響を導入した、以下の3つのシステム方程式に変換する。なお以下の3つのシステムにおいて観測データは y(t)=x(t)+v(t)、すなわち $G(x(t))^{(z_k)}=x(t)$ により生成されると仮定する。

z1:ロジスティック写像の変形

$$F(x(t)) = 3x(t)(1 - x(t)). (10)$$

z2:ベルヌーイシフト写像の変形

$$F(x(t)) = \begin{cases} 1.5x(t), & (0 < x(t) < 0.5), \\ 1.5x(t) - 0.5, & (0.5 \le x(t) < 1). \end{cases}$$
(11)

z3:テント写像の変形

$$F(x(t)) = \begin{cases} 1.5x(t), & (0 \le x(t) \le 0.5), \\ 1.5 - 1.5x(t), & (0.5 < x(t) \le 1). \end{cases}$$
 (12)

3つのシステムについて $\sigma_u = 0.01, \sigma_v = 0.01$ とする.

Case II:プライシング時系列

もともとプライシング時系列は、本論文で示すモデル推定の方法論においては、システムの変化を記述するためのジャンプ系列の生成に用いられている。しかしながら、容量 μ を変化させると、ジャンプの発生間隔の平均値が異なる時系列となるので、この性質を利用した非線形時系列の生成システムを定義する。 具体的には、式 (8) に示すプライシング時系列の生成モデルにおいて、容量を $\mu=0.35,0.55,0.75$ の 3 つに変化させた場合を 3 種類のカオス変動として定義する (K=3 となる)。また分析対象としては、特にカオス時系列である必要はないので、ランダム変数を加えた表現としている。 $\sigma_u=\sigma_v=0.1$ とする.

Case III:製品の発注の時系列

Case III においては、時系列 y(t) が小売業者から製造業者へと伝達される製品製造の数量を表現する場合に、小売業者が用いている需要予測を推定する問題を取り扱う [30][31][32]. 記号を次のように定義するx(t):小売業者の推定する需要、y(t):小売業者が製造業者に出す注文、関数 G(x(t)) で

変形される. 小売業者は市場の需要を x(t) にしたがって予測し, これをもとに製造業者に対して製造数量を伝える. しかしながら小売業者は, そのままの最適な数量ではなく, 自身が修正を加えた数量 y(t) を, 製造業者に伝達する (3 つの方法で適切な需要は変形されるので, K=3 となる). 小売業者がとるべき最適な行動を含めて, この詳細をここで述べるのは適切ではないので, 付録 A.2 にまとめておく.

変数のプロセスx(t)は1次の自己回帰モデルにしたがって次のように生成される.

$$x(t+1) = d + \rho x(t) + u(t). \tag{13}$$

パラメータ d, ρ はある範囲で自由に選ぶことはできるがシミュレーションの結果には大きな影響を与えないので、以下では $d=3, \rho=0.5, \sigma_u=1, \sigma_v=0.5$ としておく.時間遅れの導入のための変数を s(t)=x(t-1) としておく.小売業者が注文すべき最適な製品数量 r(t) は決定される (付録 A.2 に示す) が、これが小売業者の意思により関数を介して変形される.小売業者が何らかの変更を加えない場合には、数量 y(t)=r(t) として製造業者に注文される (l は小売業者があらかじめ販売を予測する期間であり、リードタイムと呼ばれる).

$$r(t) = x(t) + c_R[x(t) - x(t-1)], c_R = \frac{\rho(1 - \rho^{l+1})}{1 - \rho}.$$
 (14)

しかしながら小売業者は、自身の利益を考慮して、最適な製造数量を修正する。この場合の行動は、次に示す変換関数 G(z(t)) として記述できると仮定する。このような変換は、次のような単調増加関数からランダムに選択され、状態変数から観測時系列が生成される。

$$z_1:G(r)^{(z_1)} = r.$$

$$z_2:G(r)^{(z_2)} = 6 - r^{1.2}/2.0(0 \le r \le 6),$$

$$G(r)^{(z_2)} = 6 + r^{1.2}/2(r > 6).$$

$$z_3:G(r)^{(z_3)} = 0(\sum_{d=D_L}^{0} x(t-d) < 40),$$

$$G(r)^{(z_3)} = r(r > 40).$$

クラス z_1 は予測される需要を変更しないで発注する場合であり、クラス z_2 は予測される需要を 6 を境界として 6 より大きい (小さい) ときに、より多く (少なく) 発注する処理を行う場合である. クラス z_3 は需要の予測量を保留しておいて、前回発注を行った時刻 (時間 D_L 前としておく) から 需要予測の累計をとり、その累計が一定の値 (40 としておく) を超過した時点で、一気に注文を出す 場合を示している.

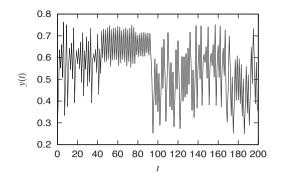
Case IV:構造変化検証のためのモデル

Case IV において用いる非線形システムは、これまで非線形システムにおける構造変化の推定問題において、例題として用いられたものである。システムの挙動は、以下の式により記述される。本論文では、このシステム記述において、パラメータりの値を3つに設定した場合の時系列を用いる。

$$x(t) = ax(t-1) + b\frac{x(t-1)}{(1+x(t-1)^2)} + 8\cos[1.2(t-1)] + u_{t-1}.$$
 (15)

$$y(t) = 0.05x(t)^2 + v_t. (16)$$

それぞれのクラスにおけるパラメータの設定は、以下のようにする. $z_1:b=25, z_2:b=12.5, z_3:b=1.2.\sigma_u=0.5, \sigma_v=0.5$ とする.



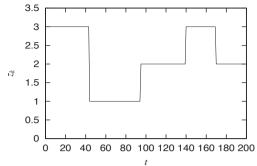
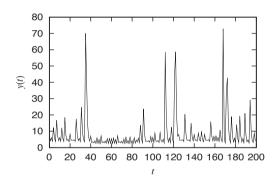


図 2: プライシングにしたがって生成された観測時系列とクラス z_k の例 (Case I)



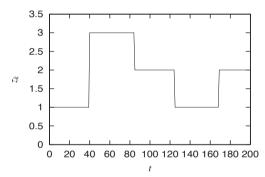


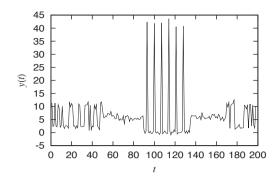
図 3: プライシングにしたがって生成された観測時系列とクラス z_k の例 (Case II)

4.3 4つのケースについての時間変化と時系列

これまで定義した 4 つの非線形システムにおける時間変化について, あらかじめ定められたプライシングにしたがってシステム変化の境界を設定し, これらの時点においてクラスの切り替え (ランダムにクラスを選択する) を行った場合の, 観測時系列 y(t) の事例を示しておく. 以下に示す図 2, 3, 4, 5 の上の部分は, それぞれ Case I, II, III, IV に対応する観測時系列 y(t) であり, 図の下の部分にはクラス z_k の時間変化を示している. 話を簡単にするために, クラス z_k を数値の k に対応させて, 示している. これらの図より分かるように, 時間的に非線形システムの切り替えが行われ, プライシングにしたがって生成される境界を挟んで, 様相が異なる時系列が出現している.

4.4 人工データを用いたパラメータ変化の推定

本論文の目的は、PF およびプライシングによる時系列区分化を用いた時系列モデルの時間変化の推定を目的としているが、これらの検討を進める前に、プライシング時系列について、生成モデルの推定結果をまとめておく、具体的には、次のような手順で行う.



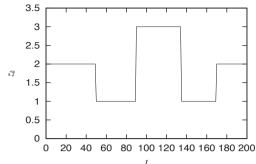
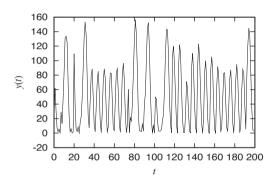


図 4: プライシングにしたがって生成された観測時系列とクラス z_k の例 (Case III)



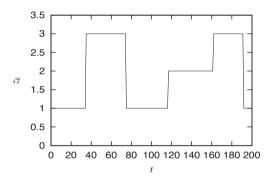


図 5: プライシングにしたがって生成された観測時系列とクラス z_k の例 (Case IV)

(1) パラメータ既知のプライシング時系列の生成

まず、プライシング時系列の生成に必要なパラメータを、既知の値として設定する。シミュレーションでは、推定により得られた結果をこの既知の値と比較することで、検証を行う。 既知である時系列の初期値 $\pi(0)$ 、容量 μ を用いてプライシングの時系列 $\pi(t)$ を生成し、 閾値 P_J を用いてセグメントを生成する.

(2) プライシングが与えるセグメントとクラス設定

次に、セグメント s_j のそれぞれにランダムに特定のクラス z_k を割り当てて、観測データ y(t) を生成する.

(3) $\pi(0)$, μ , P_I の推定

次に、この観測データ y(t) をもとにして、プライシング時系列からセグメントを生成する際の $\pi(0), \mu, P_J$ を推定できるかを検証する.

シミュレーションの条件を,以下のように設定する

容量であるパラメータ: $\mu = 0.3, 0.5, 0.7$

GA 個体における変数のストリング長:8 ビット

表 1 には、パラメータ $\pi(0)$ 、 μ 、 P_I の値を既知として与えた場合の推定結果 $(\hat{\pi}(0), \hat{\mu}, \hat{P}_I)$ として

	() /	, ,	止入しかロント	(// \		
		Case I				Case II	
μ	R_I	R_{μ}	R_J	μ	R_I	R_{μ}	R_J
0.3	0.08	0.06	0.07	0.3	0.08	0.08	0.08
0.5	0.08	0.07	0.07	0.5	0.09	0.08	0.08
0.7	0.08	0.07	0.08	0.7	0.09	0.09	0.09
		Case III				Case VI	
μ	R_I	R_{μ}	R_J	μ	R_I	R_{μ}	R_J
0.4	0.08	0.06	0.07	0.4	0.08	0.08	0.08
0.55	0.08	0.07	0.07	0.55	0.09	0.08	0.08
					l		
0.7	0.08	0.07	0.08	0.7	0.09	0.09	0.09

表 1: パラメータ $\pi(0)$, μ , P_I の推定結果 (相対誤差), (パラメータ設定 P を利用).

おく) について、容量 μ をいくつかの値に設定した場合ごとに示している. Case IV のそれぞれについて、プライシングの初期値を変えながら、それぞれのクラスが 7 個程度含まれるような非線形時系列を生成し、シミュレーションの結果を求めている。シミュレーションでは、単独プライシングにおけるパラメータ設定 P を用いる。なお、シミュレーションにおいては、最初にプライシングを生成するモデルのパラメータの値を既知として与えて推定を実施するので、その推定結果がどれくらい正しいかは、ただちに計算することができる。表 1 においては、相対誤差である $R_I = |\pi(0) - \hat{\pi}(0)|/\pi(0)$ 、 $R_{\mu}|\mu - \hat{\mu}|/\mu$ 、 $R_J = |P_J - \hat{P}_J|/P_J$ の平均値を示している。この結果から分かるように、パラメータの推定誤差は小さく抑えられており、GA 手法による推定は良好であると言える。

4.5 セグメントごとの最適なクラスの推定

次に、本論文の手法を用いて非線形時系列生成モデルをセグメントごとに推定した結果について整理する。シミュレーションにおいては、あらかじめ初期設定において、セグメントごとに割り当てている関数のクラスが分かっているので、本論文の手法により推定したセグメントごとのクラス推定値と比較することにより、性能を評価することができる。すでに述べたように、容量 μ をいくつかの値に設定した場合ごとに Case I \sim Case IV のそれぞれについて、プライシングの初期値を変えながら、それぞれのクラスが7個程度含まれるような非線形時系列を生成し、シミュレーションの結果を求めている。シミュレーションでは、単独プライシングにおけるパラメータ設定 P を用いる.

表 2 には、それぞれのケースについてクラスごとに、セグメントが正しく検出された割合を示している。セグメントの真のクラスは既知であるので、時刻 t ごとに推定されたセグメントのクラスと、既知のクラスとの比較を行うことで、正しい検出の割合を求めている。この表から分かるように、セグメントの識別は Case I \sim Case IV ごとに多少の変化はみられるが、ほぼ $0.8\sim0.95$ の範囲に収まっており、本論文の手法の妥当性を示すものとなっている。

	Case I				Case II		
μ	z_1	z_2	z_3	μ	z_1	z_2	z_3
0.3	0.94	0.80	0.92	0.3	0.90	0.92	0.92
0.5	0.95	0.81	0.93	0.5	0.90	0.92	0.92
0.7	0.95	0.82	0.94	0.7	0.92	0.92	0.93
		Case III				Case VI	
μ	z_1	z_2	z_3	μ	z_1	z_2	z_3
0.4	0.90	0.91	0.91	0.4	0.89	0.91	0.80
0.55	0.90	0.91	0.92	0.55	0.90	0.92	0.81
0.7	0.89	0.91	0.92	0.7	0.90	0.93	0.81

表 2: セグメントに割り当てたクラスの推定結果 (パラメータ設定 P を利用).

4.6 現実データの時系列牛成モデル変化の実時間推定

次に、現実に観測される時系列データに対して、本論文の手法を適用して非線形モデルの推定を行い、その適用可能性を考察する。これまで人工データとして Case I \sim Case IV をとりあげた。しかし、Case I、Case IV は、数値例としては意味があるが、現実に観測される時系列ではない。このようなことを考慮して、しかも、比較的入手しやすい現実データへの適用を考え、以下の2つのケース (Case A、Case B) について、非線形モデルの変化を実時間で推定する方法を適用し、その結果をまとめておく、Case A、B は、Case III ないしは Case II に対応する現実データであると考えられる。

現実の時系列データの場合には、パラメータ時間変化がカオス (プライシング) で記述できるとは限らないことを考慮する必要がある。すなわち、パラメータ時間変化がなんらかの規則性をもっていると仮定したとしても、その変動が、カオス変動 (プライシング) であるとは限定できない。したがって、プライシングによりジャンプ発生をモデル化することには限界がある。ここでは、分析手法について、以下のように制限を緩和することで、近似的にパラメータの同時変化を推定する方法へと拡張を行う。具体的には、擬似的なジャンプの導入と、評価におけるペナルティの導入である。いま、プライシング時系列においてジャンプが発生し、第j番目のセグメントと第j+1番目のセグメントにおける尤度に差異があると推定された場合に、この差異が小さい場合には、プライシングによるパラメータ変化の検出が正確でないことに対応する。したがって、この差異に逆比例して、プライシングのパラメータとして推定された値の評価値を減じる。

(1) 現実に観測される時系列データ

Case A:2 つの製品の実際の需要データと発注データ

ここで用いるのは文献 [31][32] に示されている, 実際に観測された製品の需要および発注データであるが, ここでは少し加工を行っている. 元の文献 [31][32] では, 需要予測をする小売業者の推定する理想的な需要時系列 $y_1(t)$ と, これを小売業者が独自に判断で修正して製造業者に伝達する時系列 $y_2(t)$ の 2 つが示されている. これらの時系列においては, $y_1(t)$ は比較的緩やかな傾向を示すのに対して, 時系列 $y_2(t)$ は, 小売業者のより有利に商品を調達したい小売業者の行動が反映され,変動の大きなものになっている. 本論文では, これらの 2 種類の時系列を $y_1(t), y_2(t)$ の順番で接続

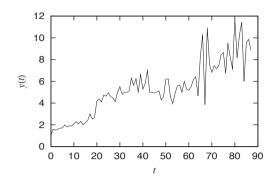


図 6: Case A における時系列の例

して、1つの時系列とみなして、これらの接続時点の推定と、非線形モデルの関数形状の推定がなされるかを同時に検証する。

Case B:対個人・対事業所サービス業界データ

日本における 2000 年から 2013 年までの対個人・対事業所サービス業界のデータであり、対個人サービスではテーマパークなど 8 つの業種のデータ (Case B, Data I としておく)を、対事業サービスではリース業など 15 の業種における来客数 (会社にサービスを依頼する件数)の月次データ (Case B, Data II としておく)として集計されている [33]. 客がでかける時期や、その増減には関連性があると思われる (来客や件数が公表されていない場合には、対応する売上高で代替する). 具体的には、対個人サービスについては、映画館、劇場・興行場、ゴルフ練習場、遊園地・テーマパーク、結婚式場業、カルチャーセンターなどについて、客の到着の集計データを整理し、到着 (来客)率の変化における変化のメカニズムを推定する. 対事業サービスについては、会社を相手にしたサービス提供の発生率を集計して、同様な分析を行っている。業種としては、物品賃貸(リース)業、業物品賃貸(レンタル)業、情報サービス業、広告業、エンジニアリング業、インターネット附随サービス業などである. 景気変動などで、会社 (事業)へのサービスが増減する時間的な経過が、時間的に変化するかを検証する.

図 6,7には, Case A, Case B における時系列の例を示している (2 つの時系列のみを示す).

(2) GP による非線形システムの推定

現実に観測される時系列データから、それを生成する非線形システムを推定する問題において、本論文では GP による関数形状の推定を用いる. GP 手法はさまざまな分野において適用され、未知である関数形状の推定に有効であることが示されている. しかしながら、何も基準がないケースにおいて、GP を適用して未知である関数形状を推定することは、計算量の増大という問題を発生する. したがって、ここでは、経済時系列の生成モデルに関する性質を用いて、より簡潔な方法で、非線形システムにおける関数推定を行う. 具体的には、経済時系列生成モデルにおける、季節性を利用する方法である.

国の経済のサイクルは、1 年を通じて、極めて明確なサイクルが存在することが知られており、例えばデパートの売り上げは、毎年 12 月に大きなピークがある。これを季節性と呼んでいる (S(t) としておく)、また、長年にわたって時系列のベースとなる数値が緩やかに変化するトレンドと呼ばれ

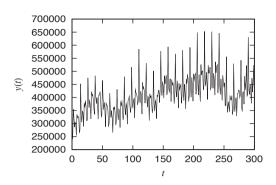


図 7: Case B における時系列の例

るものが存在する (T(t) としておく). これらの 2 つを, 観測時系列から除去したあとには, 不規則な変動を示す時系列が残る (不規則成分と呼ばれ, R(x(t)) としておく). このような成分への分解を行うことにより, GP による関数形状の推定も, やや単純な問題に帰着させることができる.

本論文では、季節性への分解において、成分の積で時系列を表現する乗法モデルを用いる.これまで述べた非線形時系列の生成モデルを、この乗法モデルで表現すると、以下のようになる.

$$F(x)^{(z_k)} = R(x), G(x)^{(z_k)} = S(t)T(t)F(x)^{(z_k)}.$$
(17)

季節性 S(t) は明確な周期性をもっているので、周期関数により近似する. R(x(t)) は不規則な動きをするので、通常行われる近似方法にしたがって、1次の自己回帰 (Auto-regressive:AR) モデルを適用して近似する. トレンド T(t) は単調な線形、あるいは低次の多項式の関数で近似する.

なお人工データの場合と異なり、プライシング生成のパラメータの真の値は分からないので、ここでは推定結果の特性についてのみ説明する. 観測される時系列の長さは、以下のようになる (No. は時系列の識別番号を意味する).

Case A:No.1 の時系列について 90, No.2 について 112

Case B(Data I):No.1~No.8 のすべてで 159

Case B(Data II):No.1~No.6 では 302, No.7~No.15 では 59

Case A, Case B について, 以下のような時系列の同時変化を記述するパラメータの推定結果が得られる.

まず表 3 においては、Case B(Data I, Data II) において検出されるセグメントの個数 N_S を示している。なお Case A は、もともと 2 つのセグメントを結合したものとなっており、シミュレーションによる結果においても、推定されたセグメント個数は 2 となっている。現実のデータ (Case B) では時系列の長さは人工データと比較して相対的に短いので、識別されるセグメント個数とクラス数は同じになる。検出されるセグメントの個数 N_S は、時系列の長さに関連しているが、それほど長い時系列ではないので、 $N_S=2$, $N_S=3$ に分布している。次に表 4 においては、図 6 と図 7 に示した時系列の事例についてだけ、GP 手法により推定された関数形状 $F(.)^{(z_k)}$, $G(.)^{(z_k)}$ を、セグメントごとに示している。ただしそれぞれの関数形状についてセグメントの開始時刻を、t=1 としている.推定された関数形状は、S(t) については周期関数、T(t) については緩やかに変化する単調な関数と

表 3: Case B の時系列に含まれるセグメントの個数 N_S の推定結果 (パラメータ設定 P を利用).

Data I:No.	1	2	3	4	5	6	7	8
Data 1.No.	1		0	4	0	U	1	0
N_S	3	2	2	3	3	2	3	3
Data II:No.	1	2	3	4	5	6	7	8
N_S	3	4	3	2	2	2	2	2
Data II:No.	9	10	11	12	13	14	15	-
N_S	2	2	2	2	3	2	2	-

表 4: GP 手法により推定された関数形状 $F(.)^{(z_k)}, G(.)^{(z_k)}$

Case A	z_1
F(x)	0.433x + 0.569
S(t)T(t)	[0.893 + 0.126t]
Case A	z_2
F(x)	-0.080x - 0.045
S(t)T(t)	$[0.984 + 0125t][1.5 + \sin(0.3t + .05)]$
Case B	z_1
F(x)	0.551x + 5.79
S(t)T(t)	$[3 + \sin(t + 0.1)] \times [1 + 0.35\delta(t - 12i)]$
Case B	z_2
F(x)	-0.424x + 8.54
S(t)T(t)	$[6 + \sin(2t + 1.4)] \times [1 + 0.35\delta(t - 12i)]$
Case B	z_3
F(x)	-0.045x + 7.29
S(t)T(t)	$[6 + \sin(2t + 2)] \times [1 + 0.35\delta(t - 12i)]$

なっている。また R(x(t)) は、通常経済時系列を近似するモデルとして用いられている 1 次の自己回帰モデルが、適切であることが示されている。

なお、時系列を区分化するために用いられる (推定される) プライシングのパラメータ μ , P_J , π (0) については、真の値が知られていないので、その範囲だけを述べておく、これらの値は、人工データのシミュレーション分析に用いているパラメータの値の範囲と、ほぼ同じであることが確認できる.

 μ :0.45 \sim 0.62

 $P_J:12.2 \sim 15.4$

 $\pi(0):0.5 \sim 0.9$

5 むすび

本論文では、PFとプライシングによる時系列区分化の手法を用いた非線形時系列生成モデル変化の推定について述べた. 時系列の観測データから PFにより状態を推定する問題を定義し、尤度の時間変化を用いて時系列を区分化する場合に、プライシングを用いて効率を向上させた. 現実データに対して推定を行うため GPの手法により、変化の発生と変化後の関数を推定する方法を提案した. 応用例として人工的なデータを用いた分析と、現実データからのモデル変化と変化後の状態方程式推定を議論した.

今後、より複雑な関数形のシステムへの適用可能性を検討する予定である.

謝辞

本研究の一部は、日本学術振興会科学研究費基盤研究 (B)23310104 により実施されている. ここに感謝の意を表する.

参考文献

- E.Punskaya, C.Andrieu, A.Doucet and W.J.Fitzgerald, "Bayesian curve fitting using MCMC with applications to signal segmentation," *IEEE Transactions on Signal Processing*, vol.50, no.3, pp.747-758, 2002.
- [2] N.Dobigeon, J.Y.Tourneret and J.D.Scargle, "Joint segmentation of multivariate astronomical time series:Bayesian sampling with a hierarchical model," *IEEE Transactions on Signal Processing*, vol.55, no.2, pp.414-423, 2007.
- [3] J.Y.Tourneret, M.Doisy, and M.Lavielle, "Bayesian retrospective detection of multiple change points corrupted by multiplicative noise. Application to SAR image edge detection," Signal Processing, vol.83, pp.115-131, 2004.
- [4] E.Fox, E.B.Sudderth, M.I.Jordan and A.S.Willsky, "Bayesian nonparametric inference of switching dynamic linear models," *IEEE Transactions on Signal Processing*, vol.59, no.4, pp.1569-1585, 2011.
- [5] F.Caron, M.Davy, A.Doucet, E.Duflos and P.Vanheeghe, "Bayesian inference for linear dynamic models with Dirichlet process mixtures," *IEEE Transactions on Signal Processing*, vol. 56, no. 1, pp. 71-84, 2008.
- [6] G.C.Chow, "Tests of equality between sets of coefficients in two linear regressions," *Econometrica*, vol.28, no.3, pp.591-605, 1960.
- [7] N.Vaswani, "Additive change detection in nonlinear systems with unknown change parameters," *IEEE Transactions on Signal Processing*, vol.55, no.3, pp.859-872, 2007.

- [8] N.Vaswani, A.RoyChowdhury and R.Chellapa, "Shape activity: A continuous state HMM for moving/deforming shapes with application to abnormal activity detection," *IEEE Trans. Image Processing*, vol.14, no.10, pp.1603-1616, 2005.
- [9] A.Andreiu, A Doucet, S.S.Singh and B.Tadic, "Particle methods for change detection," system identification and control, *Proc.IEEE*, vol.93, pp.423-438, 2004.
- [10] V.V.Kadirkamanathan, P.Li, M.H.Jaward and S.G.Farbi, "Particle filtering based fault detection in nonlinear stochastic systems," *International Journal of System Science*, vol.33, pp.259-265, 2002.
- [11] A.Doucet, N.J. Gordon and V. Krishnamurthy, "Particle filters for state estimation of jump Markov linear systems," *IEEE Transactions on Signal Processing*, vol. 49, no. 3, pp. 613-624, 2001.
- [12] A.Doucet, N.de Freitas and N.Gordon, Sequential Monte Carlo Methods in Practice, New York, Springer Verlag, 2001.
- [13] F.Gustafsson, F.Gunnarsson, N.Bergman, U.Forssell, J.Jansson, R.Karlsson and P.Nordlund, "Particle filters for positioning, navigation, and tracking," *IEEE Transactions on Signal Processing*, vol.50, no.2, pp.425-437, 2002.
- [14] J.H.Kotecha and P.M.Djuric, "Gaussian sum particle filtering," *IEEE Transactions on Signal Processing*, vol.51, no.10. pp.2602-2612, 2003.
- [15] 時永祥三, 譚 康融, "遺伝的プログラミングによる方程式近似に基づく粒子フィルタを用いた時系列からの状態推定とその変動抑制への応用," 電子情報通信学会論文誌,vol.J93-A, no.11, pp.739-755, 2010.
- [16] W.Yi, M.R.Morelande, L.Kong and J.Yang, "A computationally efficient particle filter for multitarget tracking using an independent approximation," *IEEE Transactions on Signal Processing*, vol.61, no.4, pp.843-856, 2013.
- [17] 時永祥三, "3 つのネットワーク結合離散系カオス変動モデルにおける edge snapping による同期化とその応用," 情報処理学会論文誌, 数理モデルとその応用, vol.6, no.2, pp.36-52, 2013.
- [18] 時永祥三, 池田欽一, "結合型プライシングによる時系列パラメータの同時変化のモデル化とベイズ手法によるその推定," 電子情報通信学会技術研究報告, vol.113, no.78, SIS2013-11, pp.53-58, 2013.
- [19] C.M.Rump and S. Stidham, Jr., "Stability and chaos in input pricing for a service facility with adaptive customer response to congestion," *Management Science*, vol.44, no.2, pp.246–261, 1998.

- [20] 池田欽一, 陳暁栄, 時永祥三, "GP による学習を基礎としたマルチエージェント・システムによるプライシング時系列のカオス性分析とその応用," 電子情報通信学会論文誌, vol.J89-A, no.4, pp.298-307, 2006.
- [21] 池田欽一, 時永祥三, "ネットワーク構成されたノードにおけるフロー入力調整と退去を含むプライシング時系列のカオス解析とその抑制," 電子情報通信学会論文誌, vol.J93-A, no.1, pp.1-10, 2010.
- [22] 池田欽一, 時永祥三, D" "ノードへのフロー入力調整を含むネットワークにおけるプライシング のカオス性変動の解析とその抑制,"情報処理学論文誌, TOM0202004, vol.2, no.2, pp.22-37, 2009.
- [23] 陳暁栄, 時永祥三, "共進化 GP を用いたマルチエージェントシステムの構成とその人工市場分析への応用,"電子情報通信学会論文誌, volJ86-A, no.10, pp.1038-1048, 2003.
- [24] Y.Ikeda and S.Tokinaga, "Chaoticity and fractality analysis of an artificial stock market by the multi-agent systems based on the co-evulutionary Genetic Programming," *IEICE Trans.Fundamentals*, vol.E87-A, no.9, pp.2387-2394, 2004.
- [25] 池田欽一, 陳暁栄, 時永祥三, "GP による学習を基礎としたマルチエージェント・システムによるプライシング E 時系列のカオス性分析とその応用," 電子情報通信学会論文誌, vol.J89-A, no.4, pp.298-307, 2006.
- [26] 池田欽一, 時永祥三, 呂建軍, "遺伝的プログラミングと遅延とモグララフィを用いたネットワーク構成の同定と内部遅延時間の推定," 情報処理学会論文誌, vol.47, no.SIG 1(TOM 14), pp.12-18, 2006.
- [27] Y.Ikeda and S.Tokinaga, "Multi-fractality Analysis of time series in artificial stock market generated by multi-agent systems based on the Genetic Programming and its applications," *IEICE Trans.Fundamentals*, vol.E90-A, no.9, pp.2212-2222, 2007.
- [28] 時永祥三, 池田欽一, "局所的交流による行動決定と状態遷移を行うマルチエージェントから なる平面上の エージェント・クラスタ形成分析," 情報処理学会論文誌, TOM, vol.4, no.4, pp.19-36, 2011.
- [29] 時永祥三, 岸川善紀, "遺伝的プログラミングと多段ファジイ推論に基づくジャンプ過程を含む 時系列生成モデルの推定,"電子情報通信学会論文誌, vol.J93-A, no.5, pp.365-374, 2010.
- [30] H.L.Lee, V.Padmanabhan and S.Whang, "Information distortion in a supply chain:The bullwhip effect," *Management Science*, vol.43, no.4, pp.546-558, 1997.
- [31] H.L.Lee, K.C.So and C.S.Tang, "The value of information sharing in a two-level supply chain," *Management Science*, vol.46, no.5, pp.626-643, 2000.

- [32] Z.J.Ren, M.A.Cohen, T.H.Ho and C.Terwiesch, "Information sharing in a long-term supply chain relationship: The role of customer review strategy," *Operations Research*, vol.56, no.1, pp.81-93, 2010.
- [33] 総務省統計局データ,http://www.meti.go.jp/statistics/tyo/tokusabido/result/result 1.html

付録 A:GP による関数の変形の推定

GP 手法を関数近似に適用する問題においては,近似するための非線形関数の候補を多数の GP 個体として準備しておいて,これらの関数の近似度に対応する適合度が相対的に大きな個体どうしに対して交差処理を適用し,新たに生成された個体を,個体プールの中で適合度が相対的に小さい個体と置き換える.このような操作を繰り返すことにより,個体の関数近似の精度を向上させる. 時系列を予測する方程式は木構造で表現できるが,これを前置表現 (prefix representation) により置き換えておく. 前置表現のそれぞれを,個体とよぶ.

(ステップ1)

乱数を用いて被演算子, 演算記号の並びからなる初期個体のプール P-S を構成する.

(ステップ2)

個体に表現された関数をもとに、それぞれの個体により得られる予測値を求める。これをもとにして、個体における適合度 S_i を求める。

(ステップ3)

適合度に比例する確率に応じて、2つの個体iが選択され、この2つの個体に対して遺伝的操作を行い、生成された新しい個体を次のステップにおける代替個体のプールである P-B に格納しておく、このような新しい個体の生成を、規定回数繰り返す。新規個体の生成が終了したら、プール P-A(最初の繰り返しではこれはプール P-S に等しい)の個体の中で、相対的に適合度の低い個体を、プール P-B の個体により置き換える。この結果、プール P-A は、もとから存在する適合度の高い個体と、生成された適合度が高い個体から構成されることになる。

(ステップ4)

ステップ2からステップ4までの交差処理をすべての個体に適用し,新しい個体のプールを作成したあとに,次に示す突然変異を実施する. 突然変異は任意に個体を選択して,この個体の被演算子,演算記号の部分を,任意に選択した被演算子,演算記号により置き換える.

(ステップ5)

ステップ2からステップ4までの操作を規定回数繰り返す.

なお、個体における演算子と被演算子の相互配置を調べるためにカウンタを用いて個体が関数を表現しているかどうか調べたり、交差処理において個体の切断と接続を行う必要性があり、また局所的な解を回避する突然変異を用いるが、これらの詳細はここでは省略する.

しかしながら、本論文で議論するような構造変化をしたあとのシステムの状態方程式を推定する場合には、やや異なる手順で GP 手法における交差処理を利用する. すなわち、現在までシステムの挙動を記述している関数 C が、ある時刻において別の関数に変化 (構造変化) することを仮定するが、この関数の形状は現在の関数 C の形状に類似する側面をもち、C から派生すると仮定し、GP 手法における交差処理により、この変形後の関数形を推定する方法を適用する. すなわち GP 手法に

おける個体プールには、ランダムに生成された関数の集合があり、関数 C を変形する相手の関数であるとみなすことができる。現在の関数形 C に対して、ランダムに生成された GP の個体プールのそれぞれの個体に対して交差処理をほどこすことにより、関数 C から変形された複数の関数形 A_i を個体プールの個体の個数だけ求めることができる。なお交差処理により複数の個体が生成されたあとでは、これらの個体 A_i と C とを用いて更に GP 手法が適用される。このように生成された関数 (個体) のすべてに対して PF を適用することにより、予測誤差が最小となる関数 A_m を見出すことができる。

付録 B: 小売業者と製造業者の間での製造数量の伝達

x(t) を時刻 t に見込む需要とし、以下の式にしたがって生成されると仮定する.

$$x(t+1) = d + \rho x(t) + \epsilon(t) \tag{18}$$

ただし, $\epsilon(t)$ は, 平均がゼロで標準偏差が σ である正規乱数である. 小売業者は時刻 t から t+l+1 までの間に発生する需要に相当する数量を製造業者に注文すると仮定すると, その数量 $r(t) = \sum_{i=1}^{l+1} x(t+i)$ は次のようになる [31].

$$r(t) = \frac{1}{1-\rho} \left[d \sum_{l=1}^{l+1} (1-\rho^l) + \rho (1-\rho^{l+1}) x(t) \right]$$

$$+\epsilon_i (t+l+1) + (1+\rho)\epsilon_i (t+l) + \dots +$$

$$+ (1+\rho+\rho^2 + \dots + \rho^l)\epsilon_{t+1}.$$
(19)

ここで、小売業者が保持すべき最適な在庫数量を媒介変数として用いて、式 (19) を変形すると、式 (19) は次のような簡潔な形に帰着することができる.

$$r(t) = x(t) + c_R(x(t) - x(t-1)), c_R = \frac{\rho(1 - \rho^{l+1})}{1 - \rho}.$$
 (20)

更に小売業者は、本来の注文数量を自身の都合により変形することが知られている。これを簡潔にまとめると、1)需要予測の誤差、2)注文の形式、3)商品不足のへの反応、4)価格への反応があるとされている [30].1 番目はエージェントが行う需要予測の精度に関するものであり回避できないが、2番目は注文を数期分をまとめて出すためであり、3番目は商品の需要が増加する局面で販売数量を確保できなくなることを懸念し、多めに注文を出すことに対応する。また第4番目は製品の価格が相対的に安い時期と高い時期が混在する場合には安い(高い)時期により多く(少なく)注文することに対応する。これらを関数で表現するために、以下のケースとして仮定する。

z1:小売業者は特に注文数量を変形しない

z₂:一定量を超える(超えない)とある関数にしたがって注文を増やす(減らす)

23:一定数量になるまで注文をしないで、あとでまとめて注文する

 z_2, z_3 のケースでは理想的な注文数量を r とした場合に、それぞれ、本文中に示すように、次のような変形を施すことで近似できる.

$$z_2:G(r) = 6 - r^{1.2}/2.0(0 \le r \le 6),$$

 $G(r) = 6 + r^{1.2}/2(r > 6)$

$$z_3:G(r) = 0(\sum_{d=D_L}^0 x(t-d) < 40),$$

 $G(r) = r(r > 40)$

時永 祥三〔九州大学名誉教授〕 池田 欽一〔北九州市立大学経済学部 准教授〕